

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE ENGENHARIA DO CONHECIMENTO**

Jean Carlo Rossa Hauck

**UM MÉTODO DE AQUISIÇÃO DE CONHECIMENTO
PARA CUSTOMIZAÇÃO DE MODELOS DE
CAPACIDADE/MATURIDADE DE PROCESSOS DE
SOFTWARE**

Florianópolis

2011

Jean Carlo Rossa Hauck

**UM MÉTODO DE AQUISIÇÃO DE CONHECIMENTO
PARA CUSTOMIZAÇÃO DE MODELOS DE
CAPACIDADE/MATURIDADE DE PROCESSOS DE
SOFTWARE**

Tese submetida ao Programa de Pós
Graduação em Engenharia e Gestão do
Conhecimento da Universidade
Federal de Santa Catarina para a
obtenção do Grau de Doutor em
Engenharia e Gestão do Conhecimento
Orientador: Prof. Dr. rer. nat. Aldo von
Wangenheim

Florianópolis

2011

Catálogo na fonte elaborada pela biblioteca da
Universidade Federal de Santa Catarina

Ficha catalográfica a ser confeccionada pela Biblioteca Central.

Jean Carlo Rossa Hauck

**UM MÉTODO DE AQUISIÇÃO DE CONHECIMENTO
PARA CUSTOMIZAÇÃO DE MODELOS DE
CAPACIDADE/MATURIDADE DE PROCESSOS DE
SOFTWARE**

Esta Tese foi julgada adequada para obtenção do Título de “Doutor em Engenharia e Gestão do Conhecimento”, e aprovada em sua forma final pelo Programa de Pós Graduação em Engenharia e Gestão do Conhecimento

Florianópolis, 25 de Fevereiro de 2011.

Prof. Dr. Paulo Maurício Selig
Coordenador do Programa

Banca Examinadora:

Prof. Dr. rer. nat. Aldo von Wangenheim
Presidente - UFSC

Prof^a. Dr. rer. nat. Christiane Gresse
von Wangenheim - UFSC

Prof. Mario Antônio Ribeiro Dantas, PhD
Membro - UFSC

Prof. Dr. Clenio Figueiredo Salviano
Externo - CTI

Prof. Dr. Vinícius Medina Kern
Membro - UFSC

Prof. Dr. Fergal Mc Caffery
Externo - DkIT

À minha esposa Jane
e ao meu filho Yan

AGRADECIMENTOS

Ao meu orientador prof. Aldo von Wangenheim pela oportunidade e apoio durante todo este doutorado.

À profa. Christiane Gresse von Wangenheim pela dedicação dispensada a este trabalho e por toda pesquisa realizada em conjunto.

Ao prof. Fergal Mc Caffery pela oportunidade de participar do projeto Medi SPICE e por toda orientação recebida.

A todos os demais membros do projeto Medi SPICE pelas valiosas contribuições para a evolução do método proposto nesta tese.

Aos pesquisadores membros do LERO – Irlanda pelas diversas oportunidades de discutir e evoluir a pesquisa em conjunto.

Aos pesquisadores membros do *Regulated Software Research Group* – Irlanda pelas importantes contribuições para esta pesquisa.

Aos colegas pesquisadores e funcionários do Cyclops/INCoD por todo apoio.

À pesquisadora Maiara Heil Cancian por ter participado ativamente na aplicação desta pesquisa.

Aos pesquisadores que participaram do *Expert Panel* pelo esforço empregado na revisão criteriosa do método.

À direção da empresa SPECTO Tecnologia por ter apostado e investido neste trabalho.

À CAPES que me proporcionou a oportunidade de estágio no exterior.

Aos meus pais Wilson e Miriam que me deram tudo o que eu precisava para chegar até aqui: fé e valores.

A Deus por mais esta grande conquista.

*“Crying, cockles and mussels,
alive, alive, oh!”.*

James Yorkston

RESUMO

A Engenharia do Conhecimento provê métodos que possibilitam o entendimento das estruturas e processos utilizados por especialistas, no intuito de criar uma melhor integração da tecnologia da informação em suporte ao trabalho intelectual. Um dos principais processos da Engenharia do Conhecimento é a aquisição de conhecimento, que consiste em extrair o conhecimento necessário a partir de suas diversas fontes, de modo a poder codificá-lo e reutilizá-lo. O conhecimento representado na forma de melhores práticas constitui-se no encapsulamento de experiências que, quando repetidas, levam a alcançar resultados semelhantes. Nesse sentido, os Modelos de Capacidade/Maturidade de Processo de Software (SPCMMs) são frameworks de melhores práticas de desenvolvimento de software e têm sido customizados para atender as necessidades específicas de qualidade de cada domínio de desenvolvimento de software. Neste sentido, esta tese apresenta um método de aquisição de conhecimento para customização de SPCMMs para domínios específicos, desenvolvido com base nas experiências de desenvolvimento de SPCMMs relatadas na literatura, nos processos e técnicas de aquisição de conhecimento, processos de desenvolvimento de normas de qualidade e em frameworks de desenvolvimento de modelos de qualidade de processo. O método é avaliado por especialistas e utilizado na customização de dois SPCMMs. Os resultados observados revelam primeiros indícios de que o método é adequado e aplicável à aquisição de conhecimento para a customização de SPCMMs. O método desenvolvido contribui para a Engenharia do Conhecimento na pesquisa atual em aquisição do conhecimento a partir de fontes não estruturadas e na área de aplicação em Engenharia de Software fornecendo um suporte sistemático para a customização de SPCMMs.

Palavras-chave: Aquisição de Conhecimento. Melhores Práticas. Modelos de Capacidade/Maturidade de Processos de Software. Qualidade de Processo.

ABSTRACT

The Knowledge Engineering provides methods that enable the understanding of the structures and processes used by experts leading to a better integration of information technology in support of intellectual work. One of the key processes of Knowledge Engineering is the knowledge acquisition, which is extracting the necessary knowledge from its various sources, in order to encode it and reuse it. The knowledge represented in the form of best practices is the package of experiences that, when repeated, lead to achieve similar results. The Software Process Capability/Maturity Models (SPCMMs) are frameworks of software development best practices and have been customized to meet specific quality needs of each domain of software development. Thus, this thesis presents a knowledge acquisition method for SPCMMs customization for specific domains, developed based on the experiences of developing SPCMMs reported in the literature, techniques and processes of knowledge acquisition, development processes of quality standards and development frameworks of process reference models. The method is evaluated by experts and used in customizing two SPCMMs. The observed results reveal early evidence that the method is appropriate and applicable to the knowledge acquisition for SPCMMs customization. The developed method contributes to the Knowledge Engineering on current research in knowledge acquisition from unstructured sources and in the application area in Software Engineering by providing a systematic support for SPCMMs customization.

Keywords: Knowledge Acquisition. Best Practices. Software Process Capability/Maturity Models. Process Quality.

LISTA DE FIGURAS

Figura 1: Contexto da pesquisa no método científico	36
Figura 2: Etapas da Pesquisa	38
Figura 3: Alinhamento à KE do método de aquisição de conhecimento de SPCMMs	40
Figura 4: Estrutura da Tese	41
Figura 5: Ciclo de vida de KE	45
Figura 6: Visão Geral do modelo CommonKADS	46
Figura 7: Visão Geral do modelo MIKE.....	47
Figura 8: Aquisição de conhecimento no ciclo de vida de KE	50
Figura 9: Aquisição de Conhecimento a partir de melhores práticas	56
Figura 10: Modelo de DeLone e McLean.....	58
Figura 11: As duas dimensões do SPCMM definido na ISO/IEC 15504.....	62
Figura 12: Os 50 processos definidos na norma ISO/IEC 15504.....	66
Figura 13: Extrato de detalhamento de um processo na ISO/IEC 15504.....	67
Figura 14: Processo de Revisão Sistemática da Literatura.....	74
Figura 15: Domínios dos SPCMMs identificados na SLR	78
Figura 16: Percentagem de uso dos modelos como base para os SPCMMs identificados na SLR.....	79
Figura 17: Metodologia utilizada para desenvolver os SPCMMs.....	86
Figura 18: Grau de execução das fases e passos típicos	88
Figura 19: Grau de uso prático dos modelos.....	88
Figura 20: Número de modelos desenvolvidos / pelo número de representantes envolvidos.....	89
Figura 21: Pirâmide metodológica.....	98
Figura 22: Desenvolvimento do método	100
Figura 23: Estrutura de definição do método.....	102
Figura 24: Apresentação do método	104
Figura 25: Papéis envolvidos no método	105
Figura 26: Fases e Atividades do Método.....	106
Figura 27: Extrato do formulário web contendo o questionário	127
Figura 28: Formação acadêmica dos especialistas.....	131
Figura 29: Experiência dos Especialistas.....	131
Figura 30: Modelos desenvolvidos pelos participantes.....	132
Figura 31: MQ1.1: Quantidade de erros identificados.....	137
Figura 32: MQ3.1, MQ3.2, MQ3.2 e MQ 3.4 por fase do método	139
Figura 33: MQ4.1: Impressão subjetiva sobre a capacidade do método de suportar a customização de SPCMMs.	141
Figura 34: MQ5.1: Impressão subjetiva sobre a capacidade do método de capturar as melhores práticas no domínio.....	142
Figura 35: MQ6.1: Impressão subjetiva sobre a completude do método em relação às demais abordagens existentes.	143
Figura 36: MQ7.1: Impressão subjetiva sobre a generalidade do método.	144

Figura 37: Atividades executadas no Estudo 1.....	153
Figura 38: Critérios de qualidade e fontes.....	154
Figura 39: Critérios de qualidade priorizados.....	155
Figura 40: Extrato do mapeamento entre critérios e melhores práticas.....	156
Figura 41: Site web onde o modelo SaaS está publicado.....	157
Figura 42: Utilização de Mapas mentais na Fase 1.....	162
Figura 43: Portal do Medi SPICE.....	165

LISTA DE QUADROS

Quadro 1: Classificação da Pesquisa	37
Quadro 2: Comparação entre frameworks de KE	48
Quadro 3: Requisitos dos SPCMMs	59
Quadro 4: Características dos SPCMMs.....	60
Quadro 5: Atributos de processo e níveis de capacidade	64
Quadro 6: Protocolo de Revisão	75
Quadro 7: Fases realizadas no desenvolvimento de SPCMMs	80
Quadro 8: Modelos e normas analisados no survey	85
Quadro 9: Fases e passos genéricos de desenvolvimento de.....	86
Quadro 10: Abordagens propostas para desenvolvimento de SPCMMs.....	93
Quadro 11: Detalhamento da atividade A1.2.....	112
Quadro 12: Detalhamento da técnica TE07	114
Quadro 13: Detalhamento do papel RL01	116
Quadro 14: Perguntas e Medidas do Objetivo de Avaliação 1.....	120
Quadro 15: Perguntas e Medidas do Objetivo de Avaliação 2.....	122
Quadro 16: Coleta de dados para o Objetivo de Avaliação 1.	125
Quadro 17: MQ2.1: Inconsistências identificadas.	137
Quadro 18: Lista das Atividades, Tarefas e Técnicas consideradas faltantes ..	140
Quadro 19: MQ8.1: Pontos fortes do método	145
Quadro 20: MQ9.1: Pontos fracos do método	146
Quadro 21: Coleta de dados para o Objetivo de Avaliação 2.	150
Quadro 22: Resumo da utilização do método observada no estudo 1.....	157
Quadro 23: Resumo da utilização do método observada no estudo 2.....	166
Quadro 24: Publicações em Periódicos.....	174
Quadro 25: Publicações em Conferências	174

LISTA DE TABELAS

Tabela 1: Modelos e Normas desenvolvidos pelos especialistas	132
Tabela 2: Classificação dos especialistas.....	133
Tabela 3: Tabulação das respostas da avaliação geral do método.....	135

LISTA DE ABREVIATURAS E SIGLAS

- GQM – *Goal Question Metric*
- IEEE – *Institute of Electrical and Electronics Engineers*
- IEC – *International Electrotechnical Commission*
- ISO – *International Organization for Standardization*
- KE – *Knowledge Engineering*
- KBE – *Knowledge-Based Engineering*
- KBS – *Knowledge-Based System*
- PAM – *Process Assessment Model*
- PRM – *Process Reference Model*
- SaaS – *Software as a Service*
- SE – *Software Engineering*
- SOA – *Service-Oriented Architecture*
- SPCMM – *Software Process Capability/Maturity Model*
- SPEM – *Software Process Engineering Metamodel*
- SPI – *Software Process Improvement*
- SPICE – *Software Process Improvement and Capability dEtermination*
- UML – *Unified Modeling Language*
- XP – *Extreme Programming*

SUMÁRIO

1	INTRODUÇÃO	27
1.1	<i>Apresentação do Problema de Pesquisa</i>	27
1.1.1	Objetivo Geral	32
1.1.2	Objetivos Específicos	32
1.2	<i>Justificativa e Relevância do Tema</i>	32
1.3	<i>Ineditismo do Trabalho</i>	33
1.4	<i>Contribuição Teórica</i>	34
1.5	<i>Escopo do Trabalho</i>	34
1.6	<i>Aspectos Metodológicos</i>	35
1.7	<i>Aderência a Engenharia do Conhecimento e a Multidisciplinaridade do Trabalho</i>	39
1.8	<i>Estrutura da Tese</i>	41
2	FUNDAMENTAÇÃO TEÓRICA	43
2.1	<i>Engenharia do Conhecimento</i>	43
2.2	<i>Aquisição de Conhecimento</i>	48
2.3	<i>Software Process Capability/Maturity Models (SPCMMs)</i>	56
2.4	<i>Discussão</i>	69
3	ESTADO DA ARTE	71
3.1	<i>Quais SPCMMs estão sendo desenvolvidos atualmente</i>	71
3.2	<i>Como os SPCMMs têm sido desenvolvidos</i>	83
3.3	<i>Quais são os métodos para desenvolvimento/customização de SPCMMs</i>	90
3.4	<i>Discussão</i>	96
4	MÉTODO DE AQUISIÇÃO DE CONHECIMENTO PARA CUSTOMIZAÇÃO DE SPCMMs	98
4.1	<i>Desenvolvendo um método para engenharia do conhecimento</i>	98
4.2	<i>Estrutura do método</i>	100
4.3	<i>Detalhamento do método</i>	105
4.3.1	Detalhamento de uma atividade	112

4.3.2	Detalhamento de uma técnica	114
4.3.3	Detalhamento de um papel	116
4.4	<i>Comentários do Capítulo</i>	116
5	VALIDAÇÃO DO MÉTODO	118
5.1	<i>Definição da Validação</i>	118
5.2	<i>Avaliação pelo Expert Panel</i>	122
5.2.1	Seleção dos especialistas	123
5.2.2	Design do Questionário	125
5.2.3	Análise da validade do instrumento	127
5.2.4	Coleta de dados	130
5.2.5	Análise da Confiabilidade do instrumento	134
5.2.6	Resultados do <i>Expert Panel</i>	136
5.3	<i>Estudos de Caso</i>	147
5.3.1	Planejamento	148
5.3.2	Execução, Análise e Interpretação	150
5.4	<i>Ameaças à validade</i>	167
5.4.1	Ameaças à validade do Expert Panel	167
5.4.2	Ameaças à validade do Estudo Empírico	168
5.5	<i>Conclusões do Capítulo</i>	169
6	CONCLUSÕES E TRABALHOS FUTUROS	170
6.1	<i>Principais Contribuições e Resultados</i>	173
6.2	<i>Trabalhos Futuros</i>	175
	REFERÊNCIAS	177
	APÊNDICE A – Software Process Capability/Maturity Models	195
	APÊNDICE B – Dimensão de Maturidade/Capacidade de SPCMMs	205
	APÊNDICE C – Questionário Survey	213
	APÊNDICE D – Questionário Expert Panel	223
	ANEXO A – Certificado Comitê de Ética	229

1 INTRODUÇÃO

1.1 Apresentação do Problema de Pesquisa

Novos desafios para o desenvolvimento de sistemas computacionais vêm surgindo especialmente a partir da emergência da chamada era do conhecimento onde o capital intelectual passa a ser componente essencial de agregação de valor para as organizações (DAVID, 2003). Os Sistemas Baseados em Conhecimento – *Knowledge-Based Systems* (KBS), em decorrência, têm evoluído juntamente com a conseqüente demanda crescente por suporte tecnológico à gestão do conhecimento nas organizações, refletindo em uma necessidade por abordagens e metodologias que apoiassem o desenvolvimento deste tipo de sistemas.

No sentido de atender a esta necessidade, a Engenharia do Conhecimento - *Knowledge Engineering* (KE) tem o objetivo de transformar o processo de desenvolvimento de KBSs de arte em uma disciplina de engenharia (SCHREIBER et al., 2000), por meio da utilização de processos de construção e manutenção, métodos, linguagens e ferramentas para o desenvolvimento de KBSs (STUDER, BENJAMINS & FENSEL, 1998). A KE provê os métodos para a obtenção de um entendimento completo das estruturas e processos utilizados por especialistas, no intuito de criar uma melhor integração da tecnologia da informação em suporte ao trabalho intelectual (SCHREIBER et al., 2000).

Inicialmente a KE foi entendida como uma forma de transferir o conhecimento presente na mente de um especialista para um ambiente computacional (HAYES-ROTH et al., 1983), onde o papel do engenheiro do conhecimento consistia em capturar o conhecimento do especialista e armazená-lo em uma base de conhecimento. Esta visão, no entanto restringia as possibilidades da KE representar corretamente os diversos tipos de conhecimento (STUDER, BENJAMINS & FENSEL, 1998). Em decorrência disso, a KE passou a ver o processo de desenvolvimento de KBSs como um processo de modelagem do conhecimento (STUDER, BENJAMINS & FENSEL, 1998). Neste novo paradigma o contexto do conhecimento tem importante papel na criação

do modelo que possa representar o conhecimento do especialista em uma determinada área de domínio (SCHREIBER et al., 2000).

Preston et al. (2005) propõem um ciclo de vida genérico para o desenvolvimento de aplicações de KE que estabelece as fases de: identificação do conhecimento, justificação da necessidade de estabelecimento do KBS, captura do conhecimento, formalização, empacotamento do conhecimento e ativação.

Existem atualmente diversas metodologias, *frameworks* e abordagens que implementam, ao menos em parte, um ciclo de vida de KBSs, apoiando a KE no desenvolvimento deste tipo de sistemas nas organizações. Dentre estas, as que melhor se alinham à proposta de uma KE como processo de modelagem do conhecimento são o CommonKADS (SCHREIBER et al., 2000), o Protégé (GENNARI et al., 2003) e o MIKE (STUDER, BENJAMINS & FENSEL, 1998). Isso se deve ao fato de que estas procuram englobar as características ambientais onde o KBS está sendo desenvolvido, indo além dos aspectos unicamente relacionados ao sistema de software em si.

O objeto da KE é o conhecimento. Entretanto, ainda persiste a ausência de uma definição universal para este termo (SCHREIBER et al., 2000). Sob o ponto de vista da KE, conhecimento pode ser definido como informações colocadas em prática na realização de tarefas (SCHREIBER et al., 2000). A KE interessa-se especialmente pelo conhecimento explícito, que pode ser com mais facilidade expresso em palavras e números, comunicado e compartilhado na forma de fórmulas científicas, procedimentos codificados ou princípios (NONAKA & TAKEUCHI, 1995).

Na perspectiva da KE, o processo de criação do conhecimento pode ocorrer por meio (i) da organização de conhecimento anterior em novas formas, (ii) da combinação de informações relevantes, ou mesmo (iii) de *insights* acerca da aplicação de conhecimento existente em novos contextos (CALHOUN & STARBUCK, 2005). O conhecimento existente pode ser extraído a partir de suas fontes, codificado e reutilizado, por meio do processo de aquisição de conhecimento (HUA, 2008; SCHREIBER et al., 2000; MOTODA et al., 1991; GROVER, 1983).

A aquisição de conhecimento abrange desde a identificação, coleta e análise até a modelagem e validação do conhecimento (HUA, 2008), envolvendo neste processo diversas técnicas, tais como: técnicas de geração e análise de protocolo, técnicas baseadas em matriz e

técnicas de ordenação (HUA, 2008). Assim, a aquisição de conhecimento permeia boa parte do ciclo de KE, fragmentada entre as suas diversas etapas e atividades (HUA, 2008; ANGELE, FENSEL & STUDER, 1998) ou níveis (SCHREIBER et al., 2000) dependendo da abordagem utilizada.

O resultado produzido pelo processo de aquisição de conhecimento pode constituir-se em diferentes produtos: desde simples documentos, passando por ontologias, e até complexas redes de conhecimento (MILTON, 2007). Assim, o conhecimento extraído pode ser representado das mais diversas formas. A representação do conhecimento é fundamentalmente um substituto, um meio de expressão humana utilizado para permitir que uma entidade raciocine sobre o mundo em vez de agir nele (DAVIS, HOWARD & SZOLOVITS, 1993). No entanto, independentemente do formato em que se apresente, esse produto final da aquisição de conhecimento precisa ser: útil aos seus usuários, correto, completo e relevante, e sem consumir excessivos recursos de tempo e esforço dos especialistas (MILTON, 2007).

Percebe-se, então, o grande desafio da aquisição de conhecimento, que é agravado pelo fato de que o elemento humano é normalmente chave no processo (SCHREIBER et al., 2000) uma vez que o conhecimento é, em grande parte, capturado de especialistas. O raciocínio, a memória e a representação do conhecimento nas mentes dos especialistas, estão sujeitos a diferentes falhas possíveis, que apesar de muitas vezes sutis, precisam ser levadas em conta durante o processo de aquisição de conhecimento (SCHREIBER et al., 2000). O contexto no qual o conhecimento é capturado também influencia fortemente (SCHREIBER et al., 2000), contexto esse que pode não estar acessível a quem posteriormente receber este conhecimento codificado de alguma forma.

Assim, apesar da sua importância, a aquisição de conhecimento é um dos principais gargalos da KE (HUA, 2008; SCHREIBER et al., 2000). A utilização de técnicas para automatizar a descoberta de conhecimento, por exemplo, têm resolvido apenas parte dos problemas, pois para o conhecimento representado em relações e estruturas mais complexas, não existem métodos eficazes (HUA, 2008; MOTODA et al., 1991).

De forma geral, a aquisição de conhecimento a partir de fontes computacionalmente estruturadas, como ontologias ou bases de dados (FRANK et al. 2005), pode ser realizada de maneira até automatizada, enquanto a aquisição de conhecimento a partir de fontes não

estruturadas, como quaisquer documentos em texto livre nos seus mais diversos formatos, ainda apresenta elevado grau de dificuldade, exigindo robusta análise semântica (FRANK et al. 2005). Além disso, também fontes semi-estruturadas são importantes, como documentos anotados (com anotações XML – *eXtended Markup Language*¹, por exemplo), onde o processamento computacional é também facilitado (FRANK et al. 2005).

Na aquisição automática ou semi-automática de conhecimento a partir de fontes não estruturadas, utilizando técnicas como: *text mining*, *knowledge discovery in texts*, etc. (FATUDIMU et al., 2008), em geral, não se obtém um conhecimento correto e suficiente, e mesmo quando a aquisição do conhecimento ocorre, a sua confiabilidade ainda necessita ser verificada por especialistas (HUA, 2008). Ou seja, o envolvimento do engenheiro do conhecimento, é essencial no processo de aquisição do conhecimento (SCHREIBER et al., 2000; MOTODA et al., 1991).

Em domínios onde ainda não há conhecimento codificado e disponível, a aquisição de conhecimento torna-se um processo ainda mais complexo (HUA, 2008; SCHREIBER et al., 2000). Assim, o conhecimento a ser adquirido em um determinado domínio, identificado como a maneira de resolver problemas naquele domínio, precisa ser extraído das mais diversas fontes, tais como descrições, heurísticas ou procedimentos, mesmo que tácitos (ABEL, 2001), podendo então ser codificado de várias maneiras para diferentes usos (IEEE, 2004). Nestes domínios menos explorados pela KE, o conhecimento tem sido normalmente extraído e codificado na forma de melhores práticas (GRAUPNER et al., 2009), coletadas a partir da experiência acumulada de uma comunidade de especialistas e agrupada e organizada por alguma entidade ou comitê técnico.

Neste contexto, melhores práticas são o encapsulamento de experiências que, quando repetidas, levam a alcançar resultados semelhantes (GOODMAN & GOLDMAN, 2007). *Frameworks* de melhores práticas permitem a reutilização de experiências dentro de um domínio, fornecendo orientações, freqüentemente, em um nível alto de abstração, e não contendo os detalhes das tarefas e processos necessários para se realizar o trabalho (GRAUPNER et al., 2009).

Estes *frameworks* de melhores práticas são gerais e normalmente abrangem amplos domínios (GRAUPNER et al., 2009) precisando ser customizados, ou seja, alterados ou modificados conforme

¹ www.w3.org/XML

especificações individuais (MICHAELIS, 2010), quando aplicados em determinados domínios que apresentam características e necessidades específicas que precisam ser devidamente atendidas (BEECHAM, HALL & RAINER, 2005).

Quando utilizados como base comparativa para avaliação e/ou melhoria de processos de software, estes *frameworks* de melhores práticas têm sido denominados Modelos de Capacidade/Maturidade de Processos de Software (*Software Process Capability/Maturity Model – SPCMM*) (SALVIANO & FIGUEREDO, 2008). Exemplos deste tipo de modelo são: o modelo CMMI-DEV (SEI, 2010) e o modelo ISO/IEC 15504-5 *Process Assessment Model* (ISO/IEC, 2006).

Analisando a forma como os SPCMMs têm sido desenvolvidos observa-se que, com exceção daqueles que são desenvolvidos como normas (baseadas no processo definido por um órgão normativo) a maioria dos modelos tem sido desenvolvida de forma *ad hoc* (WANGENHEIM et al., 2010a).

Sob a perspectiva da KE, o desenvolvimento de SPCMMs, consiste em um problema de aquisição de conhecimento, sendo que as melhores práticas contidas neste tipo de modelo representam conhecimento na área (WANGENHEIM et al., 2010a; MATOOK & INDULSKA, 2009; GRAUPNER et al., 2009). Quando da customização destes modelos para um domínio específico, surge a necessidade de aquisição de conhecimento a partir dos especialistas do domínio, dos SPCMMs genéricos e das demais fontes de conhecimento do domínio.

Apesar da existência de diversas técnicas e métodos de aquisição de conhecimento, majoritariamente baseadas em entrevistas e análise de textos em linguagem natural (MOTODA et al., 1991), tais como (HUA, 2008; SCHREIBER et al., 2000): Geração de protocolo, Entrevistas, *Commenting*, Análise de protocolo, *Card Sorting*, *Laddering*, *Concept mapping*, *Repertory Grid*, etc., nenhuma leva em conta ou objetiva apoiar a customização de SPCMMs.

Portanto, nota-se a ausência de um método que forneça um suporte sistemático para a aquisição de conhecimento na customização de SPCMMs para domínios específicos de desenvolvimento de software. Neste contexto, a pergunta que guia a realização desta pesquisa consiste em:

Pergunta de Pesquisa: *Como sistematicamente realizar a aquisição de conhecimento para customização de modelos de*

capacidade/maturidade de processo de software para domínios específicos de desenvolvimento de software?

1.1.1 Objetivo Geral

Elaborar um método de aquisição de conhecimento para customização de modelos de capacidade/maturidade de processo de software para domínios específicos de desenvolvimento de software.

1.1.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Objetivo I: Identificar quais e como são os modelos customizados de capacidade/maturidade de processo de software;
- Objetivo II: Analisar o processo de desenvolvimento de modelos customizados de capacidade/maturidade de processos de software;
- Objetivo III: Desenvolver um método de aquisição de conhecimento para a customização de modelos de capacidade/maturidade de processos de software para domínios específicos de desenvolvimento de software;
- Objetivo IV: Validar o método de aquisição de conhecimento no desenvolvimento de modelos customizados de capacidade/maturidade de processos de software para domínios específicos de desenvolvimento de software.

1.2 Justificativa e Relevância do Tema

O processo de aquisição do conhecimento é um dos gargalos da KE (HUA, 2008). Isso ocorre porque há grande dependência do fator humano na obtenção do conhecimento que muitas vezes somente se apresenta de forma tácita. Neste sentido, é de grande relevância o esforço empregado em tornar a elicitación do conhecimento um processo eficaz (SCHREIBER et al., 2000).

A utilização de métodos e técnicas que possibilitem minimizar o esforço despendido em recolher, transcrever e analisar o conhecimento de um especialista tende a reduzir tempo e custo no desenvolvimento de KBSs. Portanto, existe uma necessidade real de se reduzir ao mínimo o tempo gasto com caros e escassos especialistas de domínio. Evidentemente, “é importante maximizar o rendimento de conhecimentos utilizáveis” (SCHREIBER et al., 2000, p. 189).

É mais simples para um engenheiro do conhecimento recolher informações a partir de recursos não humanos, tais como: livros, manuais técnicos, estudos de caso, e assim por diante (SCHREIBER et al. 2000, p. 188), do que extraí-las diretamente dos especialistas de domínio. Assim, o conhecimento explicitamente disponível na forma de melhores práticas de processo de software (em SPCMMs) de domínios específicos tende a simplificar a etapa de aquisição de conhecimento na KE.

Neste sentido, espera-se que a existência de um método de aquisição de conhecimento a partir de SPCMMs para domínios específicos venha a colaborar no estabelecimento SPCMMs que representem o conhecimento, na forma de melhores práticas, realmente válido no domínio para o qual são concebidos. Em consequência, espera-se também que o suporte sistemático ao desenvolvimento de SPCMMs contribua na qualidade de processos e projetos de software.

1.3 Ineditismo do Trabalho

A aquisição de conhecimento, enquanto extração e codificação do conhecimento para reutilização (HUA, 2008) foi inicialmente concebida como entrevistas com especialistas de domínio (GROVER, 1983) na tentativa de extrair regras para solução de problemas específicos (MOTODA et al., 1991). Com a sua evolução, tem sido realizada a partir de fontes de conhecimento estruturadas e não estruturadas (FRANK et al. 2005), tanto automaticamente utilizando *text mining*, *knowledge discovery in texts*, etc. (FATUDIMU et al., 2008), quanto utilizando técnicas de geração e análise de protocolo, baseadas em matriz e técnicas de ordenação (HUA, 2008). O elemento humano é chave no processo de aquisição do conhecimento, pois, apesar da automação possível, a maior parte do conhecimento a ser extraído encontra-se na cabeça dos especialistas (SCHREIBER et al., 2000; MOTODA et al., 1991).

A partir da revisão da literatura apresentada nesta tese, é possível observar que atualmente não há métodos disponíveis específicos para aquisição de conhecimento para customização de SPCMMs, que descrevam detalhadamente o processo, técnicas e artefatos necessários (WANGENHEIM et al., 2010a; WANGENHEIM et al., 2010b; HAUCK et al., 2010; HUA, 2008; PRESTON et al., 2005; BOOSE, 1989).

1.4 Contribuição Teórica

A presente tese contribui para a Engenharia do Conhecimento na pesquisa atual em aquisição do conhecimento a partir de fontes não estruturadas, propondo um novo método sistemático de aquisição de conhecimento na forma de melhores práticas.

Outra contribuição relevante para a Engenharia do Conhecimento se dá na codificação e reutilização do conhecimento na forma de melhores práticas de qualidade de processo, customizadas para um domínio específico.

A representação de conhecimento em SPCMMs na forma de melhores práticas é também uma contribuição para a KE no sentido prover conhecimento codificado e reutilizável para os engenheiros de conhecimento. Neste sentido, o conhecimento sobre os SPCMMs existentes e de como eles têm sido desenvolvidos, incluindo quais métodos e técnicas têm sido utilizadas para o seu desenvolvimento, também se constitui em uma contribuição.

Por fim, esta pesquisa contribui para a Engenharia de Software com um método que fornece suporte sistemático para o desenvolvimento de SPCMMs customizados para domínios específicos de processo de software.

1.5 Escopo do Trabalho

Este trabalho restringe-se à aquisição de conhecimento a partir de fontes não estruturadas, documentais e não documentais, tendo foco na representação do conhecimento na forma de melhores práticas. Outros tipos de conhecimento e/ou áreas estão fora do escopo deste trabalho.

A presente tese foca exclusivamente na aquisição de conhecimento, não considerando outros passos ou etapas do desenvolvimento de KBSs. Também não é abrangida a extração automática ou semi-automática de conhecimento (*text mining*, ou *knowledge discovery in texts*), pois esta pesquisa pressupõe a participação humana no processo de customização de SPCMMs para domínios específicos de processo de software.

A aplicação é realizada no contexto da customização de modelos de capacidade/maturidade de processo de software (SPCMMs), para domínios específicos. Outras áreas ou formas de aplicação de aquisição de conhecimento ou de representação de conhecimento estão fora do escopo deste trabalho.

1.6 Aspectos Metodológicos

Em relação ao paradigma de pesquisa adotado neste trabalho, Orlikowski & Baroudi (1991) observam a existência de três diferentes paradigmas de pesquisa: positivista, interpretativa e crítica. Guba & Lincoln (1994), no entanto, apresentam quatro paradigmas de pesquisa: positivismo, pós-positivismo, teoria crítica e construtivismo.

Myers (1997) sugere que uma pesquisa fundamentalmente qualitativa pode ser: positivista, interpretativa ou crítica. Neste sentido, Burton (2008), analisando estas possibilidades de filosofia de pesquisa, argumenta que quando da aplicação da pesquisa no desenvolvimento e validação de um SPCMM para um domínio específico e com seus resultados interpretados em relação a este contexto, a pesquisa interpretativa é a mais adequada.

Sob o ponto de vista do contexto científico, Saunders, Lewis & Thornhill (2009) propõem uma estrutura do processo da ciência, onde o método científico é representado em camadas (*research-process onion*). Com base neste sistema, a Figura 1 apresenta o contexto da metodologia da presente pesquisa no universo do método científico.

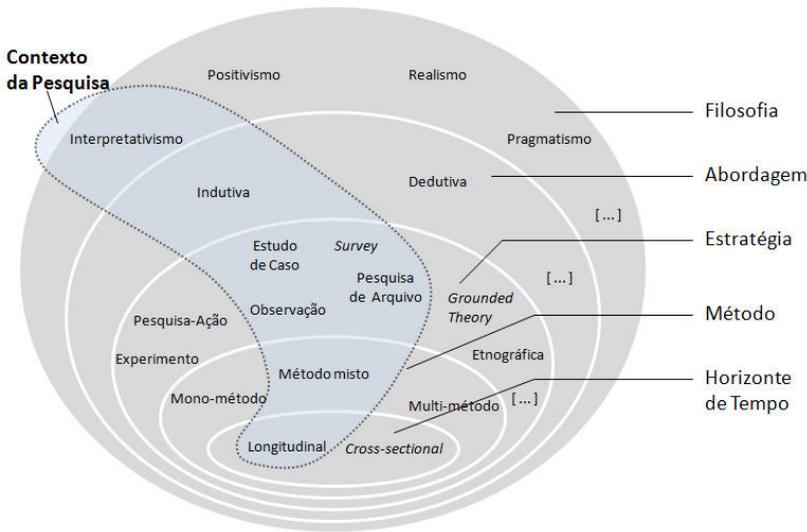


Figura 1: Contexto da pesquisa no método científico

Fonte: Baseado em (SAUNDERS, LEWIS & THORNHILL, 2009)

Na proposta de Saunders, Lewis & Thornhill (2009), a metodologia científica é envolvida pela escolha da filosofia da pesquisa, que norteia todo o trabalho científico. Neste aspecto, a presente pesquisa é predominantemente interpretativista, pois assume que o seu objeto de pesquisa (aquisição de conhecimento) é social e contextualmente construído e interpretado. Sendo assim, é difícil estabelecer uma independência do pesquisador em relação ao objeto de pesquisa. A camada inferior (vide Figura 1) consiste na abordagem científica, onde a perspectiva deste trabalho se alinha à abordagem indutiva, pois não parte de uma hipótese pré-estabelecida, mas, ao contrário, procura atingir a solução do problema com base nas conclusões extraídas do objeto pesquisado. A estratégia desta pesquisa envolve a utilização de: pesquisa de arquivo, *survey*, estudo de caso e observação, sem descartar outros. O método de pesquisa é majoritariamente qualitativo, entretanto envolve também a abordagem quantitativa em diferentes fases da pesquisa, por isso pode ser qualificado como misto (SAUNDERS, LEWIS & THORNHILL, 2009). Por fim, o horizonte de tempo da pesquisa é predominantemente longitudinal, pois a observação do método aplicado nos estudos de caso trata-se de um fenômeno distribuído no tempo. Entretanto o *survey* e a validação por meio do *Expert Panel* são *cross-sectional*, pois a coleta de dados se dá em eventos únicos no tempo.

Em relação ao seu objetivo geral este trabalho constitui-se, predominantemente, em uma pesquisa básica exploratória mista, dado o seu objetivo de desenvolver um método de aquisição de conhecimento. Segundo a classificação clássica da pesquisa, o Quadro 1 apresenta os demais critérios, classificações e justificativas:

Quadro 1: Classificação da Pesquisa

Critério	Classificação	Justificativa
Natureza	Básica	No que trata do Objetivo Geral e do Objetivo III - Desenvolver um método de aquisição de conhecimento.
	Aplicada	No que trata do Objetivo IV - Validar o método de aquisição de conhecimento.
Objetivos	Exploratória	Pela utilização de técnicas de revisão da literatura, <i>survey</i> e na aplicação do método.
Abordagem	Mista	Na aplicação de método e técnicas quantitativas e qualitativas, especialmente no atendimento dos objetivos: II - Analisar o processo de desenvolvimento de modelos customizados; e IV - Validar o método.
Procedimentos	Bibliográfica, Documental, Estudo de Caso.	Em atendimento aos objetivos: I – identificar os SPCMMs; II - Analisar o processo de desenvolvimento de modelos customizados; e IV - Validar o método.

Este trabalho propõe um novo método para a Engenharia do Conhecimento e não de uma nova metodologia, porque não cria uma nova visão de mundo, mas se aproveita das visões propostas pela Engenharia do Conhecimento (SCHREIBER et al., 2000) e pela Engenharia de Software nos SPCMMs (SALVIANO & FIGUEIREDO, 2008), estabelecendo uma estrutura sistemática de fases, atividades e ferramentas para se atingir um objetivo.

Assim, as etapas desta pesquisa para o desenvolvimento de um método de aquisição de conhecimento, seguindo uma abordagem indutiva, são apresentadas na Figura 2. O desenvolvimento do método inicia-se com uma fundamentação teórica, abrangendo os conceitos fundamentais referentes à KE e SE. A fim de obter o estado da arte no que diz respeito à forma como SPCMMs de domínio específico são desenvolvidos, é realizada uma revisão sistemática da literatura seguindo o procedimento metodológico definido por Kitchenham (2007). Esta revisão é complementada por um *survey* entre os autores dos modelos identificados na revisão sistemática seguindo o

procedimento proposto por KASUNIC (2005), com o objetivo de obter informações detalhadas sobre como esses modelos foram desenvolvidos.

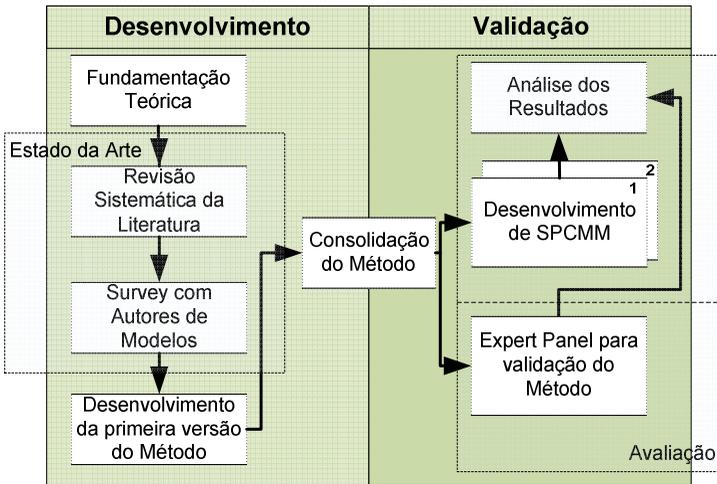


Figura 2: Etapas da Pesquisa

Com base nestes resultados é desenvolvida a primeira versão do método para aquisição de conhecimento a partir de SPCMMs para sua customização em domínios específicos de desenvolvimento de software. Este método é então consolidado pela adição de descrições detalhadas, técnicas e ferramentas. A versão *draft* já é então aplicada no desenvolvimento de SPCMMs com o intuito de permitir uma primeira observação informal de seu uso na prática.

A validação do método é realizada em duas fases. A primeira fase consiste em uma avaliação do conteúdo do método realizada utilizando-se a técnica de *Expert Panel* (BEECHAM et al., 2005). Na segunda fase, com base no procedimento para estudos empíricos definido por Wohlin et al. (2000) são realizados dois Estudos de Caso para avaliar o método desenvolvido, incluindo a definição do estudo, o planejamento, a execução e a análise dos dados coletados. Como resultado obtém-se o método consolidado e avaliado para ser aplicado na customização de SPCMMs para domínios específicos.

1.7 Aderência a Engenharia do Conhecimento e a Multidisciplinaridade do Trabalho

O objetivo do Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento (PPGEGC) consiste formação pra atuação “... na pesquisa e no desenvolvimento em codificação, gestão e disseminação dos conhecimentos (explícitos e tácitos) em organizações, públicas e privadas, e na sociedade” (EGC, 2010). Mais especificamente a área de concentração de Engenharia do Conhecimento objetiva:

“ ... a pesquisa e o desenvolvimento de técnicas e ferramentas para a formalização, codificação e gestão do conhecimento; de métodos de análise da estrutura e processos conduzidos por profissionais em atividades de conhecimento intensivo; e a pesquisa e desenvolvimento de sistemas de conhecimento” (EGC, 2010).

Neste sentido, o método definido neste trabalho alinha-se ao objetivo da área de concentração de Engenharia do Conhecimento como um método de análise da estrutura e processos conduzidos por profissionais da área de Engenharia de Software. Assim, com base nas fases típicas de KE, propostas em Preston et al. (2005), Schreiber et al. (2000) e Angele, Fensel & Studer (1998), o método contribui para a KE provendo um suporte sistemático à fase de captura e parcialmente à fase de formalização do conhecimento aplicado no desenvolvimento de SPCMMs, conforme apresentado na Figura 3.

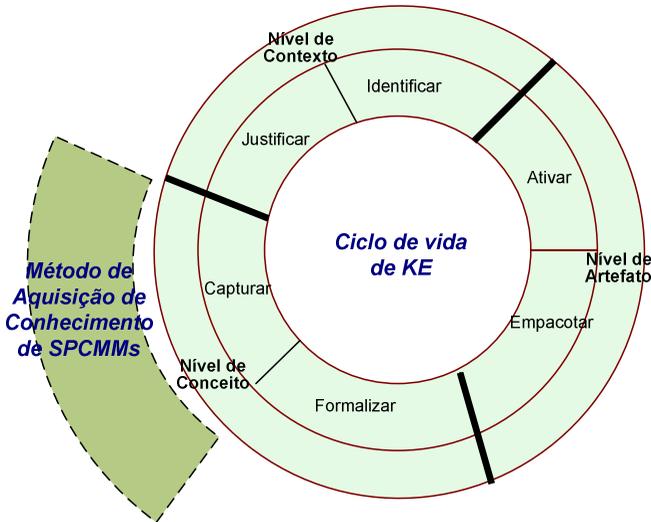


Figura 3: Alinhamento à KE do método de aquisição de conhecimento de SPCMMs

Fontes: (PRESTON et al., 2005; SCHREIBER et al., 2000; ANGELE, FENSEL & STUDER, 1998).

Desta forma, o método proposto procura sistematizar o desenvolvimento de SPCMMs para domínios específicos de processo de software, de forma a apoiar o engenheiro de conhecimento na aquisição de conhecimento necessário no processo de customização deste tipo de modelo.

O carácter multidisciplinar deste trabalho se apresenta principalmente pela interseção entre a Gestão do Conhecimento e a Engenharia do Conhecimento. O objeto deste trabalho foca nos SPCMMs, que contém melhores práticas de qualidade de processos de software para as organizações, cujo tema é coberto pela Gestão do Conhecimento (GILLINGHAM & ROBERTS, 2006). Neste sentido, no método proposto neste trabalho, a Engenharia do Conhecimento contribui, provendo as técnicas e processos de aquisição de conhecimento como apoio à extração e representação de conhecimento dos SPCMMs, na forma de melhores práticas, na sua customização para domínios específicos de processos de software.

A Engenharia de Software, como atividade intensiva em conhecimento, onde o principal ativo consiste no conhecimento dos especialistas (BJØRNSON & DINGSØY, 2008), e, mais

especificamente, a área de Melhoria de Processos de Software (SPI), servem de domínio de aplicação do método proposto e contribuem para a base conceitual de SPCMMs.

1.8 Estrutura da Tese

Além deste capítulo introdutório, esta tese está estruturada da seguinte forma: o capítulo 2 trata da fundamentação teórica e o capítulo 3 do estado da arte, envolvendo a revisão sistemática da literatura e o *survey* com autores do modelo; o capítulo 4 apresenta o desenvolvimento da primeira versão do método; o capítulo 5 apresenta a validação do conteúdo e do uso do método, e; no capítulo 6 as conclusões desta tese são apresentadas. A Figura 4 apresenta a estrutura desta tese em relação às etapas da pesquisa.

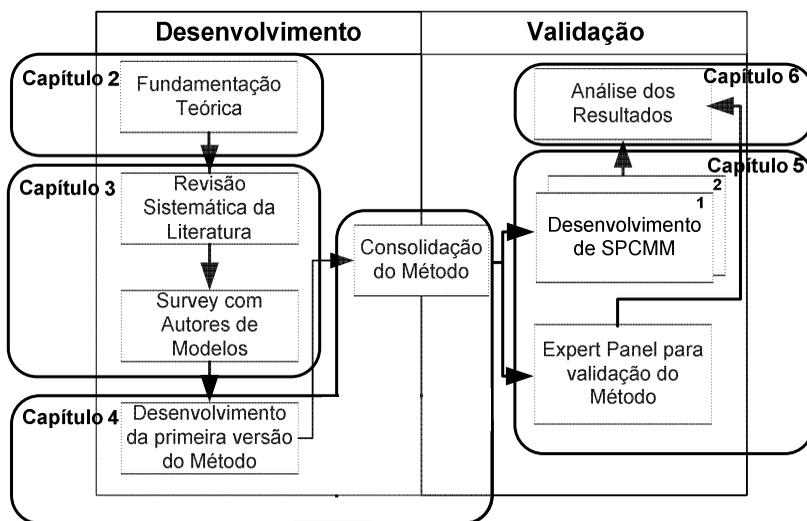


Figura 4: Estrutura da Tese

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os fundamentos teóricos relevantes para o desenvolvimento do método proposto nesta tese. Inicialmente são apresentados os conceitos referentes à Engenharia do Conhecimento (*Knowledge Engineering* - KE) e mais especificamente à aquisição de conhecimento e, na sequência, aqueles referentes à área de aplicação na Engenharia de Software (*Software Engineering* - SE), diretamente associados à definição de SPCMMs.

2.1 Engenharia do Conhecimento

Engenharia do Conhecimento (KE) é uma disciplina científica que procura aplicar métodos científicos para a construção de Sistemas Baseados em Conhecimento (*Knowledge-Based Systems* - KBS) (SCHREIBER et al., 2000). KBS são sistemas desenvolvidos para auxiliar na resolução de problemas normalmente solucionados humanos, proporcionando rapidez e qualidade na tomada de decisões e aumento na produtividade (SCHREIBER et al., 2000).

O objeto da KE é o conhecimento. Apesar da inexistência de uma definição única, no contexto da KE, conhecimento pode ser definido como todo o conjunto de dados e informações colocados em prática na realização de tarefas (SCHREIBER et al., 2000). Conhecimento também supõe a capacidade de intencionalmente criar novas informações, sendo assim considerado como um fator de produção (SCHREIBER et al., 2000).

Sob a ótica de fator de produção, o conhecimento pode ser tácito ou explícito. O conhecimento tácito é aquele conhecimento pessoal, difícil de ser externalizado e normalmente aprendido pela experiência; enquanto o conhecimento explícito é aquele que pode ser com mais facilidade expresso em palavras e números, comunicado e compartilhado na forma de fórmulas científicas, procedimentos codificados ou princípios (NONAKA & TAKEUCHI, 1995). A KE interessa-se especialmente pelo conhecimento explícito e, quando possível, busca transformar o conhecimento tácito em conhecimento explícito para que possa ser adequadamente manipulado.

