



COPPE/UFRJ

UMA ABORDAGEM PARA BENCHMARKING EM INICIATIVAS DE
IMPLEMENTAÇÃO DE MELHORIAS EM PROCESSOS DE SOFTWARE

David Bom Zanetti

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Ciências em Engenharia de Sistemas e Computação.

Orientador(es): Ana Regina Cavalcanti da Rocha


Rio de Janeiro
Novembro de 2008


UMA ABORDAGEM PARA BENCHMARKING EM INICIATIVAS DE
IMPLEMENTAÇÃO DE MELHORIAS EM PROCESSOS DE SOFTWARE

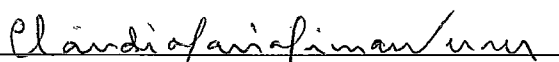
David Bom Zanetti

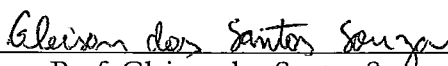
DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO
LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA
(COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:


Prof.^a Ana Regina Cavalcanti da Rocha, D. Sc


Prof.^a Sheila dos Santos Reinehr, D. Sc.


Prof.^a Claudia Maria Lima Werner, D. Sc.


Prof. Gleison dos Santos Souza, D. Sc.

RIO DE JANEIRO, RJ - BRASIL

NOVEMBRO DE 2008

Zanetti, David Bom

Uma Abordagem para Benchmarking em Iniciativas de Implementação de Melhorias em Processos de Software/ David Bom Zanetti. – Rio de Janeiro: UFRJ/COPPE, 2008.

XII, 115 p.: il.; 29,7 cm.

Orientador: Ana Regina Cavalcanti da Rocha

Dissertação (mestrado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2008.

Referencias Bibliográficas: p. 96-105.

1. Benchmarking. 2. Melhoria de Processos de Software. 3. Qualidade de Software. I. Rocha, Ana Regina Cavalcanti da. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Aos meus pais e irmãos.

AGRADECIMENTOS

Aos meus pais e irmãos, por todo amor e carinho e pelo apoio incondicional durante todas as etapas da minha formação acadêmica e também cidadã.

À professora Ana Regina, pela orientação tão necessária, por todas as portas que me abriu, e por acreditar no meu potencial, desde o ingresso no Laboratório de Engenharia de Software.

Ao amigo, professor e também orientador informal Mariano Montoni, por todos os ensinamentos que vem me fornecendo por todos os anos de COPPE/UFRJ, acreditando e apostando no meu potencial e me incentivando a melhorar cada vez mais.

Ao amigo e professor Gleison Santos, pela paciência sempre necessária, pelos ensinamentos, pela confiança depositada e pelo espírito de equipe.

À Gisele Moreira, pelo companheirismo e pela paciência para agüentar um aluno de mestrado bastante atarefado e muitas vezes mal humorado.

À amiga Anne Elise Katsurayama, pelo companheirismo e apoio durante todo o mestrado.

Às professoras Claudia Werner e Sheila Reinehr por participarem da banca e com isso contribuir com este trabalho.

Aos amigos do Laboratório de Engenharia de Software, André, Gustavo, Rômulo, Sávio, Rafael, Vitor, e outros que não me recordo agora, pela amizade e colaboração em todas as atividades do dia-a-dia.

A todos os meus amigos, pela amizade sempre necessária e confortante.

À Taisa Guidini, Flávia Decolo, Claudia Prata, Solange, Mercedes e demais funcionários do PESC pelo auxílio com toda a burocracia necessária.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

UMA ABORDAGEM PARA BENCHMARKING EM INICIATIVAS DE
IMPLEMENTAÇÃO DE MELHORIAS EM PROCESSOS DE SOFTWARE

David Bom Zanetti

Novembro/2008

Orientadora: Ana Regina Cavalcanti da Rocha

Programa: Engenharia de Sistemas e Computação

As pressões existentes no mercado de software motivam uma constante necessidade de melhorar os produtos e serviços fornecidos pelas organizações desenvolvedoras de software. Nesse contexto, iniciativas de melhoria de processos de software tornaram-se bastante difundidas. Com isso, torna-se necessário, definir estratégias adequadas para implementação de melhoria de processos de software, utilizando conhecimentos de melhores práticas e fatores de sucesso obtidos em programas de melhoria já realizados e/ou em andamento, de forma a garantir o sucesso dos projetos de melhoria.