Outra dimensão de classificação do conhecimento é em relação ao que ele aborda: conhecimento procedural ou conhecimento conceitual (MILTON, 2007). O conhecimento procedural está relacionado ao

como, aos processos, atividades e tarefas. Sempre que se diz “Eu sei como fazer ...” está se referindo à forma como as coisas são feitas, este conhecimento é dito, então, procedural. Já o conhecimento conceitual trata dos conceitos, suas propriedades e suas relações com outros conceitos. Neste sentido, sempre que se diz: “Eu sei que ...” se está normalmente falando de conhecimento conceitual. Também na classificação de determinado conceito como pertencente a uma classe ou como membro de determinada hierarquia, como em taxonomias, está se referindo ao conhecimento conceitual (MILTON, 2007).

Numa visão hierárquica – *DIKW* – *Data, Information, Knowledge, Wisdom* (ROWLEY, 2007) o conhecimento é considerado no conjunto de uma relação que inicia com dados (*Data*), que são os símbolos ou sinais, informação (*Information*) que são descrições úteis obtidas a partir dos dados, conhecimento (*Knowledge*) que pode ser definido como o elemento que torna possível a transformação de informações em instruções (SANTOS & SOUZA, 2010) e sabedoria (*Wisdom*) que implica no conhecimento acerca do conhecimento.

Diversas são as críticas a esta estrutura hierárquica atualmente, como apontado em SANTOS & SOUZA (2010): o foco restrito da visão operacionalista que reduz o conhecimento possível às deduções ou inferências lógicas e a confiança da objetividade dos dados, em despeito da natureza interpretativa inerente às formas de representação convencionadas para os mesmos dados.

Neste contexto, Melhores Práticas (*Best-practices*) são o encapsulamento de experiências que, quando repetidas, levam a alcançar resultados semelhantes (GOODMAN & GOLDMAN, 2007). Desta forma, as melhores práticas representam conhecimento em um domínio, armazenando experiências que podem ser reutilizadas. Entretanto, as melhores práticas de um domínio apesar de geralmente bem difundidas, muitas vezes não são verificadas empiricamente (LIMING, STAPLES & GORTON, 2007).

Na evolução da KE, diversas foram as metodologias e abordagens propostas, tais como: Protégé (GENNARI et al., 2003), CommonKADS (SCHREIBER et al., 2000), SPEDE (SHADBOLT, 1999), MIKE (ANGELE, FENSEL & STUDER, 1998), MOKA (OLDHAM et al., 1998) EXPECT (SWARTOUT & GIL, 1995), VITAL (DOMINGUE et al., 1993), e Componentes de Expertise (STEELS, 1990), apoiando a KE no desenvolvimento de KBSs nas organizações. De forma geral, pode-se afirmar que estas abordagens procuram implementar, de maneiras diferentes, um ciclo de vida de KE.

Um Ciclo de vida de KE pode ser entendido como o processo pelo qual um KBS é concebido e desenvolvido. O ciclo de vida proposto por Preston et al. (2005), conforme citado anteriormente, completa o ciclo de vida genérico previamente proposto por (SCHREIBER & WIELINGA, 1998), estabelecendo fases de: (i) identificação do conhecimento, (ii) justificação da necessidade de estabelecimento do KBS, (iii) captura do conhecimento, (iv) formalização, (v) empacotamento do conhecimento e (vi) ativação do conhecimento, conforme apresentado na Figura 5.



Figura 5: Ciclo de vida de KE

Fonte: Baseado em (SUREEPHONG et al., 2007; PRESTON et al., 2005).

São diversas as abordagens que implementam de forma ou de outra estas fases de desenvolvimento de um KBS, algumas focando mais no desenvolvimento do produto de software/KBS, outras incorporando também aspectos organizacionais relevantes.

Dentre estas diversas metodologias, *frameworks* e abordagens que abrangem, ao menos em parte, um ciclo de vida de desenvolvimento KBSs, segundo Studer, Benjamins & Fensel (1998) as que melhor se alinham à proposta de uma KE como processo de modelagem do conhecimento são o CommonKADS (SCHREIBER et al., 2000), o Protégé (GENNARI et al., 2003) e o MIKE (ANGELE, FENSEL & STUDER, 1998). Isso se deve ao fato de que estas possuem um foco

mais abrangente, que procura englobar as características ambientais onde o KBS está sendo desenvolvido além dos aspectos unicamente relacionados ao sistema de software em si, mais comum nas demais abordagens citadas.

O framework metodológico CommonKADS (SCHREIBER et al., 2000) dá suporte ao desenvolvimento de KBSs pelo estabelecimento de uma ampla visão do contexto onde o sistema será desenvolvido. Prescreve a análise em três níveis: de contexto, conceito e artefato, e para cada nível, busca estabelecer uma compreensão do problema a partir do desenvolvimento de modelos: organizacional, da tarefa, dos responsáveis, de conhecimento e de comunicação necessários e por fim um projeto de sistemas de conhecimento. A figura 6 apresenta uma visão geral da metodologia CommonKADS.

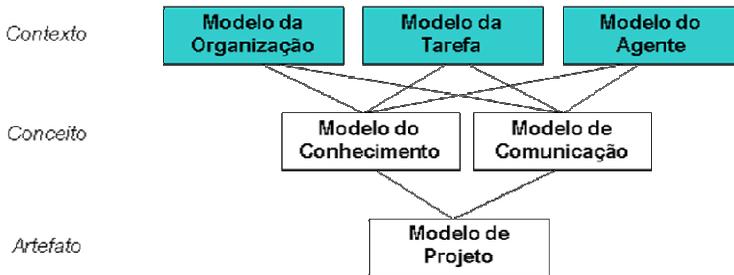


Figura 6: Visão Geral do modelo CommonKADS

Fonte: (SCHREIBER et al., 2000)

A abordagem MIKE (*Model-based and Incremental Knowledge Engineering*) (ANGELE, FENSEL & STUDER, 1998) fornece suporte ao desenvolvimento de aplicações KBS desde a sua concepção até implementação. Estabelece atividades de Elicitação, Interpretação, Formalização, Operacionalização, Design e Implementação (vide Figura 7). Diferentemente do CommonKADS, a abordagem MIKE já parte da abordagem ao especialista, buscando extrair dele o conhecimento, por meio de técnicas de aquisição de conhecimento.

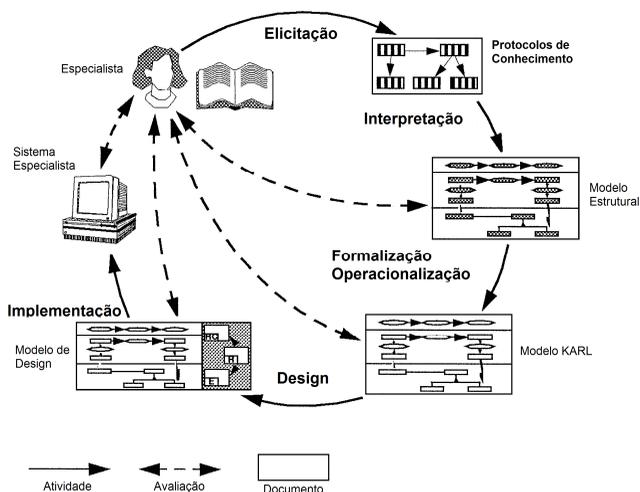


Figura 7: Visão Geral do modelo MIKE

Fontes: (STUDER, BENJAMINS & FENSEL, 1998; ANGELE, FENSEL & STUDER, 1998).

A abordagem Protégé (MUSEN, 1993) suporta o desenvolvimento de KBSs por meio da utilização de ferramentas de aquisição do conhecimento que permitem o reuso de ontologias oferecendo, inclusive, uma biblioteca de métodos de resolução de problemas que podem ser reutilizados. Os problemas a serem resolvidos por KBSs são decompostos em estruturas de tarefas e sub-tarefas até que sejam encontrados os mecanismos primitivos que possibilitem a solução destes problemas. A ontologia de domínio é utilizada para compartilhar a conceitualização como componente reutilizável para o desenvolvimento de KBSs.

Dentre os citados, o framework metodológico do CommonKADS possui maior abrangência na contextualização do conhecimento a ser extraído. Em (SUREEPHONG et al., 2007) os autores comparam três abordagens (MOKA, SPEDE e CommonKADS) utilizando como base o *Knowledge-Based Engineering (KBE) Application Lifecycle*. Esta avaliação demonstra que o CommonKADS atende a quase todas as etapas do ciclo de vida de um KBS (Quadro 2).

Quadro 2: Comparação entre frameworks de KE

Fonte: (SUREEPHONG et al., 2007)

<i>KBE Lifecycle</i>	<i>MOKA</i>	<i>SPEDE</i>	<i>CommonKADS</i>
1. Identificar	-	Entendendo o Projeto	Nível de Contexto
2. Justificar	-	Entendendo o Projeto	Modelo OTA
3. Capturar	Modelo Informal	Design do Processo	Nível de Conceito
4. Formalizar	Modelo Formal	Avaliação do novo Processo	Nível de Conceito
5. Empacotar	-	Comunicar o Processo	Nível de Artefato
6. Ativar	-	-	-

Desta forma, o framework metodológico do CommonKADS é escolhido como referência para este trabalho devido à sua abrangência na contextualização do conhecimento.

2.2 Aquisição de Conhecimento

Na perspectiva da KE, aquisição de conhecimento consiste em extrair o conhecimento necessário a partir de suas fontes (estruturadas ou não) de modo a poder codificá-lo e reutilizá-lo (HUA, 2008; SCHREIBER et al., 2000; MOTODA et al., 1991; GROVER, 1983). Ela é uma das práticas-chave e também um dos principais gargalos da KE (HUA, 2008; MOTODA et al., 1991).

Inicialmente a aquisição de conhecimento consistia basicamente na realização de entrevistas com especialistas de domínio (GROVER, 1983) no sentido de obter as regras necessárias para resolver determinado problema (MOTODA et al., 1991). As regras eram então codificadas e armazenadas em uma base de conhecimento. Com a evolução da Inteligência Artificial, técnicas automáticas de extração de conhecimento a partir de texto, apesar de muitas vezes limitadas, foram sendo introduzidas (MOTODA et al., 1991). Atualmente entende-se que a aquisição de conhecimento possui uma abrangência muito mais ampla (HUA, 2008).

Ao investigar as diversas técnicas de aquisição de conhecimento existentes, Hua (2008) propõe uma classificação em:

- Técnicas de geração de protocolo: o objetivo deste tipo de técnica consiste na produção de um protocolo, ou seja, um registro de comportamentos, em texto, áudio, vídeo ou outra mídia eletrônica. A gravação de áudio é o método usual, que é transcrita para produzir uma descrição. Há quatro principais técnicas: entrevistas, comentários, *teach back* e observação, mas também inclui entrevistas (não estruturadas, semi-estruturadas e estruturadas), técnicas de relato (como auto-relato e *shadowing*) e técnicas de observação.
- Técnicas de análise de protocolo: consiste em identificar os diferentes tipos de conhecimento, tais como metas, decisões, relacionamentos e atributos a partir de transcrições de entrevistas ou informações baseadas em texto.
- Técnicas baseadas em matriz: este tipo de técnica se baseia na construção de redes indicando problemas encontrados contra possíveis soluções. Inclui as técnicas de uso de *frames* para representar as propriedades dos conceitos e de *repertory grid*, para obter, avaliar, analisar e classificar as propriedades dos conceitos.
- Técnicas de ordenação: procuram identificar a forma como os especialistas comparam e ordenam conceitos, e pode levar à identificação de conhecimento sobre as classes, propriedades e prioridades. Inclui técnicas do tipo *card sorting*, onde o especialista recebe um número de cartões, sendo cada um exibindo o nome de um conceito. O especialista classifica repetidamente os cartões em pilhas de tal forma que os cartões em cada pilha tenham algo em comum.

Grover (1983) propôs inicialmente um ciclo de aquisição do conhecimento composto por três fases principais: Definição do Domínio, Corpo Fundamental de Conhecimento e Base de Conhecimento. Daí percebe-se que nas suas proposições iniciais a aquisição de conhecimento esteve fortemente restrita à obtenção do conhecimento do domínio. Nas abordagens mais atuais de KE, no entanto, este processo de aquisição de conhecimento está fragmentado

entre as diversas etapas, atividades (HUA, 2008; ANGELE, FENSEL & STUDER, 1998) ou níveis (SCHREIBER et al., 2000), dependendo da abordagem/metodologia. Isso implica que a aquisição do conhecimento permeia boa parte do processo de KE.

Entretanto, conforme mostra a Figura 8, num contexto mais estrito considerando um ciclo completo de KE, a aquisição do conhecimento abrange desde a identificação, coleta e análise até a modelagem e validação do conhecimento (HUA, 2008). Entende-se, no entanto, que a modelagem e formalização do conhecimento pode não ser completamente definida durante a aquisição do conhecimento, estando sujeita a aspectos técnicos e operacionais posteriores.

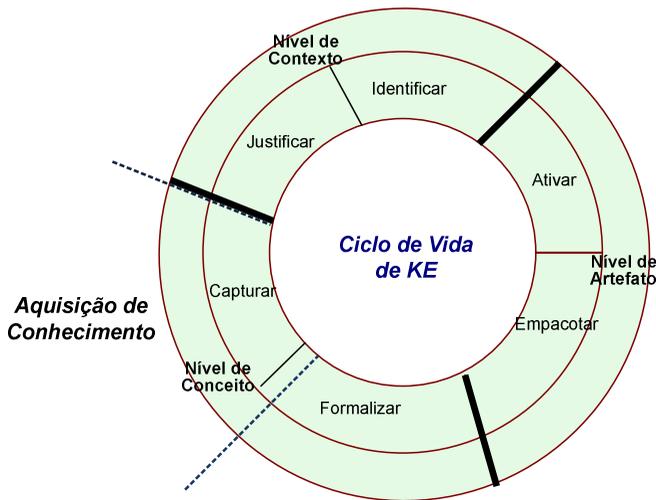


Figura 8: Aquisição de conhecimento no ciclo de vida de KE

Fontes: Baseado em (HUA, 2008; PRESTON et al., 2005; SCHREIBER et al., 2000; ANGELE, FENSEL & STUDER, 1998).

O processo de aquisição de conhecimento pode produzir diferentes resultados: desde simples documentos, passando por ontologias, até complexas redes de conhecimento (MILTON, 2007). O produto final da aquisição de conhecimento, no entanto, precisa levar em conta alguns fatores, tais como (MILTON, 2007):

- O produto deve ser útil para os seus utilizadores finais;

- Para ser útil, o produto final deve estar cheio de conhecimento de alta qualidade que seja correto, completo e relevante e que seja armazenado de forma estruturada;
- O uso dos recursos disponíveis deve ser maximizado;
- Não deve envolver muito tempo de especialistas.

É possível perceber por estes fatores que a aquisição de conhecimento é uma tarefa difícil, pois necessita minimizar o uso dos recursos disponíveis, sendo os especialistas caros para as organizações, produzindo um produto final de conhecimento de alta qualidade que possa ser reutilizado.

Ainda outras dificuldades são encontradas na aquisição de conhecimento, especialmente pelo fato de que o elemento humano é normalmente chave no processo (SCHREIBER et al., 2000). A principal dificuldade consiste em como fazer um especialista contar ou mostrar adequadamente o que e como ele faz, sendo que a maior parte do conhecimento é representada por heurísticas – regra ou método geralmente aceitas para obter um resultado dada uma informação particular (SCHREIBER et al., 2000).

Portanto, a natureza dos especialistas não pode ser ignorada, uma vez que há um importante viés em qualquer processo humano de tomada de decisão. Este viés está fortemente relacionado ao contexto no qual a decisão foi tomada, contexto esse que pode não estar acessível a quem posteriormente receber este conhecimento codificado de alguma forma, influenciando sensivelmente as probabilidades de sucesso de uma nova decisão baseada em uma anterior de sucesso. Também outros aspectos do comportamento humano, tais como o raciocínio, a memória e a representação do conhecimento, estão sujeitos a muitas diferentes falhas possíveis, apesar de mais sutis, mas que precisam ser levadas em conta durante o processo de aquisição de conhecimento (SCHREIBER et al., 2000).

Ainda tratando-se dos problemas relacionados ao elemento humano na aquisição do conhecimento, há de se refletir sobre a qualidade dos especialistas. Nem todos os que se apresentam como especialistas em um domínio realmente o são, ou possuem diferentes graus de expertise. Assim, a forma como os especialistas são identificados e categorizados pode ter um grande impacto na qualidade do conhecimento a ser extraído (BRUIN & ROSEMAN, 2007; SCHREIBER et al., 2000).

Neste sentido, Schreiber et al. (2000) identificam três tipos de especialistas: (i) o Acadêmico: que vê o seu domínio sob o ponto de vista de uma estrutura lógica organizada, a partir de generalizações sobre as leis que regem o domínio e possuindo um conhecimento teórico que é transmitido com facilidade; (ii) o Profissional: é envolvido na resolução dos problemas do domínio no dia-a-dia, possuindo, portanto, um ponto de vista prático e realista, muitas vezes dispensando a teoria como algo não aplicável; (iii) o Samurai: é o especialista focado na ação, que muitas vezes não possui qualquer formação acadêmica, mas é capaz de responder automaticamente na solução de problemas do domínio.

Como forma de adequadamente identificar e categorizar os especialistas, Powell (2003) propõe um procedimento que se inicia com a classificação dos potenciais especialistas, preparando uma categorização e estabelecendo critérios para o envolvimento de especialistas. A seguir, parte para a identificação dos potenciais especialistas utilizando as fontes disponíveis. No próximo passo, os especialistas são ranqueados, de forma a estabelecer um índice de relevância dos especialistas, a partir do qual as opiniões extraídas podem ser também categorizadas. Por fim os especialistas são formalmente abordados, mantendo-se um controle do seu envolvimento.

Mesmo tendo-se a opinião dos especialistas corretos, muitas vezes é necessário estabelecer-se um consenso para que o conhecimento extraído possa ser considerado válido. Diversos estudos (BRUIN & ROSEMAN, 2007; OKOLI & PAWLOWSKI, 2004; POWELL, 2003) têm sugerido o uso da técnica *Delphi*, que consiste numa série de rodadas seqüenciais, intercaladas por comentários mediados, para se obter um consenso confiável das opiniões de um grupo de especialistas (POWELL, 2003). Também neste sentido, Ullman & Ambrosio (1998) apresentam um modelo de ambiente de trabalho colaborativo para busca de consenso, baseado em um modelo natural de deliberação em grupo, onde as questões e os critérios de solução aceitável são previamente definidos, as alternativas conhecidas são apresentadas aos especialistas, seguidas da avaliação e a negociação que é realizada com vistas a um acordo para a melhor solução.

Tendo por base estas diversas dificuldades apresentadas, Milton (2007) propõe alguns princípios que podem auxiliar na solução das dificuldades inerentes à aquisição de conhecimento:

- A aquisição de conhecimento deve ser sistemática. Deve-se fazer uso de métodos para acompanhar e *frameworks* devem ser utilizados. Estes ajudam a concentrar-se no conhecimento que é necessário e obtê-lo da forma mais eficiente.
- Deve-se, sempre que possível, reaproveitar o conhecimento de projetos de aquisição de conhecimento anteriores ou a partir de modelos genéricos de conhecimento, como taxonomias. Assim não é necessário recomeçar cada vez, mas sim ter um esqueleto para preencher que forneça instruções para o tipo de conhecimento a capturar.
- Uma variedade de técnicas deve estar à disposição para usar na aquisição, análise e modelagem de conhecimento. Em outras palavras, é necessário um kit de ferramentas a partir do qual possa ser escolhida a ferramenta certa para o trabalho. Como existem diferentes tipos de conhecimento, diferentes técnicas são necessárias para lidar com eles.
- Fazer uso, quanto possível, de ferramentas de software que possam ajudar a tornar o trabalho mais fácil, mais rápido e eficaz.

Aquisição de conhecimento em Melhores Práticas

O conhecimento extraído pode ser representado das mais diversas formas, pois a representação do conhecimento é fundamentalmente um substituto, um meio de expressão humana utilizado para permitir que uma entidade raciocine sobre o mundo em vez de agir nele (DAVIS, HOWARD & SZOLOVITS, 1993). Em domínios menos explorados, o conhecimento tem sido normalmente extraído e codificado na forma de melhores práticas (GRAUPNER et al., 2009), coletadas a partir da experiência acumulada de uma comunidade de especialistas e agrupada e organizada por alguma entidade ou comitê técnico.

Mesmo quando são adequadamente documentadas, as informações sobre a eficácia das melhores práticas, ou contexto no qual a sua eficácia que tem sido observada, normalmente não está disponível (SHULL & TURNER, 2005). Melhores práticas deveriam ser definidas

de uma forma que pessoas diferentes daquelas que as definiram, pudessem implementá-las e demonstrar a sua capacidade de serem repetidas (SHULL & TURNER, 2005).

No sentido de estruturar estas melhores práticas, os *Frameworks* de melhores práticas são normas e modelos de qualidade de processo que permitem a reutilização das experiências de um domínio (GRAUPNER et al., 2009). Estes *Frameworks* são normalmente genéricos e abrangem amplos domínios por meio de orientações, freqüentemente, em um nível de detalhe que não é suficiente para que as pessoas possam diretamente executar suas tarefas e processos (GRAUPNER et al., 2009).

Shull & Turner (2005) propõem uma abordagem para instanciar as abstrações disponíveis nos *frameworks* de melhores práticas. Na abordagem proposta os autores identificam, categorizam e representam o conhecimento presente nos *frameworks* de melhores práticas, incorporando a descrição de evidências observadas que validam as práticas. O trabalho estabelece que:

- *Atributos úteis e precisos das práticas podem ser resumidos a partir de evidências:* é possível sintetizar uma variedade de tipos de evidências em um conjunto razoável de resumos padrões;
- *A efetividade da prática depende do contexto onde ela é aplicada:* fatores como tamanho, nível de habilidade e proximidade da equipe profissional, a criticidade do produto, o domínio da aplicação e do orçamento disponível podem influenciar a eficácia de uma prática;
- *A descrição da prática é melhorada com o uso de narrativas:* os usuários da prática aprendem melhor a partir de narrativas; e
- *Variedade de formas de acesso às melhores práticas:* existe uma grande diversidade de razões que levam alguém a procurar por uma prática, portanto diferentes formas de acesso devem estar disponíveis.

No modelo proposto por Shull & Turner (2009), cada prática é representada por um conjunto de atributos, incluindo: o nome da prática, descrição, status (que permite identificar em que grau a prática já foi efetivamente evidenciada), resumo e monitoramento da prática (contendo narrativas do uso da prática e a compatibilidade com *frameworks*). As evidências empíricas de uso da prática são registradas

numa estrutura que contém: os benefícios primários do uso da prática, fases do ciclo de vida onde a prática é aplicável, escopo organizacional, capacitadores, barreiras à aplicação da prática, entre outros.

Uma forma de operacionalizar a captura e reutilização de conhecimento de melhores práticas é por meio da abordagem de *Experience Factory*, que define um *framework* para o gerenciamento das experiências (BASILI, LINDVALL & COSTA, 2001), tendo sido aplicada com sucesso em organizações de desenvolvimento de software (BASILI, LINDVALL & COSTA, 2001; DINGSØYR, 2000). Esta abordagem foca na aprendizagem organizacional pela reutilização do conhecimento anteriormente obtido no desenvolvendo projetos de software para melhorar a qualidade dos novos projetos, envolvendo desde a simples reutilização de artefatos de software diretamente, até a criação de um ambiente de aprendizagem entre os desenvolvedores de software (DINGSØYR, 2000). Entretanto, o relato de análises empíricas dos resultados obtidos com a utilização de *Experience Factories* é muito pequeno e os métodos de validação utilizados são principalmente relatórios de lições aprendidas (DINGSØYR, 2000).

Quando se trata do conhecimento encapsulado na forma de melhores práticas, dentre as diversas técnicas de aquisição de conhecimento existentes, nem todas parecem ser apropriadas. Milton (2007) propõe uma classificação de técnicas de aquisição de conhecimento organizadas de acordo com o tipo de conhecimento ao qual se aplica. Conforme apresentado na Figura 9, algumas técnicas se aplicam melhor ao conhecimento procedural e explícito enquanto outras são mais apropriadas ao conhecimento conceitual tácito. As melhores práticas estão normalmente disponíveis como conhecimento explícito, variando do conceitual ao procedural.

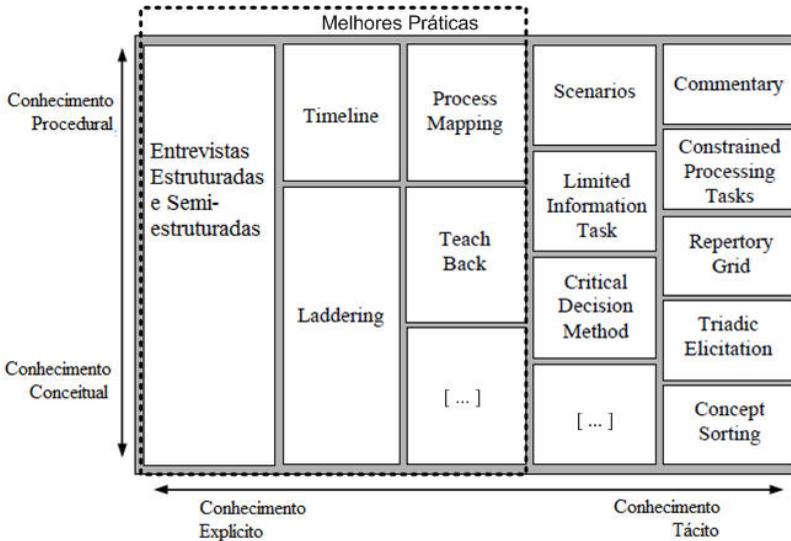


Figura 9: Aquisição de Conhecimento a partir de melhores práticas

Fonte: Baseado em (MILTON, 2007)

Especialmente o conhecimento procedural é de interesse para a aquisição de conhecimento na forma de melhores práticas, pois estas se referem à forma como as coisas são feitas (vide Figura 9). Assim as técnicas utilizadas com conhecimento na forma de melhores práticas devem ser adaptadas à forma como este conhecimento se apresenta e ao conhecimento essencialmente procedural ao qual se referem.

2.3 Software Process Capability/Maturity Models (SPCMMs)

Nesta seção é introduzido o contexto de aplicação desta pesquisa na área de Engenharia de Software, mais especificamente relacionado à customização de SPCMMs para domínios específicos de desenvolvimento de software.

Um dos produtos da evolução da Engenharia de Software foi o estabelecimento de normas e modelos que representam as melhores práticas de qualidade de processo e de produto para o desenvolvimento de sistemas de software.

Antes, no entanto, de definir os modelos para qualidade de processo é necessário primeiramente definir o que significa qualidade. Existem diversas definições para qualidade, dependendo de onde ela é

aplicada e do ponto de vista analisado (se do cliente ou do fornecedor, se de produto ou de processo, etc.). Numa visão genérica, qualidade é um conceito subjetivo que depende da forma como um indivíduo percebe um objeto e a relação deste com as suas necessidades e expectativas. O dicionário Aurélio (FERREIRA, 2008) apresenta uma definição geral para qualidade:

“Qualidade é uma propriedade, atributo ou condição das pessoas, capaz de distingui-las das outras e de lhes determinar a natureza; numa escala de valores, a qualidade é uma propriedade, atributo ou condição que permite avaliar e, conseqüentemente, aprovar, aceitar ou recusar qualquer coisa.”

Entretanto, qualidade pode ser definida como a adequação do produto ou serviço à necessidade do consumidor (JURAN, 1990); características que determinam o grau de satisfação do cliente (FEIGENBAUM, 1986); percepção e satisfação das necessidades do mercado, adequação ao uso e homogeneidade dos resultados do processo (ISHIKAWA, 1986); conformidade com os requisitos (CROSBY 1990) ou o quão bem o produto satisfaz o cliente (JALOTE, 1999).

No contexto do desenvolvimento de sistemas de software, é utilizada neste trabalho a definição de qualidade da NBR/ISO 9126 (2003) que define qualidade como “a totalidade dos atributos que determinam a capacidade de um produto de satisfazer as necessidades explícitas e implícitas quando utilizado em condições específicas”.

Entretanto, pode-se afirmar que a qualidade é um conceito complexo e multidimensional que não contém um conjunto universal de atributos (GARVIN, 1984). Portanto, em cada área de conhecimento onde a qualidade é necessária, os atributos que a caracterizam precisam ser elencados e definidos. No domínio de desenvolvimento de software, os atributos de qualidade são aqueles relacionados às necessidades explícitas e implícitas dos usuários finais. As necessidades explícitas dos usuários são normalmente traduzidas nos requisitos desejados para o software, que devem ser atendidos quando este estiver em funcionamento.

A literatura apresenta atributos de qualidade específicos para produtos de software, oferecendo inclusive normas para avaliar o atendimento destes atributos pelos produtos e processos de software, determinando o seu alinhamento ou não a estes atributos de qualidade. A norma ISO/IEC 25000 (2005), por exemplo, estabelece uma visão atualizada dos atributos de qualidade de produtos de software por meio da definição dos requisitos de qualidade e avaliação - SQuaRE - *Software Product Quality Requirements and Evaluation* (ABRAN et al., 2005).

DeLone e McLean (2003) apresentam uma taxonomia destes diversos atributos de qualidade elencados para produtos e processos de software, (vide Figura 10) que demonstra uma relação de causa e efeito entre a qualidade da informação, a qualidade do sistema e a qualidade do serviço sobre satisfação do usuário final de um sistema (qualidade percebida).

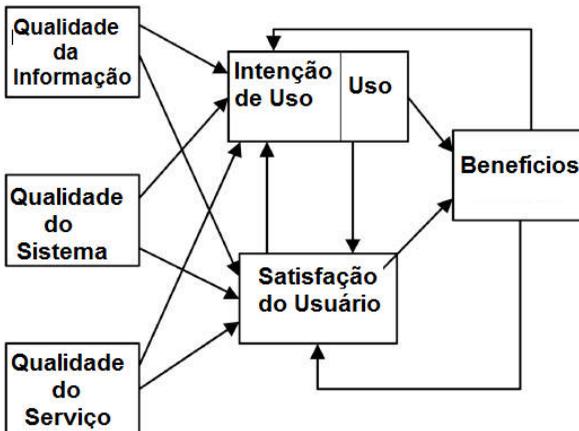


Figura 10: Modelo de DeLone e McLean

Fonte: (DELONE & MCLEAN, 2003)

No sentido de estabelecer o que representa a qualidade de processo para um determinado domínio, os *frameworks* de melhores práticas de processo têm, em geral, seu conhecimento baseado nos bons princípios de processos. Estes modelos e normas diferem em estrutura e formalização, mas possuem alguns elementos e conceitos básicos que os caracterizam. Na seção seguinte são apresentadas estas estruturas básicas e os conceitos relacionados.

Estrutura dos SPCMMs

Os modelos de qualidade de processo de software são tipicamente estruturados na forma de capacidade e/ou maturidade de processo. Como não há um nome padrão para este tipo de modelo, o termo *Software Process Capability / Maturity Model* (SPCMM) é utilizado. Este termo e o seu significado são derivados do proposto por Salviano & Figueiredo (2008) e completado em Wangenheim et al. (2010a):

“*Software Process Capability/Maturity Models* são modelos que descrevem as melhores práticas para processos de ciclo de vida do software, baseadas em bons princípios de engenharia e gerenciamento de processos, e conjuntos atributo-processo para aspectos de *design* de capacidade/maturidade.”

Sob o ponto de vista de aquisição de conhecimento, um SPCMM constitui-se em uma representação de conhecimento (DAVIS, HOWARD & SZOLOVITS, 1993) na forma de melhores práticas. Para representar o conhecimento na forma de um conjunto de melhores práticas de um determinado domínio de desenvolvimento de software (WAGENHEIM et al., 2010a), um SPCMM inclui aspectos de maturidade/capacidade de processo, que contribuem para a satisfação das necessidades relevantes de qualidade de um domínio. Assim, ao focar na melhoria da qualidade do processo, os SPCMMs objetivam, em última instância, melhorar a qualidade do produto desenvolvido (SEI, 2010).

De acordo com Matook & Indulska (2009), os modelos de referência para qualidade de processo de software devem atender os seguintes requisitos (vide Quadro 3).

Quadro 3: Requisitos dos SPCMMs

Fonte: (MATOOK & INDULSKA, 2009)

Característica	Significado
Generalidade	Grau em que o modelo executa uma ampla gama de funções e é útil em diferentes casos.
Flexibilidade	Facilidade com que o modelo se adapta e se acomoda a mudanças dos requisitos diferentes daqueles para os quais

	foi projetado especificamente
Compleitude	Grau em que todos os componentes do modelo estão presentes em um escopo pré-definido.
Usabilidade	Facilidade com que um dos utilizadores ou empresa podem operar, implementar e aplicar o modelo.
Compreensibilidade	Grau em que a finalidade, conceitos e estrutura do modelo é claro para os usuários.

Além destas características, os SPCMMs precisam ser avaliados de forma a garantir a sua consistência, aplicabilidade e validade (MATOOK & INDULSKA, 2009). Da mesma forma, uma fundamentação empírica é essencial no desenvolvimento de quaisquer modelos de referência (AHLEMANN & GASTL, 2006), embasando, especificamente, o mapeamento das melhores práticas nas expectativas de qualidade do domínio, de forma que possa realmente contribuir para a melhoria final da qualidade do produto desenvolvido.

Em relação à sua estrutura os modelos de capacidade/maturidade de processo de software, em geral possuem diversas características. Lahrman & Marx (2010) propõem uma fundamentação básica das características estruturais deste tipo de modelo, de forma a classificá-los quanto à sua estrutura, conforme apresenta o Quadro 4.

Quadro 4: Características dos SPCMMs

Fonte: Baseado em (LAHRMANN & MARX, 2010)

Critério	Características		
Dimensões	Unidimensional	Multidimensional	Hierárquico
Representação	Contínua		Por estágios
Audiências	Única		Múltiplas
Abordagem de avaliação	Qualitativa		Quantitativa

Segundo esta classificação, uma dimensão na estrutura deste tipo de modelo representa uma visão do modelo focada em um dos aspectos da melhoria de processos de software (LAHRMANN & MARX, 2010), como por exemplo: processos, avaliação, indicadores, recursos humanos, etc. A representação contínua permite que uma organização escolha uma determinada área de processo (ou grupo de áreas de processo) e melhore processos relacionados a ela (SEI, 2010). Já a representação por estágios utiliza conjuntos predefinidos de áreas de processo para definir um caminho de melhoria para uma organização.

Esse caminho de melhoria é caracterizado por níveis de maturidade. (SEI, 2010). Quanto à audiência, existem modelos que são específicos para um domínio de desenvolvimento de software e outros que são mais genéricos e são utilizados como referência de melhores práticas em mais de um domínio (LAHRMANN & MARX, 2010). A avaliação da maturidade/capacidade pode ser realizada utilizando abordagens qualitativas ou quantitativas (LAHRMANN & MARX, 2010).

Seguindo a classificação estrutural proposta por (LAHRMANN & MARX, 2010), os SPCMMs são modelos de referência para qualidade de processos de software: multidimensionais; possuem representação contínua, por estágios ou ambas; podem atender a audiências únicas ou múltiplas e podem apresentar tanto abordagens de avaliação qualitativa quanto quantitativa. Desta forma é possível perceber que o que caracteriza a estrutura de normas e modelos de referência de processo de software de forma que possam ser classificados como SPCMMs é o fato de que eles possuem, necessariamente, ao menos duas dimensões: a dimensão de processos e a dimensão de capacidade/maturidade de processo.

A **dimensão de capacidade/maturidade** define os critérios que, com base num framework de avaliação definido, indicam a habilidade que um processo ou uma organização possui para alcançar seus objetivos de negócio, atingindo aos atributos de processo associados a cada nível (ISO/IEC, 2006). Enquanto a capacidade refere-se às características específicas dos processos individualmente, definindo conjuntos de atributos a serem alcançados pelo processo em cada nível (ISO/IEC, 2006), a maturidade estabelece patamares de evolução organizacional, que identificam o conjunto de processos associados a cada um dos níveis numa escala organizacional (ISO/IEC, 2008).

A **dimensão de processos** define um conjunto de entidades que descrevem as melhores práticas dos processos do ciclo de vida com base em princípios de gestão de processos. A norma ISO/IEC 15504-2 (2003) define os requisitos mínimos de um modelo de referência de processo, indicando que, para cada processo, devem ser definidos: uma declaração de propósito e os resultados de sua execução (ISO/IEC, 2003). O propósito do processo descreve os objetivos de execução do processo, mensuráveis e de alto nível, e os prováveis resultados da aplicação efetiva do processo (ISO/IEC, 2003).

A norma ISO/IEC 15504, além de definir os atributos de um modelo de referência de processo, conforme já citado, também define

um SPCMM de referência. Este é apresentado como um exemplo de SPCMM em detalhes a seguir.

ISO/IEC 15504

Iniciada em 1993 pelo projeto SPICE (*Software Process Improvement and Capability Determination*), a ISO/IEC 15504 é uma norma internacional que tem como objetivos: (i) melhoria dos processos: gerando um perfil dos processos, identificando os pontos fracos e fortes que serão utilizados para a elaboração de um plano de melhorias; (ii) determinação da capacidade dos processos: viabilizando a avaliação de um fornecedor em potencial e/ou obtendo o seu perfil de capacidade (ISO/IEC, 2003).

O SPCMM estabelecido na parte 5 da norma (ISO/IEC, 2006) serve de base para avaliações e possui as dimensões de: processos e capacidade, conforme ilustra a Figura 11.

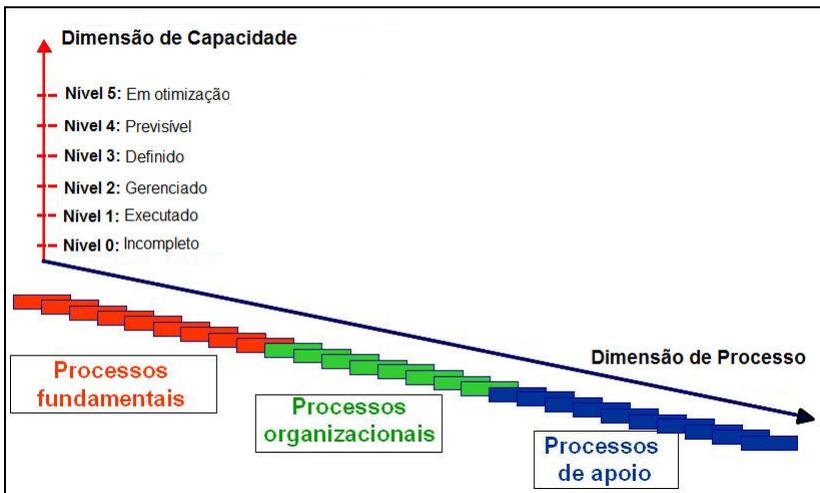


Figura 11: As duas dimensões do SPCMM definido na ISO/IEC 15504

Fonte: (ISO/IEC, 2006)

A dimensão de capacidade/maturidade de processo do SPCMM definido na norma ISO/IEC 15504 (2006) estabelece inicialmente uma abordagem contínua na avaliação de processos de software, onde não

existe uma definição fixa da seqüência na qual as áreas de processo devem ser implementadas, em oposição à abordagem clássica por estágios, onde se oferece uma seqüência fixa de processos pré-definida. Portanto, a norma ISO/IEC 15504 permite a avaliação da capacidade de cada processo individualmente, por meio de um método de avaliação de processos de software (ISO/IEC, 2006).

São cinco os níveis de capacidade de processo definidos na norma (ISO/IEC, 2006):

- **Nível 0** (incompleto): o processo não é executado nem consegue alcançar a sua finalidade. Neste nível não há quase nenhuma evidência de uma realização sistemática da finalidade do processo.
- **Nível 1** (executado): o processo é executado e consegue alcançar sua finalidade.
- **Nível 2** (gerenciado): o processo é executado de uma forma controlada (de forma planejada, monitorada e ajustada) e seus produtos de trabalho são apropriadamente estabelecidos, controlados e mantidos.
- **Nível 3** (estabelecido): o processo é executado utilizando um processo definido, baseado em um processo padrão, de forma que seja capaz de alcançar os resultados esperados.
- **Nível 4** (previsível): o processo estabelecido opera-se agora dentro de limites definidos para conseguir seus resultados esperados.
- **Nível 5** (otimização): o processo previsível é melhorado continuamente para alcançar os objetivos de negócio relevantes atuais e futuros.

Em uma avaliação segundo a norma ISO/IEC 15504 são utilizados nove atributos de processo (PA) para avaliar a capacidade de cada um dos processos. Estes atributos de processo são usados para determinar se um processo alcançou ou não um dado nível de capacidade. Cada atributo mede um aspecto particular da capacidade do processo (ISO/IEC, 2006). O Quadro 5 apresenta os atributos de processo de cada nível de capacidade.

Quadro 5: Atributos de processo e níveis de capacidade

Fonte: (ISO/IEC, 2006)

Atributo de Processo	Níveis de Capacidade e Atributos de Processo
	Nível 0: Processo Incompleto
	Nível 1: Processo Executado
PA 1.1	Desempenho do Processo
	Nível 2: Processo Gerenciado
PA 2.1	Gerenciamento do Desempenho do Processo
PA 2.2	Gerenciamento dos Produtos de Trabalho
	Nível 3: Processo Estabelecido
PA 3.1	Definição do Processo
PA 3.2	Distribuição do Processo
	Nível 4: Processo Previsível
PA 4.1	Medição do Processo
PA 4.2	Controle do Processo
	Nível 5: Processo em Otimização
PA 5.1	Inovação do Processo
PA 5.2	Otimização Contínua

Além da capacidade dos processos, a norma também oferece a possibilidade de uma avaliação de maturidade organizacional. A maturidade organizacional indica o quanto uma determinada organização implementa de forma consistente os seus processos em um escopo definido, que contribui para a realização de seus objetivos de negócio (ISO/IEC, 2008). Assim, um nível de maturidade representa um ponto em uma escala ordinal que caracteriza a maturidade da organização no âmbito do modelo de maturidade organizacional utilizado, sendo que cada nível tem por base o cumprimento do nível imediatamente inferior (ISO/IEC, 2008). Os níveis de maturidade estabelecidos na parte 7 da norma são (ISO/IEC, 2008):

- **Nível 0 (Imaturo):** a organização não demonstra a aplicação eficaz dos processos que são fundamentais para suportar o seu negócio.

- **Nível 1 (Básico):** a organização demonstra o alcance do propósito dos processos que são fundamentais para suportar o seu negócio.
- **Nível 2 (Gerenciado):** a organização demonstra que consegue gerenciar os processos que são fundamentais para suportar o seu negócio.
- **Nível 3 (Estabelecido):** a organização demonstra a efetiva definição e implantação dos processos que são fundamentais para suportar o seu negócio.
- **Nível 4 (Previsível):** a organização demonstra uma compreensão quantitativa dos processos relevantes que são fundamentais para apoiar os seus objetivos, a fim de estabelecer um desempenho consistente e previsível.
- **Nível 5 (Inovação):** a organização demonstra a habilidade de mudar e adaptar-se ao desempenho dos processos que são fundamentais para apoiar os seus objetivos de negócio de uma forma sistematicamente planejada e previsível.

Na dimensão de processo, a norma inclui um modelo exemplo onde são estabelecidos 50 processos (48 processos na parte 5 mais 2 novos processos na parte 7), agrupados em três grandes categorias. A Figura 12 mostra os processos e suas categorias.

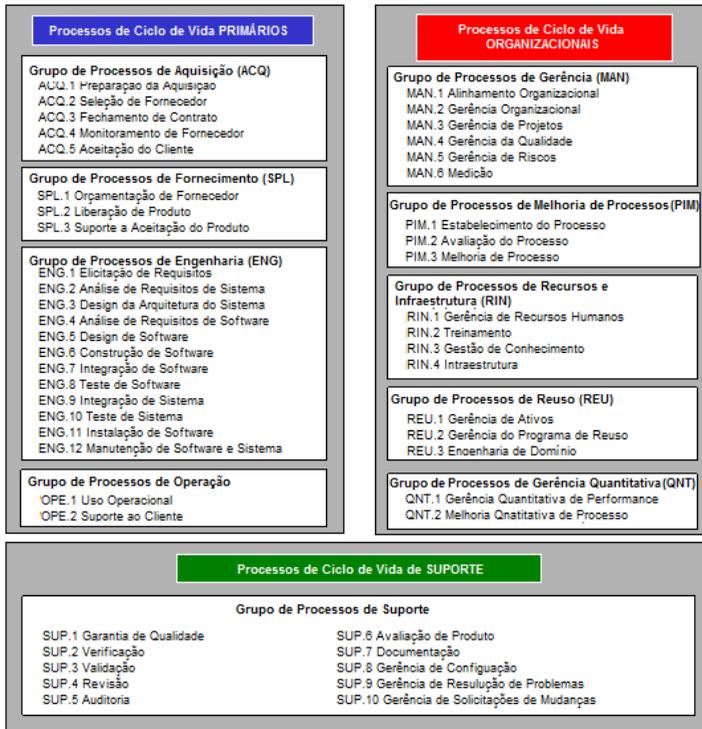


Figura 12: Os 50 processos definidos na norma ISO/IEC 15504

Fonte: Baseado em (ISO/IEC, 2008; ISO/IEC, 2006)

Conforme demonstra a Figura 12, os processos **primários de ciclo de vida**, aqueles mais diretamente ligados ao desenvolvimento do produto de software, são formados pelos grupos de processos: processos de aquisição, de fornecimento, de engenharia e processos operacionais. Os **processos de suporte** são aqueles que auxiliam e suportam a realização dos processos primários do ciclo de vida, estando indiretamente vinculados à geração do produto de software, são eles: processos de controle de configuração, garantia da qualidade do processo e garantia da qualidade do produto. O grupo de processos de **ciclo de vida organizacional** são aqueles responsáveis por proverem os recursos organizacionais necessários à realização dos demais processos, são eles: processos de gerenciamento da organização, melhoria do processo, recursos e infra-estrutura, processos de reuso e de gerência quantitativa.

No modelo de exemplo da norma ISO/IEC 15504-5 (2006), um processo é detalhado em termos de propósito (*Process Purpose*), resultados esperados (*Process Outcomes*), práticas base (*Base Practices*) e produtos de trabalho (*Work Products*). Para que um processo seja executado, ele deve utilizar as práticas base e gerar os produtos de trabalho para atingir os resultados esperados. A Figura 13 mostra a estrutura de definição do processo de Preparação da Aquisição na norma.

Process ID	ACQ.1	
Process Name	Preparação da Aquisição	
Process Purpose	O objetivo do processo de Preparação de Aquisição é estabelecer as necessidades e objetivos da aquisição e comunicá-las aos fornecedores em potencial.	
Process Outcomes	Como resultado da implementação bem sucedida do processo de Preparação de Aquisição:: 1) o conceito ou a necessidade de aquisição, desenvolvimento ou melhoria é estabelecido; ... 5) critérios de seleção de fornecedores são definidos;	
Base Practices	<p>ACQ.1.BP1: Estabelecer a necessidade. Estabeleça uma necessidade de adquirir, desenvolver ou melhorar um sistema, produto de software ou serviço. [Outcome 1]</p> <p>...</p> <p>ACQ.1.BP6 Comunicar a necessidade. Comunique a necessidade de aquisição às partes interessadas, através dos canais identificados. [Process Purpose; Outcome: 1]</p>	
Work Products		
	Inputs	Inputs
	05-02 Business Goals [Outcome: 1]	08-02 Acquisition plan [Outcome: 4]

Figura 13: Extrato de detalhamento de um processo na ISO/IEC 15504
Fonte: (ISO/IEC, 2006)

Além desta estrutura bidimensional, algumas experiências são relatadas na literatura utilizando mais de duas dimensões na definição estrutural de SPCMMs. Um exemplo é o modelo eGov MM (IRIBARREN et al. 2008), utilizado no domínio de melhoria de processos em governo eletrônico. Esse modelo inova pela introdução de uma dimensão de características de negócio no intuito de abranger mais do que as questões relacionadas à tecnologia da informação

(IRIBARREN et al., 2008). Entretanto ainda não há relatos dos possíveis benefícios da introdução de mais dimensões na estrutura de SPCMMs.

Customização de SPCMMs para domínios específicos

A partir do uso dos SPCMMs, percebeu-se que o domínio da aplicação de SPCMMs influencia na sua eficácia (SHULL & TURNER, 2005). Assim, o conhecimento representado na forma de melhores práticas é também influenciado pelo contexto onde é aplicado.

Determinados domínios de desenvolvimento de software apresentam características e necessidades específicas que precisam ser devidamente atendidas (BEECHAM, HALL & RAINER, 2005). Para estes domínios os SPCMMs precisam ser customizados, ou seja, alterados ou modificados conforme as especificações individuais (MICHAELIS, 2010) do domínio. Desta constatação surgem iniciativas de desenvolvimento de SPCMMs específicos de domínio, procurando atender a sua necessidade específica. Alguns exemplos são Spice4Space (CASS et al., 2004) para o domínio software aeroespacial, AutomotiveSPICE (SPICE USER GROUP, 2010) voltado para o domínio de software embarcado em automóveis, OOSPICE (TORGERSSON & DORLING, 2002) para o desenvolvimento baseado em componentes, o modelo brasileiro MPS.BR (SOFTEX, 2009) para micro e pequenas empresas e diversos outros (WANGENHEIM et al., 2007b).

Observando a tendência de criação de SPCMMs para diferentes domínios de desenvolvimento de software (WANGENHEIM et al., 2010a), a principal questão consiste em como estes modelos têm sido desenvolvidos, de forma a procurar garantir que SPCMMs específicos de domínio realmente contenham as melhores práticas para aquele domínio.

Os SPCMMs específicos de domínio têm sido baseados nas melhores práticas ou fatores de sucesso derivados de projetos que têm demonstrado bons resultados dentro de uma organização, segmento ou indústria (NIAZI, WILSON & ZOWGHI, 2005). A crítica mais importante a SPCMMs, no entanto, tem sido a sua fraca base teórica (MATOOK & INDULSKA, 2009), decorrendo daí a dúvida de onde vêm estas melhores práticas e se o processo que foi usado para defini-las contribuiu para a criação de modelos válidos e confiáveis

(WANGENHEIM, 2010a). Além disso, a falta de esforço empregado para avaliar os modelos em termos de eficácia, confiabilidade e generalização, é uma das razões para os resultados muitas vezes ambíguos, que vêm sendo obtidos da utilização dos SPCMMs na prática (BRUIN & ROSEMAN, 2005).

2.4 Discussão

Este capítulo apresenta os conceitos relacionados à engenharia do conhecimento, aquisição de conhecimento e à área de aplicação no desenvolvimento de SPCMMs. Assim, é possível perceber que diversos métodos, técnicas e abordagens têm sido desenvolvidas para suportar o ciclo de vida de KE e, mais especificamente, a aquisição do conhecimento, envolvendo desde a extração até a sua codificação.

O conhecimento extraído em um domínio tem sido codificado de diversas maneiras, sendo muitas vezes representado na forma de melhores práticas. Neste sentido, os modelos e normas de capacidade/maturidade de processo de software podem ser considerados frameworks de melhores práticas, pois eles agrupam sob diferentes formas e estruturas, um grande corpo de conhecimento (IEEE, 2004) sobre as melhores práticas de software em um formato que permite a sua reutilização (WANGENHEIM et al. 2010a). A principal finalidade destes modelos é fazer com que as organizações possam aplicar o conhecimento que está representado na forma de melhores práticas (MATOOK & INDULSKA, 2009), mesmo que possa não estar sempre adequadamente estruturado (SHULL & TURNER, 2009).

Assim, a partir da revisão destes conceitos sob o ponto de vista da KE e da SE, pode-se constatar que não foram encontrados na literatura, métodos e técnicas voltados para a aquisição de conhecimento para customização de SPCMMs para domínios específicos.

Tendo por base os conceitos definidos nesta seção, o próximo capítulo procura identificar o estado da arte e prática no desenvolvimento de SPCMMs. Para tanto foram levantados os trabalhos correlatos, e realizada uma revisão sistemática da literatura e um *survey*.

3 ESTADO DA ARTE

Este capítulo procura identificar o estado da arte e prática no desenvolvimento de SPCMMs. Para tanto foram levantados os trabalhos correlatos por meio de uma revisão sistemática da literatura completada por um survey.

Embora um grande volume customizações de SPCMMs já tenha sido realizado (WANGENHEIM et al., 2010b), o estudo sobre como realizar de forma sistemática a aquisição de conhecimento necessária a tais customizações é escasso. Como este trabalho trata da definição de um método para aquisição de conhecimento de SPCMMs com o intuito de fornecer um suporte sistemático para a sua customização para domínios específicos, esta seção apresenta as experiências relatadas na literatura que contém estudos que se aproximam desta pesquisa.

Foram considerados trabalhos correlatos para o estado da arte aqueles que apresentam algum grau de similaridade quanto: ao seu (i) objeto: trabalhos que enfoquem aquisição de conhecimento a partir de SPCMMs; (ii) objetivos: trabalhos que definam metodologias, métodos ou abordagens que tratem do desenvolvimento ou customização de SPCMMs; ou (iii) resultados: trabalhos que apresentem análises de quais são os SPCMMs existentes e como eles têm sido desenvolvidos.

Desta forma, com o objetivo de identificar o estado da arte e prática em relação ao objeto, objetivos e resultados desta tese, foram levantadas três perguntas que norteiam o levantamento do estado da arte:

1. Quais SPCMMs estão sendo desenvolvidos atualmente?
2. Como os SPCMMs têm sido desenvolvidos?
3. Quais são os métodos para desenvolvimento/customização de SPCMMs?

As próximas seções deste capítulo apresentam os trabalhos identificados que procuram responder a estas perguntas.

3.1 Quais SPCMMs estão sendo desenvolvidos atualmente

Vários autores têm analisado o estado da arte no que diz respeito aos modelos e normas de referência para Engenharia de Software, não se restringindo a SPCMMs somente. Dentre estes, destaca-se o trabalho “*Frameworks Quagmire*” (atoleiro de frameworks), apresentado

inicialmente em (SHEARD, 1997) e atualizado em (SHEARD, 2001), que realiza uma análise de: normas, padrões de processos, práticas recomendadas, diretrizes, modelos de maturidade e outros frameworks relacionados a software e sistemas. Esta interessante pesquisa reconhece o valor dos novos modelos integrados, como CMMI e o FAA ICMM, que facilitam a adoção e compatibilidade com outros modelos, mas a conclusão do trabalho recomenda às empresas definirem os seus próprios processos, a fim de atingir seus objetivos de negócio. Entretanto, este trabalho não apresenta foco na customização de modelos genéricos para domínios específicos.

No trabalho apresentado em (MOORE, 1999) pelo IEEE *Software Engineering Standards Committee*, é proposta uma coleção integrada de normas de engenharia de software. Este trabalho apresenta uma pesquisa que descobriu 315 normas, guias, manuais e outros documentos normativos mantido por 46 diferentes organizações, demonstrando que os profissionais de engenharia de software têm dificuldade em encontrar as normas e padrões de que precisam para uma necessidade particular dentre os inúmeros disponíveis. Neste trabalho é oferecido um guia em quatro volumes com todas as normas mais importantes encontradas na pesquisa, numa visão integrada sob um guarda-chuva da norma IEEE / EIA 12.207. Mesmo não focando estritamente em SPCMMs, este trabalho denota a expressiva quantidade de normas e padrões existentes relacionados a software, demonstrando a clara necessidade de um maior apoio metodológico para o seu desenvolvimento.

Paulk (2004) propõe algumas alternativas para sobreviver ao “atoleiro de modelos de processos, modelos integrados e normas”. O documento tenta descobrir porque os modelos de processos e padrões de referência são diferentes, e propor estratégias para a integração de diversos modelos e padrões sob a perspectiva de desenvolvedores e implementadores de modelos. Apesar de este trabalho não focar em modelos para domínios específicos de software, o autor afirma que modelos de âmbito mais restrito provêm requisitos e orientações mais detalhados sobre a execução de partes de um modelo mais amplo

O *Guide to Software Engineering Standards and Specifications* (Guia de Normas e Especificações de Engenharia de Software) apresentado em (MAGEE & TRIPP, 1997) descobriu 55 organizações desenvolvedoras de padrões de engenharia de software, que produziram cerca de 300 padrões de desenvolvimento e manutenção de software para mais de 1,5 milhões de engenheiros de software. O estudo demonstra que menos de mil pessoas foram envolvidas no

desenvolvimento destes padrões para a população mundial de engenheiros de software. Os autores afirmam que um diretório é claramente necessário, a fim de facilitar o acesso dos profissionais ao conhecimento publicado nesta infinidade de normas. Este é mais um estudo que apresenta o grande volume de normas e padrões nesta área, reforçando a urgência de apoio sistemático ao seu desenvolvimento.

Em relação ao estado da arte em modelos de processos e normas, Magee & Thiele (2004) avaliam normas e padrões de processo sob a perspectiva de sete critérios, abrangendo pontos de vista organizacionais e profissionais. Em decorrência desta análise, este trabalho aponta alguns desafios no processo de como os desenvolvedores de normas e padrões e os usuários destas normas e padrões podem: selecionar, usar e evoluir-los. Como conclusão, os autores percebem a necessidade de um sistema atualizado e mais eficiente para a produção de normas que possam "capturar as vozes do pequeno desenvolvedor de software e do usuário de software".

Apesar de bastante abrangentes, estes trabalhos não enfocam especificamente SPCMMs, mas, em geral, abrangem outros modelos e normas de referência para engenharia de software. Assim, no sentido de identificar o estado da arte sobre quais os SPCMMs existentes atualmente, foi realizada uma revisão da literatura, na forma de uma *Systematic Literature Review* – Revisão Sistemática da Literatura (SLR), conforme relatado em (WANGENHEIM et al., 2010a) e em (WANGENHEIM et al., 2010b). Esta SLR seguiu o processo proposto por Kitchenham (2007) onde são realizadas três fases principais, iniciando pelo planejamento da revisão sistemática, seguindo a realização da revisão sistemática e a apresentação dos resultados da revisão da literatura. A Figura 14 ilustra o processo de SLR.



Figura 14: Processo de Revisão Sistemática da Literatura

Fonte: Baseado em (BRERETON et al., 2007)

A **pergunta de pesquisa** da revisão da literatura, elaborada com base na pergunta de pesquisa geral deste trabalho, definida no capítulo 1, é:

Quais *Software Capability/Maturity Models* (SPCMMs) têm sido desenvolvidos, expandidos e/ou adaptados/harmonizados?

O **escopo** da SLR (PETTICREW & ROBERTS, 2005) é assim definido:

População: Organizações ou grupos de pesquisa desenvolvedores de SPCMMs.

Intervenção: Desenvolvimento de *Software Process Capability / Maturity Models*.

Comparação: Avaliação e/ou Melhoria da Qualidade de Processo e Produto de Software.

Resultados: *Software Process Capability/Maturity Models* desenvolvidos.

Contexto: Publicações realizadas a partir de 1990.

O **Protocolo da Revisão** (*Review Protocol*) é apresentado no Quadro 6.

Quadro 6: Protocolo de Revisão

PASSO	DESCRIÇÃO
Background	Estabelecimento de um método de aquisição de conhecimento de SPCMMs para o desenvolvimento de SPCMMs para domínios específicos de processo de software.
Perguntas de Pesquisa	Quais Software Capability / Maturity Models (SPCMMs) têm sido desenvolvidos, expandidos e/ou adaptados/harmonizados?
Estratégia	<p style="text-align: center;">Fontes de Pesquisa:</p> <p>IEEEExplore, the ACM Digital Library, Compendex EI, ScienceDirect e WILEY Interscience database</p> <p style="text-align: center;">Termos de Pesquisa:</p> <p>IEEE XPLORE: <i>(standard <or> model <or> framework) <and> ("software process" <or> "software processes" <or> "software engineering") <and> (assessment <or> improvement <or> capability <or> maturity) <and> (CMMI <or> 15504 <or> 12207 <or> "MPS.BR" <or> CMM <or> SPICE <or> iso <or> standards) published since 1990</i></p> <p>ACM Digital Library: <i>((Abstract:standard) or (Abstract:model) or (Abstract:framework)) and ((Abstract:"software process") or (Abstract:"software processes") or (Abstract:"software engineering")) and ((Abstract:assessment) or (Abstract:improvement) or (Abstract:capability) or (Title: maturity)) and (CMMI or 15504 or 12207 or "MPS.BR" or CMM or SPICE or iso or standards) published since 1990</i></p> <p>Compendex/Engineering village: <i>((standard OR standards OR model or framework) AND ("software process" OR "software processes" OR "software engineering") AND (assessment OR improvement OR capability OR maturity) AND (CMMI OR 15504 OR 12207 OR "MPS.BR" OR CMM OR SPICE OR ISO OR standards)) wn KY {english} WN LA</i></p> <p>ScienceDirect: <i>title-abstr-key((standard OR standards OR model or framework) AND ("software process" OR "software processes" OR "software engineering") AND (assessment OR improvement OR capability OR maturity) AND (CMMI OR 15504 OR 12207 OR "MPS.BR" OR CMM OR SPICE OR ISO OR standards))</i></p> <p>WILEY Interscience: <i>((title: standard*) OR (abstract:</i></p>

	<i>standard*) OR (title: model) OR (abstract: model) OR (title: framework) OR (abstract: framework)) AND ((abstract: "software process") OR (abstract: "software processes") OR (abstract: "software engineering")) AND ((abstract: assessment) OR (abstract: improvement) OR (abstract: capability) OR (title: maturity)) AND (CMMI OR 15504 OR 12207 OR "MPS.BR" OR CMM OR SPICE OR iso OR standards)</i>
Cr�terios Prim�rios de Sele�o	<ol style="list-style-type: none"> 1. Todos os artigos publicados em l�ngua Ingl�s sobre modelos de capacidade/maturidade de processo de software dispon�veis na Web (atrav�s de bibliotecas digitais e bases de dados), publicados entre janeiro de 1990 e abril de 2009. 2. Somente artigos revisados (<i>peer reviewed</i>), incluindo apenas os trabalhos publicados em revistas ou anais de confer�ncias. 3. Excluir qualquer publica�o, que n�o descreva explicitamente um modelo de capacidade/maturidade de processo de software ou norma, tais como: mapeamentos entre os modelos, an�lises de modelos de qualquer tipo, modelos com um foco diferente daqueles relacionados a processo de software, etc.
Cr�terios de Qualidade	<ol style="list-style-type: none"> 1. O modelo � detalhadamente apresentado? 2. S�o apresentados detalhes de como o modelo foi desenvolvido? 3. O modelo j� foi aplicado em um ambiente real de desenvolvimento de software? 4. Foram empregadas pr�ticas/t�cnicas/metodologias sistem�ticas no desenvolvimento do modelo? 5. Foram utilizados outros modelos e/ou normas de qualidade de processo/produto de software como base para a elabora�o do modelo?

As palavras dos termos de pesquisa apresentados no Quadro 6 foram retiradas dos termos constantes nas perguntas de pesquisa, seguindo o formato proposto por (KITCHENHAM, 2007), identificando ao menos tr s sin nimos para cada um dos crit rios PICOC (PETTICREW & ROBERTS, 2005).

Os termos de pesquisa s o concatenados em um s  termo utilizando AND entre cada um dos crit rios PICOC. Ap s esta jun o os termos de pesquisa s o traduzidos para a l ngua inglesa de forma a permitir a pesquisa nas fontes de pesquisa internacionais selecionadas.   importante ressaltar que, conforme apontado por Brereton et al. (2007),

o termo de pesquisa necessita ser adaptado à sintaxe específica de cada fonte de pesquisa, resultando em strings diferentes para cada ferramenta, sem que, no entanto, sejam perdidos os critérios originais de pesquisa.

Brereton et al. (2007) também observam que a ordem dos termos de pesquisa deve ser preservada, pois nas fontes de pesquisa por ele levantadas, a ordem destas palavras altera consideravelmente a ordenação dos resultados obtidos.

Resultados da Revisão Sistemática

A execução da pesquisa inicial nas fontes mencionadas retornou 1.477 artigos no total. Em uma primeira etapa, foram os revisados títulos e resumos destes artigos. Artigos irrelevantes e duplicados foram removidos, seguindo os critérios de seleção do *Review Protocol*. Esta filtragem resultou em 61 publicações, que foram então incluídas na revisão (vide Apêndice A). A fim de organizar os modelos identificados, eles foram classificados pelo domínio para o qual são desenvolvidos e foram também identificados os modelos-fonte nos quais eles se baseiam.

Nestes 61 artigos, foram identificados 52 SPCMMs diferentes. Estes modelos são detalhadamente listados no Apêndice A, onde cada modelo é caracterizado por seu domínio, uma identificação seqüencial (de M01 a M52), seu nome e/ou iniciais, uma referência para os artigos onde o modelo foi encontrado, e uma lista de modelos-fonte nos quais ele se baseia. Alguns dos modelos foram descritos em mais de um artigo. Nestes casos (m08, m21, m24, m30, m31, m38, m41 e m44), são listadas as duas referências.

Foram identificados 29 domínios para os quais modelos estão sendo desenvolvidos. Três modelos têm foco no domínio mais genérico de Engenharia de Software e Sistemas, incluindo o desenvolvimento de serviços e aquisição (m09, m11 e m39). É possível observar que, além da evolução das novas versões de modelos já existentes, existe uma clara tendência para a especialização de modelos de domínios específicos. Atualmente, existe uma grande variedade de modelos específicos para os domínios mais diversos, incluindo, por exemplo, gestão do conhecimento, sistemas automotivos, XP, *e-learning*, etc. Alguns domínios parecem ter recebido uma maior atenção da comunidade, para os quais vários modelos específicos têm sido desenvolvidos, incluindo: o domínio de serviço de engenharia de segurança, modelos orientados para as MPEs (Micro e Pequenas

Empresas) e o domínio de teste e garantia da qualidade. A Figura 15 apresenta o foco destes modelos.

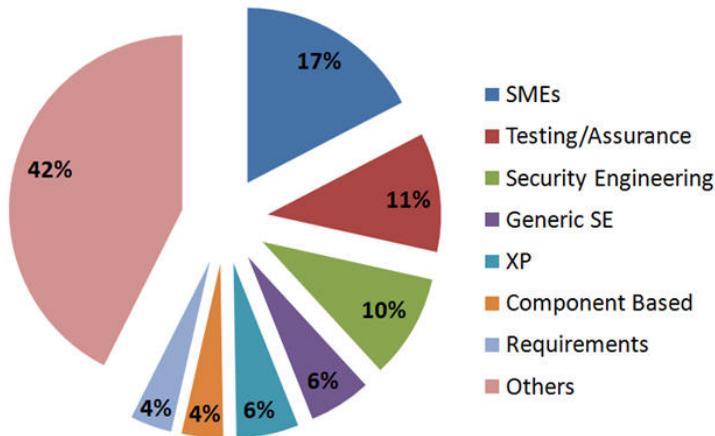


Figura 15: Domínios dos SPCMMs identificados na SLR

Fontes: (WANGENHEIM et al., 2010a; WANGENHEIM et al., 2010b)

Foi possível observar, após a análise, que os SPCMMs têm sido desenvolvidos para domínios específicos tendo por base normas e modelos já existentes, tanto genéricos para desenvolvimento de software, como normas específicas a serem atendidas para o desenvolvimento de software no domínio. No total, foram identificados 45 normas e modelos diferentes utilizados como base para o desenvolvimento dos 52 modelos apresentados no apêndice A. Analisando os modelos utilizados como base, pode-se observar (Figura 16) que a maioria é baseada no modelo CMM (31 de 52, 58%), seguido pelo uso da norma ISO/IEC 15504 e seu modelo exemplo (19 de 52, 36%) e pelo uso do CMMI-DEV (11 de 52, 21%). Vários modelos também são baseados na norma ISO/IEC 12207 (8 de 52, 15%) e ISO 9000 (9 de 52, 17%). Os 40 modelos-base restantes são usados apenas em um, dois ou três modelos, conforme apresentado na Figura 16.

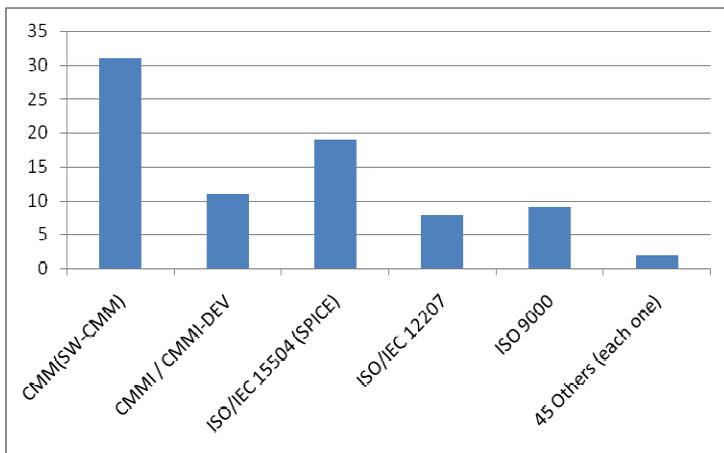


Figura 16: Percentagem de uso dos modelos como base para os SPCMMs identificados na SLR

Fontes: (WANGENHEIM et al., 2010a; WANGENHEIM et al., 2010b)

A dimensão de maturidade/capacidade é detalhada para 79% dos modelos (41 de 52). A partir da análise destes artigos que detalham a dimensão de maturidade/capacidade dos modelos (vide Apêndice B) foi possível perceber que 7% (3 de 41) dos modelos possuem tanto níveis de capacidade quanto de maturidade definidos e 46% (19 de 41) possuem níveis de capacidade na sua estrutura sendo que a mesma quantidade (46% -16 de 41) possui níveis de maturidade definidos. Quanto à quantidade de níveis de capacidade, 37% (15 de 41) apresentam seis níveis, 10% (4 de 41) apresentam cinco níveis e 7% (3 de 41) possuem menos de cinco níveis. Nos níveis de maturidade, 37% (15 de 41) apresentam cinco níveis, 10% (4 de 41) apresentam cinco níveis e 2% (2 de 41) possuem mais de cinco níveis e 2% (1 de 41) menos de quatro níveis. Isso demonstra uma clara tendência de alinhamento aos modelos nos quais a sua estrutura foi baseada, uma vez que para capacidade, o modelo mais seguido é o da ISO/IEC 15504 com seis níveis de capacidade e para maturidade o mais seguido é o do CMMI com cinco níveis de maturidade. O Apêndice B apresenta a lista completa dos modelos e seus níveis de maturidade.

Esta SLR fornece uma visão abrangente do estado da arte sobre quais são os SPCMMs atualmente existentes ou sendo desenvolvidos. Entretanto, conforme pode ser observado no Apêndice A, salvo raras exceções, os artigos analisados não apresentaram detalhamento à cerca

de como os modelos foram desenvolvidos. Na grande maioria dos casos, os artigos restringem-se em apresentar o modelo desenvolvido e alguns resultados observados de sua aplicação inicial. Apenas 21% dos trabalhos apresentaram informação detalhada sobre como o modelo foi desenvolvido, 27% apresentaram algumas informações superficiais e 52% não forneceram qualquer informação relevante sobre este aspecto.

A partir dos artigos onde foram encontradas informações relevantes sobre como o SPCMM apresentado foi desenvolvido, foram extraídas as principais fases de desenvolvimento dos modelos. Quadro 7 apresenta um resumo destas fases executadas no desenvolvimento dos SPCMMs. Os identificadores dos modelos no Quadro 7 referem-se aos códigos dos mesmos no apêndice A.

Quadro 7: Fases realizadas no desenvolvimento de SPCMMs

ID	SPCMM	FASES EXECUTADAS
m02	Business Process Maturity Model (BPMM)	1) Definição da Estrutura 1.1) Definição dos Níveis de Maturidade 1.2) Definição do Escopo de Influência das áreas de processo 2) Identificação das áreas-chave de processo de negócio 2.1) Definição de Áreas-chave de processo 2.2) Mapeamento do Processo à cadeia de valor 2.3) Pesquisa com as partes interessadas 2.4) Harmonização com a norma ISO 12207, CMM e ISO 15288
m24	Requirement CMM	1) Especificar os critérios para o modelo baseado em regras 2) Processo de abstração: 2.1) Abstrair dados empíricos sobre os requisitos principais problemas 2.2) Abstrair questões de requisitos a partir do CMM 2.3) Abstrair melhores práticas da literatura de requisitos 3) Criar o modelo especializado: RequirementCMM 4) Validar o modelo
m32	International Standard for Very Small Enterprises	1) Fase de proposta 2) Fase de preparação 3) Fase de comitê 4) Fase de inquérito 5) Fase de aprovação 6) Fase de publicação

m39	CMM (or SW-CMM) / CMMI-DEV	1.1) Patrocinador da disciplina requisita nova disciplina a ser adicionada
		1.2) SG determina se a nova disciplina pode ser integrada ao SMMI
		1.3) SG recomenda aos patrocinadores do CMMI para aprovação
		2.1) Preparação da CMMI A-Spec <i>modification</i>
		2.2) Equipes prepara o plano
		2.3) SG revisa o plano
		2.4) SG modifica A-Spec
		2.5) SG aprova a disciplina para inclusão
m09	ISO/IEC 15504-5 Process Assessment Model	2.6) Patrocinador da disciplina provê recursos e financiamento para um caso de negócio de ampla utilização
		2.7) Grupo de autores é recrutado
		3.1) B-Spec criada
		3.2) Novos elementos desenvolvidos
		3.3) Novos componentes e integração verificados com base na B-Spec
		4) Interessados/Revisores e testes-piloto testam e validam novos componentes e integração com CMMI
		5) SG revisa novos componentes para aprovação da adição ao CMMI
6) CMMI <i>Product Suite</i> distribuído e publicado		
7) Suporte planejado e implementado		
m16	Framework for assessing the use of third-party software quality assurance standards	1) Fase de proposta
		2) Fase de preparação
		3) Fase de comitê
		4) Fase de inquérito
		5) Fase de aprovação
		6) Fase de publicação
m16	Framework for assessing the use of third-party software quality assurance standards	1) Estudo de regulamentos específicos de domínio
		1.1) Revisão dos requisitos específicos de domínio
		1.2) Categorização dos requisitos de acordo com a estrutura dos SPCMMs genéricos
		1.3) Revisão e harmonização dos requisitos com o vocabulário de Engenharia de Software
2) Mapeamentos direto e inverso dos requisitos específicos de domínio para ISO e CMMI		
m42	SPI implementation maturity model	1) Análise de Fatores Críticos de Sucesso (CSF)
		1.1) Revisão da literatura sobre os CSF
		1.2) Entrevistas com interessados (<i>stakeholders</i>)
		1.3) Análise dos dados
		2) Design da dimensão de maturidade
3) Design da dimensão de CSFs		
4) Design da dimensão de avaliação		

		5) Validação do modelo
m43	Trillium	1) Obtenção de práticas do CMM 1.1) Generalização de Práticas 1.2) Harmonização de terminologia 1.3) Alteração de referências a produtos de trabalho por referências a processos
		2) Mapeamento entre as práticas do CMM com cláusulas da ISO 9001 e ISO 9000-3
		3) Extração, adição e integração das novas práticas não mapeadas da ISO e CMM
		4) Mapeamento das cláusulas das Bellcore standards para as práticas do CMM
		5) Extração, adição e integração das cláusulas das Bellcore standards não mapeadas
		6) Adição das práticas da IEC 300
		7) Adição de pontos relevantes de normas IEEE
		8) Adição de práticas específicas de domínio Trillium
		9) Validação
		10) Atribuição de roteiros ao níveis
m44	Test Maturity Model (TMM)	1) Especificação da estrutura do TMM 1.1) Especificação da estrutura do TMM com base na estrutura do CMMI 1.2) Identificação de áreas de processo, objetivos de maturidade e práticas-chave
		2) Definição dos níveis de maturidade 2.1) Definição dos níveis de maturidade alinhados ao modelo evolucionário de teste
		3) Definição das melhores práticas 3.1) Pesquisa das melhores práticas da indústria
		4) Definição das atividades e tarefas para melhorar a capacidade de teste 4.1) Identificação de atividades e tarefas alinhadas aos compromissos organizacionais 4.2) Atribuir responsabilidades às atividades e tarefas a: gerentes; desenvolvedores e testadores, usuários e clientes.
m48	Criticality-Based V&V Capability Model(CB-VVCM)	1) Criação de o modelo inicial (níveis de capacidade e conjunto de tarefas) 1.1) Decisão da estrutura de níveis de capacidade 1.2) Avaliação do conjunto mínimo de tarefas para cada nível 1.3) Criação do modelo inicial do CB-VVCM
		2) Extensão do modelo inicial para refletir o domínio de software para segurança-crítica

		2.1) Construção da estrutura do modelo
		3) Análise e extensão do modelo com as áreas de processo do CMMI
		3.1) Atribuição de tarefas
		4) Mapeamento com as práticas da ISO 9001:2000
m52	AHAA-reference model for Agile, hybrid assessment method for automotive, safety critical SMEs	1) Adoção das questões do método como fundamento para o modelo
		1.1) Adição das questões para permitir cobertura dos processos relevantes do Automotive SPICE e práticas ágeis
		2) Definição de cinco fatores para análise nas áreas de processo do CMMI
		3) Examinar as áreas de processo do CMMI escolhidas em relação aos 15 processos HIS Automotive SPICE
		4) Mapeamento das práticas ágeis relevantes para as áreas de processo do CMMI escolhidas
		5) Desenvolvimento das questões de entrevistas (AHAA <i>questions</i>)

A partir da análise detalhada dos modelos descritos nos trabalhos obtidos na SLR, percebeu-se a necessidade de se investigar com maior profundidade o processo de elaboração destes SPCMMs. Neste sentido, foi realizada uma pesquisa entre os autores dos modelos encontrados, de forma a atender a esta necessidade. Esta pesquisa e os seus resultados são apresentados na seção seguinte.

3.2 Como os SPCMMs têm sido desenvolvidos

Devido à ausência na literatura de informações detalhadas de como os SPCMMs atualmente vêm sendo desenvolvidos, foi realizado um *survey* com autores dos modelos, procurando identificar o processo pelo qual estes foram desenvolvidos, conforme relatado em (WANGENHEIM et al., 2010a). Esta seção apresenta como foi realizado este *survey* e quais foram os principais resultados observados. O *survey* foi realizado seguindo a abordagem proposta em (KASUNIC, 2005). Para planejar o *survey* foram definidos: o objetivo da pesquisa, o público-alvo e realizado o design do questionário.

Objetivo de Pesquisa: Mais uma vez partindo da pergunta de pesquisa apresentada no capítulo 1 deste trabalho, o objetivo de pesquisa para o *survey* é definido como: *Entender o processo de desenvolvimento dos SPCMMs existentes atualmente.*

Público-alvo e amostra: O público-alvo do estudo consiste nos autores/desenvolvedores de SPCMMs. Para definir a amostra a ser considerada foi realizada uma amostragem não-probabilística por julgamento, uma vez que o conjunto de todos os autores dos modelos identificados durante a revisão sistemática da literatura define uma amostra representativa da população-alvo. Foram também incluídos, por proximidade, outros autores de SPCMMs ainda não publicados que estavam dispostos a participar da pesquisa.

Aprovação da pesquisa: O projeto de pesquisa foi submetido e aprovado pelo Comitê de Ética em Pesquisa com Seres Humanos da Universidade Federal de Santa Catarina (CEPSH/UFSC), conforme pode ser visto no Anexo A – Certificado de Aprovação - Comitê de Ética.

Design do questionário: Com o intuito de obter informações gerais sobre o modelo e seu uso, no questionário (vide Apêndice C) foram incluídas questões sobre a caracterização do modelo em si, os seus autores/desenvolvedores, os detalhes de qualquer material publicado, assim como a escala de sua utilização. Uma questão foi relacionada à qual método foi adotado para o desenvolvimento do modelo. Como já se esperava (com base na escassa informação identificada na revisão de literatura) que a maioria dos modelos não fosse desenvolvida com base em algum método existente, no próprio questionário foi pré-definido um processo genérico de referência que identifica um conjunto típico de fases e etapas básicas para a criação de modelos e padrões, sob uma visão de KE. No questionário, os autores foram convidados a mapear seu processo de desenvolvimento para este processo de referência, no intuito de harmonizar a descrição dos processos.

As fases e etapas de desenvolvimento deste processo de referência foram identificadas com base nas etapas de desenvolvimento de normas internacionais ISO (ISO, 2010), PRO2PI (SALVIANO et al., 2009), o framework para o processo de desenvolvimento de modelos de maturidade proposto em (BRUIN & ROSEMAN, 2005), e em outras experiências relatadas na literatura (CANCIAN et. al, 2010, MCCAFFERY et al., 2008; MCCAFFERY & COLEMAN, 2007; SILVA et. Al, 2007; WANGENHEIM et al., 2006a; WANGENHEIM et al., 2006b), sob uma perspectiva de KE com base no processo de aquisição de conhecimento (HUA, 2008; SCHREIBER et al., 2000). O questionário completo pode ser encontrado no Apêndice C.

Execução da pesquisa: O survey foi realizado durante o período de 05/2009 a 07/2009, convidando os autores de 60 SPCMMs via e-mail (os endereços de e-mail foram retirados dos artigos encontrados na revisão sistemática da literatura). No total, foram recebidas 18 respostas, representando uma taxa de retorno de 30%.

Resultados: Dentre os autores convidados foram recebidas informações sobre os modelos listados no Quadro 8. No entanto, como nem todos os autores autorizaram a publicação de informações detalhadas, somente são apresentados os resultados da pesquisa de forma agrupada.

Quadro 8: Modelos e normas analisados no survey

Fonte: (WANGENHEIM et al., 2010a)

AgileSPI-SME
ADEPT
Automotive SPICE
Federal Aviation Administration integrated Capability Maturity Model (FAA-iCMM, also known as iCMM)
SPI Implementation Maturity Model
Integrated Maturity Model - Component Based Software Engineering
ISO/IEC 29110 Software Engineering — Lifecycle Profiles for Very Small Entities (VSEs)
MARES
MoProSoft
MR MPS (<i>Modelo de Referência para Melhoria de Processo do Software Brasileiro</i>)
Multi-model Assessment Instrument (MMAI)
OOSPICE - Software Process Improvement and Capability dEtermination for Object Oriented / Component Based Software Development
Reference Guide Proposal for Software as a Service Providers
RequirementCMM
Software Maintenance Maturity Model (S3M)
SataSPIN/SPINI/ADPM
TIPA (Tudor's IT Service Management Process Assessment)

Com base nas informações fornecidas pelos autores do modelo, pode-se observar que a maioria destes modelos foi desenvolvida de forma *ad hoc* (Figura 17). Dois dos modelos analisados estão sendo desenvolvidos como normas ISO e seguem as diretivas de normalização

da ISO (ISO, 2010). Em quatro casos, os autores do modelo desenvolveram sua própria metodologia (ADEPT, RequirementCMM, S3M, ICMM).

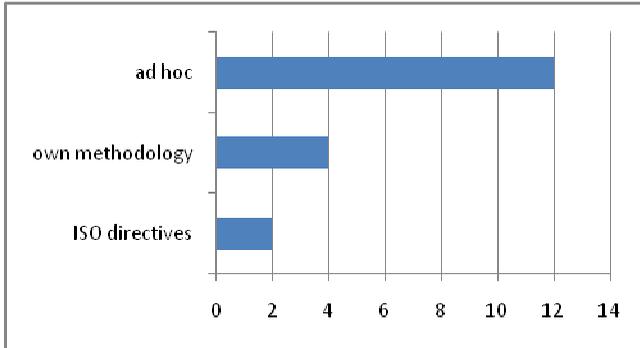


Figura 17: Metodologia utilizada para desenvolver os SPCMMs

Fonte: (WANGENHEIM et al., 2010a)

Tentando obter uma compreensão mais detalhada de como esses modelos foram desenvolvidos – mesmo na ausência de uma metodologia explicitamente definida – foi solicitado aos autores que descrevessem o que foi feito e como, de acordo com um modelo de referência pré-definido que inclui um conjunto típico de fases e etapas (Quadro 9), sob a perspectiva de aquisição de conhecimento.

Quadro 9: Fases e passos genéricos de desenvolvimento de SPCMMs sob perspectiva da KE

Fonte: (WANGENHEIM et al., 2010a)

Identificação do Conhecimento	1. Familiarização com o Domínio <i>para o qual o modelo foi desenvolvido.</i>
	2. Identificação das fontes de informação <i>que serão utilizadas como entrada para o desenvolvimento do modelo</i>
	3. Definição do escopo e dos objetivos <i>do modelo a ser desenvolvido</i>
	4. Formalização do grupo de trabalho <i>para o desenvolvimento do modelo, incluindo, por exemplo, a alocação de um patrocinador/coordenador, regras de trabalho, procedimentos, etc.</i>
Especificação do Conhecimento	5. Desenvolver o design/ arquitetura do modelo <i>identificando os elementos principais do modelo</i>
	6. Desenvolvimento do rascunho do modelo – <i>dimensão de processo como a dimensão de processo foi desenvolvida?</i>
	7. Desenvolvimento do rascunho do modelo – <i>dimensão de capacidade/</i>

	maturidade <i>como a dimensão de capacidade/maturidade foi desenvolvida?</i>
Refinamento do Conhecimento	8. Validação do rascunho do modelo (antes da publicação). <i>Se e como o rascunho do modelo foi validado com respeito a que aspectos (corretude, completude, etc.)</i>
	9. Consolidação do rascunho do modelo <i>até que o grupo de trabalho esteja convencido de que foi desenvolvida a melhor solução técnica para o problema</i>
	10. Votação no modelo consolidado <i>descreva que votou e como o consenso foi atingido.</i>
	11. Aprovação do modelo <i>indique os critérios de aprovação e o que aconteceu quando foi aprovado (ou não)</i>
	12. Publicação <i>atividades envolvidas na publicação do modelo</i>
Utilização do Conhecimento	13. Suporte ao uso do modelo <i>que tipo de suporte foi disponibilizado para o uso do modelo? Tais como: treinamentos, fóruns, etc.</i>
	14. Validação do modelo em uso <i>atividades que foquem na validação do uso prático do modelo em si (corretude, completude, etc.).</i>
Evolução do Conhecimento	15. Gerenciamento de solicitações de alteração <i>solicitações de alteração foram coletadas por que e como foram gerenciadas?</i>
	16. Confirmação, revisão ou retirada <i>como as decisões foram tomadas e como as novas versões foram desenvolvidas?</i>

Devido à falta de formalidade no desenvolvimento dos SPCMMs, as informações foram analisadas de forma qualitativa. A interpretação da informação fornecida e classificação do grau em que cada uma das fases e passos típicos que foram executados foi realizada da seguinte forma:

- 2 - executado de forma sistemática (quando técnicas sistemáticas foram aplicadas);
- 1 - executado (a etapa foi executada, mas aparentemente sem a utilização de uma técnica sistemática);
- 0 - nenhuma (o passo não foi executado) ou ainda não foi executado (quando o modelo está em desenvolvimento de forma que ainda não tenha chegado nesta fase/ passo).

A Figura 18 ilustra os resultados em um diagrama de bolha para o valor médio usando essa classificação.

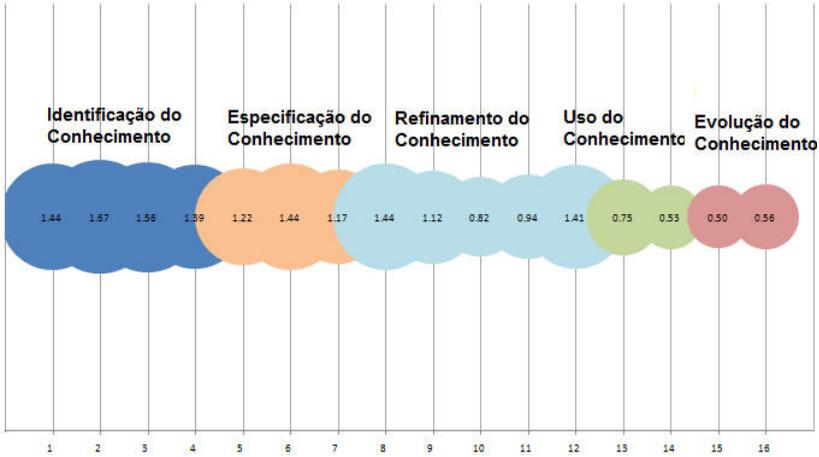


Figura 18: Grau de execução das fases e passos típicos

Fonte: (WANGENHEIM et al., 2010a)

Conforme pode ser observado na Figura 18, a maioria dos modelos foi desenvolvida executando todas as fases e etapas típicas. Uma exceção são, principalmente, os modelos desenvolvidos no âmbito de um mestrado ou tese de doutorado que, normalmente, não cobrem todas as etapas, tais como: votação, aprovação, etc. Grande parte destes desenvolvimentos também não executou (ou ainda não) as etapas relacionadas ao uso do conhecimento e as fases de evolução do conhecimento. Isso pode estar relacionado com o número estimado de usuários, o que indica que alguns modelos não estão sendo ainda aplicados na prática (Figura 19).

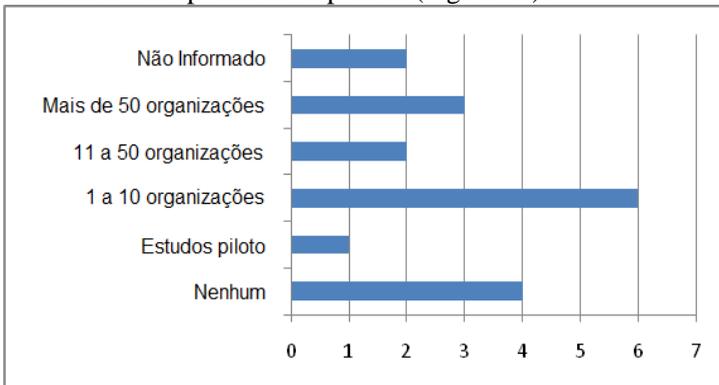


Figura 19: Grau de uso prático dos modelos

Fonte: (WANGENHEIM et al., 2010a)

É possível observar também que as técnicas aplicadas nas etapas variaram amplamente, tanto em termos do tipo quanto do grau de formalismo. Como exemplo, o primeiro passo: Familiarização com o Domínio é realizado em alguns casos simplesmente reunindo-se especialistas por meio de debates, entrevistas, ou por meio de revisões de literatura, já em outros casos é executada uma pesquisa de alcance mundial envolvendo uma grande amostra.

O grau de formação do grupo de trabalho também varia dependendo do tipo de iniciativa, em relação ao número de representantes envolvidos no desenvolvimento do modelo. A Figura 20 demonstra que o número de envolvidos varia, sendo que a maioria dos modelos está sendo desenvolvida por um grupo de cerca de 10 a 19 representantes.

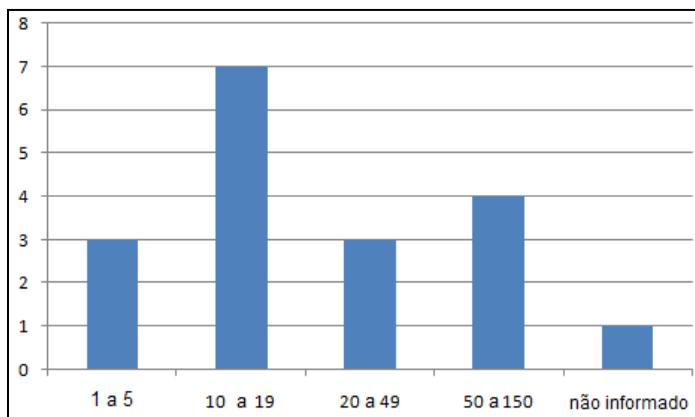


Figura 20: Número de modelos desenvolvidos / pelo número de representantes envolvidos

Fonte: (WANGENHEIM et al., 2010a)

De forma geral, é possível observar neste *survey* que, apesar do grande esforço envolvido na adaptação e personalização de SPCMMs, não existe ainda um suporte metodológico geral. Uma exceção é o desenvolvimento de normas, onde um conjunto de estágios básicos e suas diretrizes são fornecidos pela organização de normalização. No entanto, até agora, nenhum apoio mais detalhado sobre como

personalizar os modelos específicos de domínio com base em normas e modelos existentes está disponível.

Da mesma forma é possível observar que, em geral, não há aplicação de métodos e técnicas de KE no desenvolvimento ou customização deste tipo de modelo. Neste sentido, a existência de um suporte sistemático para a aquisição de conhecimento no desenvolvimento de SPCMMs específicos de domínio constitui-se numa importante contribuição tanto para a KE quanto para a SE.

3.3 Quais são os métodos para desenvolvimento/customização de SPCMMs

Conforme observado nos trabalhos que apresentam SPCMMs desenvolvidos, poucos são aqueles que apresentam detalhes sobre como eles foram desenvolvidos. Raros também são os trabalhos que apresentam propostas metodológicas para o desenvolvimento de SPCMMs.

Um dos trabalhos que trata de como SPCMMs devem ser desenvolvidos é proposto por Bruin & Rosemann (2005), que apresentam uma seqüência de etapas para o desenvolvimento e avaliação de Modelos de Maturidade. As etapas propostas são: (i) a definição do escopo do modelo, (ii) o *design* de um novo modelo, (iii) a população do modelo, usando componentes de domínio como fonte de necessidades específicas, (iv) testes, (v) implantação e manutenção do modelo. Embora, este trabalho considere as necessidades específicas de domínio no desenvolvimento de um modelo de maturidade, ele não aborda em detalhes a personalização das melhores práticas para um domínio específico a partir de modelos genéricos. Também não são adequadamente abordadas as questões referentes à estrutura e validade do conhecimento armazenado e como ele deve ser transformado ou adequado para um domínio específico.

Salviano et al. (2009) propõem o framework de método PRO2PI-MFMOD na metodologia PRO2PI para o desenvolvimento de modelos de maturidade/capacidade de processo. Com base nas suas experiências anteriores de desenvolvimento deste tipo de modelo, os autores apresentam uma série de passos para o desenvolvimento de SPCMMs. O framework é composto por sete etapas: (i) as decisões iniciais, (ii) análise de fontes (de boas práticas), incluindo: literatura, as pesquisas e outros, (iii) estratégia de desenvolvimento, incluindo a forma como a

comunidade de interesse será envolvida (iv) o *design* do modelo utilizando ISO / IEC 15504 como estrutura geral para modelar, (v) o desenvolvimento do modelo proposto; (vi) a validação do modelo *draft*, e (vii) consolidação do modelo de análise com base nos resultados da validação do modelo *draft*. Este trabalho representa uma importante pesquisa para se alcançar uma abordagem para a customização de modelos de capacidade/ maturidade de processo de software. Por tratar-se de um framework, ele aborda o desenvolvimento de SPCMMs sob um nível mais alto de abstração indicando as principais atividades e técnicas genéricas a serem utilizadas por métodos no desenvolvimento deste tipo de modelo.

Matook e Indulska (2009) propõem uma abordagem baseada em QFD - *Quality Function Deployment* (QFDI, 2010) para o desenvolvimento de modelos de processo de referência que incorpora a voz dos usuários e apresentação de uma medida para mensurar a qualidade de tais modelos. Sua abordagem também fornece um meio para gerenciar o desenvolvimento do modelo de referência, incluindo as seguintes fases: (i) definição do problema, (ii) a análise de requisitos, (iii) coleta de informação, (iv) criação de convenções e regras; (v) documentação; (vi) construção e design e, (vii) avaliação. Esta pesquisa apresenta um passo importante no desenvolvimento de um apoio mais sistemático para o desenvolvimento de modelos de referência. No entanto seu foco principal é sobre a construção do modelo, sem cobertura à sua utilização e evolução. Da mesma forma, eles não fornecem suporte metodológico detalhado para a personalização do SPCMMs, nem tampouco envolvem a aquisição do conhecimento dos SPCMMs genéricos.

Becker, Knackstedt & Pöppelbuß (2009) propõem um modelo geral de desenvolvimento de Modelos de Referência de Processos. Inicialmente os autores propõem oito requisitos para um modelo de referência de processos: (i) comparação com modelos existentes, (ii) desenvolvimento iterativo, (iii) avaliação do modelo, (iv) procedimento multi-metodológico, (v) identificação da relevância do problema, (vi) definição do problema, (vii) resultados publicados e (viii) documentação científica. A partir destes requisitos, diversos modelos obtidos da literatura são analisados. A principal conclusão obtida é de que existe pouca documentação acerca de como este tipo de modelo é desenvolvido. Os autores propõem então um processo de desenvolvimento de modelos que procura cobrir todos os requisitos definidos (vide Quadro 10). Este processo tem um importante foco na

produção da documentação científica referente ao SPCMM a ser desenvolvido, dada a carência de documentação percebida no estudo realizado. Entretanto, o trabalho não aborda a questão da evolução do modelo após a sua publicação e não possui foco na aquisição do conhecimento.