Esta dissertação trata da definição de uma abordagem para a realização de atividades de *benchmarking* no contexto de iniciativas de implementação de melhoria de processos de software sob o ponto de vista das organizações que prestam serviços de consultoria em melhoria de processos de software. Este trabalho propõe um processo com as etapas necessárias às atividades de *benchmarking* e uma ferramenta de apoio à realização destas atividades.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M. Sc.).

AN APPROACH TO BENCHMARKING IN SOFTWARE PROCESS
IMPROVEMENT INITIATIVES

David Bom Zanetti

November/2008

Advisor: Ana Regina Cavalcanti da Rocha

Department: System and Computing Engineering

The pressure made by the market motivates a constant necessity in improving the products and services provided by software organizations. In this context, software process improvement has become a common approach. So, it is necessary to define adequate software process improvement strategies, applying best practices and knowledge related to critical success factors acquired in past and ongoing projects in order to succeed in these initiatives.

This dissertation presents the definition and development of a Benchmarking approach that supports the execution of Benchmarking activities in the context of software process improvement initiatives, from the point of view of the software process improvement consulting organizations. This approach is composed by a Benchmarking process and an integrated tool that supports the process.

ÍNDICE

CAPÍTULO 1 – INTRODUÇÃO	1
1.1 Motivação	1
1.2 Objetivos	4
1.3 Estrutura da Dissertação	5
CAPÍTULO 2 – MELHORIA CONTÍNUA DE PROCESSOS DE SOFTWARE.....	7
2.1 Introdução	7
2.2 O Ciclo PDCA.....	9
2.3 QIP.....	10
2.4 Fábrica de Experiências.....	13
2.5 Normas e Modelos de Referência.....	16
2.6 Fatores Críticos de Sucesso	25
2.7 Considerações finais.....	32
CAPÍTULO 3 – <i>BENCHMARKING</i>	33
3.1 Introdução	33
3.2 O que é Benchmarking?	34
3.3 Definições.....	38
3.4 Tipos de Benchmarking	39
3.5 Modelos de Processo de Benchmarking.....	41
3.6 Trabalhos Correlatos.....	46
3.7 Considerações finais.....	50
CAPÍTULO 4 – UMA PROPOSTA PARA <i>BENCHMARKING</i> DE INICIATIVAS DE MELHORIAS EM PROCESSOS DE SOFTWARE.....	52
4.1 Introdução	52
4.2 Uma Abordagem para Condução de Iniciativas de Melhorias em Processos de Software	53
4.3 Uma Proposta para Benchmarking de Iniciativas de Melhorias em Processos de Software	58
4.4 Processos de Apoio ao Benchmarking de Iniciativas de Melhorias em Processos de Software	62
4.5 Considerações finais.....	71

CAPÍTULO 5 – UMA FERRAMENTA DE APOIO ÀS ATIVIDADES DE <i>BENCHMARKING</i> DE INICIATIVAS DE MELHORIAS EM PROCESSOS DE SOFTWARE.....	72
5.1 Introdução	72
5.2 CORE-KM: Um ambiente de Gestão de Conhecimento	73
5.3 Requisitos do Apoio Ferramental às Atividades de Benchmarking em Iniciativas de Melhorias em Processos de Software	76
5.4 Uma Ferramenta de Apoio às Atividades de Benchmarking de Iniciativas de Melhorias em Processos de Software	78
5.5 Considerações finais.....	90
CAPÍTULO 6 – CONCLUSÃO	91
6.1 Considerações finais.....	91
6.2 Contribuições.....	92
6.3 Limitações.....	93
6.4 Perspectivas futuras	94
REFERÊNCIAS BIBLIOGRÁFICAS	96
ANEXO I - ESPECIFICAÇÃO DE REQUISITOS	106
I.1 Necessidades do Cliente	106
I.2 Modelo de Domínio	107
I.3 Requisitos do Cliente.....	107
I.4 Requisitos Funcionais de Software.....	108
I.5 Requisitos Não-Funcionais de Software	110
I.6 Caso de Uso e Processos de Apoio.....	111
I.7 Requisitos de Produto do CORE-KM	114