Mettler (2009) realiza uma análise mais profunda sobre os fundamentos dos modelos de maturidade de processo, colocando as principais fases descritas em (BRUIN & ROSEMAN, 2005), sob uma perspectiva de pesquisa em *design science*. Neste contexto, Mettler analisa as fases de desenvolvimento de modelos de maturidade, indicando a necessidade de métodos mais formais e maiores estudos. Apesar de apresentar uma importante crítica à forma como os modelos de maturidade de processo têm sido elaborados e aos resultados obtidos por estes, este trabalho também não aborda a questão da customização de SPCMMs para domínios específicos.

Sob um ponto de vista do desenvolvimento de modelos de referência de processo de uma forma geral, o trabalho de Ahlemann & Gastl (2006) enfoca a necessidade de fundamentação empírica no desenvolvimento de modelos de referência de processo. Os autores estabelecem os requisitos para um modelo de referência derivados da literatura e das suas experiências. É então proposto um processo para o desenvolvimento de modelos de referência de forma a atender os requisitos identificados. Uma importante contribuição deste trabalho consiste na preocupação com a fundamentação empírica do modelo de referência de processo. Os modelos devem ser, segundo os autores, empiricamente fundamentados e validados na prática. A abordagem também trata os modelos de referência de processo como ativos de conhecimento, procurando extrair o conhecimento dos especialistas de domínio. Entretanto o processo trata de modelos de referência de processos em geral, não levando em conta as especificidades dos SPCMMs.

Maier, Moultrie & Clarkson (2009) definem um guia para desenvolvimento de *Maturity Grids*, que consistem em ferramentas para avaliar a capacidade requerida de uma organização para entregar um produto ou um serviço. *Maturity Grids* são normalmente estruturados em uma grade ou matriz, composta de uma série de células onde são atribuídos níveis de maturidade para aspectos-chave de desempenho ou de atividades-chave. O guia propõe quatro fases: planejamento, desenvolvimento, avaliação e manutenção, sendo que em cada fase diversas atividades são definidas e seqüenciadas, permitindo, no entanto,

a possibilidade de iterações. O propósito cobre uma gama mais ampla de modelos, não focando em SPCMMs. Também não são analisados os aspectos de aquisição de conhecimentos no desenvolvimento, apesar do envolvimento da comunidade ser estimulado.

Alguns SPCMMs têm sido desenvolvidos na forma de normas, apoiados por algum órgão regulador ou grupo de normatização internacional (WANGENHEIM et al. 2010a). As normas são, em geral, desenvolvidas de acordo com alguns princípios (ISO, 2009): (i) Consenso: no desenvolvimento de normas diferentes interesses precisam ser levados em conta: fabricantes, fornecedores, usuários, governos, organizações de pesquisa, etc., (ii) Indústria: as normas devem fornecer soluções globais para as indústrias e clientes, (iii) Participação voluntária: a normalização é uma atividade baseada na participação voluntária de todos os interesses da comunidade. As normas ISO são desenvolvidas em um processo de três fases (ISO, 2009): Fase 1: a percepção da necessidade de uma nova norma vem de um setor da indústria, que comunica essa necessidade de um membro do corpo nacional. Esta necessidade é então avaliada e, uma vez aprovada, o escopo da nova norma é definido, envolvendo grupos de trabalho compostos por especialistas de diferentes países; Fase 2: esta é a fase de construção de consenso. Após a definição do escopo, começa a negociação entre os membros do grupo para detalhar o conteúdo da norma. Fase 3: a aprovação final e a geração do *draft* da norma é dado nesta fase, onde a norma precisa receber a aprovação de pelo menos dois terços de todos os membros do grupo e 75% dos votantes. Após este processo de votação, a primeira versão da norma é publicada.

O Quadro 10 apresenta um resumo dos processos de desenvolvimento de SPCMMs encontrados na literatura. É possível observar que, em geral, as abordagens não encaram os modelos como repositórios de conhecimento na forma de melhores práticas, nem tampouco procuram extrair o conhecimento existente, tanto que também que não há referências ao uso de técnicas de aquisição de conhecimento.

Quadro 10: Abordagens propostas para desenvolvimento de SPCMMs

Referência	Processo Proposto	Análise
Bruin & Rosemann (2005)	1) Definição do escopo do modelo	Não utiliza métodos ou técnicas de aquisição do conhecimento para customização do modelo.
	2) <i>Desing</i> do novo modelo	
	3) População do modelo, usando componentes de domínio como fonte de necessidades específicas	

	4) Testes	
	5) Implantação e manutenção do modelo	
Salviano et al. (2009)	1) Decisões iniciais	Por ser um framework, possui um nível mais alto de abstração indicando as principais atividades e técnicas genéricas no desenvolvimento deste tipo de modelo.
	2) Análise de fontes (de boas práticas), incluindo a literatura, pesquisas e outros	
	3) Definição da estratégia de desenvolvimento, incluindo a forma como a comunidade de interesse será envolvida	
	4) <i>Design</i> do modelo utilizando ISO / IEC 15504 como a estrutura geral para modelar	
	5) Desenvolvimento do modelo proposto	
	6) Validação do modelo	
	7) Consolidação de um modelo de análise da validação dos resultados do modelo	
Matook e Indulska (2009)	1) Definição do problema	Abrange especificamente a construção do modelo, sem cobertura à sua manutenção e suporte à utilização e evolução. Não enfoca o aspecto de aquisição de conhecimento.
	2) Análise de requisitos	
	3) Coleta de informação	
	4) Criação de convenções e regras	
	5) Documentação	
	6) Construção e design	
	7) Avaliação.	
Becker, Knackstedt & Pöppelbuß, (2009)	1) Definição do problema	Principal foco na produção de documentação científica do modelo. Não utiliza métodos ou técnicas de aquisição de conhecimento para customização do modelo.
	2) Comparação com modelos existentes	
	3) Determinação da estratégia de desenvolvimento	
	4) Desenvolvimento do modelo	
	4.1) Selecionar o nível de <i>design</i>	
	4.2) Selecionar a abordagem	
	4.3) Elaborar a seção do modelo	
	4.4) Testar os resultados	
	5) Concepção da transferência e avaliação	
6) Implementação da mídia de transferência		
7) Avaliação		
8) Rejeição do modelo (em caso de		

	descontinuação)	
Ahlemann & Gastl (2006)	1) Identificação do Problema	Enfoca a necessidade de fundamentação empírica. Trata os modelos como artefatos de conhecimento. Abrange modelos de referência de processo de forma geral, sem tratar das especificidades dos SPCMMs.
	2) Planejamento	
	2.1) Planejamento do modelo	
	2.2) Planejamento da organização	
	2.3) Planejamento do método	
	2.4) Planejamento da tecnologia	
2.5) Planejamento do projeto		
3) Construção do Modelo	Enfoca a necessidade de fundamentação empírica. Trata os modelos como artefatos de conhecimento. Abrange modelos de referência de processo de forma geral, sem tratar das especificidades dos SPCMMs.	
3.1) Captura do conhecimento do domínio		
3.2) Construção do <i>frame</i> de referência		
3.3) Preparação da primeira investigação empírica		
3.4) Execução da investigação empírica		
3.5) Construção inicial do modelo		
4) Validação	Enfoca a necessidade de fundamentação empírica. Trata os modelos como artefatos de conhecimento. Abrange modelos de referência de processo de forma geral, sem tratar das especificidades dos SPCMMs.	
4.1) planejamento e execução da segunda investigação empírica		
4.2) Refinamento do modelo		
5) Teste prático		
5.1) Aplicação do modelo		
5.2) Refinamento do modelo		
6) Documentação	Enfoca a necessidade de fundamentação empírica. Trata os modelos como artefatos de conhecimento. Abrange modelos de referência de processo de forma geral, sem tratar das especificidades dos SPCMMs.	
Maier, Moultrie & Clarkson (2009)		1) Planejamento
		1.1) Objetivo
		1.1.1 Sensibilização
		1.1.2 Benchmarking
		1.2 Finalidade
	1.3 Requisitos	
1.4 Âmbito	Define passos bem fundamentados em análise de desenvolvimento de <i>Maturity Grids</i> . Enfoca amplamente quaisquer tipos de Grids de Qualidade, não focando especificamente em SPCMMs. Também não possui foco de aquisição do conhecimento.	
1.5 Público-alvo		
2) Desenvolvimento		
2.1) Áreas de processo		
2.1.1 Lógica Seleção		
2.2) Rotulagem e número		
2.2.1 Identificando razão		
2.2.2 Definindo rótulos		
2.2.3 Numeração		
2.2.4 Definição de layout		
2.3) Descrições das Células	Define passos bem fundamentados em análise de desenvolvimento de <i>Maturity Grids</i> . Enfoca amplamente quaisquer tipos de Grids de Qualidade, não focando especificamente em SPCMMs. Também não possui foco de aquisição do conhecimento.	

	2.3.1 Prescritiva ou descritiva 2.3.2 Fonte de informação 2.3.4 Mecanismo de Formulação 2.5) Mecanismo de administração 2.5.1 Foco no processo 2.5.2 Foco em resultados finais	
	3) Avaliação 3.1) Validação 3.2) Verificação 3.3) Refinamento iterativo	
	4) Manutenção 4.1) Áreas de processo 4.2) Descrições das Células 4.3) Armazenamento de dados	
ISO (2009)	1) Percepção da necessidade 1.1) Proposta 1.2) Preparação	Trata-se de um processo genérico para desenvolvimento de normas que não incorpora aspectos específicos de SPCMMs ou aquisição de conhecimento. Entretanto, é um importante referencial de obtenção de consenso entre especialistas de um domínio de conhecimento.
	2) Construção de consenso 2.1) Comitê 2.2) Inquérito	
	3) Aprovação final e a geração do <i>draf</i> 3.1) Aprovação 3.2) Publicação	

3.4 Discussão

Neste capítulo são apresentados os conceitos fundamentais da Engenharia do Conhecimento e da Engenharia de Software no âmbito da aquisição de conhecimento no desenvolvimento e customização de *Software Process Capability/Maturity Models* (SPCMMs) como representação de conhecimento na forma de melhores práticas. Também é apresentada a revisão da literatura sobre quais são os SPCMMs existentes e como eles têm sido desenvolvidos.

A partir deste estudo dos conceitos fundamentais e da revisão sistemática da literatura, de forma geral pode-se observar que existe uma grande variedade de SPCMMs sendo desenvolvidos atualmente, demonstrando uma real tendência na customização de SPCMMs para domínios específicos de desenvolvimento de software. Também é possível observar que, apesar desta tendência, pouca documentação

detalhada sobre quais abordagens têm sido utilizadas no desenvolvimento destes modelos está disponível na literatura.

Tanto as experiências relatadas de desenvolvimento de SPCMMs identificadas por meio do *survey* quanto as propostas atualmente existentes que procuram abordar o tema, não cobrem totalmente o processo de customização dos SPCMMs para domínios específicos. Enquanto algumas focam no desenvolvimento de SPCMMs sem abordar o suporte a sua manutenção, outras não tratam da customização das melhores práticas dos SPCMMs.

Da mesma forma, não foram encontradas referências indicando que um enfoque de aquisição de conhecimento tenha sido aplicado na customização das melhores práticas de SPCMMs genéricos para domínios específicos até o momento. Portanto, pode-se observar a falta de abordagens que ofereçam apoio sistemático para a customização de SPCMMs enquanto representações de conhecimento na forma de melhores práticas.

No intuito de preencher esta lacuna, o próximo capítulo trata do desenvolvimento de um método de aquisição de conhecimento para customização SPCMMs para domínios específicos.

4 MÉTODO DE AQUISIÇÃO DE CONHECIMENTO PARA CUSTOMIZAÇÃO DE SPCMMs

Este capítulo apresenta a proposta de um método de aquisição de conhecimento para customização de modelos de capacidade/maturidade de processos de software para domínios específicos. Nas seções seguintes o desenvolvimento do método é apresentado, seguindo a definição da sua estrutura, uma visão geral das fases e atividades do método e também o detalhamento de uma atividade como exemplo.

4.1 Desenvolvendo um método para engenharia do conhecimento

Método, sob a perspectiva da ciência, consiste em um conjunto de atividades sistemáticas e racionais que permite alcançar um objetivo (LAKATOS & MARCONI, 1995). No contexto de metodologias de KE um método representa um dos seus componentes. Uma metodologia de KE é composta por uma série de elementos, conforme demonstra a pirâmide metodológica apresentada em (SCHREIBER et al., 2000). A **visão de mundo** são os *slogans*, fundamentados num conjunto de princípios que formam a linha de base da metodologia. Esta visão de mundo é fundamentada nas **teorias** que fornecem os conceitos essenciais para o estabelecimento da metodologia. Os **métodos** e as **ferramentas** fornecem o essencial para possibilitar a aplicação prática da metodologia. O **uso** desta metodologia produz o *feedback* que realimenta as demais “camadas” da pirâmide e possibilita a evolução da metodologia. A Figura 21 apresenta a pirâmide metodológica.



Figura 21: Pirâmide metodológica

Fonte: (SCHREIBER et al., 2000)

O método de aquisição de conhecimento para customização de SPCMMs desenvolvido neste trabalho e detalhado na seção seguinte encaixa-se nesta pirâmide metodológica, agregando suporte ao uso da metodologia. Os fundamentos para o desenvolvimento deste método são:

- As experiências de desenvolvimento/customização de SPCMMs relatadas na literatura (WANGENHEIM et al., 2010b) e obtidas a partir de um *survey* com autores dos modelos (WANGENHEIM et al., 2010a); e
- Os processos e técnicas de aquisição de conhecimento da KE (HUA, 2008; SCHREIBER et al., 2000);
- O framework de método PRO2PI-MFMOD (SALVIANO et al., 2009);
- O framework de processo de desenvolvimento de modelos de maturidade proposto em (BRUIN & ROSEMAN, 2005);
- As fases e etapas do processo de desenvolvimento de normas da *ISO International Standard* (ISO, 2010) e o processo de desenvolvimento de normas da IEEE (IEEE, 2010).

A Figura 22 ilustra como, seguindo os passos da pesquisa definidos neste trabalho, o método é desenvolvido a partir destas fontes.

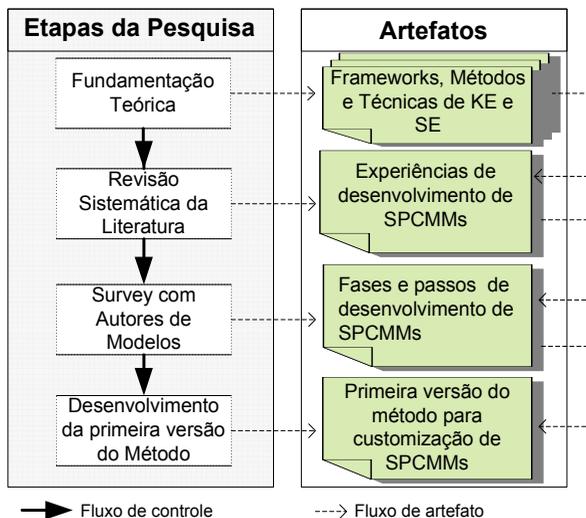


Figura 22: Desenvolvimento do método

Conforme pode ser observado na Figura 22, a fundamentação teórica identifica os frameworks, métodos e técnicas de aquisição de conhecimento. Tendo esses por base, a revisão sistemática da literatura aponta as primeiras impressões sobre como os SPCMMs atuais têm sido desenvolvidos. O survey com os autores dos SPCMMs completa as primeiras impressões adicionando detalhes sobre os métodos e técnicas utilizados no desenvolvimento de SPCMMs, incluindo os processos genéricos de desenvolvimento de normas. Com base nestes artefatos, o resultado deste processo é a primeira versão do método, que é detalhada nas seções seguintes.

4.2 Estrutura do método

Nesta seção a estrutura do método desenvolvido é apresentada. Um método enquanto conjunto de atividades sistemáticas que permite alcançar um objetivo pode ser representado na forma de um processo. Neste sentido, Benali (1990), Finkelstein (1994) e Acuña et al. (2000) definem os principais elementos que compõem a descrição de um processo, de forma que ele possa ser adequadamente modelado.

Os elementos que tipicamente compõem a descrição de um processo são (ACUÑA et al, 2000; BENALI, 1990; FINKELSTEIN, 1994):

- **Papel e Grupo:** é o conjunto de responsabilidades de um ator ou de um grupo, necessárias para se realizar determinada atividade. Um papel é decorrente da associação entre os atores e as atividades realizadas por eles;
- **Atividade:** é o estágio ou ação que produz modificações externamente visíveis no estado de um produto. Uma atividade pode ter uma entrada, uma saída e alguns resultados intermediários, denominados produtos ou artefatos;
- **Tarefa:** as tarefas consistem na realização específica das atividades descritas no método. Possuem uma granularidade mais fina;
- **Artefato ou Produto de Trabalho:** são as entradas e saídas de uma atividade durante o processo. Um artefato de um processo pode ser utilizado por outro processo para produzir outros artefatos. Estes artefatos produzidos e consumidos ao longo do processo podem ter também longos ciclos de vida, sendo criados, acessados e modificados.

Além destes componentes básicos da representação de um processo, a estrutura do método apresenta outros elementos específicos:

- **Fase:** representa um conjunto de atividades agrupadas em etapas, coletadas a partir da experiência prática de desenvolvimento de SPCMMs (WANGENHEIM et al., 2010a) e representadas conforme as fases típicas de KE (SCHREIBER et al., 2000);
- **Técnica:** servem de apoio à realização das atividades e tarefas. São utilizadas principalmente as técnicas de aquisição de conhecimento (HUA, 2008; SCHREIBER et al., 2000);
- **Passos:** são detalhamentos das técnicas, representando uma seqüência de ações em alto nível para a sua execução;

- **Ferramenta:** consiste no suporte necessário para a execução das atividades, tarefas e técnicas. O apoio ferramental é considerado essencial para a realização de determinadas tarefas e atividades do método;
- **Seção:** parte descritiva de um artefato. Auxilia na definição da estrutura de um artefato ou produto de trabalho;
- **Nota:** somente complementam a descrição das atividades, acrescentando outros detalhes.

A Figura 23 apresenta a estrutura do método utilizando a notação de um diagrama de classes UML – *Unified Modeling Language* (Linguagem de Modelagem Unificada) (OMG, 2007), contendo as classes, atributos e relações.

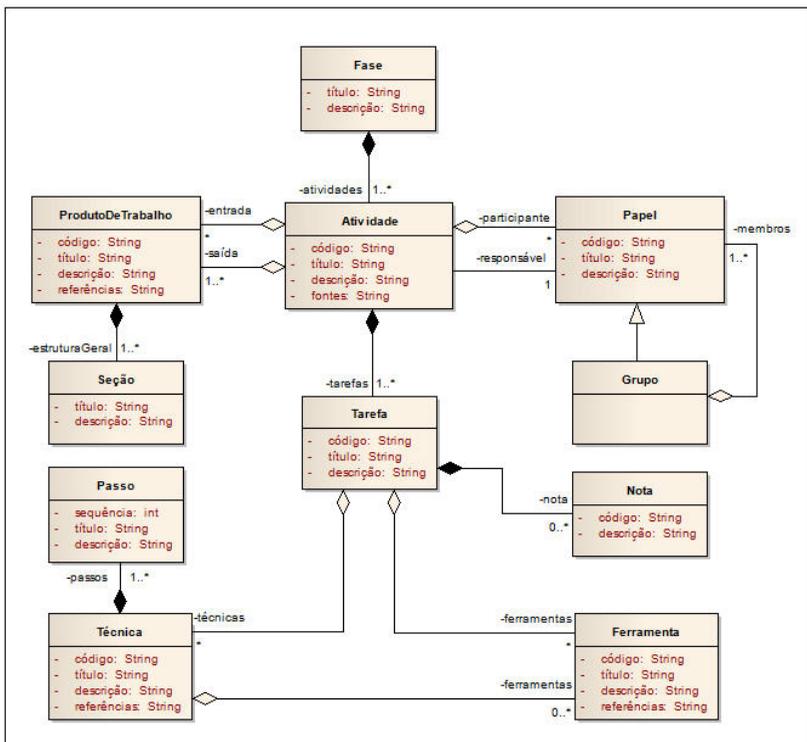


Figura 23: Estrutura de definição do método

O método é, assim, estruturado em: fases, atividades, tarefas, notas, papéis, grupos, técnicas, produtos de trabalho e ferramentas (Figura 23). As atividades são o conceito central do método sendo realizadas por papéis (ou grupos de papéis) e produzindo ou consumindo produtos de trabalho quando executadas. Cada atividade é formada por uma série de tarefas e cada tarefa se utiliza de ferramentas e técnicas. As técnicas, por sua vez, são formadas por uma série de passos. Atividades podem conter notas, que adicionam outras relações externas ao método, informando compatibilidade com normas e outros modelos, por exemplo.

O relatório técnico que descreve o método apresenta formatação semelhante à exibida na Figura 24 (HAUCK, WANGENHEIM & WANGENHEIM, 2011). A figura exhibe o original em língua inglesa com explicações de cada parte em português. Para cada atividade que compõe o método são indicados o código e o título, seguidos de uma pequena descrição. Na seção seguinte são apresentados os papéis envolvidos na execução da atividade, sendo que o papel responsável é indicado com um sinal de (*). No corpo principal da descrição da atividade encontram-se as tarefas que formam a atividade, seqüencialmente apresentadas, incluindo também código e descrição. Para cada tarefa ainda podem ser indicadas as técnicas e ferramentas aplicáveis. Ainda no corpo das tarefas, notas indicam o alinhamento da atividade a normas de qualidade de processo de software, quando aplicável.

Na seção seguinte do método os artefatos produzidos e consumidos pela execução da atividade são apresentados na forma de produtos de trabalho de entrada e saída, sendo todos codificados. As técnicas e ferramentas utilizadas em toda a execução da atividade são apresentadas na seção seguinte. Por fim, são indicadas as referências que fornecem suporte, embasamento ou exemplos de execução da atividade, encontrados na literatura.

Activity	A2.1 - Develop the design/architecture of the model	Código e título
Description	Identifying the main elements of the model. Typically, within customizations, the structure of one of the principal source models is adopted. The standard ISO/IEC 15504 establishes a general structure for model design. This structure includes a Process Reference Model and a Process Assessment Model. Therefore, the structure of those models has to be analyzed and if necessary, modified appropriately in order to satisfy the requirements of the project.	Descrição
Roles	* GR03 - Operational Team RL03 - Technical Reviewer GR04 - Advisory Board	Papéis envolvidos na execução da atividade.
Tasks	2.1.1 Choose the representation. One of the first decisions is about the representation. There are two types of representation: Continuous and Staged [2.1-2]: (i) A model with the continuous representation is a model that is developed and maintained as a single, continuous artifact. (ii) A model with the staged representation is a model that is developed and maintained as a series of artifacts, each representing a different view of the model. 2.1.5 Review Architecture. The architecture is reviewed by a Technical Reviewer. If there is some incoherence, they are corrected. A technique can be used to obtain coherence: Delphi [TE07] Checklist-Based Reading [TE21]	Divisão da atividade em tarefas, contendo a descrição de cada uma.
	Notas indicando o alinhamento a normas de qualidade de processo.	
	NOTE 2.1-1: This activity provides coverage of ISO/IEC 15504-2 6.2.4-(a,c); 6.3.2.1; 6.3.2.2; 6.3.2.3-(a,b,c); 6.3.4-(a,b); 6.3.5]	
Work Products	Input WP01 - Domain Analysis Report Output WP04 - Process Reference Model (with the layout defined) WP05 - Process Assessment Model (with the layout defined)	Produtos de trabalho consumidos e produzidos na execução da atividade
Techniques and Tools	TE07 - Delphi TE17 - Perspective-Based Reading TE21 - Checklist-based Reading	Lista de técnicas e ferramentas utilizadas
Sources	[2.1-1] Maier A. M., Moultrie J. and Clarkson, P. J. "Developing maturity grids for assessing organizational capabilities: a practitioner guidance." Proceedings of 4th International Conference on Management (MCD09), 2009. [2.1-2] Software Engineering Institute, "CMMI for Development, version 1.3." Technical Report, Software Engineering Institute, 2010.	Referências que suportam a atividade ou dão exemplos de sua execução

Figura 24: Apresentação do método

Diversos são os papéis envolvidos na customização de um SPCMM para um domínio específico. No método são definidos os papéis que são envolvidos direta ou indiretamente na execução das atividades (vide Figura 25). As responsabilidades são atribuídas a todos os papéis na definição de atividades nas quais eles são os principais responsáveis e/ou participantes. Na maioria dos casos uma mesma pessoa pode desempenhar papéis diferentes simultaneamente. A exceção ocorre quando um papel é responsável por revisar o trabalho realizado pelo outro. Nestes casos, cada trabalho deve ser executado por pessoas diferentes. Na descrição do método procurou-se que as características definidas para cada papel não fosse demasiado prescritiva, deixando espaço para a interpretação de acordo com cada caso. A estrutura geral dos papéis é apresentada utilizando a notação UML com estereótipos

SPEM – *Software Process Engineering Metamodel* (Meta-modelo de Engenharia de Processos de Software) (OMG, 2008) na Figura 25.

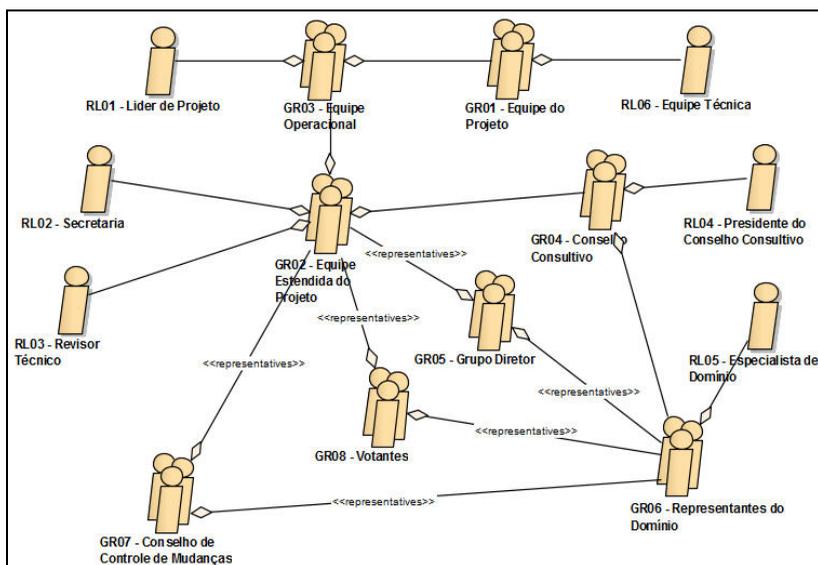


Figura 25: Papéis envolvidos no método

4.3 Detalhamento do método

Nesta seção o método de aquisição de conhecimento é apresentado e suas fases e atividades são brevemente descritas. O método proposto é composto por cinco fases baseadas no ciclo de vida de KE (considerando a aquisição de conhecimento): (i) Identificação do Conhecimento, (ii) Especificação do Conhecimento, (iii) Refinamento do Conhecimento, (iv) Uso do Conhecimento, e (v) Evolução do Conhecimento (vide Figura 26). Cada fase é composta por um conjunto de atividades que podem não ser necessariamente executadas em seqüência.

Método de Aquisição de Conhecimento para Customização de SPCMMs				
Fase 1: Identificação do Conhecimento	Fase 2: Especificação do Conhecimento	Fase 3: Refinamento do Conhecimento	Fase 4: Uso do Conhecimento	Fase 5: Evolução do Conhecimento
1.1 - Familiarizar-se com o Domínio	2.1 - Desenvolver a arquitetura do modelo	3.1 - Avaliar o modelo <i>draft</i>	4.1 - Fornecer suporte ao uso do modelo	5.1 - Administrar as solicitações de mudança
1.2 - Identificar fontes de conhecimento	2.2 - Analisar e integrar modelos relacionados	3.2 - Consolidar o modelo	4.2 - Validar o modelo em uso	5.2 - Confirmação, revisão ou rejeição
1.3 - Definir escopo e objetivos	2.3 - Desenvolver o modelo <i>draft</i> - dimensão de processo	3.3 - Votar o modelo consolidado		
1.4 - Formalizar o grupo de trabalho	2.4 - Desenvolver o modelo <i>draft</i> - dimensão de capacidade/ maturidade	3.4 - Aprovar o modelo		
		3.5 - Publicar		

Figura 26: Fases e Atividades do Método

Somente uma breve descrição de cada atividade é apresentada nesta seção, mas a descrição detalhada de todas as atividades está disponível no relatório técnico do modelo em (HAUCK, WANGENHEIM & WANGENHEIM, 2011). A seguir cada uma das fases e atividades do método apresentado na Figura 26 é resumidamente descrita:

Fase 1: Identificação do Conhecimento

O principal objetivo da fase 1 é obter uma familiarização com o domínio e uma caracterização do contexto para o qual o SPCMM será customizado. As atividades relacionadas são:

Atividade 1.1 – Familiarizar-se com o domínio: Consiste em uma contextualização do domínio para o qual o modelo será desenvolvido. Uma análise da literatura relacionada ao domínio provê, em primeiro lugar, um profundo entendimento de exatamente o que é o domínio e as

suas características, fornecendo definições dos principais conceitos e da terminologia, bem como a identificação do processo subjacente geral.

Atividade 1.2 - Identificar as fontes de conhecimento: que serão usadas como insumo para o desenvolvimento do modelo SPCMM. Fontes de conhecimento importantes consistem em: recursos humanos, normas relacionadas a desenvolvimento de software específicas do domínio, SPCMMs genéricos, ou relatórios/documentos que identifiquem, por exemplo, aspectos de qualidade e desempenho que sejam relevantes para o domínio. A identificação de fontes humanas requer a definição do perfil dos agentes de conhecimento, que neste contexto significa descrever os especialistas do domínio de desenvolvimento de software. Também é necessário identificar quais SPCMMs genéricos serão customizados para o domínio específico. A escolha depende do quão expressivo cada SPCMM é para o domínio/setor alvo, em termos de confiabilidade, aplicabilidade e impacto no mercado.

Atividade 1.3 - Definir escopo e objetivos: do modelo a ser desenvolvido. O escopo do SPCMM deve definir com precisão os limites do domínio de aplicação, e definir, sem ambigüidade, o objeto do modelo e os aspectos abrangidos, indicando os limites da sua aplicabilidade ou de partes específicas do mesmo. É também importante a identificação dos objetivos específicos que devem ser atingidos pelo SPCMM, determinando as metas e os interesses que podem ser afetados.

Atividade 1.4 - Formalizar o grupo de trabalho: para o desenvolvimento do modelo. Isso inclui a definição da atribuição de um patrocinador / coordenador e as regras e procedimentos de trabalho que serão utilizados durante o desenvolvimento do novo modelo. Também inclui o convite de todos os interessados a participar e define quem tem o direito de voto para aprovar o modelo dentro do grupo de trabalho, que pode fazer solicitações de mudança e que tem a capacidade de contribuir com o desenvolvimento do modelo.

Fase 2: Especificação de Conhecimento

Durante esta fase central, uma primeira versão do SPCMM é desenvolvida. As atividades propostas para esta fase são:

Atividade 2.1 - Desenvolver a arquitetura do modelo: identificar os principais elementos do modelo. A norma ISO/IEC 15504 estabelece uma estrutura geral para o design de um modelo. Esta estrutura inclui um Modelo de Referência de Processo e um Modelo de Avaliação de Processo. Normalmente, em customizações, a estrutura de um dos principais modelos base é adotada. Portanto, a estrutura desses modelos tem de ser analisada e, se necessário, modificada adequadamente. A definição da estrutura do modelo é um passo essencial, pois dela depende como as melhores práticas de desenvolvimento de software para o domínio serão representadas no SPCMM desenvolvido.

Atividade 2.2 - Analisar e integrar os modelos relacionados: uma vez especificados o escopo e da arquitetura do SPCMM, os modelos relevantes são definidos e analisados. Isso geralmente envolve um mapeamento dos modelos relacionados e/ou um esforço de harmonização ou integração dos modelos existentes em um modelo unificado. O mapeamento ou harmonização é um processo semântico complexo, pois envolve o entendimento profundo de todos os modelos envolvidos e a resolução de possíveis disparidades terminológicas de forma que as melhores práticas contidas nesse modelo possam ser adequada e inequivocamente representadas.

Atividade 2.3 - Desenvolver o modelo *draft* - dimensão de processo: nesta atividade principal, a dimensão de processo de SPCMM é desenvolvida. Definir a dimensão de processo do SPCMM implica em identificar os processos relevantes que contêm as melhores práticas para o domínio específico. Para identificar os processos relevantes, é necessário identificar quais são as necessidades de qualidade de software mais importantes para o domínio. Isso pode ser realizado por meio da extração do conhecimento dos agentes de conhecimento do domínio (identificados na fase 1). Esta extração é realizada utilizando-se diversas técnicas, tais como: entrevistas, *survey*, engenharia de ontologias, grupos focais, etc., individualmente ou combinadas, de forma iterativa e incremental. Então, num próximo passo, é necessário

relacionar esses atributos qualitativos aos processos pertinentes. Uma versão adaptada do QFD - *Quality Function Deployment*, que envolve também especialistas de SPI, pode ser usada para mapear sistematicamente as necessidades de qualidade com processos, os resultados/melhores práticas necessárias e os produtos de trabalho típicos. O mapeamento dos modelos relacionados realizado na Atividade 2.2 é utilizado para apoiar a customização do SPCMM como uma base de re-utilização dos processos existentes, adaptando-os ou completando-os com novos processos quando necessário.

Atividade 2.4 - Desenvolver o modelo *draft* – dimensão de capacidade/maturidade: a fim de produzir um modelo que possa servir de referência para avaliação do processo, a dimensão de capacidade / maturidade é desenvolvida. Portanto, é necessário definir os atributos de processo e agrupá-los em níveis de capacidade. Isto significa definir os atributos aplicáveis a todos os processos que descrevem uma faceta da capacidade global de alcançar a finalidade do processo e pode ser avaliada em uma escala de desempenho, fornecendo uma medida da capacidade do processo. Os níveis de capacidade podem ser definidos como conjuntos de atributos que trabalham juntos para fornecer a melhoria na capacidade de realizar o processo. Como exemplo, a ISO/IEC 15504 define seis níveis de capacidade. Se for adequado, os processos também podem ser agrupados em níveis de maturidade, a fim de definir uma dimensão de maturidade organizacional, seguindo a ordem de prioridade definida pela priorização das necessidades de qualidade obtidas dos agentes de conhecimento. Novamente, a dimensão de capacidade/maturidade dos modelos base pode ser utilizada como modelo, e sendo adaptada se necessário. Como resultado desta fase um modelo *draft* é desenvolvido.

Fase 3: Refinamento do Conhecimento

Nesta fase, o modelo *draft* é validado, votado e refinado para desenvolver um modelo aprovado pela maioria da comunidade do domínio.

Atividade 3.1 - Avaliar o modelo *draft*: o modelo *draft* é validado, a fim de demonstrar que ele cumpre com as características gerais esperadas de um SPCMM. Nesta etapa, tal validação é

normalmente baseada em um consenso das partes interessadas na revisão do modelo. Várias técnicas podem ser utilizadas, incluindo *Expert Panel*, *Delphi*, etc.

Atividade 3.2 – Consolidar o modelo: Com base no *feedback* obtido, o modelo *draft* é evoluído de forma iterativa, até que seja alcançado um consenso entre os membros do grupo de trabalho. Isto requer: a discussão, negociação e resolução de divergências técnicas significativas a fim de produzir um modelo que seja aceito e amplamente utilizado pela comunidade do domínio.

Atividade 3.3 – Votar o modelo consolidado: Durante esta atividade, o modelo desenvolvido é distribuído, e os interessados votam na aprovação ou rejeição do modelo. Eventuais mudanças solicitadas são analisadas e, quando apropriado, implementadas para obter-se o consenso necessário para a aprovação. Somente melhores práticas que sejam amplamente aceitas como capazes de atender às necessidades de qualidade no domínio têm significado para o domínio e, portanto, devem fazer parte do SPCMM.

Atividade 3.4 - Aprovar o modelo: critérios claros para a aprovação devem ser definidos, bem como procedimentos para o que acontece após a aprovação ou não aprovação. Se necessário, revisões do modelo são repetidas até que o modelo seja aprovado. O ponto chave é obter-se a aprovação da comunidade para que o modelo seja efetivamente utilizado quando disponibilizado.

Atividade 3.5 – Publicar o modelo: o modelo resultante é então disponibilizado em um lugar acessível para a comunidade de interesse no domínio. Uma estrutura adequada é disponibilizada de forma que o modelo possa ser facilmente acessado pela comunidade de interesse.

Fase 4: Uso do Conhecimento

Após a publicação, o modelo está sendo colocado em uso e os resultados da sua utilização são coletados e analisados.

Atividade 4.1 – Suporte à utilização do modelo: é necessário definir que tipo de apoio será fornecido para os usuários do modelo, tais como: treinamento, fóruns de usuários, etc. As políticas de suporte ao modelo são definidas e no mesmo ambiente onde o modelo foi publicado, a criação de um fórum web é importante para manter ativa a comunidade de desenvolvimento do SPCMM. Também a infra-estrutura para receber sugestões de melhoria e manter um contato vivo com a comunidade de usuários do modelo é planejada e implementada.

Atividade 4.2 – Validar o modelo em uso: a fim de validar o modelo baseado em seu uso na prática, um *framework* para a sua validação é definido, e os dados coletados e analisados. Os resultados irão complementar aqueles obtidos da validação por especialistas que foi realizada anteriormente e podem ser utilizados para desenvolver versões futuras do modelo. É somente neste momento, a partir do uso do SPCMM, que é possível avaliar se o modelo realmente atinge os seus objetivos e se as melhores práticas representadas no modelo realmente correspondem às melhores práticas de desenvolvimento de software necessárias para o domínio.

Fase 5: Evolução do conhecimento

Devido a várias razões, os SPCMMs evoluem constantemente (amadurecimento do conhecimento do domínio, avanços tecnológicos, etc.) Por isso, é necessário também prestar apoio metodológico para a evolução contínua do modelo, uma vez que o modelo esteja em uso no domínio para o qual foi destinado.

Atividade 5.1 – Gestão de solicitações de mudança: é necessário definir como as solicitações de mudança das diferentes partes interessadas serão coletadas de forma sistemática e como eles são gerenciadas.

Atividade 5.2 - Confirmação, revisão ou revogação: o grupo de desenvolvimento do modelo define como as mudanças serão analisadas e como novas versões do modelo serão publicadas. Cada conjunto de alterações necessárias deve seguir todo o processo da fase 3 para

possibilitar a validação das mudanças. Este processo deve ser suportado por um processo regular de gerenciamento de configuração.

4.3.1 Detalhamento de uma atividade

Com o objetivo de descrever o processo de aquisição de conhecimento para a customização de SPCMMs, o método apresenta cada uma das atividades descritas acima de forma detalhada. Conforme definido na estrutura do método, a descrição de uma atividade inclui: o título da atividade, uma breve descrição, a identificação dos papéis responsáveis e envolvidos, as tarefas, produtos de trabalho, técnicas e ferramentas e as fontes. O detalhamento de todas as atividades está disponível em (HAUCK, WANGENHEIM & WANGENHEIM, 2011). Nesta seção, como exemplo de descrição de atividade, é apresentada atividade A1.2 – *Identificar Fontes de Informação* por conter todos os tipos de elementos definidos para uma atividade no método. O Quadro 11 apresenta em detalhes a atividade.

Quadro 11: Detalhamento da atividade A1.2

Atividade	A1.2 - Identificar Fontes de Informação
Descrição	Identificar fontes de informação que serão utilizadas como insumo para o desenvolvimento do modelo. Fontes de informação importantes consistem em: recursos humanos, normas específicas para desenvolvimento de software no domínio, SPCMMs genéricos, ou de relatórios/documentos que identifiquem, por exemplo, aspectos importantes para de qualidade/desempenho para o domínio. A identificação de fontes humanas requer a definição do perfil dos agentes de conhecimento, que neste contexto significa descrever especialistas de desenvolvimento de software no domínio. O objetivo desta atividade é identificar e não analisar as fontes de informação, que será realizado em outras atividades.
Papéis	* GR01 - Equipe do Projeto GR06 - Representantes do Domínio
Tarefas	<p>1.2.1 <i>Definir o perfil dos especialistas.</i></p> <p>Consiste em estabelecer o perfil esperado para os agentes de conhecimentos que serão futuramente listados. Um perfil de especialista define quem vai ser considerado um especialista para o desenvolvimento do SPCMM. Isso pode incluir: <i>background</i> (academia, indústria, etc.), representatividade (uma média de: experiência, formação e qualificações), o papel na organização e participação no grupo.</p> <p>Técnicas Aplicadas:</p> <p style="padding-left: 40px;">Seleção de Especialistas [TE06] (passo um).</p> <p>1.2.2 <i>Identificar os Especialistas de Domínio.</i></p> <p>Com o perfil de especialistas criado, é hora de identificá-los.</p>

Estes terão um papel fundamental no desenvolvimento do modelo. O número de especialistas pode variar de acordo com o escopo do problema e dos recursos disponíveis, mas é importante que a comunidade de interesse seja devidamente representada, especialmente em termos de representatividade. Os peritos devem ser escolhidos por seu trabalho e credibilidade junto à comunidade de interesse.

Técnicas Aplicadas:

Seleção de Especialistas [TE06]

1.2.3 Identificar as normas de desenvolvimento de software relacionadas ao domínio.

Com base nos resultados da familiarização com o domínio (consulte Atividade A1.1 - Familiarização com o domínio), as normas relacionadas com o domínio são identificadas. A opinião dos especialistas é de extrema relevância no aspecto da classificação das normas mais relevantes.

A fim de obter consenso sobre as normas relevantes relacionadas ao domínio, as seguintes técnicas podem ser aplicadas:

Delphi [TE07]

Focus Groups [TE12]

1.2.4 Identificar os SPCMMs genéricos.

Consiste em identificar quais SPCMMs genéricos serão utilizados como base para o desenvolvimento do novo modelo. É importante justificar porque serão utilizados. O parecer dos especialistas deve ser consultado para identificar e priorizar esses modelos genéricos.

A fim de obter consenso sobre SPCMM (s) genéricos, as seguintes técnicas podem ser aplicadas:

Delphi [TE07]

Focus Groups [TE12]

1.2.5 Identificar outras fontes de literatura relevantes.

Com base nos resultados da familiarização do domínio (consulte Atividade A1.1 – Familiarização com o domínio), esta tarefa consiste em descobrir outras fontes de informações na literatura, incluindo trabalhos relevantes, relatórios ou livros que identifiquem importantes aspectos de qualidade/desempenho para o domínio.

As técnicas também podem ser aplicadas a fim de obter o consenso sobre a literatura pertinente:

Delphi [TE07]

Focus Groups [TE12]

| NOTE 1.2-1: Esta atividade prove cobertura dos requisitos da norma ISO/IEC 15504-2 6.2.3.2|

Produtos de Trabalho	Entrada: WP01 - Relatório de Análise do Domínio
----------------------	--

	Saída: WP01 - Relatório de Análise do Domínio com as fontes de informação identificadas WP03 – Matriz de Rastreabilidade (com as fontes de informação)
Técnicas e Ferramentas	TE06 - Seleção de Especialistas TE07 - Delphi TE12 - Focus Groups
Fontes	[1.2-1] Schreiber, G.; Akkermans, H.; Anjewierden A.; De Hoog, R., Shadbolt, N.; Van De Velde, W; Wielinga, B. "Knowledge Engineering and Management - The CommonKADS Methodology." The MIT Press, 2000. [1.2-2] Bruin T., Rosemann M. "Using the Delphi Technique to Identify BPM Capability Areas." In 18th Australasian Conference on Information Systems, pp. 42, 2007. [1.2-3] Bovee M. et al. "A Framework for Assessing the Use of Third-Party Software Quality Assurance Standards to Meet FDA Medical Device Software Process Control Guidelines." IEEE Transactions on Engineering Management, vol. 48, no. 4, pp. 465-478, 2001. [1.2-4] Salviano C. F. et al. "A Method Framework for Engineering Process Capability Models." 16th European Systems and Software Process Improvement and Innovation, Alcala, Spain, 2009. [1.2-5] Matook S.; Indulska, M. "Improving the Quality of Process Reference Models: A Quality Function Deployment-Based Approach." Decision Support Systems, v. 47, 2009.

Além das demais atividades, o método também inclui o resumo dos passos indicados para cada uma das técnicas citadas (e.g. TE07 - *Delphi*). Um exemplo de técnica é apresentado na seção seguinte.

4.3.2 Detalhamento de uma técnica

No método de aquisição de conhecimento para customização de SPCMMs, uma técnica é detalhada em termos de: título, descrição, passos a serem executados, atividades que utilizam essa técnica e referências. A técnica TE07 - *Delphi*, citada na atividade apresentada na seção anterior, é detalhada aqui como exemplo de como uma técnica é descrita. O Quadro 12 apresenta os detalhes da técnica.

Quadro 12: Detalhamento da técnica TE07

Técnica	TE07 - Delphi
Descrição	A técnica Delphi constitui-se em uma série de rodadas seqüenciais, intercaladas por comentários mediados visando obter o mais confiável consenso da opinião de um painel de especialistas [TE07-1].
Passos Gerais	1. Definir o objetivo do estudo

É necessário identificar claramente o objetivo do estudo, a fim de fornecer uma base de avaliação e tornar possível verificar se o objetivo do estudo foi alcançado.

2. Determinar a condição de parada

Estudos demonstraram que três ou quatro iterações podem ser suficientes para a obtenção de convergência, mas para qualquer caso o limite de interações deve ser determinado em função de diversos fatores como a disponibilidade de recursos ou limitações de tempo. Outra condição de parada pode ser o alcance de uma convergência aceitável (tais como média, desvio e / ou variação).

3. Seleção dos Especialistas

Veja TE06 - Seleção de Especialistas

4. Definir o instrumento de coleta de dados e métodos de pontuação

O instrumento de coleta de dados mais utilizado tem sido questionários, mas outros podem ser utilizados, como: planilhas, páginas web, etc. O método de pontuação que vai ser utilizada para quantificar as contribuições dos especialistas deve ser definido, bem como os métodos de agregação utilizados para avaliar os pareceres dos especialistas.

5. Coletar dados

Os dados são coletados e agrupados por painel (ver TE06 – Seleção de Especialistas) e no total.

6. Verificar a convergência

O mediador calcula a convergência com o sistema de pontuação e métodos de agregação definidos e compara com a convergência esperada. O resultado é enviado para todos os especialistas.

7. Repetir até que a convergência seja obtida ou o número de interações seja alcançado

Se não houver convergência suficiente ou o número de interações seja alcançado, o instrumento de coleta de dados é enviado novamente para todos os peritos, incluindo as novas informações obtidas a partir da interação anterior.

8. Documentar os resultados

Após o término das interações todos os resultados devem ser documentados, incluindo os resultados do estudo e os processos utilizados para obtê-lo. Também é importante para documentar o sistema de pontuação e os métodos de agregação utilizados.

Atividades Afetadas

A1.2 - Identificar as fontes de informação

A3.1 - Validar o *draft* do modelo

Referências

[TE07-1] Powell C. "The Delphi technique: myths and realities." Journal of Advanced Nursing, 41(4), 376-382, Blackwell Publishing, 2003.

[TE07-2] Okoli C., Pawlowski S. D." The Delphi method as a research tool: an example, design considerations and applications." Inf. Manage, pp. 42, 2004

[TE07-3] Bruin T., Rosemann M. "Using the Delphi Technique to Identify BPM Capability Areas." In 18th Australasian Conference on Information Systems, 42, 2007.

Além da descrição detalhada destes elementos, os papéis desempenhados são também descritos, conforme o exemplo apresentado a seguir.

4.3.3 Detalhamento de um papel

Cada papel é detalhado no método na forma de: título, descrição e responsabilidades. As responsabilidades de um papel incluem as atividades pelas quais ele é responsável e aquelas nas quais ele só participa. O Quadro 13 mostra como o papel RL01 - Líder do Projeto é definido no método.

Quadro 13: Detalhamento do papel RL01

Papel	RL01 - Project Leader
Descrição	Principal responsável pelo projeto de desenvolvimento do SPCMM. É indicado pelo patrocinador do projeto, quando aplicável.
Responsabilidades	
(*) Atividades Responsável	A1.3 - Definir escopo e objetivos A1.4 - Formalizar o grupo de trabalho
Atividades Participante	A1.1 - Familiarização com o domínio

O documento completo do método na forma de um relatório técnico está disponível em (HAUCK, 2011).

4.4 Comentários do Capítulo

Este capítulo apresenta o Método de Aquisição de Conhecimento para Customização de SPCMMs, incluindo a sua estrutura e um resumo do conteúdo do mesmo. Também o detalhamento de alguns de seus elementos é apresentado como exemplo.

O método é desenvolvido com base nas experiências de desenvolvimento de SPCMMs relatadas na literatura, nos processos e técnicas de aquisição de conhecimento, processos de desenvolvimento de normas de qualidade e em frameworks de desenvolvimento de modelos de qualidade de processo. Como resultado, diversos elementos

compõem a estrutura o método: atividades, tarefas, notas, técnicas, ferramentas, papéis e referências.

O método desenvolvido representa uma primeira proposta para aquisição de conhecimento na customização de SPCMMs para domínios específicos de desenvolvimento de software. O método apresentado neste capítulo é avaliado por especialistas e aplicado em projetos reais de customização de SPCMMs para a sua validação. O capítulo seguinte apresenta em detalhes esta validação.

5 VALIDAÇÃO DO MÉTODO

Este capítulo apresenta a validação do método proposto de forma a avaliar a sua adequação e caracterizar a sua aplicabilidade. Conforme definido nos aspectos metodológicos desta tese, esta validação se dá por meio de duas abordagens diferenciadas: a avaliação em um *Expert Panel* e a aplicação prática em dois estudos de caso de customização de SPCMMs. Na primeira parte deste capítulo a definição da validação é detalhada, seguida pela apresentação do *Expert Panel* e das características e resultados observados nos estudos de caso. Por fim os resultados gerais da validação são então sintetizados nas considerações finais.

5.1 Definição da Validação

Para sistematicamente apoiar a definição dos objetivos da validação do método, a abordagem GQM - *Goal Question Metric* (BASILI et al., 1994) é utilizada. Resumidamente, a GQM consiste em uma abordagem de medição orientada a metas, que auxilia na definição e análise de medição, por meio da definição de metas e no desdobramento destas metas em medidas operacionalmente coletáveis (BASILI et al., 1994). Na abordagem GQM primeiramente são definidos os objetivos de medição e com base nestes objetivos são, então, definidas perguntas que, quando respondidas, atendem a estes objetivos. O passo seguinte consiste em definir, para cada pergunta, quais medidas devem ser coletadas para respondê-las. Para cada medida também são estabelecidos os meios de coleta de dados e as formas de análise e interpretação.

Desta forma, como primeiro passo na direção da validação do método são definidos os objetivos de avaliação. Estes objetivos são definidos no sentido de cobrir tanto as características intrínsecas do método quanto a observação da sua aplicação prática. A abordagem GQM estabelece um template para a definição do objetivo de medição composto por: (i) objeto: o que vai ser estudado/mensurado; (ii) propósito: para que está sendo realizada a medição (avaliar, entender, caracterizar, etc.); (iii) foco: qual característica do objeto será estudada; (iv) ponto de vista: sob qual ponto de vista o objeto será estudado, e; (v) contexto: em que local, situação, condições ambientais o estudo será realizado.

Sob este paradigma, os dois objetivos de avaliação são assim definidos:

- **Objetivo de Avaliação 1:** Avaliar a adequação do Método de Aquisição de Conhecimento para Customização de SPCMMs, sob o ponto de vista de especialistas em modelos de maturidade no contexto de um painel de especialistas.
- **Objetivo de Avaliação 2:** Caracterizar a aplicabilidade do Método de Aquisição de Conhecimento para Customização de SPCMMs, sob o ponto de vista de usuários do método no contexto do desenvolvimento do modelo 'X'.

Para o Objetivo 1, o propósito e o foco consistem em: avaliar a adequação. O termo adequação consiste na qualidade do que é “apropriado, adaptado, que corresponde perfeitamente a um objetivo [...]” (FERREIRA, 2008). Assim, o objetivo final do método em si é o de capturar o conhecimento do domínio de forma a sistematicamente produzir SPCMMs de qualidade, que realmente contenha as melhores práticas de desenvolvimento de software para o domínio. Desta forma, para avaliar se o método adequadamente proporciona o atendimento destes objetivos, é utilizado julgamento de especialistas por meio da realização de um *Expert Panel*.

Para o Objetivo 2, o propósito e o foco consistem em caracterizar a aplicabilidade. Caracterizar implica em “pôr em relevo o caráter de, distinguir: as virtudes que o caracterizam [...]” e aplicabilidade significa a qualidade daquele que “se aplica, aderente [...]” (FERREIRA, 2008). Assim, o atendimento deste objetivo de avaliação passa pela realização de estudos empíricos, onde os resultados observados possam distinguir as características do método enquanto aplicado no desenvolvimento de SPCMMs.

Após a definição dos objetivos de validação, seguindo a abordagem GQM, o passo seguinte consiste em se definir as questões (*Questions*) que, quando respondidas possibilitem o atendimento dos objetivos de avaliação definidos. Para cada questão, são também definidas as medidas que serão coletadas para responder as questões propostas.

Em relação ao Objetivo de Avaliação 1, pra que se possa avaliar a adequação do método, são derivadas as características desejadas para um método adequado ao seu objetivo. Neste sentido, Milton (2007) define que o resultado da aquisição de conhecimento deve ser: correto, completo e relevante. A estas características são acrescentadas a

consistência e a generalidade (DAVIS, HOWARD & SZOLOVITS, 1993). Assim, espera-se que um método adequado possua as seguintes características:

- **Corretude:** o quanto o método é correto, ou seja, qual é a extensão dos erros existentes;
- **Completude:** refere-se à cobertura do método, se o método é suficientemente completo;
- **Relevância:** a extensão da utilidade do método, procurando identificar o quanto o método atende às necessidades de aquisição de conhecimento para a customização de SPCMMs;
- **Consistência:** tem a intenção de identificar se o método é uniforme e possui coerência no seu conteúdo;
- **Generalidade:** indica se o método pode ser aplicado na customização de modelos para diferentes domínios de desenvolvimento de software.

Para cada uma destas características são então definidas as questões, conforme apresentado na Quadro 14:

Quadro 14: Perguntas e Medidas do Objetivo de Avaliação 1.

OBJETIVO 1:	Avaliar a adequação do Método de Aquisição de Conhecimento para Customização de SPCMMs, sob o ponto de vista de especialistas em modelos de maturidade no contexto de um painel de especialistas.
Corretude	
Perguntas	
Pergunta Q1:	O método é correto?
Medidas:	MQ1.1: Quantidade de erros identificados.
	MQ1.2: Quantidade de erros corrigidos.
Consistência	
Perguntas	
Pergunta Q2:	O método é consistente?
Medidas:	MQ2.1: Quantidade de inconsistências identificadas.
	MQ2.2: Quantidade de inconsistências corrigidas.
Completude	

Perguntas	
Pergunta Q3:	O método é completo?
Medidas:	MQ3.1: Quantidade de atividades faltantes.
	MQ3.2: Quantidade de tarefas faltantes.
	MQ3.3: Quantidade de técnicas faltantes.
	MQ3.4: Quantidade de produtos de trabalho faltantes.
Pergunta Q4:	O método contém o que é necessário para suportar a customização de SPCMMs?
Medidas:	MQ4.1: Impressão subjetiva sobre a capacidade do método de suportar a customização de SPCMMs.
Relevância	
Perguntas	
Pergunta Q5:	A utilização do método é capaz de produzir SPCMMs que realmente contenham as melhores práticas de desenvolvimento de software para o domínio?
Medidas:	MQ5.1: Impressão subjetiva sobre a capacidade do método de capturar as melhores práticas no domínio
Pergunta Q6:	O método apresenta suporte mais amplo para o desenvolvimento de SPCMMs do que os atualmente existentes?
Medidas:	MQ6.1: Impressão subjetiva sobre a completude do método em relação às demais abordagens existentes.
Pergunta Q7:	O método é genérico o suficiente para permitir a captura de conhecimento no desenvolvimento de SPCMMs para diferentes domínios de desenvolvimento de software?
Medidas:	MQ7.1: Impressão subjetiva sobre a generalidade do método.
Generalidade	
Perguntas	
Pergunta Q8:	Quais são os pontos fracos mais importantes do método?
Medidas:	MQ8.1: Pontos fracos do método
Pergunta Q9:	Quais são os pontos fortes mais importantes do método?
Medidas:	MQ9.1: Pontos fortes do método

Desta forma foram definidas 14 medidas a serem coletadas para avaliar o método sob o ponto de vista do Objetivo de avaliação 1. As medidas definidas são coletadas por meio de um questionário no contexto de um *Expert Panel*, conforme definido na seção seguinte (4.2).

Para o Objetivo de Avaliação 2 são também levantadas as questões e medidas, seguindo a abordagem GQM, conforme apresenta a Quadro 15. A coleta de dados para as três medidas que respondem às perguntas definidas para o Objetivo 2 se dá por meio da realização de estudos de caso, que é apresentado na seção 4.3.

Quadro 15: Perguntas e Medidas do Objetivo de Avaliação 2.

OBJETIVO 2:	Caracterizar a aplicabilidade do Método de Aquisição de Conhecimento para Customização de SPCMMs, sob o ponto de vista de usuários do método no contexto do desenvolvimento do modelo 'X'.
Perguntas	
Pergunta Q10:	Como o método foi aplicado?
Medidas:	MQ10.1: Impressão subjetiva da extensão da aplicação do método no desenvolvimento do modelo.
	MQ10.2: Detalhamento de como as fases foram executadas.

Estando definidos os objetivos, questões e medidas de avaliação, a validação do método se dá, então, das duas formas previstas: *Expert Panel* e Estudos de Caso. O conteúdo do método é validado por meio do *Expert Panel* (BEECHAM et al., 2005), sendo apresentado em detalhes na seção 5.2 deste capítulo. Com base no procedimento para estudos empíricos definido por Wohlin et al. (2000) são realizados dois estudos de caso de customização de SPCMMs para validar o método desenvolvido, apresentadas na seção 4.3. Como resultado obtém-se o método consolidado e avaliado para ser aplicado na customização de SPCMMs para domínios específicos.

5.2 Avaliação pelo Expert Panel

Um *Expert Panel* (painel de especialistas) é uma técnica baseada na opinião de especialistas, bastante utilizada na validação de novos métodos e técnicas (BEECHAM et al., 2005), fazendo extenso uso de *surveys* (LITWIN, 1995) para coleta de dados e da técnica *Delphi* (OKOLI & PAWLOWSKI, 2004) para obtenção de consenso. Assim,

um *Expert Panel* consiste em reunir especialistas de determinada área de conhecimento de forma a obter a sua opinião em relação a algum aspecto do objeto de pesquisa.

O valor de *Expert Panels* na avaliação de novos métodos e técnicas é observado em diversas pesquisas. Em (DYBA, 2000) onze especialistas são utilizados no processo de revisão, já em (EMAM & MADHAVJI, 1996) trinta especialistas foram responsáveis por avaliar o trabalho. Alta confiabilidade tem sido observada na utilização de *Expert Panels* por meio de análises estatísticas, também para outros fins como predição (LAUESEN & VINTER, 2001) e estimativa (KITCHENHAM et al. 2002), conforme relatado em (BEECHAM et al., 2005).

Esta seção apresenta a realização de um *Expert Panel* para validação do conteúdo do método desenvolvido nesta tese. Primeiramente são apresentados detalhes de como foram selecionados os especialistas, como foi elaborado o instrumento de coleta de dados e por fim a realização do *Expert Panel* e os resultados observados.

5.2.1 Seleção dos especialistas

A seleção de especialistas consiste em determinar de forma sistemática quem será convidado a fazer parte do corpo de especialistas, e pode determinar o sucesso de um esforço que envolve opinião de especialistas (BRUIN & ROSEMANN, 2007). Neste sentido, de forma a corretamente identificar os especialistas responsáveis pela avaliação do método, foi seguido o procedimento indicado por Powell (2003) na seleção dos especialistas:

1. Definir a classificação potencial dos especialistas: Esta primeira etapa tem a intenção de preparar uma categorização dos especialistas antes de identificá-los, na intenção de definir quais são as importantes classes de especialistas. A Norma para a Educação e Testes Psicológicos da *American Educational Research Association* estabelece três critérios para membros de *Expert Panels* envolvidos no processo de revisão de conteúdo: formação adequada, experiência e qualificações (SENDJAYA, 2003). Assim foram identificadas como características relevantes de forma a possibilitar uma avaliação do método: (i) publicação de artigos relevantes descrevendo criação ou metodologia de desenvolvimento de SPCMMs, (ii) experiência no

desenvolvimento de modelos de capacidade e/ou maturidade de processo, (iii) formação acadêmica e, (iv) experiência prática em melhoria de processos.

2. Preencher uma planilha com os especialistas em potencial: Este é um passo iterativo onde são preenchidas as categorias com nomes reais dos potenciais peritos tentando cobrir todas as classes de especialistas identificados no primeiro passo. Para obtenção dos nomes e e-mails dos especialistas a serem convidados foi utilizada a lista dos artigos encontrados na revisão da literatura (vide capítulo 3). Destes artigos foram extraídos nomes e e-mails dos primeiros autores de cada artigo que descrevia: (i) o desenvolvimento de um SPCMM onde foram apresentados detalhes de como o modelo foi desenvolvido; (ii) uma técnica ou metodologia pra desenvolvimento de modelos. Como resultado, quarenta especialistas foram identificados e catalogados em uma planilha.
3. Estabelecer critério de ranking dos especialistas: Este passo visa definir grau de representatividade (formação, experiência e qualificação) de cada perito. Esta representatividade é quantificada por meio da avaliação dos critérios definidos no primeiro passo. Neste sentido é necessário especificar quais os pesos de cada uma das características. Os pesos foram definidos conforme o grau de importância dado a cada um: 0.20 – formação; 0.40 – experiência em melhoria de processos, e; 0.40 – experiência no desenvolvimento de SPCMMs. Para formação foi considerado o nível máximo como doutorado (4 - 0.20), seguido de mestrado (3 - 0.15), bacharel (2 - 0.10) e outros (1 - 0.05), sendo o percentual atribuído proporcionalmente conforme indicado. Para a experiência em melhoria de processos quatro faixas foram também definidas: mais de dez anos de experiência (4 - 0.40), entre cinco e dez anos de experiência (3 - 0.30), entre um e cinco anos (2 - 0.20) e menos de um ano de experiência (1 - 0.10). Por último, para a experiência no desenvolvimento de SPCMMs foi estabelecido que o peso fosse proporcional à quantidade máxima de modelos que um dos especialistas tivesse participado, da seguinte forma: ($\text{<número de modelos>/quantidade máxima de modelos} * 0.40$).

4. Convidar os especialistas: Os especialistas são convidados a participar do *Expert Panel* de validação. Todos são convidados diretamente por meio de um e-mail explicando o caráter da pesquisa e contendo um link para download do método e outro para o preenchimento do formulário web contendo o questionário. Os e-mails foram obtidos diretamente das publicações citadas no passo 2. Cinco e-mails retornaram erro após o envio indicando possível obsolescência dos endereços de e-mail. Dois foram substituídos por outros aparentemente mais novos encontrados em publicações mais recentes e não foram exibidos mais erros. Para dois especialistas convidados, no entanto, não foi possível encontrar novos e-mails e o erro persistiu, sendo o total de convites realmente enviados de 38.

5.2.2 Design do Questionário

O instrumento escolhido para coletar as medidas definidas foi o questionário, pela sua aplicabilidade ao cenário do *Expert Panel* (vide seção 5.2) onde os especialistas encontram-se distribuídos e não estão diretamente disponíveis para entrevista. As medidas definidas são traduzidas em perguntas do questionário, conforme apresentado no Quadro 16.

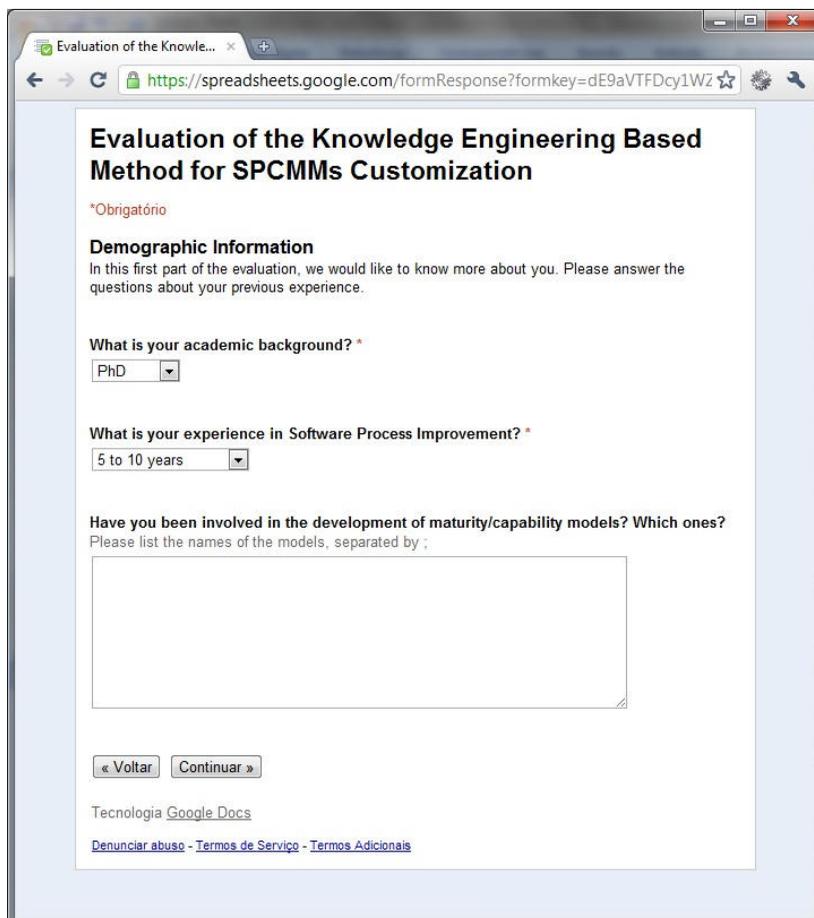
Quadro 16: Coleta de dados para o Objetivo de Avaliação 1.

Medida	Coleta de Dados
MQ1.1: Quantidade de erros identificados.	Questionário - perguntas: 4, 7, 10, 13 e 16
MQ2.1: Quantidade de inconsistências identificadas.	Questionário - perguntas: 5, 8, 11, 14 e 17
MQ2.2: Quantidade de inconsistências corrigidas.	Contagem das correções feitas no texto do método
MQ3.1: Quantidade de atividades faltantes.	Questionário - perguntas: 6, 9, 12, 15 e 18
MQ3.2: Quantidade de tarefas faltantes.	Questionário - perguntas: 6, 9, 12, 15 e 18
MQ3.3: Quantidade de técnicas faltantes.	Questionário - perguntas: 6, 9, 12, 15 e 18
MQ3.4: Quantidade de produtos de trabalho faltantes	Questionário - perguntas: 6, 9, 12, 15 e 18
MQ4.1: Impressão subjetiva sobre a capacidade do método de	Questionário - pergunta 19

suportar a customização de SPCMMs.	
MQ5.1: Impressão subjetiva sobre a capacidade do método de capturar as melhores práticas no domínio	Questionário - pergunta 20
MQ6.1: Impressão subjetiva sobre a completude do método em relação às demais abordagens existentes.	Questionário - pergunta 21
MQ7.1: Pontos fracos do método	Questionário - pergunta 26
MQ8.1: Pontos fortes do método	Questionário - pergunta 27
MQ9.1: Impressão subjetiva sobre a generalidade do método	Questionário - perguntas: 22 e 24

Conforme pode ser observado no Quadro 16, foram definidas perguntas para todas as medidas. Foram elaboradas no total 28 perguntas, sendo 19 perguntas abertas, 2 de múltipla escolha e 7 utilizando a escala Likert (CHANG, 1993). As perguntas abertas foram utilizadas especialmente para coletar erros, inconsistências e omissões das diversas fases do método (vide Apêndice D) e também para identificar os pontos fortes e fracos do método e as observações gerais. As perguntas de múltipla escolha foram utilizadas na coleta de dados demográficos, como experiência e formação acadêmica, na forma de faixas ou itens individuais. Por fim, as perguntas de escala Likert, foram utilizadas para coletar dados do nível de opinião dos especialistas sobre vários aspectos a serem avaliados no modelo por serem apropriadas para este tipo de pergunta (CHOMEYA, 2010).

O questionário foi desenvolvido na forma de um formulário web utilizando a plataforma *Google Docs* (<http://docs.google.com/>). Por meio desta plataforma é possível elaborar um formulário web utilizando a ferramenta *Google Forms* (<http://www.google.com/google-d-s/forms/>). A Figura 27 apresenta um extrato do questionário original disponibilizado em língua inglesa, que pode ser acessado em: https://spreadsheets.google.com/viewform?hl=en_GB&formkey=dE9aVTFDcy1WZlITWnVIZFA1aHVYOF6MQ#gid=0. Entretanto, não é mais possível submeter novas avaliações no formulário. A versão completa do questionário em português encontra-se no Apêndice D.



The image shows a web browser window displaying a Google Form. The browser's address bar shows the URL: <https://spreadsheets.google.com/formResponse?formkey=dE9aVTFDcy1Wz>. The form title is "Evaluation of the Knowledge Engineering Based Method for SPCMMs Customization". Below the title, there is a red asterisk and the word "Obrigatório" (Mandatory). The section is titled "Demographic Information" and includes the introductory text: "In this first part of the evaluation, we would like to know more about you. Please answer the questions about your previous experience." The first question is "What is your academic background? *", with a dropdown menu currently showing "PhD". The second question is "What is your experience in Software Process Improvement? *", with a dropdown menu showing "5 to 10 years". The third question is "Have you been involved in the development of maturity/capability models? Which ones? Please list the names of the models, separated by ;", followed by a large text input area. At the bottom of the form, there are two buttons: "« Voltar" and "Continuar »". Below the buttons, it says "Tecnologia Google Docs" and provides links for "Denunciar abuso", "Termos de Serviço", and "Termos Adicionais".

Figura 27: Extrato do formulário web contendo o questionário

5.2.3 Análise da validade do instrumento

Validade consiste no grau em que as conclusões (interpretações) derivadas dos resultados de qualquer avaliação são bem fundamentadas ou justificáveis, sendo ao mesmo tempo relevantes e significativas (COOK & BECKMAN, 2006). Ainda segundo Cook & Beckman (2006), uma boa forma de verificar a validade de um instrumento de coleta de dados se dá por meio da avaliação da validade do seu conteúdo. A validade de conteúdo é uma medida subjetiva que identifica

o quanto são apropriados os itens de um instrumento na opinião de revisores que tenham conhecimento na área (LITWIN, 1995).

Segundo Litwin (1995), a avaliação da validade de conteúdo envolve a realização de uma revisão estruturada onde o instrumento de coleta é revisado por especialistas de forma a assegurar que: (i) ele contém tudo que é esperado para contemplar o seu objetivo e (ii) ele não inclui nada que não seja desejável. Também é essencial que o questionário seja submetido a um teste piloto em uma pequena amostra da população de forma a identificar eventuais erros e inconsistências (LITWIN, 1995).

Desta forma, a validade do conteúdo do questionário eletrônico foi avaliada por meio da coleta de opinião de seis especialistas, sendo cinco estudantes de doutorado na área de engenharia de software (um brasileiro, três irlandeses e um alemão), e um pesquisador sênior brasileiro na área de desenvolvimento de SPCMMs com nível de doutorado. Estes revisores foram escolhidos pelos critérios de disponibilidade e proximidade, pois a maior parte deles é membro do mesmo grupo de pesquisas de engenharia de software no *Dundalk Institute of technology* (DKIT, 2010). Os objetivos da pesquisa, o método e o questionário foram apresentados aos revisores, que foram então instruídos a responder normalmente o questionário e paralelamente avaliar o questionário em si, respondendo às seguintes questões abertas:

- O questionário contém tudo que é esperado para contemplar o seu objetivo?
- O questionário contém quaisquer informações não desejáveis ou desnecessárias ao contexto e objetivo da pesquisa?
- Você conseguiu compreender adequadamente as perguntas?
- Existe algum erro ou inconsistência no questionário?

O questionário foi então avaliado e os revisores responderam às perguntas por e-mail diretamente ao autor do questionário. Os resultados observados foram, na sua maior parte, relacionados a pequenas correções de gramática da língua inglesa. Além destes aspectos, as perguntas foram respondidas da seguinte forma:

- *O questionário contém tudo que é esperado para contemplar o seu objetivo?*

Todos os avaliadores consideraram que o questionário é completo o suficiente para contemplar o seu objetivo de avaliar

o método. No entanto, alguns termos em inglês foram corrigidos na escala por sugestão de um dos revisores.

- *O questionário contém quaisquer informações não desejáveis ou desnecessárias ao contexto e objetivo da pesquisa?*

Três revisores apontaram a desnecessidade de uma das questões que solicitava ao avaliador com quantos modelos o entrevistado já havia tido contato e a pergunta seguinte questionava quais eram esses modelos. Assim, acatando a sugestão, as duas perguntas foram unificadas.

- *Você conseguiu compreender adequadamente as perguntas?*

Todos os revisores concordaram que o questionário é fácil de ser compreendido. Foi sugerida a realocação da seção de avaliação geral do método, que estava antes da avaliação detalhada, para ser movida ao final da avaliação. Esta alteração foi também realizada.

- *Existe algum erro ou inconsistência no questionário?*

Nenhuma inconsistência foi encontrada pelos revisores. Entretanto alguns erros de redação foram apontados, especialmente pelos nativos de língua inglesa (três irlandeses) em relação a algumas perguntas. Os problemas apontados foram também corrigidos, sem que a semântica original das questões fosse alterada.

Uma vez corrigidos os problemas encontrados, o teste piloto do formulário web contendo o questionário foi realizado por uma pesquisadora de nível de doutorado na área de engenharia de software. O tempo de preenchimento do questionário relatado pela pesquisadora foi de 70 minutos e somente foi observado que algumas informações apareciam em língua portuguesa, tais como: indicação de campos obrigatórios e rótulos de botões. Após algumas simulações foi concluído, no entanto, que o formulário web se adapta à linguagem do navegador do usuário, de forma que, em navegadores configurados em língua inglesa esse problema não ocorria. Foi possível observar também neste teste piloto que os registros submetidos por meio do formulário web foram corretamente inseridos na planilha possibilitando a adequada coleta de dados.

Tendo sido analisada a validade do instrumento de coleta de dados e realizado o teste piloto, o questionário foi aplicado no *Expert Panel* para validação do conteúdo do modelo, conforme apresenta a seção seguinte.

5.2.4 Coleta de dados

Os dados foram coletados de novembro/2010 a dezembro/2010 por meio do formulário web já apresentado. Dos 38 especialistas convidados, 15 participaram do painel, avaliando o método e respondendo o questionário, representando uma taxa de resposta de 39,5%.

As respostas dos especialistas foram automaticamente armazenadas pelo formulário web em uma planilha de cálculo no *Google Docs*, o que facilitou a tabulação e tratamento dos dados. Como não houve registros com campos obrigatórios não preenchidos, pois o formulário web bloqueava o não preenchimento destes campos, não foi necessário tratar valores nulos.

Análise demográfica

Conforme pode ser observado no apêndice D, as três primeiras perguntas do questionário aplicado visam caracterizar os especialistas que avaliaram o método. Os especialistas foram questionados quanto à sua formação, experiência com SPI e em quais SPCMMs participaram do desenvolvimento. Estes dados coletados são também utilizados para calcular o índice de classificação dos especialistas conforme definido na seção 4.2.1. A seguir são apresentados os resultados.

Quanto à formação, a maior parte dos especialistas (73% - 11) possui titulação de doutor enquanto 20% (3) possuem título de mestre, conforme apresenta a Figura 28. Com base nesses dados é possível afirmar que os participantes do painel possuem alta formação acadêmica em sua maioria.

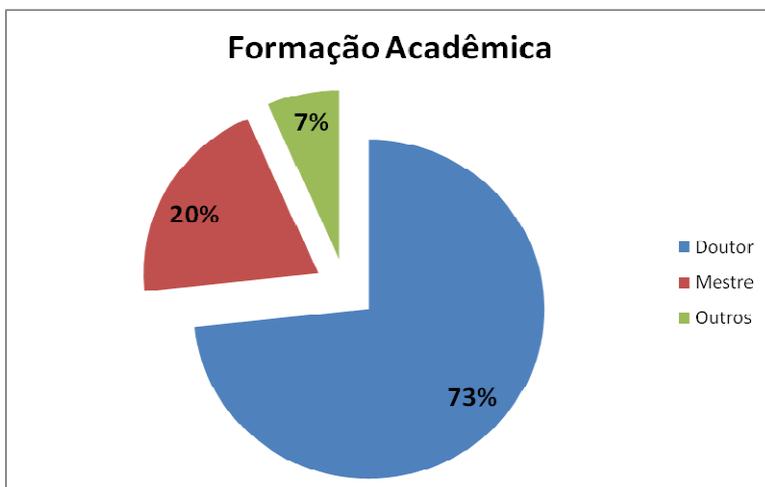


Figura 28: Formação acadêmica dos especialistas

Quanto à experiência em SPI, a maior parte dos especialistas (67% - 10) possui mais de 10 anos de experiência, 13% (2) possuem entre 5 e 10 anos, 7% (1) possui entre 1 e 5 anos e 13% (2) possuem menos de 1 ano de experiência (vide Figura 29). Estes dados demonstram que os participantes do painel possuem experiência prática na melhoria de processos de software, com poucas exceções.



Figura 29: Experiência dos Especialistas

Por fim, os participantes do painel foram questionados acerca de quais SPCMMs eles estiveram envolvidos no desenvolvimento. Como resultado 23 modelos e normas diferentes foram citados pelos participantes, sendo a média geral, considerando modelos que tiveram mais de um participante envolvido, de 2,6 modelos desenvolvidos por participante. Com a exceção de 2 participantes (13%), todos os demais (13 - 87%) estiveram diretamente envolvidos na elaboração de pelo menos um modelo ou norma, sendo que dois participantes (13%) estiveram envolvidos na elaboração de mais de quatro modelos, conforme pode mostra a Figura 30.

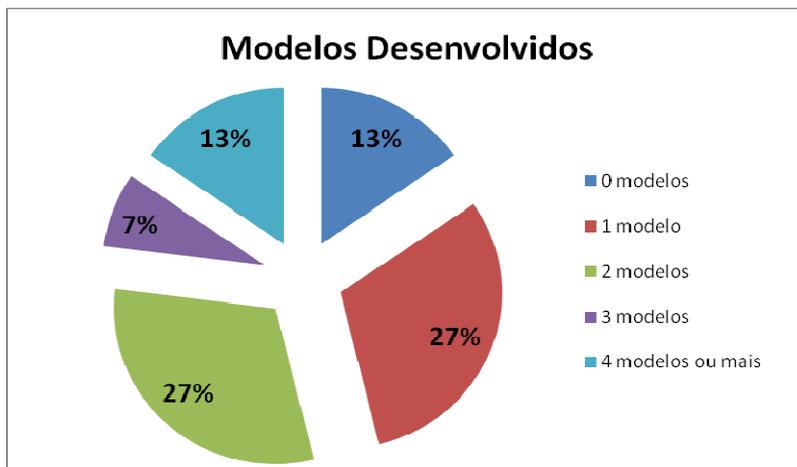


Figura 30: Modelos desenvolvidos pelos participantes

Conforme pode ser observado na Tabela 1, das 23 normas e modelos desenvolvidos pelos especialistas participantes, 4 possuem mais de um envolvido no seu desenvolvimento (Medi SPICE, ISO/IEC 15504-5, Enterprise SPICE e ISO/IEC 29110-2), sendo que 9 (60%) participantes diferentes estiveram envolvidos em alguma destas normas e modelos.

Tabela 1: Modelos e Normas desenvolvidos pelos especialistas

Modelo	Número de Participantes Envolvidos

Enterprise SPICE	4
Medi SPICE	3
ISO/IEC 15504-5	3
ISO/IEC 29110-2	2
Situational Maturity Model for Collaboration	1
Health Supplier Relationship Management Maturity Model	1
E ³ M	1
Process Reference Model and Assessment for the Leadership of Complex Virtual Teams	1
ISO/IEC 15504-6	1
ISO/IEC 15504-8	1
ISO/IEC 15504-10	1
CMMI v1.0	1
CMMI v1.2	1
IT Performance Measurement Maturity Model	1
Business Process Management (BPM) maturity	1
ISO/IEC 20000	1
SPINI	1
ISO/IEC 29110-5.1 based assessment model	1
Mares	1
Spice for Research	1
Software Público Brasileiro	1
Requirements CMM	1

Seguindo o cálculo definido na seção 5.2.1, onde são atribuídos pesos para cada uma das características dos especialistas participantes, o índice geral de relevância dos especialistas é apresentado na Tabela 2.

Tabela 2: Classificação dos especialistas

Especialista	Experiência no Desenv. de Modelos	Experiência com SPI	Formação	Índice Geral
Especialista 1	0,03	0,40	0,20	0,6
Especialista 2	0,06	0,20	0,20	0,5
Especialista 3	0,06	0,40	0,20	0,7
Especialista 4	0,00	0,10	0,20	0,3

Especialista 5	0,03	0,40	0,20	0,6
Especialista 6	0,03	0,30	0,20	0,5
Especialista 7	0,25	0,40	0,15	0,8
Especialista 8	0,03	0,30	0,20	0,5
Especialista 9	0,06	0,10	0,15	0,3
Especialista 10	0,09	0,40	0,05	0,5
Especialista 11	0,06	0,40	0,20	0,7
Especialista 12	0,00	0,40	0,20	0,6
Especialista 13	0,40	0,40	0,20	1,0
Especialista 14	0,06	0,40	0,15	0,6
Especialista 15	0,03	0,40	0,20	0,6
Média geral				0,6

Por meio da análise da Tabela 2 é possível perceber que as opiniões de alguns especialistas poderiam ter uma maior relevância em relação a outros, devido à sua formação e experiências (e.g. o Especialista 13 em relação ao Especialista 4). Entretanto, como todos os especialistas foram convidados por terem sido autores de artigos científicos que relatavam o desenvolvimento de modelos de capacidade/maturidade e também devido à baixa taxa de resposta obtida no painel (vide seção ameaças à validade) esses pesos não são diretamente utilizados para ponderar os dados obtidos. Somente nas discussões de alguns dos resultados esses dados são utilizados, conforme pode ser observado na seção 5.2.6.

5.2.5 Análise da Confiabilidade do instrumento

A confiabilidade (*reliability*) de um instrumento de coleta de dados refere-se à reprodutibilidade e consistência dos resultados obtidos de uma coleta para outra, sendo um componente necessário, mas não suficiente, da sua validade (COOK & BECKMAN, 2006). Existem diversas formas de avaliar a confiabilidade de um instrumento, dentre elas, Cook & Beckman (2006) citam as mais importantes: consistência interna, estabilidade temporal, formas paralelas, concordância inter-avaliador e generalização de teoria.

Neste sentido, Gliem & Gliem (2003) indicam que ao se utilizar escalas do tipo *Likert*, é imperativo calcular e comunicar a consistência interna para quaisquer escalas ou sub-escalas utilizadas. Assim, dentre as possibilidades disponíveis, a consistência interna, que implica em avaliar se todos os itens de um instrumento medem o mesmo construto (COOK & BECKMAN, 2006), apresenta-se como uma interessante alternativa.

Considerando-se que todas as perguntas do questionário têm o objetivo de medir o mesmo construto da *Adequação do Método* (vide seção 5.1), sete destas perguntas foram elaboradas para realizar a avaliação geral do método de acordo com a opinião dos especialistas, na forma de uma afirmação cada, com cinco opções em escala *Likert* (perguntas: 19, 20, 21, 22, 23, 24 e 25). As demais correspondem a perguntas abertas buscando detalhes de erros, inconsistências e ausências percebidas para cada uma das cinco fases do método, conforme mostram o Quadro 16 e o Apêndice D.

Gliem & Gliem (2003) indicam o cálculo do coeficiente *Cronbach Alpha* para análise da consistência interna quando são utilizadas escalas do tipo *Likert* (GLIEM & GLIEM, 2003). O teste de confiabilidade *Cronbach Alpha* é uma técnica que exige a administração de apenas um teste simples em uma amostra para fornecer uma estimativa de confiabilidade para um determinado instrumento (GLIEM & GLIEM, 2003). Entretanto, como o *Expert Panel* não é baseado em amostragem, todos os dados reais coletados foram utilizados para o cálculo do coeficiente. A Tabela 3 apresenta os dados coletados dos 15 especialistas e o coeficiente calculado, sendo que cada coluna representa um especialista e cada linha uma das perguntas. Os itens da escala foram indexados de 1 a 5, correspondendo às alternativas *Discordo Fortemente* a *Concordo Fortemente*.

Tabela 3: Tabulação das respostas da avaliação geral do método.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	4	3	4	4	5	4	4	4	2	4	4	3	5	2	4
2	4	2	5	5	5	5	2	4	3	4	4	3	4	3	4
3	5	3	5	3	5	3	3	3	3	4	4	3	4	4	5
4	4	4	4	4	5	5	2	3	4	3	4	3	4	4	4
5	5	4	5	4	5	4	3	3	2	5	4	2	3	5	3
6	5	2	5	4	5	5	3	3	1	4	4	2	4	5	4

7	4	3	4	4	5	4	4	4	2	5	4	3	4	3	5
<i>Cronbach's Alpha</i>												0,871			

Com base nos dados apresentados na Tabela 3, o coeficiente *Cronbach Alpha* foi calculado utilizando-se o *Free Statistics Software* (WESSA, 2011), sendo que o cálculo foi validado por meio de uma planilha implementada no Microsoft Excel², onde o coeficiente resultante foi o mesmo.

Conforme mostra a Tabela 3, o cálculo do *Cronbach Alpha* resultou no coeficiente 0,871. O coeficiente *Cronbach Alpha* é um número que varia de 0 a 1, sendo que o valor 0 indica que o instrumento não tem confiabilidade e o valor 1 indica que o instrumento tem a confiabilidade perfeita (GOLDENSON et al., 1999). Goldenson et al. (1999), citando outras pesquisas anteriores, concluem que valores acima de 0,8 são considerados suficientes para fins de pesquisa aplicada em processos de software.

Desta forma, o coeficiente obtido indica que as sete perguntas da avaliação geral tendem a representar concisamente e com confiabilidade o construto da *Adequação do Método*. A partir desta avaliação os resultados do *Expert Panel* podem ser então analisados na seção seguinte.

5.2.6 Resultados do *Expert Panel*

Os resultados observados são apresentados nesta seção seguindo a ordem das perguntas elaboradas de acordo com a abordagem GQM (vide item 5.1).

Pergunta Q1: O método é correto?

Em resposta a esta pergunta os especialistas encontraram diversos erros, na sua maior parte tipográficos e referentes ao uso da língua inglesa (vide Figura 31). Os erros foram relatados pelos especialistas em resposta a perguntas abertas (4, 7, 10, 13 e 16 do questionário – correspondentes às 5 fases do método), de forma que foi necessário categorizá-los. Foram relatados 58 erros no total, os quais foram

² Disponível em: <http://www.brunopedroso.com.br/cronbach.html>

classificados em 3 categorias: (i) Erros Textuais, quando referentes a erros ortográficos/gramaticais relacionados ao uso da língua inglesa, digitação, etc.; (ii) Erros de Formatação, quando erros relacionados a figuras, quadros ou impressão foram relatados; e (iii) Erros Conceituais, quando o uso de algum conceito ou referência foi considerado inadequado, conforme apresentado na Figura 31.

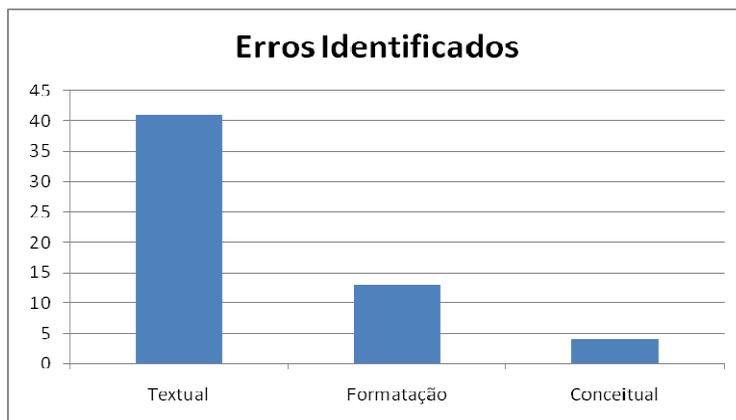


Figura 31: MQ1.1: Quantidade de erros identificados

Dos 58 erros relatados, em relação à medida *MQ1.2: Quantidade de erros corrigidos*, todos foram corrigidos na versão 1.0 do método.

Pergunta Q2: O método é consistente?

Poucas inconsistências foram relatadas nas perguntas 5, 8, 11, 14 e 17 do questionário. Somente 7 inconsistências no método foram percebidas pelos especialistas, todas relacionadas às atividades das fases 1 e 2 do método, conforme mostra o Quadro 17.

Quadro 17: MQ2.1: Inconsistências identificadas.

No.	Inconsistência Relatada	Atividades Envolvidas
1	Um especialista indica que a tarefa 1.1.3 - Identificar o <i>Gap</i> está inconsistente com as demais tarefas da atividade e deveria ser movida para a atividade A1.2, pois está relacionada à identificação de fontes de	A1.1

	conhecimento.	
2	Na opinião de um dos especialistas, a principal preocupação no desenvolvimento de um PRM ou PAM está na divisão global do domínio em processos. O especialista acredita que as definições das atividades não são consistentes em relação a esta preocupação e acredita que o enfoque dominante sobre requisitos da qualidade do domínio nem sempre pode fornecer um resultado eficaz.	A2.1, A2.2, A2.3
3	Para outro especialista a atividade A2.2 é inconsistente pois parece contemplar modelos de um mesmo domínio (ou uma sobreposição do mesmo). O especialista argumenta que podem ser processos fortemente relacionados (normalmente de suporte ou do escopo da empresa) pertencentes a outros domínios, que precisam ser integrados.	A2.2
4	Um especialista relata que as atividades 2.1 e 2.3 são inconsistentes, pois, enquanto a tarefa 2.1.2 Design da arquitetura de capacidade/dimensão maturidade da atividade 2.1 dá a entender que qualquer arquitetura possa ser utilizada, a atividade 2.3 prevê a definição de objetivos e resultados, que são específicos da arquitetura da ISO/IEC 15504.	A2.1, A2.3
5	Outro especialista indica que, na tarefa 2.4.1 Definir indicadores de desempenho do processo, é discutível se o mapeamento de um indicador a mais de um resultado processo deve ser permitido (mesmo que os modelos atuais como ISO/IEC 15504-5 permitam fazê-lo)	A2.4
6	Segundo outro especialista, as atividades 1.3 e 1.4 devem vir antes de 1.1 e 1.2, fazendo parte de uma fase de criação, que ocorre antes do início do projeto de customização.	A1.3, A1.4
7	Ainda é relatado que a fase 2 é claramente dependente da fase 1. A extração do conhecimento dos agentes de conhecimento do domínio seria reforçada pela identificação de uma pergunta de pesquisa que requeresse atendimento nas fases iniciais da abordagem.	A2.1, A2.2, A2.3

Após a análise das inconsistências relatadas, somente as inconsistências 3, 4, 5 e 7 puderam efetivamente ser corrigidas com

pequenas alterações no texto do método de forma a contemplar a sugestão dos especialistas. A inconsistência 2 não pode ser corrigida por não apresentar detalhes suficientes para que alguma alteração pudesse ser realizada no texto do método. Já para a inconsistência 1 o autor entende que a identificação do *Gap* fica mais evidente logo após a revisão (sistemática) de literatura, onde é possível perceber que não existem modelos que contemplem as maiores necessidades de qualidade do domínio. Desta forma, em relação à medida MQ2.2: *Quantidade de inconsistências corrigidas*, na versão atual do método (1.0) foram corrigidas 4 das 7 inconsistências relatadas.

Pergunta Q3: O método é completo?

Os especialistas foram perguntados se perceberam a ausência de alguma atividade, tarefa, técnica ou produtos de trabalho relevantes (nas perguntas 6, 9, 12, 15 e 18 do questionário). Foram recebidas 20 sugestões de novas tarefas, 4 de novas técnicas e 2 de novas atividades para o método. Nenhuma sugestão foi recebida para novos produtos de trabalho. As sugestões são apresentadas por fase do método na Figura 32. Nota-se que a fase 1 é a que mais recebeu sugestões de melhoria no conjunto de tarefas (8) e só na fase 2 foram recebidas sugestões de atividades, tarefas e técnicas.

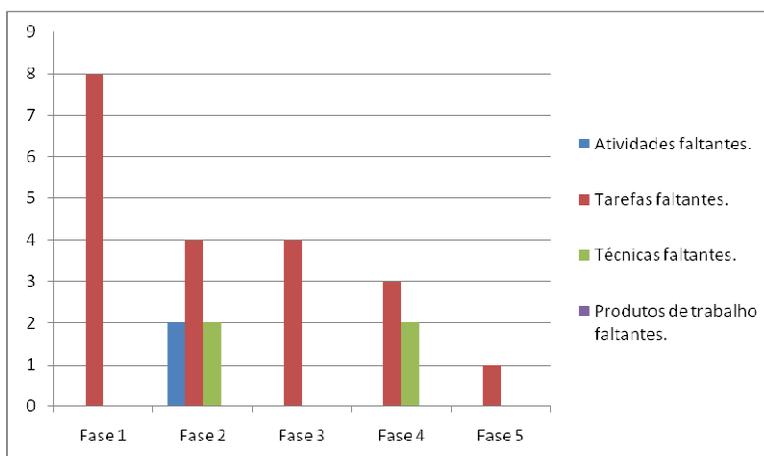


Figura 32: MQ3.1, MQ3.2, MQ3.2 e MQ 3.4 por fase do método

Uma sugestão que não está relacionada a um novo componente foi relatada por 4 diferentes especialistas: definir iterações de forma mais explícita. Apesar de não ser a pergunta ideal para receber este comentário, no entanto, esta recomendação feita por 26% dos especialistas evidencia que uma mudança mais profunda precisa ser feita na apresentação das atividades, de forma a deixar a realização de iterações, que na versão atual do método é indicada no corpo da descrição de diversas tarefas, mais evidente. O Quadro 18 apresenta a lista das sugestões de atividades, tarefas e técnicas recebidas.

Quadro 18: Lista das Atividades, Tarefas e Técnicas consideradas faltantes

Atividades
<ul style="list-style-type: none"> • Analisar as capacidades culturais (valores e incentivos para um comportamento) • Analisar a capacidade tecnológica (infra-estrutura)
Tarefas
<ul style="list-style-type: none"> • Publicações revisadas por pares quanto à definição do escopo e dos objetivos do modelo • Benchmarking de melhores práticas existentes • Identificação dos principais <i>stakeholders</i> • Nível de cobertura da empresa, e sua relação com a arquitetura corporativa • Possibilidade de variações de múltiplos perfis baseados em diferentes contextos de operação • Abordagem sistemática para a reutilização de modelos existentes • Iteração mais explícita • Obter <i>feedback</i> de usuários em fases iniciais • Estudos empíricos como parte da avaliação • Procedimento de obtenção de comentários dos usuários antes da votação • Separar os aspectos técnicos do modelo de governança e políticas distintas das sugestões de procedimento • Programas de estudos empíricos formais na fase 4 • Uso de pilotos antes da votação, como parte da atividade A3.1 • Repetição da votação na fase 5 • Familiarização com CMMi e ISO/IEC 15504 • Verificar se o domínio de interesse já foi personalizado • Determinação dos objetivos (ou <i>business cases</i>) para criar o modelo customizado • Determinação dos recursos disponíveis • Criação de uma pergunta de pesquisa focada para a customização do modelo • Verificar se o modelo em uso é adequado para o propósito
Técnicas
<ul style="list-style-type: none"> • Métodos estatísticos para calcular níveis de maturidade • Engenharia situacional de método

- Guias eletrônicos de processo (EPGs)
- Repositórios de experiência

As sugestões recebidas não foram ainda implementadas na versão atual do método, mas serão incorporadas a futuras versões.

Pergunta Q4: O método contém o que é necessário para suportar a customização de SPCMMs?

Para responder a esta pergunta, a medida: *MQ4.1: Impressão subjetiva sobre a capacidade do método de suportar a customização de SPCMMs* foi coletada por meio de uma afirmação no questionário (pergunta 19) com perguntas na forma de escala Likert de cinco pontos, variando de *Concordo Fortemente* (código 5) até *Discordo Fortemente* (código 1). Os percentuais das respostas são apresentados na Figura 33.