ÍNDICE DE FIGURAS

Figura 2.1 – Ciclo do PDCA	9
Figura 2.2 – Ciclos do QIP	12
Figura 2.3 – Infra-estrutura da abordagem de Fábrica de Experiências	15
Figura 2.4 – Estrutura de processos da norma ISO/IEC 12207	19
Figura 2.5 – Processo baseado no sistema de gestão da qualidade	22
Figura 4.1 – Componentes da Abordagem para Condução de Iniciativas de Melhorias em Processos de Software.....	56
Figura 4.2 – Cenário de execução de atividades de <i>Benchmarking</i> durante o planejamento dos projetos de implementação de melhorias em processos de software	59
Figura 4.3 – Cenário de execução de atividades de <i>Benchmarking</i> durante o monitoramento dos projetos de implementação de melhorias em processos de software.	60
Figura 4.4 – Perspectivas de visualização.....	64
Figura 4.4 – Processo de Gerência da Base de <i>Benchmarking</i>	66
Figura 4.5 – Processo de Apoio à Realização de <i>Benchmarking</i>	67
Figura 5.1 – Representação esquemática da infra-estrutura do ambiente CORE-KM	74
Figura 5.2 – Infra-estrutura em camadas do ambiente CORE-KM	75
Figura 5.3 – Requisitos de Cliente X Requisitos Funcionais	79
Figura 5.4 – Apoio à atividade “Criar Base de <i>Benchmarking</i> ”	80
Figura 5.5 – Itens de Caracterização dos Projetos de <i>Benchmarking</i>	81
Figura 5.6 – Escalas de Itens de Caracterização	82
Figura 5.7 – Perspectivas de Visualização da Base de <i>Benchmarking</i>	83
Figura 5.8 – Projeto de Implementação de Melhorias em Processos de Software.....	84
Figura 5.9 – Caracterização do Projeto “CCA-SJ Nível E”	85
Figura 5.10 – <i>Benchmarks</i> do Projeto “CCA-SJ Nível E”	86
Figura 5.11 – Empacotamento dos <i>Benchmarks</i> do Projeto “CCA-SJ Nível E”	87
Figura 5.12 – <i>Baselines</i> de Desempenho dos projetos.....	87
Figura 5.13 – Seleção dos Critérios de Busca dos Projetos.....	88
Figura 5.14 – Resultados do <i>Benchmarking</i>	89
Figura 5.15 – Relatório detalhado do <i>Benchmarking</i>	90
Figura I.1 – Diagrama de domínio.....	107
Figura I.2 – Processo de Gerência da Base de <i>Benchmarking</i>	111

Figura I.3 – Atores e Casos de Uso do processo de Gerência da Base de <i>Benchmarking</i>	112
Figura I.4 – Processo de Realização de <i>Benchmarking</i>	113
Figura I.5 – Atores e Casos de Uso do processo de Realização de <i>Benchmarking</i>	114
Figura I.6 – Rastreabilidade entre os Requisitos e Casos de Uso.....	115

ÍNDICE DE TABELAS

Tabela 2.1 – Níveis de Capacidade e Maturidade do CMMI	19
Tabela 2.2 – Principais Componentes de Fatores Críticos de Sucesso	31
Tabela 3.1 – Processos de <i>benchmarking</i>	47
Tabela I.1 – Requisitos de Cliente.....	108
Tabela I.2 – Requisitos Funcionais	109
Tabela I.3 – Requisitos Não-Funcionais.....	110
Tabela I.4 – Casos de uso do processo de Gerência da Base de <i>Benchmarking</i>	112
Tabela I.5 – Casos de uso do processo de Realização de <i>Benchmarking</i>	113
Tabela I.6 – Requisitos de Produto do CORE-KM.....	114

CAPÍTULO 1 – INTRODUÇÃO

Este capítulo apresenta o contexto que motiva a elaboração desta dissertação de mestrado. São também detalhados neste capítulo os objetivos que se espera atingir, assim como os trabalhos relacionados a esta dissertação de mestrado.