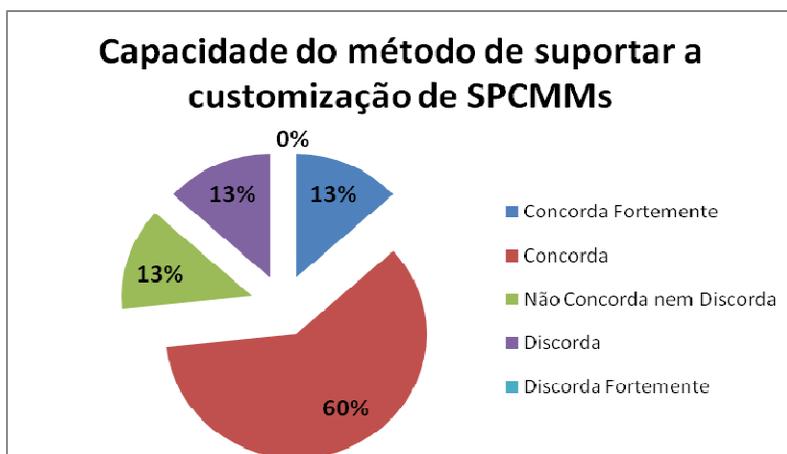


Figura 33: MQ4.1: Impressão subjetiva sobre a capacidade do método de suportar a customização de SPCMMs.

Como resultado, dos especialistas participantes, 73% (11) concordam que o método fornece o suficiente para possibilitar a customização de SPCMMs e 13% (2) discordam que o método forneça o suficiente. Outros 13% (2) mantêm-se neutros e nenhum especialista discorda fortemente desta afirmação.

Pergunta Q5: A utilização do método é capaz de produzir SPCMMs que realmente contenham as melhores práticas de desenvolvimento de software para o domínio?

Esta é, talvez, a pergunta mais importante do ponto de vista da utilidade do método na aquisição de conhecimento para a customização das melhores práticas de desenvolvimento de software para o domínio. Para responder a esta pergunta, a medida: *MQ5.1: Impressão subjetiva sobre a capacidade do método de capturar as melhores práticas no domínio* foi também coletada por meio de uma afirmação no questionário (pergunta 20) com perguntas na mesma forma de escala Likert. Conforme mostra a Figura 34, 73% (11) dos especialistas afirmam que o método é capaz de produzir SPCMMs que contenham as melhores práticas de desenvolvimento de software para o domínio, enquanto 7% (1) entendem que o método não tem essa capacidade e 20% (3) permaneceram neutros. Nenhum especialista discorda fortemente desta afirmação.

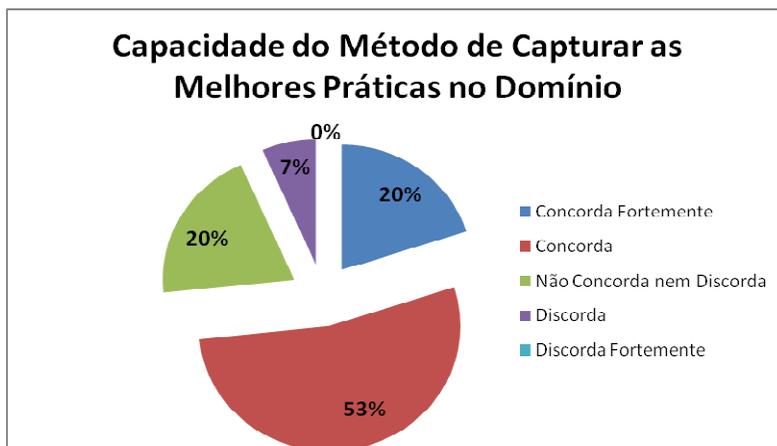


Figura 34: MQ5.1: Impressão subjetiva sobre a capacidade do método de capturar as melhores práticas no domínio.

Assim, esta afirmação dá indícios de que, na opinião da maioria dos especialistas, um modelo que seja desenvolvido utilizando o método tende conter as melhores práticas capturadas das fontes de conhecimento do domínio. É importante observar que o único especialista que discorda da capacidade do método de capturar as melhores práticas possui índice

de relevância de 0,3 (vide seção 4.2.4), o que corresponde à metade da média geral dos especialistas.

Pergunta Q6: O método apresenta suporte mais amplo para o desenvolvimento de SPCMMs do que os atualmente existentes?

Para esta pergunta, a medida: *MQ6.1: Impressão subjetiva sobre a completude do método em relação às demais abordagens existentes* foi coletada da mesma forma, por meio de uma afirmação no questionário (pergunta 21) com perguntas na forma de escala Likert. Conforme apresentado na Figura 35, 60% (9) concordam que o método apresenta suporte mais amplo para o desenvolvimento de SPCMMs do que os atualmente existentes, 27% (4) ficaram neutros, enquanto 13% (2) afirmam que o método não é mais completo que os demais. Nenhum especialista discorda fortemente desta afirmação.

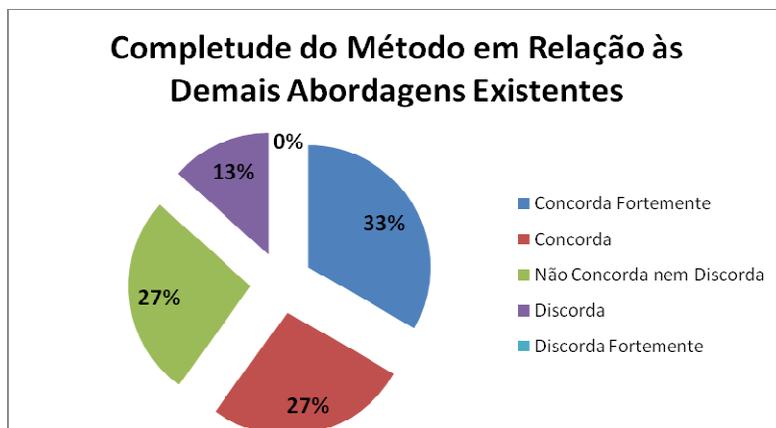


Figura 35: MQ6.1: Impressão subjetiva sobre a completude do método em relação às demais abordagens existentes.

É interessante ressaltar que a afirmação de que o método não é mais completo que os atuais foi observada nas respostas de especialistas que, no entanto, não sugeriram quaisquer outras atividades, tarefas e técnicas adicionais que deveriam ser incorporadas ao método (vide pergunta Q3). Também quanto à qualificação dos especialistas que responderam negativamente a esta questão, os índices gerais de

relevância são (respectivamente 0,3 e 0,6) baixos em relação à média de experiência dos demais especialistas participantes.

Pergunta Q7: O método é genérico o suficiente para permitir a captura de conhecimento no desenvolvimento de SPCMMs para diferentes domínios de desenvolvimento de software?

Em resposta a esta pergunta, a medida: *MQ7.1: Impressão subjetiva sobre a generalidade do método* foi também coletada por meio de uma afirmação no questionário (pergunta 24) com perguntas na forma de escala Likert. A Figura 36 mostra que 73% (11) concordam que o método é genérico o suficiente, 20% (3) ficaram neutros, e 7% (1) discordam o método seja genérico o suficiente. Nenhum especialista discorda fortemente desta afirmação.

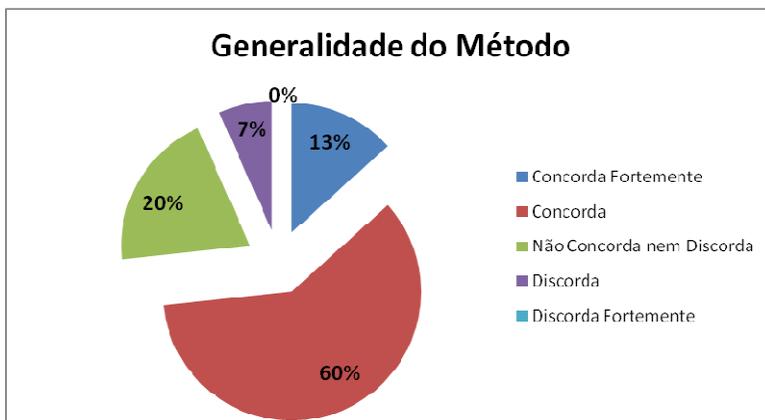


Figura 36: MQ7.1: Impressão subjetiva sobre a generalidade do método.

A forte aceitação da generalidade do método por parte dos especialistas é bastante positiva, pois reforça a sua possível aplicabilidade na captura de conhecimento em diferentes domínios de desenvolvimento de software.

Perguntas Q8 e Q9: Quais são os pontos fortes e fracos mais importantes do método?

Em resposta a estas perguntas, duas medidas foram coletadas *MQ8.1: Pontos fortes do método* e *MQ9.1: Pontos fracos do método* em duas perguntas abertas no questionário (perguntas 26 e 27). No Quadro 19 são apresentados os pontos fortes encontrados pelos especialistas, enquanto no Quadro 20 os pontos fracos são apresentados.

Quadro 19: MQ8.1: Pontos fortes do método

Pontos Fortes do Método
Apresenta uma boa visão
Fornecer uma abordagem global
Preenche a necessidade deste tipo de método
Exemplos práticos de técnicas e ferramentas
Referências
Estrutura
Abrangência
Uniformidade das práticas
Apoio para a melhoria de processos em ambiente real
Estruturação
Repetibilidade
Fornecer uma estrutura real para o desenvolvimento de um novo SPCMM
Pode ser usado para desenvolver um SPCMM para qualquer domínio
Reúne em um lugar que tudo que precisa ser levado em consideração ao desenvolver um SPCMM
Amplitude de aplicação
Abrangente na abordagem
Fornecer um ponto de partida para grupos que queiram desenvolver modelos em um novo domínio
Harmonizado com as abordagens de normatização disponíveis
Se for seguido, deverá proporcionar bom comprometimento das partes interessadas
Tentativa de documentar o processo
Abrangente
Abordagem sistemática
Cobertura adequada do escopo
Estrutura clara
Um conjunto completo de referências
Pronto para testes
Customizável
Tentativa de definir uma abordagem sistemática
Agrupar de material de referência relevante
O método utilizado para desenvolvê-lo

O detalhamento de cada atividade
O conjunto de atividades e tarefas
É uma boa visão
Fornecer uma abordagem global
Abrange as áreas que podem não ser óbvias, se não forem seguidas todas as diretrizes
Dá uma ordem à forma de se desenvolver o modelo
Todas as atividades são claramente definidas.

Quadro 20: MQ9.1: Pontos fracos do método

Pontos Fracos do Método
É centrado somente na norma 15504
É de alto nível em sua abordagem
Fica preso no velho paradigma de que a melhoria contínua se baseia apenas no processo
Mais orientação é necessária em determinadas etapas do método
Não define os fatores situacionais a fim de criar avaliações específicas do contexto
Não deve ser limitado a "S" de software, mas estar aberto a qualquer tipo de domínio
Não deve ser estritamente direcionado para a ISO/IEC 15504-2
Falta de uma ferramenta de apoio ao método
Insuficiente foco em testes iterativos e incorporação do <i>feedback</i> em vários protótipos
Falta de precisão e de forma adequada da distinção entre a capacidade do processo e maturidade organizacional
Incapacidade de resolver questões de arquitetura empresarial
Ausência de harmonização entre modelos de diferentes (mas relacionados) domínios
Aparente falta de consideração do contexto operacional
Visão possivelmente não realista do processo de votação/liberação de uma norma formal
Não é realmente sobre a customização, mas sobre o desenvolvimento de novos modelos
Pode permitir a criação de um modelo válido, mas não a garante
Falta de clarificação do objetivo do método na introdução
Só suporta a utilização de modelos de referência padronizados de processo
Difícil de aprender todas as técnicas necessárias
Falta de exemplos
Faltando alguns grandes corpos de outros trabalhos relevantes
Pouca consideração de reutilização sistemática
Mistura um pouco de orientações escolhidas arbitrariamente sobre a política/procedimentos, e também questões de gestão/técnicas

Deveria começar bem de alto nível com um guia passo a passo que levasse a uma descrição mais detalhada.
Muitas referências cruzadas - o que naturalmente é necessário numa versão em papel. Mas a navegação pelo documento fica bastante difícil.
Há alguns erros de digitação que dificultam a compreensão

Conforme pode ser observado nos Quadro 19 e Quadro 20, diversos pontos fortes e pontos fracos do método foram observados pelos especialistas. Os pontos fortes citados mais recorrentes parecem ser mais relacionados à estrutura do método, sua fundamentação teórica e sua completude. Já os pontos negativos aparentemente são mais relacionados ao seu alinhamento à norma ISO/IEC 15504-2, falta de atenção ao contexto organizacional e operacional e na ausência de iterações explícitas.

Acredita-se que o acréscimo das atividades, tarefas e técnicas indicadas pelos mesmos especialistas (vide pergunta Q3), deve contornar a maior parte destes pontos fracos em versões futuras do método. Até o momento da publicação deste documento, estes pontos fracos ainda não foram corrigidos.

A seção seguinte continua a validação do método apresentando os estudos de caso realizados e os resultados observados na sua realização.

5.3 Estudos de Caso

Conforme definido nos objetivos de avaliação (vide seção 5.1), e nas fases da pesquisa deste trabalho (vide capítulo 1), o método desenvolvido foi utilizado em estudos empíricos ao longo do seu desenvolvimento para caracterizar a sua aplicabilidade. Estudos empíricos são aplicações práticas utilizadas para investigar os efeitos de alguma intervenção sobre o objeto de estudo, sendo adequados para pesquisas aplicadas a: ciências sociais, psicologia, engenharia de software e outras onde o comportamento humano é envolvido (WOHLIN et al., 2000).

Wohlin et al. (2000) estabelece as principais fases para um estudo empírico: Planejamento, Execução e Análise e Interpretação dos dados coletados. Esta seção apresenta dois estudos de caso onde o método foi utilizado. As fases definidas por Wohlin et al. (2000) são apresentadas

nas duas próximas seções: (i) Planejamento e (ii) Execução, Análise e Interpretação.

5.3.1 Planejamento

Uma parte do planejamento de um estudo empírico consiste na definição dos seus objetivos, que foram contemplados pela utilização da abordagem GQM, onde os objetivos, perguntas e medidas são estabelecidas (vide Quadro 15). Entretanto, uma parte importante desta fase de planejamento é a definição do design do estudo empírico, que consiste em escolher qual estratégia de pesquisa será utilizada na aplicação. Neste sentido, estudos de caso são recomendados para estudos empíricos em várias ciências, desde sociologia, medicina e psicologia até engenharia de software (WOHLIN et al., 2000).

Estudo de Caso é uma estratégia de pesquisa que envolve investigação empírica de um fenômeno particular no seu contexto real, utilizando múltiplas fontes de evidência, onde objetivos específicos são definidos e dados são coletados para atingi-los (YIN, 2003; ROBSON, 2002). Eles são especialmente adequados quando o objeto em estudo não pode ser separado do seu contexto, como por exemplo, em estudos de avaliação de projetos ou programas (YIN, 2003).

Desta forma, entende-se que a presente avaliação da aplicação do método de aquisição de conhecimento no desenvolvimento de um SPCMM é realizada no próprio contexto do projeto, mostrando-se assim o estudo de caso como uma estratégia adequada para o problema.

A evidência coletada um estudo de caso é tipicamente de natureza qualitativa e se concentra em um estudo em profundidade (em cada caso) e não em largura (muitos casos) que poderia gerar um entendimento generalizável (ELLIS & LEVY, 2009). Estudos de caso possuem diversas características que os diferenciam de outras estratégias de pesquisa, tais como (ELLIS & LEVY, 2009; YIN, 2003; ROBSON, 2002):

- Um ou poucos casos são observados;
- Aplicação é realizada em um contexto real;
- Não há controle sobre o contexto e as variáveis;
- Não há manipulação de variáveis independentes;
- Coleta de dados é detalhada para cada caso;

- Relação de causalidade não pode ser devidamente estabelecida;
- A amostra é pequena e normalmente não suficientemente representativa.

Existem, no entanto, diferentes tipos de estudos de caso que apresentam variações nestas características (YIN, 2003): (i) Estudos de Caso Descritivos: descrevendo o fenômeno observado; (ii) Exploratórios: procurando investigar inicialmente futuras hipóteses e (iii) Explanatórios: que tentam explicar como e porque determinados fenômenos ocorrem.

Estudo de Caso Exploratório é um caso de estudo abreviado, realizado como um primeiro passo antes de uma investigação de maior escala. Sua função é desenvolver as questões de avaliação, as medidas, design e estratégia analítica para um possível estudo mais amplo. É especificamente útil quando existe alguma considerável incerteza sobre processos, metas e resultados e serem alcançados devido ao estado embrionário da pesquisa (GAO, 1990). Assim, para a presente avaliação, um Estudo de Caso Exploratório foi definido como design do estudo empírico, dada a incipiência da presente pesquisa quando do início do estudo.

Em um Estudo de Caso Exploratório, diferentes técnicas de coleta de dados podem ser utilizadas, tais como (ZELKOWITZ & WALLACE, 1998): Entrevista, Observação, Análise documental, Questionário, etc. Dentre essas, a normalmente menos intrusiva é a Observação, que permite a coleta de dados sem a intervenção direta. Observação é utilizada para documentar atividades, comportamento e aspectos físicos sem interferir ou depender da capacidade e disponibilidade dos participantes de responder a questões (TAYLOR-POWELL & STEELE, 1996). Além de fenômenos como aqueles que envolvem o comportamento humano, a Observação também possibilita a coleta de impressões sobre aspectos físicos e documentais de um processo em execução (TAYLOR-POWELL & STEELE, 1996). Assim, a Observação pode ser apoiada pela Análise Documental como um método para validação cruzada da informação recolhida a partir da Observação em estudos de caso (NOOR, 2008).

A Observação como técnica de coleta de dados é considerada *Participant Observation* (Observação Participante) quando o observador está envolvido no objeto observado (SLACK & ROWLEY, 2001). Também é importante a classificação da Observação quanto à forma

como os dados são coletados: se previamente se decide o que vai ser observado, então a observação é dita *Structured Observation* (Observação Estruturada), caso contrário é denominada não-estruturada.

Como nas duas aplicações o pesquisador esteve direta ou indiretamente envolvido no desenvolvimento do SPCMM em si, a Observação Participante Estruturada é considerada apropriada, apesar das limitações da técnica quanto à isenção na coleta de dados (vide ameaças à validade na seção 4.4). Assim a técnica de coleta de dados utilizada foi a Observação Participante Estruturada apoiada por Análise Documental durante todo o estudo de forma a atender as medidas que respondem às questões estabelecidas (vide seção 4.1), conforme demonstra o Quadro 21.

Quadro 21: Coleta de dados para o Objetivo de Avaliação 2.

Medida	Coleta de Dados
MQ10.1: Detalhamento de como as fases foram executadas.	Observação participante estruturada da aplicação do método apoiada por Análise Documental durante todas as fases do projeto de desenvolvimento do SPCMM
MQ10.2: Impressão subjetiva da extensão da aplicação do método no desenvolvimento do modelo.	Observação participante estruturada das atividades executadas e não executadas apoiada por Análise Documental

Definido o design, a próxima seção apresenta como os dois Estudos de Caso Exploratórios foram conduzidos.

5.3.2 Execução, Análise e Interpretação

Os dois Estudos de Caso Exploratórios que compõe este estudo empírico foram realizados em estágios diferentes da pesquisa, tanto em relação ao amadurecimento do método, quanto em relevância e abrangência da aplicação em si. O primeiro estudo de caso foi realizado no ano de 2009 no desenvolvimento de um SPCMM para o domínio de SaaS - *Software as a Service* (Software como Serviço), no contexto de uma dissertação de mestrado (CANCIAN, 2009). O segundo é aplicado no ano de 2010 no desenvolvimento de um SPCMM para o domínio de software embarcado em dispositivos médicos, no contexto de um projeto de pesquisa financiado pela EU (MEDI SPICE, 2011). Ambos são individualmente apresentados a seguir, sendo que para cada um são apresentados os dados coletados em relação às medidas definidas para o

Objetivo de Avaliação 2. Como se trata de uma observação participante estruturada, a análise e interpretação dos dados coletados já são apresentadas em conjunto com as próprias observações coletadas.

Estudo 1 - Desenvolvimento de um SPCMM para o domínio de SaaS

Neste primeiro estudo de caso Exploratório o método de aquisição de conhecimento para customização de SPCMMs foi aplicado ainda em estágio embrionário, em paralelo com o seu desenvolvimento, a partir do início de 2009 no desenvolvimento de um SPCMM para o domínio de SaaS - *Software as a Service* (Software como Serviço) (CANCIAN, 2010).

Contexto

SaaS é uma solução onde o software é oferecido como um serviço e é desenvolvido utilizando-se uma arquitetura orientada a serviços (SOA). Neste tipo de arquitetura o usuário de um software acessa um serviço provido por um fornecedor em um determinado local, ao invés de utilizá-lo diretamente no seu ambiente físico. Desta forma, o cenário SaaS possui necessidades específicas de qualidade, tais como: a disponibilidade, segurança e continuidade de serviço, devido a sua característica de produto de software distribuído como serviços.

Assim, a partir destes fatores foi constatada a necessidade de um SPCMM representando o conhecimento na forma de melhores práticas para o domínio específico de desenvolvimento de SaaS. Desta forma, como não foram encontradas fontes de conhecimento já estruturadas no domínio, para a definição de um SPCMM para SaaS, percebeu-se que seria necessário realizar a aquisição de conhecimento de forma sistemática, dos especialistas e outras fontes de conhecimento no domínio, no intuito de definir as melhores práticas de desenvolvimento de software para o domínio.

O SPCMM SaaS foi então desenvolvido por um grupo de pesquisadores da UFSC, envolvendo especialistas do domínio SaaS e especialistas de SPI. A forma como a aquisição de conhecimento foi realizada utilizando-se o método é detalhada a seguir por meio da apresentação dos dados coletados para as duas medidas definidas. O

desenvolvimento do modelo é também detalhadamente apresentado em (CANCIAN et al., 2010), (CANCIAN, 2010) e (CANCIAN, 2009).

MQ10.1: Detalhamento de como as fases foram executadas.

A partir da necessidade inicialmente percebida de um SPCMM específico para o domínio de SaaS foi realizado o desenvolvimento do modelo. O autor participou indiretamente do desenvolvimento na definição do método a ser utilizado e no acompanhamento e monitoração do desenvolvimento durante todo o processo. Desta forma, a observação de como cada uma das fases foi realizada foi possível e diversas informações foram coletadas a respeito de como, na prática, o método foi utilizado.

Como o método ainda estava em desenvolvimento durante a realização deste primeiro estudo, as atividades das fases não foram executadas na mesma seqüência definida na versão atual do método. A Figura 37 apresenta como as principais atividades foram executadas nesse estudo de caso. Conforme pode ser observado na Figura 37 as fases 4 e 5 não foram executadas pois o modelo não foi, até o momento, efetivamente colocado em uso, apesar de já publicado.

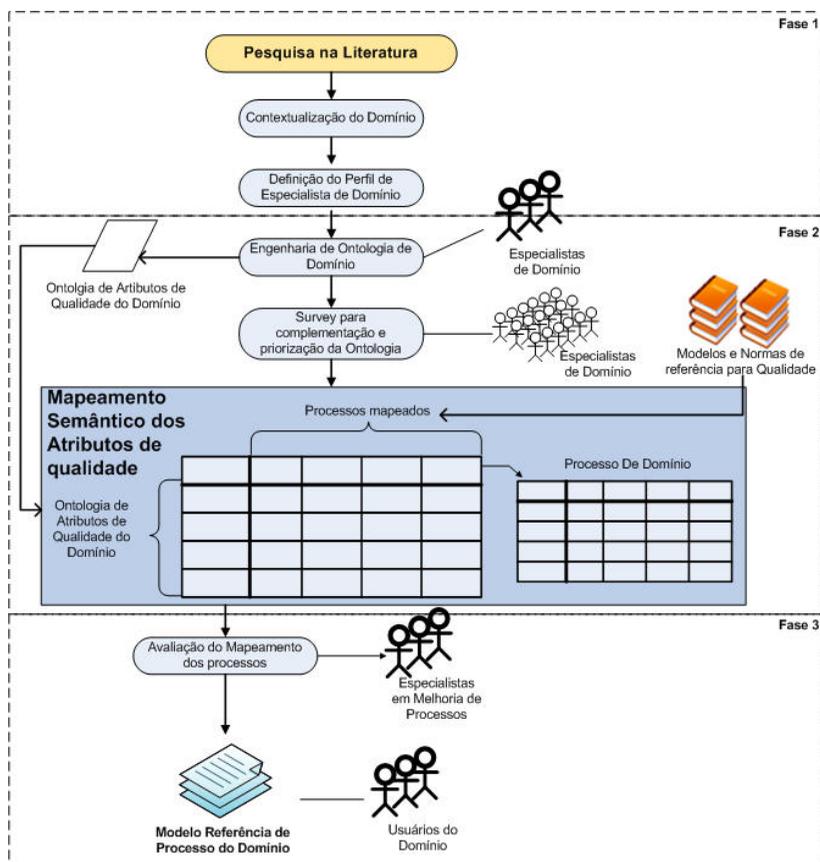


Figura 37: Atividades executadas no Estudo 1.

Fonte: Baseado em (CANCIAN et al., 2010)

Na seqüência, as observações coletadas em cada fase são apresentadas.

Fase 1: Identificação do conhecimento: nesta primeira fase, a familiarização com o domínio foi realizada na forma de uma revisão de literatura, a partir da qual as fontes de conhecimento também foram identificadas nos autores das publicações encontradas. Como consequência foi reafirmada a necessidade de desenvolvimento de um novo modelo pela percepção, por meio da revisão da literatura, da inexistência de um modelo específico para o domínio. O escopo foi

então definido como sendo focado inicialmente no PRM, ou seja, não foi desenvolvida a dimensão de capacidade ou maturidade neste primeiro estágio. Por tratar-se de uma pesquisa para dissertação de mestrado, o grupo de trabalho para o desenvolvimento do modelo foi definido de maneira informal, compondo-se principalmente da pesquisadora responsável assessorada por especialistas de SPI e SaaS.

Fase 2: Especificação do conhecimento: a fase 2 iniciou-se com o levantamento dos critérios de qualidade para o domínio. Isso se deu por meio de várias entrevistas individuais com seis especialistas de domínio selecionados inicialmente por critério de proximidade, com o objetivo de definir uma ontologia de atributos de qualidade de software para o domínio de SaaS. O resultado destas entrevistas foi uma lista de critérios de qualidade no domínio na forma de uma ontologia ainda não hierarquizada. A nomenclatura e as definições dos critérios de qualidade foram harmonizadas apoiando-se nas definições encontradas na literatura, tomando-se por base a revisão da literatura realizada na fase 1. Os critérios de qualidade obtidos nas entrevistas foram também completados por critérios de qualidade encontrados na revisão da literatura. A Figura 38 mostra um extrato do original em inglês dos critérios de qualidade elicitados, conforme relatado em (CANCIAN et al., 2010).

Criteria	Description	Source
Acquisition	Refers to the acquisition process from the point of view of the customer.	Interviewee 1 Interviewee 5
Availability	Refers to the availability of the service for immediate use.	Interviewee 2 Interviewee 4
Change control	Refers to the change control process in order to minimize the impact of changes.	Interviewee 4 Interviewee 5
Help desk	Refers to the support process focusing on the way customers will be assisted when using the service.	Interviewee 2 Interviewee 3 Interviewee 4
Infrastructure capability	Refers to the extent to which the available infrastructure is adequate and sufficient.	Interviewee 4
Interoperability	Refers to the capability of communicating as transparently as possible to other systems.	Interviewee 1
Maintenance	Refers to the maintenance process in order to perform changes in software according to requests.	Interviewee 1 Interviewee 2 Interviewee 4 Interviewee 5
Prevision of continuity of service	Refers to extent to which an organization is able to provide the continuation of technical resources and IT systems / services	Interviewee 4 Interviewee 6

Figura 38: Critérios de qualidade e fontes

Fonte: (CANCIAN et al., 2010)

Os critérios de qualidade foram então submetidos à validação de um grupo maior de especialistas do domínio identificados na fase 1 por meio da realização de um *survey*. O objetivo consistiu, além de validar os critérios, também de estabelecer uma ordem de prioridade entre os critérios de qualidade observados. Este *survey* foi suportado pela ferramenta Lime Survey (<http://www.limesurvey.org>). Assim, 280 especialistas em SaaS foram convidados a responder um questionário onde eles avaliavam a corretude e completude dos critérios, bem como a sua ordem de prioridade. Como resultado, obteve-se a lista priorizada dos critérios de qualidade, conforme pode ser observado na Figura 39.

	ESSENTIAL	VERY IMPORTANT	IMPORTANT	UNNECESSARY	ESSENTIAL	VERY IMPORTANT	IMPORTANT	UNNECESSARY	TOTAL
	4	3	2	1	weight x responses number				
	Responses number								
Integrity	57	21	5	0	228	63	10	0	301
Reliability	48	30	4	1	192	90	8	1	291
Security	45	31	7	0	180	93	14	0	287
Accessibility	49	21	13	0	196	63	26	0	285
Requirements development and management	41	29	13	0	164	87	26	0	277
Infrastructure capability	28	48	6	1	112	144	12	1	269
Quality control	32	37	14	0	128	111	28	0	267
Acquisition	28	40	15	0	112	120	30	0	262
Testing	26	44	12	1	104	132	24	1	261
Performance	22	47	13	1	88	141	26	1	256
Utilization of standards	29	34	18	2	116	102	36	2	256
Change control	26	39	16	2	104	117	32	2	255

Figura 39: Critérios de qualidade priorizados

Fonte: (CANCIAN et al., 2010)

Tendo-se os critérios de qualidade validados e priorizados, o próximo passo foi a realização de um mapeamento semântico entre os critérios de qualidade e normas e modelos genéricos de qualidade de processos de software. Este mapeamento semântico procura identificar nas práticas das normas e modelos, quais atendem aos atributos de qualidade estabelecidos e quais são as carências de melhores práticas. Para viabilizar este mapeamento semântico foi utilizada uma abordagem adaptada da metodologia QFD - *Quality Function Deployment* (AKAO & MIZUNO, 1994), seguindo as experiências relatadas em (MATOOK

& INDULSKA, 2009) e em (RICHARDSON, 1997). A Figura 40 mostra um extrato do original em inglês do mapeamento resultante. Cada critério de qualidade é associado aos processos, indicando se o processo é: essencial, muito importante, importante, fracamente importante ou desnecessário para o respectivo critério de qualidade

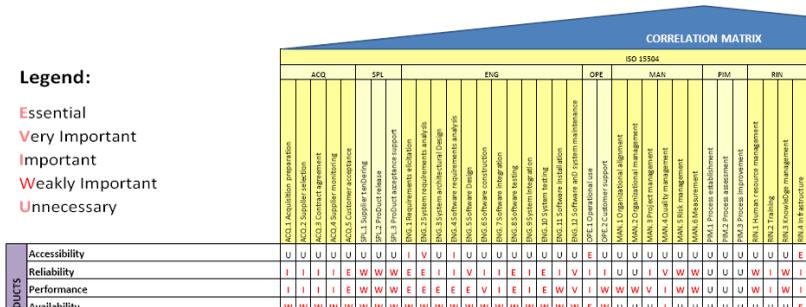


Figura 40: Extrato do mapeamento entre critérios e melhores práticas
 Fonte: (CANCIAN et al., 2010)

Fase 3: Refinamento do Conhecimento: a partir do mapeamento estabelecido, um grupo de especialistas em melhoria de processos (SPI) foi envolvido na validação do mapeamento resultante. O mapeamento semântico foi validado por meio da análise individual de cada conexão identificada entre critérios de qualidade e melhores práticas em seções coletivas de validação. Após diversas iterações, o mapeamento validado foi documentado para a sua publicação, utilizando como arquitetura para documentação das melhores práticas a mesma arquitetura da norma ISO/IEC 15504-5 (ISO/IEC, 2006). Um site web (vide Figura 41) foi então desenvolvido para suportar a publicação do modelo resultante (<http://www.gsigma.ufsc.br/~cancian/guide>). Neste site o modelo é apresentado de forma interativa e o usuário do modelo pode navegar tanto pelas melhores práticas documentadas, como a partir dos critérios de qualidade e seus respectivos mapeamentos identificados. Também uma área para contato com os principais autores do modelo foi desenvolvida para possibilitar a coleta de *feedback* (na forma de comentários, correções, etc.) a respeito do futuro uso modelo.

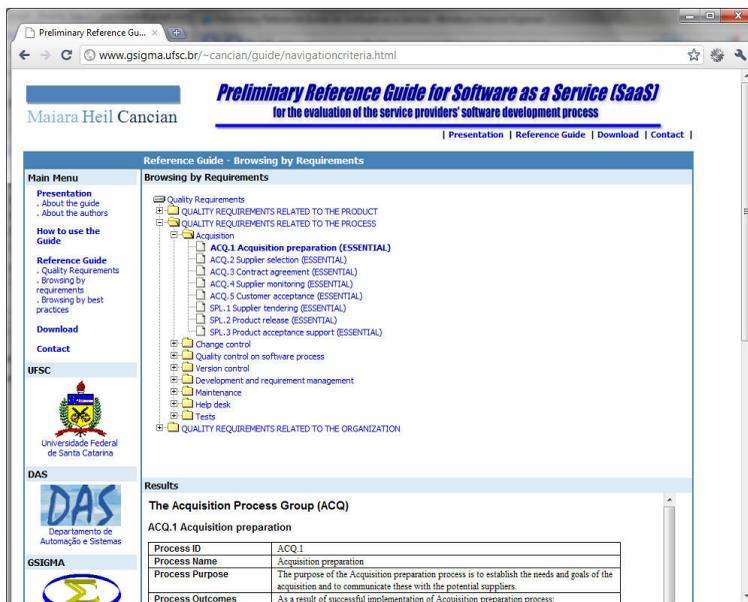


Figura 41: Site web onde o modelo SaaS está publicado

Fonte: (CANCIAN, 2010)

Conforme já salientado, a **Fase 4: Uso do Conhecimento** e a **Fase 5: Evolução do Conhecimento** não foram executadas pois, até o momento, o modelo ainda não foi utilizado em um ambiente real de desenvolvimento de software no domínio de SaaS.

MQ10.2: Impressão subjetiva da extensão da aplicação do método

Em atendimento à medida MQ10.2, o Quadro 22 mostra um resumo de como cada fase foi executada.

Quadro 22: Resumo da utilização do método observada no estudo 1.

Fases do método	Resumo de como foi executada
Fase 1: Identificação do conhecimento	A familiarização com o domínio é realizada na forma de uma revisão de literatura e as fontes de conhecimento foram identificadas a partir da mesma revisão.
Fase 2: Especificação do conhecimento	Baseada na elicitação dos critérios de qualidade do domínio por meio de entrevistas individuais e um <i>survey</i> . Um mapeamento semântico é realizado entre os critérios e melhores práticas encontradas nos modelos genéricos.

Fase 3: Refinamento do conhecimento	O mapeamento semântico é validado por especialistas em SPI e o modelo é documentado e publicado em um site.
Fase 4: Uso do conhecimento	Não executada.
Fase 5: Evolução do conhecimento	Não executada.

A partir da observação da aplicação do método, conforme demonstra o Quadro 22, foi possível perceber que o método pôde ser aplicado na prática, mesmo ainda estando em desenvolvimento. No estágio em que se encontrava durante este primeiro estudo de caso, o método não era tão prescritivo, pois a descrição das atividades ainda estava em alto nível, possibilitando que adaptações fossem feitas durante o percurso. Um ponto forte observado foi o alto envolvimento dos especialistas de domínio, especialmente no fornecimento dos critérios de qualidade para o domínio, que vieram a servir de base para o desenvolvimento do SPCMM.

Estudo 2 - Desenvolvimento de um SPCMM para o domínio de Dispositivos Médicos - Medi SPICE

Neste segundo estudo de caso exploratório, o método foi aplicado no desenvolvimento do Medi SPICE. O Medi SPICE é um projeto internacional que possui suporte financeiro da União Européia (EU) para o desenvolvimento de um SPCMM contendo as melhores práticas de processo de software no domínio de software embarcado em dispositivos médicos (MCCAFFERY, DORLING & CASEY, 2010). O projeto envolve doze empresas e sete universidades de diferentes partes da Europa e visa gerar um *Process Reference Model* (PRM) e um *Process Assessment Model* (PAM) que permitam harmonizar os modelos genéricos e as normas específicas atualmente existentes para software em dispositivos médicos.

Contexto

O desenvolvimento de software embarcado em dispositivos médicos possui diversas especificidades que o diferenciam do desenvolvimento de software em outros domínios especialmente pelos riscos envolvidos na utilização dos diversos dispositivos médicos que possuem software (MCCAFFERY & DORLING, 2010). Devido a estes

fatores o desenvolvimento de software neste domínio é uma atividade fortemente regulada por diversos órgãos, por meio de normas como: AAMI/IEC 62304, FDA *and European guidelines*, ISO 14971, IEC 60601-1-4, ISO 13485, etc. (MCCAFFERY, DORLING & CASEY, 2010).

Como a quantidade de regulamentação é grande, a maior parte dos fornecedores de soluções de software nesse domínio emprega um esforço considerável tentando alinhar-se às exigências destes órgãos regulatórios para poderem comercializar os seus produtos (MCCAFFERY, DORLING & CASEY, 2010). Assim existe um grande volume de conhecimento a respeito de como desenvolver software para este domínio específico disperso em diversas fontes, tanto nas normas quanto nos profissionais que atualmente desenvolvem produtos de software neste domínio.

SPCMMs genéricos de melhores práticas de desenvolvimento de software, tais como CMMI-DEV (SEI, 2010) e ISO/IEC 15504-5 (ISO/IEC, 2006), por possuírem um escopo de cobertura amplo em termos de processos, também não contêm as práticas necessárias para o atendimento de todas as necessidades específicas de qualidade do domínio, estabelecidas nas normas citadas. Um exemplo disso é a exigência de um documento que detalhe o histórico das modificações de design de todos os componentes envolvidos no desenvolvimento do software embarcado em dispositivos médicos, prática que não está contida em nenhum dos citados SPCMMs genéricos (MCCAFFERY, DORLING & CASEY, 2010).

Assim, a partir destes fatores foi constatada a necessidade de um SPCMM representando o conhecimento na forma de melhores práticas para o domínio específico de desenvolvimento de software para dispositivos médicos. Observou-se, no entanto, que as fontes de conhecimento inicialmente encontradas no domínio constituem-se de normas que, apesar de incluírem conhecimento sobre como desenvolver software no domínio, não se apresentam de forma estruturada o suficiente para a definição de um SPCMM. Isto levou à constatação de que seria necessário realizar a aquisição de conhecimento de forma sistemática, dos especialistas e destas outras fontes de conhecimento no domínio (tais como as citadas normas), com o intuito de definir as melhores práticas de desenvolvimento de software para o domínio. A forma como esta aquisição deste conhecimento foi realizada é detalhada a seguir.

MQ10.1: Detalhamento de como as fases foram executadas.

O método foi aplicado no desenvolvimento do PRM do Medi SPICE no período de janeiro a dezembro de 2010. Durante esse período o autor realizou estágio no exterior (PDEE/CAPES), com o objetivo de pilotar a aplicação do método em um projeto real. Assim, o desenvolvimento do PRM do Medi SPICE foi acompanhado pelo autor durante todo o período citado, como pesquisador visitante no *Regulated Software Research Group* no *Dundalk Institute of Technology* (DKIT, 2010) - Irlanda. Pelo fato do projeto ser formado por uma equipe grande, a observação da aplicação do método no desenvolvimento do modelo foi realizada por meio de participação em reuniões, análise de documentos e revisão de demais artefatos produzidos durante o período. A seguir são apresentados detalhes de como cada fase do método foi executada.

Fase 1: Identificação do conhecimento: A familiarização com o domínio já havia sido realizada antes do início do projeto, pois vários membros do grupo já haviam realizado diversos trabalhos de melhoria de processos no domínio, inclusive com diversos artigos publicados a respeito. Neste sentido, a documentação sobre o domínio foi realizada na forma dos diversos artigos e livro publicados pelos membros do grupo, tais como: (MCCAFFERY & DORLING, 2010; MCCAFFERY, DORLING & CASEY, 2010; BURTON et al., 2009; MCCAFFERY, BURTON & RICHARDSON, 2009; MCCAFFERY & COLEMAN, 2007). As fontes de conhecimento foram identificadas de duas formas: (i) em uma revisão de literatura, onde as principais normas associadas ao desenvolvimento de software no domínio foram listadas, e; (ii) por meio da identificação de especialistas do domínio, alguns dos quais já membros do grupo e outros posteriormente convidados a participar. O escopo e os objetivos do modelo já haviam sido previamente definidos quando da obtenção de suporte financeiro junto à EU. O escopo e as fontes de conhecimento identificadas foram então documentados no documento *Modus Operandi* do projeto. Este mesmo documento foi ainda completado com a definição formal do grupo de trabalho, onde cada um dos papéis foi identificado e as políticas de funcionamento do grupo foram formalmente definidas.

Fase 2: Especificação do conhecimento: a segunda etapa do desenvolvimento do modelo iniciou-se com a definição da sua

arquitetura. Foi identificada, conforme indicada no método, a necessidade de utilização de uma arquitetura que pudesse atender aos requisitos da norma ISO/IEC 15504-2 para um PRM de forma a estar alinhado a esta norma. A definição da arquitetura é uma atividade muito importante, pois ela estabelece como as melhores práticas serão efetivamente representadas e como as dimensões de processos e de maturidade/capacidade serão definidas. Em uma reunião do grupo estendido de trabalho foi definida a arquitetura de representação das melhores práticas na forma de título, descrição e propósito de cada processo. Para a dimensão de capacidade foi identificada a necessidade de utilização do modelo indicado na parte 2 da norma ISO/IEC 15504, inicialmente sem alterações. Foi também decidido que a definição de uma estrutura de maturidade organizacional seria realizada futuramente.

A proposta de arquitetura foi então revisada internamente pelo grupo e submetida à aprovação do grupo diretor. Nesta avaliação foi identificada uma dificuldade na representação das melhores práticas, quanto à independência necessária entre as duas dimensões do modelo. Isso ocorre porque, no domínio de dispositivos médicos, os softwares são classificados de acordo com um índice de risco representado pelo equipamento em categorias A, B ou C (por exemplo: um equipamento que somente exibe imagens médicas tem um índice diferente do *firmware* embarcado em um marca-passo). A consequência deste fato, em termos da estrutura de representação das melhores práticas, consiste que alguns processos ou práticas somente são executados para dispositivos de determinadas classes. Sendo que a arquitetura do primeiro nível de capacidade da norma ISO/IEC 15504-2 indica que *todos* os processos são executados, em termos de práticas-base, resultados e produtos de trabalho, foi percebida pelo grupo diretor uma inconsistência na representação. A inconsistência identificada ocorre porque um critério de classificação de produto altera a avaliação da dimensão de capacidade dos processos. A solução adotada foi a representação destas melhores práticas, associadas a categorias diferentes de produtos, na forma de práticas específicas, que podem ou não ser executadas de acordo com a categoria do produto desenvolvido, sem, no entanto, provocar alterações na dimensão de capacidade/maturidade.

Definida a arquitetura do modelo, um amplo trabalho foi então realizado nos primeiros meses de 2010 na análise dos documentos das principais normas e modelos. Um mapeamento semântico foi realizado, tomando como base os processos do PAM da norma ISO/IEC 15504-5 e

a partir destes processos, todos os conceitos presentes nas demais normas foram harmonizados e incluídos em um documento de mapeamento, que mais tarde tornou-se o PRM do modelo. No mapeamento semântico, a técnica de mapas mentais (BUZAN & BUZAN, 1990) foi utilizada pela equipe para harmonizar as arquiteturas e os conceitos das diferentes normas. A Figura 42 mostra parte de um exemplo de utilização de mapas mentais no mapeamento semântico da norma AAMI/IEC 62304 para o PAM da norma ISO/IEC 15504-5. A ferramenta utilizada para o design dos mapas mentais foi o software livre *SciPlore MindMapping* (<http://sciplore.org>). Esta ferramenta auxilia a análise de documentos no formato *pdf*, pois facilita na montagem do mapa da estrutura dos documentos ou de uma lista de documentos a serem analisados.

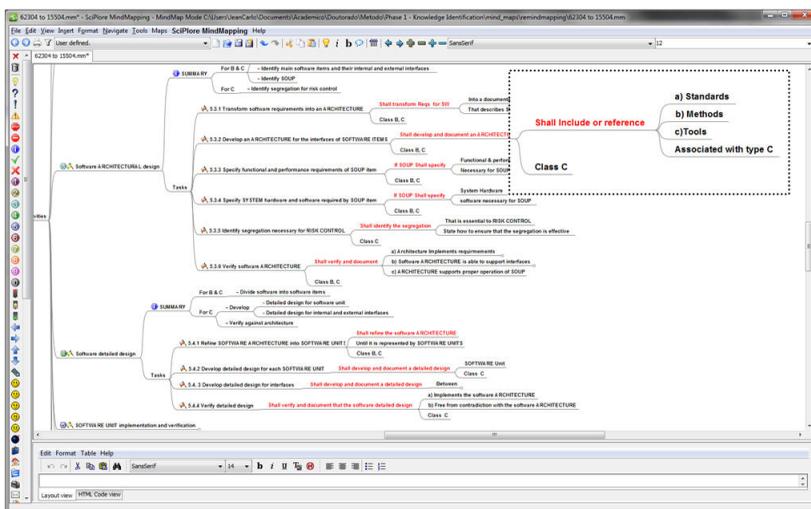


Figura 42: Utilização de Mapas mentais na Fase 1.

Fonte: Parte da documentação restrita do projeto Medi SPICE, gentilmente cedida pelo prof. Val Casey, PhD.

O mapeamento desenvolvido foi então formatado para adequar-se à arquitetura definida. Cada processo foi documentado incluindo título e propósito. Os principais *stakeholders*, indicados como fontes de conhecimento na primeira fase, foram envolvidos nesta etapa, especificamente incluindo especialistas do domínio (representantes de empresas de desenvolvimento de software para dispositivos médicos)

que forneceram os critérios de qualidade para o domínio, a partir da análise da documentação das citadas normas. Este conhecimento foi integrado ao mapeamento realizado anteriormente, completando então, por meio de diversos ciclos de revisões, as melhores práticas de desenvolvimento de software para o domínio.

De forma iterativa, cada nova versão do documento contendo os processos (futuro PRM) foi enviada para revisão dos especialistas de domínio, utilizando a técnica *Delphi* de forma adaptada, onde cada especialista revisava o documento, adicionando comentários quando necessário, até que o consenso sobre o conteúdo do documento foi obtido. O resultado desta fase foi, então, a primeira versão *draft* do PRM do modelo.

Fase 3: Refinamento do Conhecimento: a terceira fase consistiu na avaliação do conteúdo de forma a identificar se o modelo realmente contém o conhecimento a respeito das melhores práticas de desenvolvimento de software para o domínio.

O primeiro passo neste sentido consistiu em avaliar as características de qualidade do modelo, sendo executado na forma de uma leitura sistemática do documento do modelo por parte dos membros da equipe. Como resultado deste passo, uma inconsistência foi percebida em relação à compatibilidade do modelo às normas que serviram de fonte de conhecimento para o seu desenvolvimento. Algumas disposições normativas, ou seja, consideradas obrigatórias nas normas originais, foram incorporadas como notas explicativas nas melhores práticas do modelo, o que, de acordo com a arquitetura definida, não mais representavam características obrigatórias. A consequência direta consistia no não alinhamento às normas originais, ou seja, as fontes de conhecimento não haviam sido semanticamente preservadas. Este problema foi então corrigido.

Também foram realizadas diversas revisões na forma de *focus groups*, onde diversos membros do grupo e especialistas discutiam o conteúdo do texto até que o consenso fosse atingido. Nestes grupos, diversas pequenas correções textuais foram realizadas melhorando a qualidade do texto das melhores práticas. Com base no *feedback* obtido, o modelo foi refinado de forma iterativa, até que fosse alcançado um consenso entre os membros do grupo de trabalho. Para isso foi necessária, muitas vezes, a discussão, negociação e resolução das divergências técnicas significativas de forma a gerar um modelo que seja aceito e amplamente utilizado pela comunidade de desenvolvedores de software para dispositivos médicos.

Quando da elaboração desta tese, o desenvolvimento do Medi SPICE encontrava-se no estágio de votação e aprovação final do documento do modelo, sendo que até o momento a versão final do documento não havia sido ainda publicada. Entretanto algumas outras atividades das fases seguintes já foram realizadas ou planejadas, conforme indicado na descrição a seguir.

Fase 4: Uso do Conhecimento: a terceira fase não foi inteiramente executada no projeto até o momento, uma vez que, como indicado na seção anterior, o texto final do modelo ainda não foi oficialmente publicado. Entretanto algumas atividades já foram executadas previamente de forma a preparar a arquitetura para a publicação do documento e possibilitar a suporte necessário ao seu uso.

Um portal web foi desenvolvido (vide Figura 43) para suportar a comunidade de desenvolvimento e uso do modelo (MEDI SPICE, 2010), utilizando como base a plataforma de redes sociais NING (<http://www.ning.com/>). Neste portal são publicados os documentos relevantes para a comunidade e um recurso de fórum também permite a interação entre os membros da equipe. Cada membro pode construir a sua rede de relacionamento dentro da comunidade de usuários e desenvolvedores do Medi SPICE e também é possível criar um blog pessoal para discutir temas relacionados ao modelo. A política de uso do modelo também já foi definida inicialmente no documento *Modus Operandi* do projeto, onde é estabelecido o tipo de suporte que será fornecido ao usuário final do modelo por meio do portal da comunidade.

Figura 43: Portal do Medi SPICE.

Fonte: (MEDI SPICE, 2010)

A avaliação do modelo em uso ainda não foi realizada, pois, conforme já mencionado, o documento final do modelo ainda não foi publicado. Entretanto a realização de *trials* de uso do modelo já está planejada, contando com o envolvimento de empresas de desenvolvimento de software para dispositivos médicos na Irlanda e também em outros países da EU.

Fase 5: Evolução do Conhecimento: esta fase final que prevê a evolução contínua do modelo ainda não foi realizada. Entretanto o portal que dará suporte à sua evolução já existe, conforme citado, e o canal de comunicação com os usuários, por meio do qual será possível receber eventuais futuras solicitações de mudança e correções dos usuários do modelo já está devidamente definido.

MQ10.2: Impressão subjetiva da extensão da aplicação do método

Em atendimento à medida MQ10.2, o Quadro 23 apresenta um resumo de como a aquisição de conhecimento foi realizada no estudo 2.

Quadro 23: Resumo da utilização do método observada no estudo 2.

Fases do método	Resumo de como foi executada
Fase 1: Identificação do conhecimento	Familiarização com o domínio realizada anteriormente e documentada em artigos publicados. As fontes de conhecimento identificadas por meio de revisão de literatura e identificação de especialistas de domínio, documentados no documento de <i>modus operandi</i> do projeto, onde também é definido o grupo de trabalho.
Fase 2: Especificação do conhecimento	A arquitetura foi definida alinhada aos requisitos da norma ISO/IEC 15504-2 com pequenas alterações após a revisão. Um mapeamento semântico foi realizado entre as normas específicas para o domínio e o modelo de referência da ISO/IEC 15504-5, onde foram utilizados <i>mind maps</i> para harmonização das diferentes estruturas e conceitos. A versão <i>draft</i> do modelo foi revisada por meio de diversos ciclos iterativos.
Fase 3: Refinamento do conhecimento	Revisão inicial da versão <i>draft</i> do modelo realizada por meio de leitura sistemática. <i>Focus groups</i> foram utilizados para avaliação das características de qualidade do modelo. Documento do PRM do modelo encaminhado para análise e votação.
Fase 4: Uso do conhecimento	Executada parcialmente. Desenvolvido o portal web do projeto para interação entre os membros da equipe de desenvolvimento e os futuros usuários do SPCMM. A política de uso foi definida no documento <i>modus operandi</i> do projeto.
Fase 5: Evolução do conhecimento	Executada parcialmente. Canal para recebimento de solicitações de mudança e correções foi definido no portal web do projeto.

A observação da extensão de como o método foi aplicado permite perceber que, neste segundo estudo de caso, o método já se encontrava melhor definido. Outro ponto importante consiste no fato de que neste domínio outras fontes de conhecimento, como normas e modelos, tiveram papel bastante relevante na aquisição do conhecimento sobre as necessidades de qualidade do domínio, somente sendo completadas pela participação direta de especialistas do domínio no grupo. A presença de pesquisadores experientes no desenvolvimento de SPCMMs no grupo de desenvolvimento do Medi SPICE também é um fator relevante, pois a

forma como as técnicas foram aplicadas na aquisição de conhecimento foi fortemente influenciada por esta experiência anterior, vindo a enriquecer o método com outras técnicas e formas de representação (vide o exemplo da Figura 42).

5.4 Ameaças à validade

Como qualquer resultado de pesquisa, a avaliação do método apresentada neste capítulo por meio da realização do *Expert Panel* e do Estudo Empírico possui diversas limitações à sua validade. Nesta seção são apresentadas as mais relevantes identificadas pelo autor.

5.4.1 Ameaças à validade do Expert Panel

Uma das principais limitações inerentes à realização de um *Expert Panel* é a pequena amostra utilizada. Isso se deve ao fato de que este tipo de técnica não visa cobrir uma amostra relevante da população-alvo em um estudo em largura do objeto, mas sim, realizar um estudo em profundidade do objeto a partir de um grupo reduzido de especialistas adequadamente qualificados para tal. Entretanto, pode-se afirmar que a amostragem aplicada (38 especialistas), apesar de pequena, é bastante focada, pois foram convidados somente especialistas com reconhecida publicação relevante na área, e, inclusive, autores daqueles mesmos trabalhos que serviram como base para a abordagem indutiva da pesquisa. Ainda assim, a pequena amostra reduz a possibilidade de generalização dos resultados e conclusões assinaladas.

Outra limitação importante da pesquisa realizada no *Expert Panel* diz respeito à baixa taxa de respostas. Dos 38 especialistas convidados, somente 39,5% (15) responderam ao questionário. Esta baixa taxa de respostas pode ser devida a diversos fatores, como o período do ano no qual a pesquisa foi realizada (novembro-dezembro). Entretanto, uma taxa de retorno média de 55,6%, com variação de +/-19,7% é considerada normal neste tipo de pesquisa (BARUCH, 1999).

A amostragem realizada somente com pesquisadores que têm publicação científica relevante limita as conclusões obtidas. Essa limitação gera clara tendência ao ambiente acadêmico em detrimento da indústria de desenvolvimento de software. Ainda assim, a grande parte das iniciativas de customização de SPCMMs é originada no ambiente acadêmico, conforme pode ser observado nas filiações dos autores dos

modelos apresentados em (WANGENHEIM et al., 2010b) e na lista de modelos do Apêndice A.

Ainda em relação à amostragem, o fato de que foram selecionados como especialistas os autores dos mesmos trabalhos que deram base ao desenvolvimento do método, poderia gerar uma tendência positiva à aprovação do método por parte dos especialistas. Apesar disso, conforme pode ser observado, um grande número de oportunidades de melhoria no método foi relatado pelos especialistas durante a realização do painel.

5.4.2 Ameaças à validade do Estudo Empírico

A utilização de Estudos de Caso Exploratórios na realização do estudo empírico leva à incorporação das limitações típicas deste tipo de técnica. Os resultados de estudos de caso são normalmente difíceis de generalizar e interpretar (YIN, 2003; ZELKOWITZ & WALLACE, 1998). A atenção demasiada aos detalhes do estudo, característica deste tipo de técnica, pode levar a não observação de aspectos relevantes de um possível quadro geral de cada caso, ou dos casos estudados em conjunto. Assim, os efeitos de uma situação típica podem ser adequadamente observados, mas, em geral, não é possível se estabelecer uma generalização com confiabilidade para outras situações (NOOR, 2008; WOHLIN et al., 2000; ZELKOWITZ & WALLACE, 1998).

A utilização da Observação Participante na coleta de dados, apesar de ter sido identificada como adequada para as características específicas dos casos estudados, também possui uma série de limitações. Nesta técnica existe a dificuldade inerente de se registrar a experiência discretamente como ela se apresenta (SLACK & ROWLEY, 2001), especialmente no que diz respeito ao envolvimento direto ou indireto do pesquisador com o objeto estudado. Na utilização deste tipo de técnica é difícil separar as observações subjetivas das objetivas devido ao forte envolvimento do pesquisador. Este envolvimento pode comprometer as conclusões obtidas e, especialmente, a possibilidade de generalização das observações a outros casos similares, onde o pesquisador não esteja envolvido.

Outra importante limitação no estudo diz respeito ao fato de que o método somente foi aplicado de maneira incompleta, pois as fases 4 e 5 foram somente parcialmente executadas no estudo 2. Isso torna difícil a avaliação da aplicabilidade das fases finais e da eficácia das técnicas e ferramentas recomendadas. Também na literatura, de forma geral as

atividades envolvidas nessas últimas fases do desenvolvimento de SPCMMs, onde o conhecimento adquirido é utilizado e evoluído, são menos freqüentemente executadas (WANGENHEIM et al., 2010a), conforme ilustra a Figura 18 no capítulo 3.

5.5 Conclusões do Capítulo

Este capítulo apresenta a validação do método de aquisição de conhecimento para customização de SPCMMs. A validação inicia-se com o estabelecimento dos objetivos de avaliação, a partir da utilização da técnica GQM, tomando-se por base os objetivos geral e específicos deste trabalho. Como forma de alcançar os objetivos de avaliação estabelecidos, são realizados um *Expert Panel* e dois Estudos de Caso, seguindo os procedimentos indicados na literatura.

O *Expert Panel* procura avaliar a adequação do método por meio da coleta de opinião de especialistas. Os resultados observados, de forma geral, fornecem uma primeira impressão que, de acordo com as opiniões coletadas dos especialistas, o método é adequado para a aquisição de conhecimento na customização de SPCMMs, apesar de apresentar oportunidades de melhoria na sua versão atual.

Os estudos de caso realizados procuram caracterizar a aplicabilidade do método na sua aplicação em ambientes reais de desenvolvimento de SPCMMs. Por meio de observação, impressões subjetivas coletadas do uso prático do método indicam que os seus diversos componentes, de maneira geral, podem ser aplicados com sucesso na customização de SPCMMs.

Na validação do método apresentada neste capítulo, várias oportunidades de melhoria foram percebidas, especialmente considerando que esta pesquisa representa um primeiro esforço na direção de sistematizar o processo de aquisição de conhecimento na customização de SPCMMs. Da mesma forma, diversas são as limitações às quais a validação apresentada neste capítulo está sujeita, tais como as limitações típicas dos métodos e técnicas utilizados. Entretanto, após a validação realizada, é possível observar indicações de que o método desenvolvido atende ao seu objetivo na aquisição de conhecimento para customização de SPCMMs.

6 CONCLUSÕES E TRABALHOS FUTUROS

Esta tese apresenta um Método de Aquisição de Conhecimento para Customização de SPCMMs. O método é desenvolvido em uma abordagem indutiva, onde, após uma revisão dos conceitos fundamentais de KE e SE, a forma como os SPCMMs têm sido desenvolvidos é investigada por meio de uma revisão sistemática de literatura e um *survey* com pesquisadores envolvidos no desenvolvimento deste tipo de modelo. O desenvolvimento do método, então, se dá sob a perspectiva da KE, tomando por base as abordagens, *frameworks* e metodologias inicialmente propostas para desenvolvimento de normas e modelos de capacidade/maturidade de processos de software, e suprido com métodos e técnicas relacionados à aquisição de conhecimento. Por fim, o método é validado sob duas perspectivas: em relação à sua adequação por meio de um *Expert Panel* e em relação a sua aplicabilidade por meio de estudos de caso ao longo do seu desenvolvimento.

A KE foi inicialmente concebida como uma forma de transferir o conhecimento de um especialista para um ambiente computacional. No atual paradigma da KE, entretanto, o contexto do conhecimento tem também um papel importante na construção da representação do conhecimento. Assim, diversas abordagens vêm sendo desenvolvidas com a intenção de prover um ciclo de vida de desenvolvimento de KBSs que contemple as necessidades de uma sociedade baseada em conhecimento. Um dos processos fundamentais num ciclo de vida de KE é a aquisição de conhecimento.

A aquisição de conhecimento, que consiste em extrair o conhecimento necessário a partir de suas diversas fontes, de modo a poder codificá-lo e reutilizá-lo, abrangendo desde a identificação, coleta e análise até a modelagem e validação do conhecimento, ainda representa um dos principais gargalos na KE.

O conhecimento adquirido pode ser representado das mais diversas formas, desde simples documentos, passando por ontologias, até complexas redes de conhecimento. Neste sentido, o conhecimento na forma de melhores práticas representa o encapsulamento de experiências que, quando repetidas, levam a alcançar resultados semelhantes. Os SPCMMs são *frameworks* de melhores práticas de desenvolvimento de software e têm sido customizados para atender as necessidades específicas de qualidade de cada domínio de desenvolvimento de software.

A customização desses modelos para domínios específicos de desenvolvimento de software representa um problema de aquisição de conhecimento, no sentido de que existe a necessidade de se extrair o conhecimento a partir dos especialistas, dos modelos e normas genéricos e das demais fontes de conhecimento no domínio, para que possam ser definidas as melhores práticas que adequadamente atendam às necessidades de qualidade no domínio.

Assim, a presente pesquisa inicia-se com o estudo de quais e como são os modelos de capacidade/maturidade de processos de software existentes (Objetivo I) por meio da realização de uma Revisão Sistemática de Literatura. Os resultados desta revisão levam a concluir, primeiramente, que existe uma perceptível tendência à customização dos modelos genéricos de processo de software para domínios específicos. Outra constatação importante desta pesquisa inicial é de que, na literatura que relata o desenvolvimento de SPCMMs, em geral, poucos detalhes são dados de como este desenvolvimento é realizado.

Para suplementar esta carência da literatura, foi então realizado um *survey* com os autores dos modelos identificados na revisão sistemática, com o objetivo de entender melhor o processo de desenvolvimento de modelos customizados (Objetivo II). As respostas recebidas levam à conclusão de que, apesar do grande esforço envolvido, existe ainda pouco suporte metodológico sendo utilizado no desenvolvimento e customização de SPCMMs.

Os *frameworks*, métodos e abordagens relacionados ao desenvolvimento de SPCMMs são também analisados. Entretanto, não foi possível encontrar indicação de que um enfoque de aquisição de conhecimento tenha sido utilizado. Assim, são estudados os métodos, técnicas e abordagens de aquisição de conhecimento, com o intuito de identificar aqueles aplicáveis ao contexto de customização de SPCMMs. Desta forma, o estudo da literatura suporta o desenvolvimento do método, completando as experiências práticas já obtidas da revisão sistemática e do *survey*.

O método de aquisição de conhecimento para customização de modelos de capacidade/maturidade de processo de software para domínios específicos é então desenvolvido (Objetivo III). O método é dividido em cinco fases com base no ciclo de vida de KE (considerando a aquisição de conhecimento), cada uma contendo o detalhamento de atividades, produtos de trabalho, técnicas e ferramentas. O método é

documentado e publicado na forma de um relatório técnico³ contendo todos os seus componentes.

A validação do método (Objetivo IV) é conduzida tomando-se por base a abordagem GQM, a partir da qual dois objetivos de avaliação são elaborados. Assim, a validação é realizada sob duas diferentes perspectivas: a opinião de especialistas e a aplicação prática no desenvolvimento de SPCMMs. Para cada objetivo de avaliação, seguindo a abordagem GQM, diversas perguntas são definidas, bem como as medidas a serem coletadas e os instrumentos e técnicas de coleta de dados.

Em atendimento ao primeiro objetivo de avaliação, um *Expert Panel* é realizado para coletar a opinião de especialistas acerca da adequação do método de aquisição de conhecimento na customização de SPCMMs. Os especialistas selecionados para participar do painel são os autores dos mesmos trabalhos encontrados na revisão sistemática de literatura. Fecha-se assim o ciclo indutivo que se origina com a coleta das experiências de desenvolvimento de SPCMMs relatadas na literatura e obtidas no *survey*, que serviram de base para elaboração do método, e termina com a avaliação do método pelos mesmos especialistas que elaboraram aqueles trabalhos. A partir da análise dos resultados do *Expert Panel* tem-se uma primeira indicação que, de acordo com a opinião dos especialistas participantes, apesar das diversas oportunidades de melhoria percebidas, o método é adequado para a aquisição de conhecimento na customização de SPCMMs.

O segundo objetivo de avaliação envolve a realização de dois estudos empíricos, na forma estudos de caso, no intuito de caracterizar a aplicabilidade do método. O primeiro estudo de caso é realizado ainda durante a elaboração do método, no desenvolvimento de um SPCMM para o domínio de SaaS. No segundo estudo de caso, a primeira versão do método é aplicada na customização do SPCMM Medi SPICE para o domínio de dispositivos médicos. Algumas atividades propostas nas últimas fases no método, no entanto, não chegaram a ser aplicadas no estudo, devido ao estágio atual de desenvolvimento dos citados modelos. Os resultados observados nos estudos de caso levantam indícios positivos da aplicabilidade do método na customização de SPCMMs.

A partir dos resultados observados na validação, conclui-se que (Pergunta de Pesquisa) o método provê o suporte sistemático necessário

³ Disponível em: (HAUCK, 2011)

à aquisição de conhecimento na customização de SPCMMs para domínios específicos. Assim, é possível afirmar que a presente pesquisa atinge o seu Objetivo Geral de elaborar um método de aquisição de conhecimento para customização de modelos de capacidade/maturidade de processo de software para domínios específicos de desenvolvimento de software.

6.1 Principais Contribuições e Resultados

O trabalho apresentado nesta tese contribui para a KE especificamente na pesquisa atual de aquisição de conhecimento a partir de fontes não estruturadas e na codificação e reutilização do conhecimento na forma de melhores práticas de qualidade de processo, customizadas para um domínio específico. O enfoque de SPCMMs como representações de conhecimento na forma de melhores práticas também constitui uma contribuição à KE, encarando-os como conhecimento codificado e reutilizável para os engenheiros de conhecimento. Assim, uma contribuição importante dá-se também para a evolução do conhecimento a respeito dos SPCMMs existentes, na forma como são desenvolvidos, estruturados e customizados.

Para a área de aplicação na SE, o método proposto representa um suporte sistemático para a customização de SPCMMs para domínios específicos. Na SE este trabalho contribui também para o aprofundamento da discussão acerca da qualidade no desenvolvimento de SPCMMs.

Dentre estas contribuições, diversos resultados foram obtidos a partir da pesquisa realizada, tais como:

- Documentação do estado da arte em SPCMMs por meio de uma revisão sistemática da literatura (Objetivo I);
- Documentação da análise do processo de desenvolvimento de SPCMMs realizada por meio de um *survey* e da fundamentação teórica (Objetivo II);
- Primeira versão do método documentada na forma de um relatório técnico (Objetivo III);
- Documentação da validação do método (Objetivo IV);

A partir destes resultados diretos e indiretos e em decorrência deles, foram realizadas diversas publicações em conferências e

periódicos. Estas publicações são listadas nos quadros Quadro 24 e Quadro 25. Nestes quadros também é apresentada a relação das respectivas publicações com esta tese.

Quadro 24: Publicações em Periódicos

Referência	Título	Periódico	Relação com a Tese
(WANGENHEIM et al., 2010a)	Creating Software Process Capability/Maturity Models	IEEE Software	Resultado direto do estudo realizado para o capítulo 3
(CANCIAN et al., 2010)	Discovering Software Process and Product Quality Criteria in Software as a Service	Lecture Notes in Computer Science	Resultado de parte de estudo empírico relatado na seção 5.3.2
(WANGENHEIM, HAUCK & WANGENHEIM, 2009)	Enhancing Open Source Software in Alignment with CMMI-DEV	IEEE Software	Relato da customização de um software como resultado do estudo de melhores práticas de modelos e normas descritos no capítulo 3
(HAUCK et al., 2008b)	Adaptação de um Software de Gerência de Projetos de Código Aberto para o Atendimento dos Resultados Esperados do Nível G do MPS.BR	Programa Brasileiro da Qualidade e Produtividade em Software	Relato da customização de um software como resultado do estudo de melhores práticas de modelos e normas descritos no capítulo 3

Quadro 25: Publicações em Conferências

Referência	Título	Conferência	Relação com a Tese
(HAUCK et al., 2010)	Proposing a Knowledge Engineering Based Approach for Process Capability/Maturity Models Customization	EuroSPI, Grenoble França	Resultado direto do capítulo 4
(WANGENHEIM et al., 2010b)	Systematic Literature Review of Software Process Capability/Maturity Models	SPICE Conference, Pisa/Itália	Resultado direto do estudo realizado para o capítulo 3
(HAUCK & WANGENHEIM, 2008)	Development of a Process Reference Model for Telemedicine Software Development	EuroSPI, Dublin/Irlanda	Proposta de customização de SPCMM relacionado ao estudo empírico

			relatado na seção 5.3.2
(HAUCK et al., 2008c)	Experiences in Developing Requirements for a Clinical Laboratory Information System in Brazil	EuroSPI, Dublin/Irlanda	Primeira experiência de coleta de melhores práticas relacionada ao estudo empírico relatado na seção 5.3.2
(HAUCK et al., 2008d)	Utilizando algoritmos genéticos para solucionar o problema de escalonamento em ambientes multi-projetos	Congresso Brasileiro de Gerenciamento de Projetos, Porto Alegre/RS	Resultado da disciplina de EGC9001-11 - Engenharia de Ontologias
(FERNANDES et al., 2008)	Uma avaliação da interface de usuário de uma ferramenta Open Source de gerenciamento de Projetos baseado na web	Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, São Paulo/SP	Relato da avaliação ergonômica da customização de um software resultante do estudo de melhores práticas de modelos e normas descritos no capítulo 3

6.2 Trabalhos Futuros

Como trabalhos futuros que possam dar continuidade à pesquisa apresentada nesta tese, sugere-se a aplicação completa, principalmente das fases 4 e 5 do método, nos *trials* dos SPCMMs SaaS e Medi SPICE. Até o momento da elaboração deste documento, tanto o SPCMM para SaaS quanto o Medi SPICE não haviam sido ainda aplicados em um ambiente real de desenvolvimento de software no domínio, de forma que tornasse possível a execução de algumas das mais importantes atividades das fases 4 e 5, especialmente no que diz respeito à validação das melhores práticas extraídas. Isto se deve ao fato que o desenvolvimento deste tipo de modelo é, em geral, um processo bastante longo, normalmente com uma duração acima de 24 meses. A coleta de dados acerca da aplicabilidade destas duas fases finais possibilitaria a melhor adequação das atividades previstas e a validação mais completa das ferramentas e técnicas que compõem estas fases.

O desenvolvimento de um sistema de software que suporte a aplicação do método, apontado como sugestão por um dos especialistas envolvidos no painel, seria também uma importante contribuição. O sistema de software poderia dar suporte às diversas atividades de revisão e à formação de consenso durante a consolidação do conhecimento.

Também um repositório de modelos-base poderia integrar este sistema de software, facilitando a composição um novo SPCMM a partir da identificação de diferentes partes de modelos atualmente existentes. Desta forma, quando do desenvolvimento de um novo modelo, as melhores práticas documentadas em diversos SPCMMs diferentes poderiam ser consultadas, possibilitando a reutilização quando necessário. Poderia ser evitado, assim, o esforço empregado na eliciação de melhores práticas quando estas já estivessem disponíveis em modelos desenvolvidos ou customizados para outros domínios com necessidades de qualidade similares.

Durante a elaboração do método de aquisição de conhecimento, na tentativa de compor o conjunto de técnicas que formariam o método, foi percebida uma carência de técnicas que suportem a validação de um SPCMM em relação ao alcance dos seus objetivos. Isso implica, dentre outras possibilidades, em verificar se o conhecimento na forma de melhores práticas nos modelos, realmente representa as melhores práticas de desenvolvimento de software em relação às expectativas de qualidade no domínio. Neste sentido um grande campo de pesquisa parece ainda estar em aberto, pois pouca literatura foi encontrada a respeito e os relatos de desenvolvimento deste tipo de modelos, com poucas exceções, não apresentam detalhes de como essas atividades foram realizadas.

REFERÊNCIAS

- ABNT, Associação Brasileira de Normas Técnicas. **NBR 13596 - Tecnologia de Informação - Avaliação de Produto de Software: Características de qualidade e diretrizes para o seu uso**. Rio de Janeiro, RJ, 1996.
- _____. **NBR ISO/IEC 9126 Engenharia de software - Qualidade de produto**. Rio de Janeiro, RJ, 2003.
- _____. **NBR ISO/IEC 9126 - Engenharia de Software - Qualidade de Produto**. Rio de Janeiro: RJ, 2003.
- ABRAN, A.; AL-QUTAISH, A.; DESHARNAIS, J. M.; HABRA, N. **An Information Model for Software Quality Measurement with ISO Standards**. SWDC-REK International Conference on Software Development, Reykjavick, Iceland, 2005.
- ACKOFF, R. L. **From Data to Wisdom**. Journal of Applied Systems Analysis, v. 16, pp. 3-9, 1989.
- ACUÑA, A. A.; FERRE, X.; LÓPEZ, M.; MATE L. **The Software Process: Modeling, Evaluation and Improvement**. Handbook of Software Engineering and Knowledge Engineering. Argentina: World Scientific Publishing Company, 2000.
- AHLEMANN, F.; GASTL, H. **Process model for an empirically grounded reference model construction**. Fettke, P., Loos, P. (eds.), Reference Modeling for Business Systems Analysis, Idea Group Hershey, 2006.
- AKAO, Y.; MIZUNO, S. **QFD, the Customer-Driven Approach to Quality Planning and Deployment** Asian Productivity Organization 1994.
- ANGELE, J.; FENSEL, D.; STUDER, R. **Developing Knowledge-Based Systems with MIKE**. Journal of Automated Software Engineering, v. 5, n. 4, 1998.
- BARBACCI, M.; KLEIN, M. H.; LONGSTAFF, T. A.; WEINSTOCK, C. B. **Quality Attributes**. Carnegie Mellon University - Software Engineering Institute, Pittsburgh, 2001.
- BARUCH, Y. **Response Rate in Academic Studies – A Comparative Analysis**. Human Relations, v. 52, n. 4, 1999.

- BASILI, V.; LINDVALL, M.; COSTA, P. **Implementing the Experience Factory Concepts as a Set of Experience Bases.** 13th International Conference on Software Engineering & Knowledge Engineering, Buenos Aires, 2001.
- BASILI, V. R.; CALDIERA, G.; ROMBACH H. D. **The Goal Question Metric Approach.** Encyclopedia of Software Engineering, v. 1, John Wiley & Sons, 1994.
- BASS, L.; CLEMENTS, P.; KAZMAN, R. **Software Architecture in Practice.** Software Architecture in Practice, Addison-Wesley, Reading, MS, 1998.
- BECKER, J.; KNACKSTEDT, R.; PÖPPELBUß, J. **Developing Maturity Models for IT Management - A Procedure Model and its Application.** Journal Business & Information Systems Engineering, Gabler Verlag, v. 1, n. 3, 2009.
- BEECHAM, S.; HALL, T.; RAINER, A. **Defining a Requirements Process Improvement Model.** Software Quality Journal, v. 13, n. 3, Springer, 2005.
- BEECHAM, S.; HALL, T.; BRITTON, C.; COTTEE, M.; RAINER, A. **Using an Expert Panel to Validate a Requirements Process Improvement Model.** The Journal of Systems and Software, v. 76, 2005.
- BENALI, K.; DERNIAME, J. C. **Software processes modeling: what, who, and when.** Proceedings of the Second European Workshop on Software Process Technology, 1992.
- BJØRNSON, F. O., DINGSØY T. **Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used.** Information and Software Technology, v. 50, n. 11, 2008.
- BOEHM, B. W.; BROWN, J. R., KASPAR, H., LIPOW, M., MACLEOD, G. J., MERRIT, M. J. **Characteristics of Software Quality.** American Elsevier, New York, 1978.
- BOOSE, J. H. **A survey of knowledge acquisition techniques and tools.** Knowledge Acquisition, v. 1, n. 1, 1989.
- BRERETON, P, KITCHENHAM, B.A., BUDGEN, D., TURNER, M., KHALIL, M. **Lessons from applying the systematic literature**

- review process within the software engineering domain.** JSS - Journal of Systems and Software, v. 80, n. 4, 2007.
- BRUIN, T.; ROSEMANN, M. **Using the Delphi Technique to Identify BPM Capability Areas.** 18th Australasian Conference on Information Systems, 42, 2007.
- _____. **Understanding the Main Phases of Developing a Maturity Assessment Model.** 16th Australasian Conference on Information Systems, Sydney, 2005.
- BURTON, J.; RICHARDSON, I. MCCAFFERY, F.; O'HAODHA, M. **Risk Management Capability Model for Medical Device Software,** Cambridge Scholars Publishers, 2009.
- BURTON, J. **A Software Risk Management Capability Model for Medical Device Software.** Tese de Doutorado, University of Limerick, 2008.
- BUZAN, T.; BUZAN, B. **The Mind Map Book - How to use Radiant Thinking to Maximize Your Brain's Untapped Potential.** Dutton: New York, 1990.
- CALHOUN, M. A.; STARBUCK, W. H. **Barriers to Creating Knowledge.** In: Easterby-Smith, M. & Lyles M. A. (eds.) Handbook of organizational learning and knowledge management, Wiley-Blackwell, 2003.
- CANCIAN, M.; HAUCK J. C. R.; WANGENHEIM, C. G.; RABELO R. **Discovering Software Process and Product Quality Criteria in Software as a Service.** Lecture Notes in Computer Science, v. 6156, 2010.
- CANCIAN, M. **Process Reference Model for SaaS - Technical Report.** Universidade Federal de Santa Catarina, Florianópolis/Brasil. Disponível em: <http://www.gsigma.ufsc.br/~cancian/guide/>. Acessado em: Fevereiro/2010.
- _____. **Uma Proposta de Guia de Referência para Provedores de Software como um Serviço.** (Mestrado). Programa de Pós Graduação em Engenharia de Automação e Sistemas, Universidade Federal de Santa Catarina, Florianópolis, 2009.
- CASS, A; VOLCKER, C; OUARED, C.; DORLING, A.; WINZER, L.; CARRANZA, J. M. **SPICE for SPACE Trials, Risk Analysis, and**

- Process Improvement.** Software Process: Improvement and Practice, v. 9, n. 1, 2004.
- CHANG, L., **Using confirmatory factor analysis of multitrait-multimethod data to assess the psychometrical equivalence of 4-point and 6-point likert type scales.** Proceeding of the Annual Meeting of the National Council on Measurement in Education, ERIC, Atlanta, 1993.
- CHOMEYA, R. **Quality of Psychology Test Between Likert Scale 5 and 6 Points.** Journal of Social Sciences, v. 6, n. 3, 2010.
- COOK D. A.; BECKMAN T. D. **Current Concepts in Validity and Reliability for Psychometric Instruments: Theory and Application.** The American Journal of Medicine v. 119, n. 166, 2006.
- CRONBACH, L. J.; SHAVELSON, R. J. **My Current Thoughts on Coefficient Alpha and Successor Procedures.** Educational and Psychological Measurement, v. 64, n. 3, 2004.
- CROSBY, P. B. **Qualidade Falada a sério.** São Paulo: Mc Graw-Hill do Brasil, 1990.
- DAVID, P. A.; DOMINIQUE F. **Economic Fundamentals of the Knowledge Society.** Policy Futures in Education, v. 1, n. 1, 2003.
- DAVIS, R.; HOWARD, S.; SZOLOVITS, P. **What Is a Knowledge Representation?** AI Magazine, v. 14, n. 1, 1993.
- DELONE, W. H.; MCLEAN, E. R. **The DeLone and McLean Model of Information System Success: A Ten-Year Update.** Journal of Management Information Systems, v. 19, n.4, 2003.
- DINGSØYR, T. **An Evaluation of Research on Experience Factory.** 2nd International Workshop on Learning Software Organization, 2000.
- DKIT – Dundalk Institute of Technology. Disponível em: <<http://www.dkit.ie>>. Acesso em: Dezembro/2010.
- DOMINGUE, J.; MOTTA, E.; WATT, S. **The emerging VITAL workbench.** Knowledge Acquisition for Knowledge-Based Systems, Lecture Notes in Computer Science, v. 723, 1993.

- DYBA, T. **An instrument for measuring the key factors of success in software process improvement.** Empirical Software Engineering, v. 5, n. 4, 2000.
- EGC. **Programa de Pós-Graduação em Engenharia e Gestão do Conhecimento.** Disponível em: <<http://www.egc.ufsc.br>>. Acessado em: Fevereiro/2010.
- ELLIS, T. J.; LEVY, Y. **Towards a Guide for Novice Researchers on Research Methodology: Review and Proposed Methods.** Issues in Informing Science and Information Technology, v. 6, 2009.
- EMAM, E. K. **Benchmarking Kappa: Interrater Agreement in Software Process Assessments.** Empirical Software Engineering, v. 4, 1999.
- EMAM, E. K; MADHAVJI, N.H. **Does organizational maturity improve quality?** IEEE Software, v. 13, n. 5, 1996.
- ERIKSSON, H. **A survey of knowledge acquisition techniques and tools and their relationship to software engineering.** Journal of Systems and Software, v. 19, n. 1, 1992.
- FABRINI F. et al., **Integrating joint reviews with automotive SPICE assessments results.** Lecture Notes in Computer Science, v. 5007, 2008.
- FATUDIMU, I. T.; MUSA, A. G.; AYO, C. K.; SOFOLUWE, A. B. **Knowledge Discovery in Online Repositories: A Text Mining Approach.** European Journal of Scientific Research, v. 22, n. 2, 2008.
- FEIGENBAUM, A. V. **Total Quality Control.** 3a. ed. New York: McGraw-Hill, 1986.
- FERNANDES, L. S.; WANGENHEIM, C. G.; PACHECO, A. P.; HAUCK, J. C. R. **Uma avaliação da interface de usuário de uma ferramenta Open Source de gerenciamento de Projetos baseado na web.** 8º Congresso Brasileiro de Pesquisa e Desenvolvimento em Design, São Paulo, 2008.
- FERREIRA, A. B. de H. **Novo dicionário da Língua Portuguesa.** Rio de Janeiro: Nova Fronteira, 2008.
- FINKELSTEIN, A.; KRAMER, J.; NUSEIBEH, B. **Software Process Modelling and Technology.** Research Studies Press, 1994.

- FRANK, A. et al. **Querying Structured Knowledge Sources**. Workshop on Question Answering in Restricted Domains, Pittsburgh, PA, 2005.
- GAO - United States General Accounting Office. **Case Study Evaluations**. Technical Report GAO/PEMD-91-10.1.9. Program Evaluation and Methodology Division, 1990.
- GARVIN, D. A. **What does "product quality" really mean?** MIT Sloane Management Review, 1984.
- GENNARI, J. H.; MUSEN, M. A.; FERGERSON, R. W.; GROSSO, W. E.; CRUBÉZY, M.; ERIKSSON, H.; NOY, N. F. **The evolution of Protégé: an environment for knowledge-based systems development**. International Journal of Human-Computer Studies, v. 58, n. 1, 2003.
- GILLINGHAM, H.; ROBERTS, B. **Implementing Knowledge Management: A Practical Approach**. Journal of Knowledge Management Practice, v. 7, n. 1, 2006.
- GLIEM, J. A.; GLIEM, R. R. **Calculating, Interpreting, and Reporting Cronbach's Alpha Reliability Coefficient for Likert-Type Scales**. In Midwest Research-to-Practice Conference in Adult, Continuing, and Community Education, 2003.
- GOLDENSON, D. R.; EMAM, K. E.; HERBSLEB, J.; DEEPHOUSE, C. **Empirical Studies of Software Process Assessment Methods**. In Elements of Software Process Assessment & Improvement, Khaled El Emam & Nazim H. Madhavji (Eds), Wiley IEEE, 1999.
- GOLUBIÉ S. **Influence of Software Development Process Capability on Product Quality**. 8th International Conference on Telecommunications, Zagreb, 2005.
- GÓMEZ-PÉREZ, A.; MONTES, C. **Enseñanza de Inteligencia Artificial e Ingeniería del Conocimiento**. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial, v.3, 1997.
- GOODMAN, B. D.; GOLDMAN, S. N. **Freeing creativity by understanding the role of best practices**. 19th IEEE International Engineering Management Conference, Austin, 2007.
- GRAUPNER, S.; MOTAHARI-NEZHAD, H. R.; SINGHAL, S.; BASU, S. **Making processes from best practice frameworks**

- actionable.** 13th Enterprise Distributed Object Computing Conference Workshops, Auckland, 2009.
- GROVER, M. D. **A Pragmatic Knowledge Acquisition Methodology.** International Joint Conference on Artificial Intelligence, Germany, 1983.
- GUBA, E. G.; LINCOLN, Y. S. **Competing paradigms in qualitative research.** Handbook of Qualitative Research. London: Sage, 1994.
- HAMANN, D.; JARVINEN, J.; BIRK, A.; PFAHL, D. **A Product-Process Dependency Definition Method.** 24 th. EUROMICRO Conference, Sweden, 1998.
- HAUCK, J. C. R.; WANGENHEIM, C. G.; WAGENHEIM, A. **A Knowledge Engineering Based Method for SPCMMs Customization.** Relatório Técnico RT GQS 1 0. Disponível em: <http://www.inf.ufsc.br/~jeanhauck/method/RT_GQS_01_SPCMMs_Dev_Method_v_1_0.pdf>. Acessado em: Janeiro/2011.
- HAUCK, J. C. R., WANGENHEIM, C. G., MCCAFFERY, F., WANGENHEIM, A. **Proposing a Knowledge Engineering Based Approach for Process Capability/Maturity Models Customization.** EuroSPI 2010, Grenoble, 2010.
- HAUCK, J. C. R.; WANGENHEIM, C. G.; SOUZA, R. H.; THIRY, M. **Process Reference Guides Support for Improving Software Processes in Alignment with Reference Models and Standards.** Communications in Computer and Information Science, v. 16, 2008.
- HAUCK, J. C. R.; WANGENHEIM, C. G.; THIRY, M. **Adaptação de um Software de Gerência de Projetos de Código Aberto para o Atendimento dos Resultados Esperados do Nível G do MPS.BR.** Programa Brasileiro da Qualidade e Produtividade em Software, v. 5, 2008.
- HAUCK, J. C. R.; WANGENHEIM, C. G.; WANGENHEIM, A. **Development of a Process Reference Model for Telemedicine Software Development.** European Systems & Software Process Improvement and Innovation - EuroSPI 2008, Dublin, 2008.
- HAUCK, J. C. R.; CANCIAN, M. H. ; WANGENHEIM, C. G.; THIRY, M. ; WANGENHEIM, A. ; SOUZA, R. H. **Experiences in Developing Requirements for a Clinical Laboratory Information**

System in Brazil. European Systems & Software Process Improvement and Innovation - EuroSPI 2008, Dublin, 2008.

HAUCK, J. C. R.; CABRAL, R. B.; WANGENHEIM, A.; GAUTHIER, F. A. O.; TODESCO, J. L.; WANGENHEIM, C. G. **Utilizando Algoritmos Genéticos para Solucionar o Problema de Escalonamento em Ambientes Multi-projetos.** Congresso Brasileiro de Gerenciamento de Projetos, Porto Alegre, 2008.

HAYES-ROTH, F.; WATERMAN, D. A.; LENAT, D. B. **Building Expert Systems.** 4. ed., Addison-Wesley, 1983.

HUA, J. **Study on Knowledge Acquisition Techniques.** 2nd International Symposium on Intelligent Information Technology Application, Shanghai, 2008.

IEEE, Computer Society. **SWEBOK - Guide to the Software Engineering Body of Knowledge.** California: IEEE, 2004.

_____. **IEEE Standards Development Process.** Disponível em: <<http://standards.ieee.org/resources/development/index.html>>. Acessado em Fevereiro/2010.

IRIBARREN, M.; CONCHA, G.; VALDES, G.; SOLAR, M.; VILLARROEL, M. T.; GUTIÉRREZ, P.; VÁSQUEZ, A. **Capability Maturity Framework for eGovernment: A Multi-dimensional Model and Assessing Tool.** Lecture Notes in Computer Science n. 5184, 2008.

ISO - International Organization for Standardization, **ISO9001:2008 Quality management systems – Requirements,** 2008.

_____. **How ISO develops standards.** Disponível em: <http://www.iso.org/iso/how_iso_develops_standards>. Acesso em: Fevereiro/2009.

ISO/IEC - International Organization for Standardization. **ISO/IEC 15504: Information technology -- Process assessment – Part 2: Performing an assessment,** ISO/IEC International Standard, 2003.

_____. **ISO/IEC 15504: Information technology -- Process assessment – Part 5: An exemplar Process Assessment Model,** ISO/IEC International Standard, 2006.

- _____. **ISO/IEC 15504: Information technology -- Process assessment – Part 7: Assessment of Organizational Maturity**, ISO/IEC International Standard, 2008.
- _____. **ISO/IEC 25000: Software Engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Guide to SQuaRE**. ISO/IEC International Standard, 2005.
- ISHIKAWA, K. **TQC, total quality control: estratégia e administração da qualidade**. São Paulo: IMC Internacional Sistemas Educativos, 1986.
- ITSMF - Information Technology Service Management Forum. **An Introduction Overview of Itil V3**. itSMF, Earley, United Kingdom, 2007.
- JALOTE, P. **CMM in Praticce. Processes for Executing Software Projects at Infosys**. SEI, Addison Wesley, 1999.
- JURAN, J. M. **Planejando para a Qualidade**. São Paulo: Pioneira, 1990.
- KASUNIC, M. **Designing an Effective Survey**. Hadbook CMU/SEI-2005-HB-004, Software Engineering Institute / Carnegie Mellon University, Pittsburgh, 2005.
- KITCHENHAM, B. A. **Guidelines for performing Systematic Literature Reviews in Software Engineering, Version 2.3** Technical report, University of Durham, Keele, UK, 2007.
- KITCHENHAM, B.; PFLEEGER, S.L.; MCCOLL, B.; EAGAN, S. **An empirical study of maintenance and development estimation accuracy**. Journal of Systems and Software, n. 64, v. 1, 2002.
- KONITIO, J. **Introduction to Software Engineering**. Helsinki University of Technology, 1999.
- KRUEGER, R. A., CASEY, M. A. **Focus Groups: A Practical Guide for Applied Research**. Sage Publications, Thousand Oaks, 2000.
- LAKATOS, E. M.; MARCONI, M. A. **Metodologia do Trabalho Científico**. São Paulo: Atlas, 1995.
- LAHRMANN G.; MARX F. **Systematization of Maturity Model Extensions**. Lecture Notes in Computer Science, v. 6105, 2010.

- LEROUGE C.; GARFIELD, M.; HEVNER, A. **Quality Attributes in Telemedicine Video Conferencing**. Hawaii International Conference on System Sciences, IEEE Computer Society, Big Island, 2002.
- LIMING, Z.; STAPLES, M.; GORTON, I. **An Infrastructure for Indexing and Organizing Best Practices**. Second International Workshop on Realising Evidence-Based Software Engineering, 2007.
- LITWIN, M. **How to Measure Survey Reliability and Validity**. Sage Publications, 1995.
- MAGEE, S.; TRIPP, L. **Guide to Software Engineering Standards and Specifications**, Artech House, 1997.
- MAGEE, S.; THIELE, D. **Engineering Process Standards: State of the Art and Challenges**. IEEE Information Technology Professional, v. 6, n. 5, 2004.
- MAIER, A. M.; MOULTRIE, J. and CLARKSON, P.J. **Developing maturity grids for assessing organisational capabilities: practitioner guidance**. Proceedings of 4th International Conference on Management Consulting, Academy of Management, 2009.
- MATOOK S.; INDULSKA, M. **Improving the Quality of Process Reference Models: A Quality Function Deployment-Based Approach**. Decision Support Systems, v. 47, 2009.
- MCCALL, J.; RICHARDS, P.; WALTERS, G. **Factors in Software Quality**. US Rome Air Development Center Reports NITS, 1977.
- MCCAFFERY, F.; DORLING, A. **Medi SPICE Development**. Software Process Maintenance and Evolution: Improvement and Practice Journal, v. 22, n. 4, 2010.
- MCCAFFERY F.; DORLING A.; CASEY V. **Medi SPICE: An Update**. 10th International Conference on Software Process. Improvement and Capability determination, Pisa, Italy, 2010.
- MCCAFFERY, F.; BURTON, J.; RICHARDSON, I. **Risk management capability model for the development of medical device software**. Software Quality Journal, Springer Publishers, v. 18, n. 1, 2009.

- MCCAFFERY, F.; RICHARDSON, I. **MedeSPI :A Software Process Improvement Model for the medical device industry based upon ISO/IEC 15504**. International SPICE Days 2007, Germany, 2007.
- MCCAFFERY, F.; COLEMAN, G. **Developing a Configuration Management Capability Model for the Medical Device Industry**. International Journal of Information Systems and Change Management, v. 2, n. 2, 2007.
- MCCAFFERY F.; PIKKARAINEN, M.; RICHARDSON, I. **Ahaa - Agile, Hybrid Assessment Method for Automotive, Safety Critical SMEs**, 30th International Conference on Software Engineering, Leipzig, Germany, 2008.
- MEDI SPICE. **Portal do projeto Medi SPICE**. Disponível em: <<http://medispice.ning.com>>. Acessado em: Novembro/2010.
- METTLER, T. **A Design Science Research Perspective on Maturity Models in Information Systems**. Universität St. Gallen, St. Gallen, Switzerland, Technical Report BE IWI/HNE/03, 2009.
- MICHAELIS. **Moderno Dicionário Inglês**. Disponível em: <<http://michaelis.uol.com.br>>. Acessado em: Janeiro/2011.
- MILTON, N. R. **Knowledge Acquisition in Practice A Step-by-step Guide**. Springer, 2007.
- MOORE, J. W. **An Integrated Collection of Software Engineering Standards**. IEEE Software, v. 16, n. 6, 1999.
- MOTODA H., MIZOGUCHI, R.; BOOSE, J.; GAINES, B.. **Knowledge Acquisition for Knowledge-Based Systems**. IEEE Expert: Intelligent Systems and Their Applications, v. 6, n. 4, 1991.
- MYERS, M. D. **Qualitative Research in Information Systems**. MISQ Discovery, v. 21, n. 2, 1997.
- MYERS, M. E. **Qualitative Research and the Generalizability Question: Standing Firm with Proteus**. The Qualitative Report, v. 4, n. 3/4, 2000.
- NATALI A. C. C.; FALBO R. A. **Knowledge Management in Software Engineering Environments**. Simpósio Brasileiro em Engenharia de Software, Gramado/RS, 2002.
- NIAZI, M.; WILSON, D.; ZOWGHI, D. **A Maturity Model for the Implementation of Software Process Improvement: An**

- Empirical Study.** Journal of Systems and Software, v. 74, n. 2, 2005.
- NONAKA, I.; TAKEUCHI, H. **The Knowledge-Creating Company - How Japanese Companies Create the Dynamics of Innovation.** Hardback, 1995.
- NOOR, K. B. M. **Case Study: A Strategic Research Methodology.** American Journal of Applied Sciences, v. 5, n. 11, 2008.
- OKOLI C., PAWLOWSKI S. D. **The Delphi method as a research tool: an example, design considerations and applications.** Inf. Management, v. 42, n. 1, 2004.
- OLDHAM, K.; KNEEBONE, S.; CALLOT, M.; MURTON, A.; BRIMBLE, R. **MOKA - A Methodology and tools Oriented to Knowledge-based engineering Applications.** Changing the Ways We Work - Advances in Design and Manufacturing, v. 8, 1998.
- OMG – Object Management Group. **Software & Systems Process Engineering Meta-Model Specification – Version 2.0.** Technical Report formal/ 2008-04-01, 2008.
- _____. **OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2.** Technical Report formal/2007-11-02, 2007.
- ORLIKOWSKI, W. J.; BAROUDI, J. J. **Studying Information Technology in Organizations: Research Approaches and Assumptions.** Information Systems Research, v. 2, n. 1, 1991.
- PARZINGER, M.J.; NATH, R. **A Study of the Relationships Between Total Quality Management Implementation Factors and Software Quality.** Total Quality Management, 2000.
- PAULK, M. C. **Surviving the Quagmire of Process Models, Integrated Models, and Standards.** ASQ Annual Quality Congress, Toronto, 2004.
- PETTICREW, M.; ROBERTS, H. **Systematic Reviews in the Social Sciences: A Practical Guide,** Blackwell Publishing, 2005.
- PIPINO, L.; LEE, Y.; WANG, R. **Data Quality Assessment.** Communications of the ACM, v. 45, n. 4, 2002.
- POWELL, C. **The Delphi technique: myths and realities.** Journal of Advanced Nursing, v. 41, n. 4, Blackwell Publishing, 2003.