1.1 Motivação

Com a globalização do mercado e a alta competitividade existente entre as organizações desenvolvedoras de software, há uma motivação constante para melhorar os produtos e serviços fornecidos pelas organizações. É cada vez maior a necessidade de se produzir software de qualidade, utilizando o menor tempo possível e a menor quantidade de recursos, sempre visando melhor satisfazer e fidelizar os clientes. Nesse contexto, a melhoria de processos de software se tornou uma abordagem importante para melhorar a qualidade dos produtos de software (HUMPHREY, 1989).

Algumas organizações se destacam por possuírem algum tipo de programa de melhorias, baseado em um dos modelos de melhoria de processos existentes como, por exemplo, CMMI (*Capability Maturity Model Integration*) (SEI, 2006), MPS.BR (Melhoria do Processo de Software Brasileiro) (SOFTTEX, 2007), ISO/IEC 9000 (2000), ISO/IEC 12207 (2008) ou ISO/IEC 15504 (2003). Estas organizações devem adotar, junto com esses modelos, estratégias de execução de iniciativas de melhoria de processos de software que estejam alinhadas aos seus objetivos de negócio, de forma a garantir a eficiência e a eficácia dessas empreitadas.

A literatura apresenta diversos relatos de empresas apontando dificuldades em alcançar altos níveis de maturidade de seus processos (PAULK *et al.*, 2000; NIAZI *et al.*, 2006). Uma das razões para esse fato é que os programas de melhoria não são tratados como projetos reais, o que dificulta a aplicação de métodos e técnicas já estabelecidas na área de gerência de projetos para estimar custos e prazos, para implementar as melhorias ou mesmo para tratar adequadamente problemas que podem surgir durante a condução das iniciativas de melhoria (BIRK e PFAHL, 2002). Essa suposição é confirmada em vários relatos que apontam o fracasso das iniciativas de implementação de melhorias de processos

de software em atingir os resultados esperados (BADD00 e HALL, 2002; NIAZI *et al.*, 2005b).

Segundo ZAHARAN (1998), diferentes organizações adotam diferentes estratégias para implementação de melhorias em seus processos, ou seja, não há uma padronização na aplicação dessas estratégias. Isso torna difícil a previsão dos resultados da condução dessas iniciativas. Além disso, os modelos em que comumente se baseiam os programas de melhoria apenas indicam “o quê” devem conter os processos, mas não especificam “como” implementar essas atividades, o que contribui ainda mais para a dificuldade ou mesmo o insucesso dessas iniciativas de melhorias (MINGHUI *et al.*, 2004).

Outro grande problema encontrado em programas de melhoria de processos de software é a baixa utilização de uma gama de informações geradas durante essas iniciativas (SAKAMOTO *et al.*, 1998). O conhecimento adquirido durante a condução de iniciativas de melhoria de processos de software pode ser reutilizado para definir uma abordagem eficaz e eficiente, capaz de aumentar as chances de sucesso dessas iniciativas (MONTONI, 2007; MONTONI *et al.*, 2008b). SAKAMOTO (1998) destaca também que um das conseqüências da baixa utilização das técnicas de Gerência de Projetos é a falta de visibilidade do andamento e do desempenho em que se encontram os projetos de melhoria, o que pode levar ao fracasso da iniciativa de melhoria.

A implementação desses programas de melhoria exige profissionais altamente qualificados e com profundo conhecimento especializado. No entanto, esses profissionais não são facilmente encontrados dentro das organizações desenvolvedoras de software, o que faz com que estas recorram aos serviços de organizações de consultoria em melhoria de processos de software, as quais possuem os profissionais com as qualificações necessárias.