- PRESTON, S.; CHAPMAN, C.; PINFOLD, M.; SMITH, G. **Knowledge Acquisition for Knowledge-based Engineering Systems**. International Journal of Information Technology and Management, v. 4, n. 1, 2005.
- QFDI – **Quality Function Deployment Institute**. Disponível em: <<http://www.qfdi.org/>>. Acessado em: Julho/2010.
- RATNESHWER, G. **A maturity model for CBSE**. 2nd India Software Engineering Conference, 2009.
- RICHARDSON, I. **Quality Function deployment - A Software Process Tool?** Third Annual International QFD Symposium. Linköping, Sweden, 1997.
- RICHARDSON, I.; WANGENHEIM, C. G. **Why are Small Software Organizations Different?** IEEE Software, v. 24, n. 1, 2007.
- ROBSON, C. **Real world research: a resource for social scientists and practitioner-researchers**. John Wiley & Sons, 2ed, 2002.
- ROWLEY, J. **The wisdom hierarchy: representations of the DIKW hierarchy**. Journal of Information Science, v. 33, n. 2, 2007.
- SALVIANO C. F.; ZOUCAS, A.; SILVA, J. V. L.; ALVES, A. M.; WANGENHEIM, C. G.; THIRY, M. **A Method Framework for Engineering Process Capability Models**. 16th European Systems and Software Process Improvement and Innovation, Alcalá, Spain, 2009.
- SALVIANO, C. F.; FIGUEIREDO, A. M. C. M., **Unified Basic Concepts for Process Capability Models**, 20th International Conference on Software Engineering and Knowledge Engineering. SEKE, Redwood, 2008.
- SANCHEZ R. **“Tacit Knowledge” versus “Explicit Knowledge”- Approaches to Knowledge Management Practice**. Technical Report IVS/CBS Working Papers 2004-01, Department of Industrial Economics and Strategy, Copenhagen Business School, 2004.
- SANTOS, F. M. R.; SOUSA, R. P. L. **O Conhecimento no Campo de Engenharia e Gestão do Conhecimento**. Perspectivas em Ciência da Informação, v. 15, n. 1, 2010.
- SAUNDERS, M. N. K.; LEWIS P.; THORNHILL, A. **Research Methods for Business Students**. Prentice Hall, 5 ed, 2009.

- SHADBOLT, N. R.; MILTON, N. **From Knowledge Engineering to Knowledge Management**. British Journal of Management, v.10, 1999.
- SCHREIBER, G.; AKKERMANS, H.; ANJEWIERDEN A.; DE HOOG, R., SHADBOLT, N.; VAN DE VELDE, W; WIELINGA, B. **Knowledge Engineering and Management – The CommonKADS Methodology**. The MIT Press, 2000.
- SCHREIBER, A. T.; WIELINGA, B. J. **Knowledge Model Construction**, 11th Workshop on Knowledge Acquisition, Modeling and Management, Voyager Inn, Banff, Alberta, Canada, 1998.
- SEI – Software Engineering Institute. **CMMI-DEV CMMI® for Development, Version 1.3**. Carnegie Mellon University, Pittsburgh, 2010.
- SENDJAYA S. **Development and Validation of Servant Leadership Behavior Scale**. Servant Leadership Research Roundtable, Regent University, 2003.
- SHEARD, S. A. **The Frameworks Quagmire**. Crosstalk: The Journal of Defense Software Engineering, v. 10, v. 9, 1997.
- _____. **Evolution of the Frameworks Quagmire**. IEEE Computer, v. 34, n. 7, 2001.
- SHULL, F.; TURNER, R. **An empirical approach to best practice identification and selection: the US Department of Defense acquisition best practices clearinghouse**. 2005 International Symposium on Empirical Software Engineering, Noosa Heads, Australia, 2005.
- SIBISI, M.; WAVEREN, C. C. **A Process Framework for Customising Software Quality Models**. IEEE AFRICON, Windhoek, South Africa, 2007.
- SILVA, J. V. L. et. al. **Strategic Management in University Research Laboratories Towards a Framework for Assessment and Improvement of R&D Management**. International Conference on Software Process Improvement and Capability dEtermination, Seoul, South Korea, 2007.
- SLACK, F.; ROWLEY, J. **Observation: perspectives on research methodologies for leisure managers**, Management Research News, v. 24, n: 1/2, 2001.

- SMITH, E. A. **The role of tacit and explicit knowledge in the workplace.** Journal of Knowledge Management, v. 5, n. 4, 2001.
- SOFTEX. **MPS.BR – Melhoria de Processo do Software Brasileiro, Guia Geral, Versão 2009.** Brasília: Softex, 2009.
- SPICE USER GROUP. **Automotive SPICE® Process Reference Model v4.5.** Technical Report. Disponível em: <<http://www.automotivespice.com/>> Acessado em: Novembro/2010.
- STEELS, L. **Components of Expertise.** AI Magazine, v. 11, n. 2, 1990.
- STUDER, R.; BENJAMINS V. R.; FENSEL, D. **Knowledge Engineering: Principles and Methods.** Data and Knowledge Engineering, v. 25, n. 1-2, 1998.
- SUREEPHONG, P.; CHAKPITAK, N.; OUZROUTE, Y.; NEUBERT, G.; BOURAS, A. **Knowledge Engineering Technique for Cluster Development.** Knowledge Science Engineering and Management Melbourne, Australia, 2007.
- SWARTOUT, B.; GIL, Y. **EXPECT: Explicit Representations for Flexible Acquisition.** Ninth Knowledge Acquisition for Knowledge-Based Systems Workshop, Banff, Canada, 1995.
- TAYLOR-POWELL, E.; STEELE, S. **Collecting Evaluation Data: Direct Observation.** Technical Report G3658-5. Program development and Evaluation, University of Wisconsin, 1996.
- TELLIS, W. **Application of a Case Study Methodology.** The Qualitative Report, v. 3, n. 3, 1997.
- TORGERSSON, J.; DORLING, A. **Assessing CBD - What's the Difference?** 28th Euromicro Conference, Dortmund, Germany, 2002.
- ULLMAN, D. G.; AMBROSIO, B. **An Introduction to The Consensus Model of Engineering Design Decision Making.** AAAI - Association for the Advancement of Artificial Intelligence. Technical Report SS-98-03, 1998.
- WANG, R. Y.; STRONG, D. M. **Beyond Accuracy: What Data Quality Means to Data Consumers.** Journal of Management Information Systems, v. 12, n. 4, 1996.

- WANGENHEIM C. G. et al. **Standard Based Software Process Assessments in Small Companies**. Software Process: Improvement and Practice, v. 11, n. 3, 2006.
- WANGENHEIM, C. G.; HAUCK, J. C. R.; MCCAFFERY, F.; WANGENHEIM, A. **Creating Software Process Capability/Maturity Models**. IEEE Software, v. 27, n. 4, 2010.
- WANGENHEIM, C. G.; HAUCK, J. C. R. **Teaching Software Process Improvement and Assessment**. EuroSPI 2010, Grenoble, 2010.
- WANGENHEIM, C. G.; HAUCK, J. C. R.; SALVIANO, C. F.; WANGENHEIM, A. **Systematic Literature Review of Software Process Capability/Maturity Models**. SPICE Conference 2010, Pisa, Itália, 2010.
- WANGENHEIM, C. G.; HAUCK, J. C. R.; WANGENHEIM, A. **Enhancing Open Source Software in Alignment with CMMI-DEV**. IEEE Software, v. 26, n. 2, 2009.
- WANGENHEIM, C. G.; ANACLETO, A.; SALVIANO, C. F. **Helping Small Companies Assess Software Processes**. IEEE Software, v. 23, n. 1, 2006.
- WESSA, P. **Cronbach alpha (v1.0.1) in Free Statistics Software (v1.1.23-r6)**, Office for Research Development and Education. Disponível em: http://www.wessa.net/rwasp_cronbach.wasp/
Acessado em: Janeiro/2011.
- WOHLIN, C.; RUNESON, P.; HÖST, M.; OHLSSON, M. C.; REGNELL, B.; WESSLÉN, A. **Experimentation In Software Engineering: An Introduction**. International Series In Software Engineering, Springer: 1 ed, 2000.
- YIN, R. K. **Applications of Case Study Research**. Second Edition. Applied Social Research Methods Series, v. 34, Sage Publications, 2003.
- ZAHARAN, S. **Software Process Improvement: Practical Guidelines for Business Success**. Edinburgh: Addison-Wesley, 1998.
- ZANNIER, C.; MELNIK, G.; MAURER, F. **On the success of empirical studies in the international conference on software engineering**. 28th International Conference on Software engineering, Shanghai, 2006.

- ZELKOWITZ, M. V.; WALLACE, D. **Experimental models for validating computer technology.** IEEE Computer, v. 31, n. 5, 1998.
- ZHU, L.; STAPLES, L. Z.; GORTON, I. **An Infrastructure for Indexing and Organizing Best Practices.** Second International Workshop on Realising Evidence-Based Software Engineering, 2007.

APÊNDICE A – *Software Process Capability/Maturity Models*

Este apêndice apresenta os SPCMMs encontrados na revisão sistemática de literatura (WANGENHEIM et al., 2010b). Cada modelo é caracterizado por seu domínio, uma identificação sequencial (de M01 a M52), seu nome e/ou iniciais, uma referência para o artigo em que ele é descrito, e uma lista de modelos a fonte na qual se baseia. Alguns dos modelos foram descritos em mais de um artigo. Nestes casos (M08, M21, M24, M30, M31, M38, M41, M44), são listadas as duas referências.

Domain	Id	Capability/Maturity Model	Ref	Based on
Automotive systems	m01	AutomotiveSPICE Process Assessment Model	[1]	ISO/IEC 15504 (SPICE)
Business Process	m02	Business Process Maturity Model (BPMM)	[2]	CMM/CMMI, ISO/IEC 12207, and ISO/IEC 15288
Component Based Software Engineering	m03	Integrated Component Maturity Model	[3]	CMM
	m04	OOSPICE Process Assessment Model for Software Component-based Development	[4]	ISO/IEC 15504
Data Warehouse Systems	m05	Data Warehousing Process Maturity Model	[5]	CMM
Documentation	m06	Software system documentation process maturity model	[6]	CMM
E-Government	m07	EGMM - E-government maturity model	[7]	CMM, PMMM
E-Learning	m08	e-learning maturity model	[8] [9]	CMM, ISO/IEC 15504 (SPICE)
Software and System Engineering, including development, services and acquisition.	m09	ISO/IEC 15504-5 Process Assessment Model	[10]	ISO 9000, ISO/IEC 12207, ISO/IEC 15288
Information Systems	m10	Extending information system integrability index with CMM model	[11]	CMM
Software and System Engineering, including development, services and acquisition.	m11	iCMM – Integrated Capability maturity Model	[12]	CMM, ISO 9000, EIA/IS 731 Systems Engineering Capability, Malcolm Baldrige National Quality Award, CMMI, ISO/IEC TR 15504, ISO/IEC 12207, ISO/IEC CD 15288
Knowledge Management	m12	Knowledge Management Maturity Model	[13]	CMM
Maintenance	m13	Software Maintenance Maturity Model	[14]	ISO/IEC 14764 and ISO/IEC 12207
Measurement	m14	MIS-PyME software measurement maturity model-supporting the definition of	[15]	GQ(I)M

		software measurement programs		
Medical Systems	m15	CMCM - Configuration Management Capability Model	[16]	CMMI and ANSI/AAMI SW68
Software Quality Assurance	m16	Framework for assessing the use of third-party software quality assurance standards	[17]	ISO 9000-3 and CMM
Network	m17	Concepts for a network maturity model	[18]	CMM, ISO/IEC 15504 (SPICE)
Open Source Software	m18	Process Maturity Model for Open Source Software	[19]	CMMI-DEV, ISO/IEC 15504 (SPICE)
Performance Engineering	m19	PEMM - Performance Engineering Maturity Model	[20]	CMM
Product Line Management	m20	Evolving Standard Process Reference Models for Product Line Development	[21]	ISO/IEC 12207
Product Quality	m21	Product Process Dependencies	[22] [23]	ISO 15504, ISO 9126, Bootstrap
Railway/Safety	m22	CMMI RAMS extension based on CENELEC railway standard	[24]	CMMI SE-SW, CENELEC 50126, 50128, 50129
Requirements	m23	Formal Specifications Strategies Maturity Model	[25]	CMM
	m24	Requirements CMM	[26] [27]	SW CMM
Security Engineering/Service Oriented	m25	A CC-based Security Engineering Process Evaluation Model	[28]	ISO/IEC 15408, SSE-CMM
	m26	Development system security process of ISO/IEC TR 15504 and security considerations for software process improvement	[29]	ISO/IEC 15504, ISO/IEC 15408
	m27	Lessons learned with the systems security engineering capability maturity model	[30]	CMM
	m28	Representation of knowledge in Information Technology Service Capability Maturity Model (IT Service CMM)	[31]	SW CMM
	m29	Research on third party logistics service capability maturity model	[32]	CMM
SME (Small and Medium Enterprises)	m30	MARES Process Assessment Model	[33] [34]	ISO/IEC 15504
	m31	SATASPIN Software Process Improvement Network in the Satakunta Region	[35] [33]	ISO/IEC TR 15504
	m32	Developing International Standards for Very Small Enterprises	[36]	Moprosoft (ISO/IEC 12207, ISO/IEC 15504, ISO9001, CMMI, PMBOK)
	m33	Software processes in developing countries	[37]	ISO/IEC 12207, ISO/IEC 15504
	m34	Software Quality Improvement Model for Small Organizations	[38]	ISO 9000, CMM, ISO/IEC 15504 (SPICE), SPIRE and others
	m35	Dynamic CMM for Small Organizations	[39]	CMM
	m36	Competisoft Process Model for Software Process Improvement:	[40]	SW CMM, ISO 9000, ISO/IEC 15504, PMBOK, and others

	m37	Initiating Software Process Improvement in Small Enterprises: Experiment with MicroEvaluation Framework	[41]	SW-CMM, ISO/IEC 15504 (SPICE)
	m38	MPS.BR - Brazilian software process reference model and assessment method	[42] [43]	CMMI and ISO/IEC 15504 (SPICE)
Software and System Engineering, including development, services and acquisition.	m39	CMM (or SW-CMM) / CMMI-DEV	[44]	SW-CMM, The Systems Engineering Capability Model (SECM) , The Integrated Product Development Capability Maturity Model (IPD-CMM)
Software Engineering	m40	BOOTSTRAP	[45]	CMM, ISO 9000, DoD-STD 2167*, ESA Software Engineering Standard PSS-05-0
Space	m41	SPICE for SPACE	[46] [47]	ISO/IEC TR 15504, ISSO/IEC 12207, ECSS-E40: Space Software Engineering ECSS-Q-80: Space Software Product Assurance
SPI Implementation	m42	SPI implementation maturity model	[48]	CMMI
Telecom	m43	Trillium	[49]	CMM, ISO 9000, Bellcore TR-NWT-000179, Bellcore TA-NWT-001315 , Malcolm Baldrige National Quality Award, IEEE Software Engineering Standards Collection, IEC Standard Collection
Testing Assurance	m44	Test Maturity Model (TMM)	[50] [51]	CMM, Gelperin and Hetzel's Evolutionary Testing Model, Beizer's Progressive Phases of a Testers' Mental Model
	m45	SAMM - Modern software assurance and a five-level model of software assurance maturity	[52]	CMM
	m46	MB-V2M2 - Metrics Based Verification and Validation Maturity Model	[53]	TMM and CMM
	m47	TIM- Test Improvement Model	[54]	CMM, TMM
	m48	Criticality-Based V&V Capability Model(CB-VVCM)	[55]	IEEE Std.1012, IEEE 1012 and CMMI
	m49	A framework for the V&V capability assessment focused on the safety-criticality	[55]	IEEE Std.1012, IEEE Std.7-4.3.2, 1228 and RG 1.168, CMMI, ISO 9001
XP (eXtreme Programming)	m50	XPI - XP Based Process Improvement Framework	[56]	XP
	m51	extreme Programming Maturity Model (XPMM)	[57]	CMMI, PSP
	m52	AHAA-reference model for Agile, hybrid assessment method	[58]	CMMI, AutomotiveSPICE

		for automotive, safety critical smes		
--	--	---	--	--

- [1] Fabbrini F. et al., "Integrating joint reviews with automotive SPICE assessments results". Lecture Notes in Computer Science, vol.5007, 2008.
- [2] J. Lee et al., "An Overview of the Business Process Maturity Model (BPMM)". Lecture Notes in Computer Science, Volume 4537/2010, Springer, 2007.
- [3] Ratneshwer G., "A maturity model for CBSE". 2nd India Software Engineering Conference, pp. 127-128, 2009.
- [4] Torgersson J. and Dorling A., "Assessing CBD - What's the Difference?" pp.332, 28th Euromicro Conference, 2002.
- [5] Sen A. et al., "Data Warehousing Process Maturity: An Exploratory Study of Factors Influencing User Perceptions". IEEE Transactions on Engineering Management, vol. 53, no. 3, pp.440-455, 2006.
- [6] Visconti M. and Cook C., "Software system documentation process maturity model". ACM Conference on Computer Science, pp. 352-357, 1993.
- [7] Mengxing H. et al., "E-government maturity model and its evaluation". Journal of Southeast University Vol. 24, No. 3, pp. 389-392, 2008.
- [8] Marshall S. and Mitchell G., "Applying SPICE to e-Learning: An e-Learning Maturity Model". Conferences in Research and Practice in Information Technology, Vol. 30, pp.185-191, 2004.
- [9] Marshall, S. and Mitchell, G., "Benchmarking International E-learning Capability with the E-Learning Maturity Model". EDUCAUSE, 2007.
- [10] Cass A. et al., "SPICE in Action - Experiences in Tailoring and Extension," 28th Euromicro Conference, pp.352, 2002.
- [11] Pušnik M. et al., "Extending Information System Integrability Index with CMM Model. A Preliminary Proposal". 29th Int. Conference on Information Technology Interfaces. pp. 145-150, 2007.
- [12] Ibrahim L. and Pyster A., "A Single Model for Process Improvement," IT Professional, vol. 6, no. 3, pp. 43-49, 2004.
- [13] Feng J., "A Knowledge Management Maturity Model and Application". Technology Management for the Global Future, 2006. Vol. 3, pp. 1251-1255, 2006
- [14] April A. et al., "SMCMM Model to Evaluate and Improve the Quality of the Software Maintenance Process," 8th Euromicro Working Conference on Software Maintenance and Reengineering, pp.243, 2004.

- [15] Díaz-Ley M. et al., "MIS-PyME Software Measurement Maturity Model-Supporting the Definition of Software Measurement Programs", *Lecture Notes in Computer Science*, vol. 5089, 2009.
- [16] Mccaffery F. and Coleman G., "Developing a configuration management capability model for the medical device industry". *Int. Journal of Information Systems and Change Management* vol.2 , no.2, pp.139-154, 2007.
- [17] Bovee M. et al., "A framework for assessing the use of third-party software quality assurance standards to meet FDA medical device software process control guidelines". *IEEE Transactions on Engineering Management*, vol.48, no. 4, pp.465-478, 2001.
- [18] Capone J. et al., "Concepts for a network maturity model". *IEEE Workshop on Application - Specific Software Engineering and Technology*, 1998.
- [19] Ciolkowski M. and Soto M., "Towards a Process Maturity Model for Open Source Software". *32nd Annual IEEE Int. Computer Software and Applications Conference*, pp.1213-1214, 2008.
- [20] Scholz A. and Schmietendorf A., "A risk-driven Performance Engineering Process Approach and its Evaluation with a Performance Engineering Maturity Model". *15th UK Performance Engineering Workshop*, 1999
- [21] Hoyer C. and Chroust G., "Evolving Standard Process Reference Models for Product Line Development". *Software Engineering and Advanced Applications*, pp.320-327, 2006.
- [22] Taramaa J. et al., "Product-Based Software Process Improvement for Embedded Systems," *Euromicro*, vol. 2, pp.20905, 1998.
- [23] Hamann D. et al., "Dependency Definition Method". *Workshop on Software Process and Product Improvement*, 1998.
- [24] Fonseca J. and Almeida J., "CMMI RAMS Extension Based on CENELEC Railway Standard". *Lecture Notes in Computer Science*, Springer, Vol.3688, 2005.
- [25] Fraser M. and Vaqishnavi V., "A formal specifications maturity model". *Communications of the ACM*, Vol.40, no.12, pp.95-103, 1997.
- [26] Beecham S. et al., "Using an expert panel to validate a requirements process improvement model". *Journal of Systems and Software*, Vol.76, no. 3, pp.251-275, 2005.
- [27] Beecham, S. et al., "Building a requirements process improvement model". *Software Process: Improvement and Practice*, 2003.
- [28] Le J. et al., "A CC-based Security Engineering Process Evaluation Model" *27th COMPSAC*, pp.130, 2003.

- [29] Lee E. and Lee M., "Development System Security Process of ISO/IECTR15504 and Security Considerations for Software Process Improvement". Lecture Notes in Computer Science, Springer, vol.3481, pp. 363-372, 2005.
- [30] Hefner R., "Lessons learned with the systems security engineering capability maturity model", 19th Int. Conference on Software Engineering, p.566-567, 1997.
- [31] Daneshgar F. et al., "Representation of knowledge in information technology Service Capability Maturity Model (IT Service CMM)". Research Challenges in Information Science, pp.215-226, 2008.
- [32] Qiao H. and Zhao Q., "Research on Third Party Logistics Service Capability Maturity Model". IEEE International Conference on Service Operations and Logistics, and Informatics, vol.2, pp.2858-2861, 2008.
- [33] Wangenheim C. et al., "Standard based software process assessments in small companies". Software Process Improvement and Practice, vol.11, no. 3, pp.329-335, 2006.
- [34] Wangenheim C. et al., "Helping small companies assess software processes". IEEE Software, vol.23, no.1, pp.91-98, 2006.
- [35] Varkoi T. et al., "Process improvement priorities in small software companies". Portland Int. Conference on Management of Engineering and Technology, pp.555, 1999.
- [36] Laporte C. Y. et al., "Developing International Standards for Very Small Enterprises". IEEE Computer, Vol.41, Iss.3, pp.98-101, 2008.
- [37] Pino F. J. et al., "Adaptation of the standards ISO/IEC 12207:2002 and ISO/IEC 15504:2003 for the assessment of the software processes in developing countries", IEEE Latin America Transactions, vol.4, no.2, pp.85-92, 2006.
- [38] Zeineddine R. and Mansour N., "Software Quality Improvement Model for Small Organizations". Lecture Notes in Computer Science, Springer, vol.2869, 2003.
- [39] Laryd A. and Orci T., "Dynamic CMM for Small Organizations". 1. Argentine Symposium on Software Engineering, pp.133-149, 2000.
- [40] Oktaba H. et al., "'Software Process Improvement: The Competisoft Project". IEE Computer, vol.40, no.10, pp.21-28, 2007.
- [41] Laporte C. et al., "Initiating Software Process Improvement in Small Enterprises: Experiment with Micro-Evaluation Framework". Int. Conference on Software Development, pp.153-163, 2005.
- [42] Rocha A. R. et al., "Process Reference Model and Assessment Method". Lecture Notes in Computer Science, Springer, vol.3733, 2005.
- [43] Weber K. et al., "Brazilian Software Process Reference Model and Assessment Method". Lecture Notes in Computer Science, Springer, vol.3733, 2005.

- [44] Paulk M. C. et al., "Capability Maturity Model, Version 1.1", IEEE Software, vol.10 n.4, p.18-27, 1993.
- [45] Kuvaja P., "BOOTSTRAP: A software process assessment and improvement methodology". Lecture Notes in Computer Science, Springer, vol.926, pp.31-48, 1995.
- [46] Cass et al., "SPICE for SPACE trials, risk analysis, and process improvement". Software Process: Improvement and Practice, vol.9, no.1, pp.13-21, 2004.
- [47] Cass et al., "SPICE for SPACE: A Process Assessment and Improvement Method for Space Software Development", European Space Agency Bulletin 107, 2001.
- [48] Mahmood N. et al., "A maturity model for the implementation of software process improvement: an empirical study", Journal of Systems and Software, Vol.74, n.2, pp.155-172, 2005.
- [49] April A. and Coallier F., "Trillium: a model for the assessment of telecom software system development and maintenance capability", 2nd IEEE Software Engineering Standards Symposium, pp.175, 1995.
- [50] Rana K. and Ahmad S., "Bringing maturity to test". Electronics Systems and Software, vol.3, no.2, 2005.
- [51] Burnstein I. et al., "Developing a Testing Maturity Model, Part II", Crosstalk, September, 1996.
- [52] Bush M., "Modern Software Assurance and a Five-Level Model of Software Assurance Maturity", Journal of High Integrity Systems, 1.2, pp.157-169, 1994.
- [53] Jacobs J and Trienekens J., "Towards a Metrics Based Verification and Validation Maturity Model", 10th Int. Workshop on Software Technology and Engineering Practice, pp.123, 2002.
- [54] Ericson T. et al., "TIM - a test improvement model". Software Testing, Verification and Reliability, Vol.7, Iss.4, pp.229-246, 1996.
- [55] Kyung-A Y. et al., "A Framework for the V&V Capability Assessment Focused on the Safety-Criticality". 13th IEEE Int. Workshop on Software Technology and Engineering Practice, pp.17-24, 2005
- [56] Ramachandran M., "A Process Improvement Framework for XP Based SMEs". Lecture Notes in Computer Science, Springer, vol.3556, pp.202-205, 2005.
- [57] Nawrocki J. et al., "Toward maturity model for extreme programming", 27th Euromicro Conference, pp.233-239, 2001.
- [58] McCaffery F. et al., "Ahaa - Agile, Hybrid Assessment Method for Automotive, Safety Critical SMEs". 30th Int. Conference on Software Engineering, pp.551-560, 2008.

APÊNDICE B – Dimensão de Maturidade/Capacidade de SPCMMs

Este apêndice apresenta a dimensão de capacidade/maturidade dos SPCMMs encontrados na revisão sistemática de literatura (WANGENHEIM et al, 2010b). Cada modelo é caracterizado em duas colunas: a primeira apresenta seu nome e iniciais e uma referência para o artigo em que ele é descrito (*Capability/Maturity Models & Reference(s)*), e a segunda apresenta quais níveis de maturidade e/ou capacidade de processos compõem esta dimensão da sua estrutura. A referência numérica equivale à lista de publicações apresentadas no Apêndice B.

Capability/ Maturity Models & Reference(s)	Capability (CL) or Maturity (ML) Levels							
AutomotiveSPICE Process Assessment Model [3]	CL 0 Incomplete Process	CL 1 Performed Process	CL 2 Managed Process	CL 3 Established Process	CL 4 Predictable Process	CL 5 Optimizing Process		
Business Process Maturity Model (BPMM) [4]	ML 1 Initial	ML 2 Managed	ML 3 Defined	ML 4 Quantitatively Managed		ML 5 Optimizing		
Integrated Component Maturity Model [5]	ML 1 Preliminary Level	ML 2 Component Reuse Orientation Level	ML 3 Managed Quality Orientation Level	ML 4 'Formal Definition And Quantitative Analysis' Level		ML 5 Innovation Level		
Process Assessment Model for Software Component-based Development (OOSPICE) [6]	CL 0 Not Performed	CL 1 Informal	CL 2 Planned	CL 3 Defined	CL 4 Controlled	CL 5 Optimized		
Data Warehousing Process Maturity Model [7]	ML 1 Initial Level	ML 2 Repeatable Level	ML 3 Defined Level	ML 4 Managed Level		ML 5 Optimizing Level		
Software System Documentation Process Maturity Model [8]	ML 1 Ad-hoc		ML 2 Inconsistent		ML 3 Defined		ML 4 Controlled	
E-government Maturity Model (EGMM) [9]	ML 1 Network Infrastructure		ML 2 Information Serving		ML 3 Information Interactivity		ML 4 Information Sharing	ML 5 Comprehensive Integration
E-learning Maturity Model [10] [11]	CL 0 Not Performed	CL 1 Initial	CL 2 Planned	CL 3 Defined	CL 4 Managed	CL 5 Optimising		
Extending information system integrability index with CMM model [12]	Data level	Application interfaces	Business methods	Presentation	Security	B2B		
Knowledge Management Maturity Model [13]	ML 1 Initialization level	ML 2 Iteration level	ML 3 Definition level	ML 4 Management level		ML 5 Optimization level		

Software Maintenance Maturity Model [14]	CL 0 Incomplete	CL 1 Performed	CL 2 Managed	CL 3 Established	CL 4 Predictable	CL 5 Optimizing
Software measurement maturity model (MIS-PyME) [15]	ML 1 Level 1	ML 2 Level 2	ML 3 Level 3	ML 4 Level 4	ML 5 Level 5	
Configuration Management Capability Model (CMCM) [16]	CL 0 Level Med	CL 1 Level 1	CL 2 Level 2	CL 3 Level 3	CL 4 Level 4	CL 5 Level 5
Network Maturity Model (NMM) [17]	CL 1 Informal	CL 2 Structured	CL 3 Standardized	CL 4 Measured	CL 5 Optimizing	
Performance Engineering Maturity Model (PEMM) [19]	ML 1 Uncoordinated Practices	ML 2 Consideration of PE subprocesses	ML 3 Entire PE Process Definition	ML 4 Successful integrated and proved PE process	ML 5 Optimized PE process	
CMMI RAMS extension based on CENELEC railway standard [23]	CL 0 Incomplete	CL 1 Executed	CL 2 Managed	CL 3 Defined	CL 4 Quantitatively Managed	CL 5 Optimized
Formal Specifications Strategies Maturity Model [24]	ML 1 Initial	ML 2 Repeatable	ML 3 Defined	ML 4 Managed	ML 5 Optimizing	
Requirement CMM (R-CMM) [25] [26]	ML 1 Initial ad-hoc Software Processes	ML 2 Repeatable Software Process	ML 3 Defined Software Processes	ML 4 Managed Software Processes	ML 5 Optimizing Software Processes	
A CC-based Security Engineering Process Evaluation Model [27]	CL 0 Not Performed	CL 1 Performed Informally	CL 2 Planned and Tracked	CL 3 Well Defined	CL 4 Quantitatively Controlled	CL 5 Continuously Improving
Systems Security Engineering Capability Maturity Model [29]	CL 0 Initial	CL 1 Performed	CL 2 Planned and Tracked	CL 3 Well Defined	CL 4 Quantitatively Controlled	CL 5 Continuously Improving
IT Service CMM [30]	ML 1 Initial	ML 2 Repeatable	ML 3 Defined	ML 4 Managed	ML 5 Optimizing	
Research on third party logistics service capability maturity	ML 1 Bud Level	ML 2 Growing Level	ML 3 Maturity Level	ML 4 Stabilization Level	ML 5 Optimization Level	

model (3PL-SCMM) [31]								
MARES Process Assessment Model [32] [33]	CL 0 Incomplete		CL 1 Managed		CL 2 Established		CL 3 Defined	
Software Process Improvement Network (SATASPIN) [34]	CL 0 Incomplete	CL 1 Managed	CL 2 Established	CL 3 Defined	CL 4 Predictable	CL 5 Optimizing		
Software Processes in Developing Countries (MECPDS) [36]	CL 0 Incomplete Process		CL 1 Realized Process			CL 2 Managed Process		
Dynamic CMM for Small Organizations [38]	ML 1 Initial				ML 2 Repeatable			
Process Model for Software Process Improvement (Competisoft) [39]	CL 0 Incomplete	CL 1 Performed	CL 2 Managed	CL 3 Established	CL 4 Predictable	CL 5 Optimizing		
Software Process Improvement in Small Enterprises (MicroEvaluation Framework) [40]	CL 0 Score 1		CL 1 Score 1		CL 2 Score 1		CL 3 Score 1	
Brazilian software process reference model and assessment method (MPS.BR) [41] [42]	ML 1 Level G Partially Managed	ML 2 Level F Managed	ML 3 Level E Partially Defined	ML 4 Level D Largely Defined	ML 5 Level C Defined	ML6 Level B Quantitatively Managed	ML 7 Level A Optimizing	
Capability Maturity Model Integration (CMMI-DEV) [43]	Staged	ML 1 Initial	ML 2 Repeatable	ML 3 Defined	ML 4 Managed	ML 5 Optimizing		
	Continuous	CL 0 Incomplete	CL 1 Performed	CL 2 Managed	CL 3 Defined	CL 4 Quantitatively Managed	CL 5 Optimizing	
ISO/IEC 15504 Process Assessment Model (ISO/IEC 15504-5) [44]	CL 0 Incomplete	CL 1 Performed	CL 2 Managed	CL 3 Established	CL 4 Predictable	CL 5 Optimizing		
	ML 0 Organization:	ML 2 Organization:	ML 2 Organization:	ML 3 Organization:	ML 4 Organization:	ML 5 Organization:		

	Immature	Basic	Managed	Established	Predictable	Innovating
Integrated Capability maturity Model (iCMM) [45]	ML 1 Level 1	ML 2 Level 2	ML 3 Level 3	ML 4 Level 4	ML 5 Level 5	
	CL 0 Incomplete	CL 1 Performed	CL 2 Managed: planned and tracked	CL 3 Defined	CL 4 Quantitatively Managed	CL 5 Optimizing
BOOTSTRAP [46]	CL 0 Incomplete Process	CL 1 Performed Process	CL 2 Managed Process	CL 3 Established Process	CL 4 Predictable Process	CL 5 Optimizing Process
SPICE for SPACE [48] [49]	CL 0 Incomplete	CL 1 Performed	CL 2 Managed	CL 3 Established	CL 4 Predictable	CL 5 Optimizing
SPI implementation maturity model [50]	ML 1 Initial		ML 2 Aware	ML 3 Defined	ML 4 Optimizing	
Trillium [51]	ML 1 Unstructured	ML 2 Repeatable and Project Oriented	ML 3 Defined and Process Oriented	ML 4 Managed and Integrated	ML 5 Fully Integrated	
Testing Maturity Model (TMM) [52] [53]	ML 1 Initial	ML 2 Phase Definition	ML 3 Integration	ML 4 Management and Measurement	ML 5 Optimization, Defect Prevention, and Quality Control	
Metrics Based Verification and Validation Maturity Model (MB-V2M2) [55]	ML 1 Initial	ML 2 Repeatable	ML 3 Defined	ML 4 Managed and Aligned	ML 5 Optimizing.	
Test Improvement Model (TIM) [56]	ML 1 Baseline		ML 2 Cost-effectiveness	ML 3 Risk-lowering	ML 4 Optimizing	
Criticality-Based V&V Capability Model (CB-VVCM) [57]	ML 1 None	ML 2 Low	ML 3 Mediate	ML 4 High	ML 5 Rigorous	
Extreme Programming Maturity Model (XPMM) [59]	ML 1 Not compliant at all		ML 2 Initial	ML 3 Pair programming	ML 4 Project performance	

APÊNDICE C – Questionário Survey

Este apêndice apresenta o questionário utilizado para o survey entre os autores de SPCMMs. Primeiramente é apresentada a versão original do questionário em inglês seguida da versão traduzida do mesmo.

Questionário Original

Capability/maturity model name		<i>For example: SW-CMM (CMMI-DEV)</i>
Institution who developed/ coordinated the development		<i>For example: SEI/CMU</i>
Principal author(s)		<i>For example: D. Mike Phillips</i>
Primary reference documenting the model		<i>For example: A.A. Author, "Article Title Here," Computer, May 2003, pp. 40-46.</i>
Year of publication	First version	<i>When was the first version of the model published? For example: SW-CMM 1991</i>
	Current version	<i>When has the current version of the model published? For example: CMMI-DEV v1.2 2006</i>
Degree of usage		<i>How MANY organizations approximately are using the model WHERE (worldwide / in the US, in Latin America, etc.)?</i>
Scope		<i>What is the scope of the model? Such as, software development, SME, maintenance, service-orientation, CBSE....</i>
First version has been based on		<i>Such as, ISO 9000, CMM, CMMI, ISO/IEC 15504,</i>
Objective of the model		<i>What is the objective of the model?</i>
Who was involved in its development?	Range	<i>Such as, international, national, regional or local representatives</i>
	Representatives	<i>Such as, representatives from industry, government, universities</i>
	Form of participation	<i>Such as, open for public participation, by invitation only</i>
	Approx. number of representatives	<i>How many representatives participated in the development of the model?</i>

Method/procedure used for model development		<i>Was an existing method used for the development of the model? If yes, state the name of the method and a reference where the method is documented.</i>	
Process used for development and evolution of the model		<i>Describe briefly WHAT and HOW for each of the following steps. If necessary, feel free to change the order of steps, include additional steps or delete steps, which do not apply.</i>	
		WHAT was done?	HOW? (describing how and indicating techniques used, such as, interviews, brainstorming, focus groups, survey, experiments, case studies, systematic literature review, discussions, etc.)
Knowledge Identification	Familiarize with domain <i>for which the model will be developed.</i>		
	Identify information sources <i>that will be used as input for the model development.</i>		
	Define scope and goals <i>of the model to be developed.</i>		
	Formalize the working group <i>for the development of the model, including e.g., the allocation of a sponsor/coordinator, working rules and procedures, etc.</i>		
Knowledge Specification	Develop the design/ architecture of the model <i>identifying the principal elements of the model.</i>		
	Develop a draft model – process dimension - <i>how has the process dimension of the model be developed?</i>		
	Develop a draft model – capability/ maturity dimension - <i>how has the capability/maturity dimension of the model be developed?</i>		
Knowledge	Validate draft model (before publication).		

Refinement	<i>If and how the draft model itself has been validated with respect to which aspects (correctness/ completeness/ etc.)</i>		
	Consolidate draft model <i>until the working group is satisfied that it has developed the best technical solution to the problem being addressed.</i>		
	Ballot on the consolidated model – <i>describe who votes and how consensus is attained.</i>		
	Approve the model – <i>indicate the criteria for approval and what happens when (or not) approved.</i>		
	Publish – <i>activities involved in publishing the model where and how.</i>		
Knowledge Usage	Support model usage – <i>which kind of support for the model usage is provided? Such as, trainings, user forums, etc.</i>		
	Validate model in use – <i>any activities focusing on validating the model itself (correctness/ completeness/ etc.) once it is being used in practice.</i>		
Knowledge Evolution	Change request management – <i>are change request collected from whom and how are they managed?</i>		
	Confirmation, revision or withdrawal – <i>how is this decision made and how are new versions of the model developed?</i>		

Questionário Traduzido

Modelo de Capacidade/Maturidade		<i>Por exemplo: SW-CMM (CMMI-DEV)</i>
Instituição que desenvolveu/coordenou o desenvolvimento		<i>Por exemplo: SEI/CMU</i>
Principal(is) Autor(s)		<i>Por exemplo: D. Mike Phillips</i>
Principal documento de referência do modelo		<i>Por exemplo: A.A. Author, "Article Title Here," Computer, May 2003, pp. 40-46.</i>
Ano de Publicação	Primeira versão	<i>Quando a primeira versão do modelo foi publicada? Por exemplo: SW-CMM 1991</i>
	Versão atual	<i>Quando a versão atual do modelo foi publicada? Por exemplo: CMMI-DEV v1.2 2006</i>
Grau de utilização		<i>QUANTAS organizações estão utilizando o modelo (aproximadamente) e ONDE (mundo todo /no Estados Unidos, na América Latina, etc.)?</i>
Escopo		<i>Qual é o escopo do modelo? Como: software development, SME, maintenance, service-orientation, CBSE....</i>
A primeira versão foi baseada em		<i>Exemplo: ISO 9000, CMM, CMMI, ISO/IEC 15504,</i>
Objetivo do modelo		<i>Qual é o objetivo do modelo?</i>
Quem foi envolvido no seu desenvolvimento?	Faixa	<i>Tais como: internacional, nacional, regional ou local</i>
	Representantes	<i>Tais como, indústria, governo, universidades</i>
	Forma de participação	<i>Tais como: aberto à participação pública, somente por convite</i>
	Número aproximado de	<i>Quantos representantes participaram do desenvolvimento do modelo?</i>

	representantes		
Método/Processo utilizado no desenvolvimento do modelo		<i>Algum método existente foi utilizado para o desenvolvimento do modelo? Se sim, qual o nome do método e qual a referência de documentação deste método?</i>	
Processo utilizado para desenvolvimento e evolução do modelo		<i>Descreva brevemente O QUE e COMO para cada um dos passos a seguir. Sinta-se livre para alterar a ordem dos passos ou apagá-los se não se aplicarem.</i>	
		O que foi feito?	Como? <i>(descrevendo como e indicando as técnicas utilizadas, como entrevistas, brainstorming, focus groups, survey, experiments, case studies, systematic literature review, discussões, etc.)</i>
Identificação do Conhecimento	Familiarização com o Domínio <i>para o qual o modelo foi desenvolvido.</i>		
	Identificação das fontes de informação <i>que serão utilizadas como entrada para o desenvolvimento do modelo</i>		
	Definição do escopo e dos objetivos <i>do modelo a ser desenvolvido</i>		
	Formalização do grupo de trabalho <i>para o desenvolvimento do modelo, incluindo, por exemplo, a alocação de um patrocinador/coordenador, regras de trabalho, procedimentos, etc.</i>		
Especificação do Conhecimento	Desenvolver o design/arquitetura do modelo <i>identificando os elementos principais do modelo</i>		
	Desenvolvimento do rascunho do modelo – dimensão de processo <i>como a dimensão de processo foi desenvolvida?</i>		

	Desenvolvimento do rascunho do modelo – dimensão de capacidade/maturidade <i>como a dimensão de capacidade/maturidade foi desenvolvida?</i>		
Refinamento do Conhecimento	Validação do rascunho do modelo (antes da publicação). <i>Se e como o rascunho do modelo foi validado com respeito a que aspectos (corretude, completude, etc)</i>		
	Consolidação do rascunho do modelo <i>até que o grupo de trabalho esteja convencido de que foi desenvolvida a melhor solução técnica para o problema</i>		
	Votação no modelo consolidado <i>descreva que votou e como o consenso foi atingido.</i>		
	Aprovação do modelo <i>indique os critérios de aprovação e o que aconteceu quando foi aprovado (ou não)</i>		
	Publicação <i>atividades envolvidas na publicação do modelo</i>		
Utilização do Conhecimento	Suporte ao uso do modelo <i>que tipo de suporte foi disponibilizado para o uso do modelo? Tais como: treinamentos, fóruns, etc.</i>		
	Validação do modelo em uso <i>atividades que foquem na validação do uso prático do modelo em si (corretude, completude, etc).</i>		
Evolução do Conhecimento	Gerenciamento de solicitações de alteração <i>solicitações de alteração</i>		

	<i>foram coletadas por que e como foram gerenciadas?</i>		
	Confirmação, revisão ou retirada <i>como as decisões foram tomadas e como as novas versões foram desenvolvidas?</i>		

APÊNDICE D – Questionário *Expert Panel*

Este apêndice apresenta o questionário utilizado para o *Expert Panel* no qual o método foi avaliado por um grupo de especialistas. Como a versão final do questionário foi disponibilizada via formulário WEB, a apresentação final do questionário diferencia-se no formato, porém o conteúdo é exatamente o apresentado neste apêndice.

Evaluation of the Knowledge Engineering Based Method for SPCMMs customization

This form aims to evaluate the Knowledge Engineering Based Method for SPCMMs customization.

Demographic Information

In this first part of the evaluation, we would like to know more about you. Please answer the questions about your previous experience.

1. What is your academic background?

PhD, Master, Bachelor, another

2. What is your experience in Software Process Improvement?

Less than 1 year, 1 to 5 years, 5 to 10 years, more than 10 years

3. Have you been involved in the development of maturity/capability models? Which ones? (Please list the names of the models, separated by ;)

Detailed evaluation of the method

We would like to know if you have found any errors, inconsistencies or incompleteness of the method in the document.

Phase 1: Knowledge Identification

Activities of this phase:

- A1.1 - Familiarize with domain
- A1.2 - Identify information sources
- A1.3 - Define scope and goals

- A1.4 - Formalize the working group

4. Did you find any errors in the activities, techniques, work products or tools in phase 1? (Please indicate any error in the following way: activity code - description;)

5. Did you find any inconsistency in the activities, techniques, work products or tools involved in phase 1? (Please indicate any error in the following way: activity code - description;)

6. Did you notice the absence of any activities, techniques, work products or tools that you consider important in this phase 1? (Please indicate any error in the following way: activity code - description;)

Phase 2: Knowledge Specification

Activities of this phase:

- A2.1 - Develop the design/architecture of the model
- A2.2 - Analyze and integrate existing related models
- A2.3 - Develop a draft model - process dimension
- A2.4 - Develop a draft model - capability/maturity dimension

7. Did you find any errors in the activities, techniques, work products or tools in phase 2? (Please indicate any error in the following way: activity code - description;)

8. Did you find any inconsistency in the activities, techniques, work products or tools involved in phase 2? (Please indicate any error in the following way: activity code - description;)

9. Did you notice the absence of any activities, techniques, work products or tools that you consider important in this phase 2? (Please indicate any error in the following way: activity code - description;)

Phase 3: Knowledge Refinement

Activities of this phase:

- A3.1 - Evaluate draft model
- A3.2 - Consolidate draft model
- A3.3 - Ballot on the consolidated model
- A3.4 - Approve the model
- A3.5 - Publish

10. Did you find any errors in the activities, techniques, work products or tools in phase 3? (Please indicate any error in the following way: activity code - description;)

11. Did you find any inconsistency in the activities, techniques, work products or tools involved in phase 3? (Please indicate any error in the following way: activity code - description;)

12. Did you notice the absence of any activities, techniques, work products or tools that you consider important in this phase 3? (Please indicate any error in the following way: activity code - description;)

Phase 4: Knowledge Usage

Activities of this phase:

- A4.1 - Support model usage
- A4.2 - Validate model in use

13. Did you find any errors in the activities, techniques, work products or tools in phase 4? (Please indicate any error in the following way: activity code - description;)

14. Did you find any inconsistency in the activities, techniques, work products or tools involved in phase 4? (Please indicate any error in the following way: activity code - description;)

15. Did you notice the absence of any activities, techniques, work products or tools that you consider important in this phase 4? (Please indicate any error in the following way: activity code - description;)

Phase 5: Knowledge Evolution

Activities of this phase:

- A5.1 - Change request management
- A5.2 - Confirmation, revision or withdrawal

16. Did you find any errors in the activities, techniques, work products or tools in phase 5? (Please indicate any error in the following way: activity code - description;)

17. Did you find any inconsistency in the activities, techniques, work products or tools involved in phase 5? (Please indicate any error in the following way: activity code - description;)

18. Did you notice the absence of any activities, techniques, work products or tools that you consider important in this phase 5? (Please indicate any error in the following way: activity code - description;)

Overall Evaluation

We would like to know what your overall opinion on the method.

19. The method adequately represents what is necessary to customize a SPCMM.

Strongly disagree - Strongly agree

20. The method enables the creation of a model that really contains the best practices of the respective domain.

Strongly disagree - Strongly agree

21. The method provides more support than methods/processes currently available.

Strongly disagree - Strongly agree

22. The development of a SPCMM using this method can result in a valid model (considering as characteristics of a valid model: generality, flexibility, completeness, usability and understandability).

Strongly disagree - Strongly agree

23. Models developed using the method can satisfy the requirements of ISO/IEC 15504-2 for PRM and PAM.

Strongly disagree - Strongly agree

24. The method is comprehensive enough to allow customization of SPCMMs for different domains.

Strongly disagree - Strongly agree

25. When developing a SPCMM in the future, I would recommend the use of this method.

Strongly disagree - Strongly agree

26. In your opinion, what are the three main strengths of the method?

27. In your opinion, what are the three main weaknesses of the method?

28. Do you have any other comment about the method?

ANEXO A – Certificado Comitê de Ética



UNIVERSIDADE FEDERAL DE SANTA CATARINA
Pró-Reitoria de Pesquisa e Extensão
Comitê de Ética em Pesquisa com Seres Humanos

CERTIFICADO Nº 434

O Comitê de Ética em Pesquisa com Seres Humanos (CEPSH) da Pró-Reitoria de Pesquisa e Extensão da Universidade Federal de Santa Catarina, instituído pela PORTARIA N.º584/GR/99 de 04 de novembro de 1999, com base nas normas para a constituição e funcionamento do CEPSH, considerando o contido no Regimento Interno do CEPSH, **CERTIFICA** que os procedimentos que envolvem seres humanos no projeto de pesquisa abaixo especificado estão de acordo com os princípios éticos estabelecidos pela Comissão Nacional de Ética em Pesquisa – CONEP.

APROVADO

PROCESSO: 434

FR: 299922

TÍTULO: Como modelos de maturidade/capacidade de software são desenvolvidos

AUTOR: Christiane A. Gresse von Wangenheim, Jean Carlo Rossa Hauck

FLORIANÓPOLIS, 30 de Novembro de 2009.

Coordenador do CEPSH/UFSC