Baseando-se na suposição de que o fracasso das iniciativas de melhoria de processos de software se deve à falta de uma gerência adequada desses projetos, há, então, uma necessidade de apoiar essas organizações de consultoria em melhoria de processos de software na definição, planejamento e execução de estratégias adequadas de implementação de melhoria de processos. Isto pode ser realizado empregando conhecimento de melhores práticas e fatores de sucesso de iniciativas anteriores ou concorrentes no contexto das organizações que implementam a melhoria, de forma a garantir que os objetivos destas iniciativas sejam atingidos de forma eficaz e eficiente.

Para isso, é necessário que as organizações assumam uma postura de “organizações que desejam aprender”, de forma que se possa justificar o esforço investido no processo, já

que a busca pelas melhores práticas é um trabalho intensivo e que requer recursos diversos, além de um alto nível de disciplina.

Neste contexto, é importante analisar formas de se identificar, capturar, processar e disseminar as melhores práticas nas implementações de melhoria de processos de software, com o intuito de auxiliar na definição de novas estratégias de implementação e na tomada de decisão relativa aos projetos de implementação de melhorias já existentes.

O *benchmarking* é um método comumente utilizado em várias áreas da indústria para acompanhar os processos de organizações, concorrentes ou não, comparando processos de trabalho e melhores práticas com o objetivo de alcançar o melhor do melhor (CAMP, 1993). Trata-se de um método que faz uso de comparações para melhor avaliar o desempenho atual dos projetos e identificar possíveis ações para o presente e para o futuro.

O *benchmarking* deve ser considerado um processo de pesquisa, contínuo e sistemático, para avaliar produtos, serviços e métodos de trabalho, com o objetivo de atingir a melhoria organizacional. Ou seja, as atividades de *benchmarking* devem ser realizadas com frequência suficiente que possibilite a avaliação de desempenho dos projetos de implementação de melhorias de processos correntes, em relação a outros projetos similares que se tornaram marcos de referência.

A aplicação do método de *benchmarking* juntamente com o conceito de Fábrica de Experiências (BASILI *et al.*, 1994) apóia o acúmulo de conhecimento sobre as melhores práticas. Este acúmulo de conhecimento serve de mecanismo para aumentar as chances de sucesso de programas de melhoria, pois possibilita que estratégias de implementação de melhoria de processos de software sejam definidas e adaptadas considerando os fatores de sucesso de implementações anteriores ou concorrentes (NIAZI *et al.*, 2005b).

A medição é o instrumento básico para monitorar, controlar e melhorar um processo de software e seus produtos relacionados (BOFFOLI, 2006). Segundo JURAN (1994), em termos operacionais, um *benchmark* é um ponto de referência através do qual medições e comparações podem ser realizadas. Portanto, é necessário o estabelecimento de um programa de medição efetivo que permita a execução das atividades de *benchmarking*. ESTOLANO (2005) define uma série de motivações para medir software. Estas motivações podem ser adaptadas para o contexto de projetos de melhoria de processos de software da seguinte forma:

- Compreender e caracterizar os projetos de melhoria de processos de software para definir o estado ou desempenho atual do processo;

- Analisar os pontos fortes e pontos fracos dos projetos de melhoria de processos de software;
- Monitorar as tendências e a evolução do desempenho e/ou estado dos projetos de melhoria de processos de software;
- Predizer atributos externos relevantes a partir da identificação de relacionamentos entre características dos projetos de melhoria de processos de software;
- Controlar a execução dos projetos de melhoria de processos de software, minimizando seus riscos, a partir da identificação de relações causais entre características desses projetos;
- Melhorar a qualidade e a produtividade dos projetos de melhoria de processos de software futuros, com base na identificação de relacionamentos causais que influenciem o estado ou desempenho desses projetos.

As organizações necessitam, portanto, de apoio para: (i) planejar, executar e monitorar as atividades do projeto de melhoria de processos de software; (ii) armazenar as informações provenientes dos projetos de melhoria de processos de software; (iii) compartilhar o conhecimento armazenado entre os membros da equipe de implementação do projeto de melhoria de processos de software; (iv) reutilizar o conhecimento armazenado em novos projetos de melhoria de processos de software; (v) visualizar o progresso efetivo dos projetos de melhoria; e (vi) comparar o progresso efetivo dos projetos de melhoria de processos de software em relação a projetos passados da mesma natureza.

Existem outros dois trabalhos, um de mestrado e outro de doutorado, relacionados a esta dissertação. MONTONI (2007) desenvolve uma estratégia para apoiar organizações de consultoria em melhoria de processos de software na condução de iniciativas de melhoria. Inicialmente, foi realizado um estudo experimental para a identificação dos fatores que influenciam positiva ou negativamente as iniciativas de melhorias em processos de software e, a partir deste estudo foi proposta uma infra-estrutura de apoio à execução destas iniciativas. Essa infra-estrutura atende a todas as necessidades citadas acima. CERDEIRAL (2008) desenvolveu em seu trabalho de mestrado uma abordagem para o planejamento e a monitoração de iniciativas de melhoria de processos de software, o que compreende as necessidades (i) e (ii) acima citadas.

1.2 Objetivos

Este trabalho tem como foco principal atender diretamente ou indiretamente às necessidades (iv), (v) e (vi) acima citadas. Sendo assim, tem como objetivos:

1. Apoiar uma organização de consultoria em melhoria de processos de software na tomada de decisão durante a definição, planejamento e monitoração dos projetos de melhoria de processos de software, através da utilização de dados sobre projetos passados e/ou correntes;
2. Definir uma abordagem de apoio à execução de atividades de *benchmarking* em iniciativas de implementação de melhoria de processos de software em execução e/ou já executados por esta organização, e;
3. Desenvolver um apoio ferramental para a execução das atividades relativas à abordagem proposta.

A partir da identificação dos componentes da infra-estrutura de apoio à condução de iniciativas de melhorias em processos de software e da identificação dos componentes que são tratados por este trabalho, foi realizada uma revisão da literatura sobre o tema do benchmarking. Esta revisão foi baseada na busca de artigos e livros sobre o tema em questão. O estudo deste material derivou a abordagem para execução do benchmarking no contexto de projetos de implementação de melhorias em processos de software, composta pelos processos de apoio às atividades de benchmarking e pelo ferramental de software para apoiar a execução destas atividades.

1.3 Estrutura da Dissertação

Esta dissertação está organizada em seis capítulos. O presente capítulo apresentou a motivação para o desenvolvimento deste trabalho, os objetivos e a organização da dissertação.

O segundo capítulo, Melhoria Contínua de Processos de Software, apresenta os principais conceitos relacionados à melhoria de processos, os paradigmas comumente utilizados e que serviram de base para a definição da abordagem proposta neste trabalho e, também, o estudo inicial que resultou na abordagem completa envolvendo os três trabalhos de pesquisa.

O terceiro capítulo, *Benchmarking*, apresenta os principais conceitos da técnica de *benchmarking*.

O quarto capítulo, Uma Proposta para *Benchmarking* de Iniciativas de Melhorias em Processos de Software, apresenta a abordagem proposta com a aplicação dos conceitos de *benchmarking* no contexto das iniciativas de melhorias em processos de software.

O quinto capítulo, Uma Ferramenta de Apoio às Atividades de *Benchmarking* de Iniciativas de Melhorias em Processos de Software, apresenta os requisitos e o apoio ferramental desenvolvido para auxiliar a abordagem proposta. Neste capítulo, também são abordados os aspectos principais do CORE-KM, ambiente de gerência de conhecimento onde foi integrado o apoio ferramental proposto.

O sexto capítulo, Conclusão, apresenta a conclusão e as considerações finais desta dissertação, além de suas contribuições e perspectivas para a continuidade do trabalho.

CAPÍTULO 2 – MELHORIA CONTÍNUA DE PROCESSOS DE SOFTWARE

Este capítulo apresenta os principais conceitos relacionados à melhoria contínua no contexto do setor de software, os principais paradigmas de melhoria utilizados como base para a definição da abordagem proposta, as principais normas e modelos de referência de melhoria de processos de software e o estudo inicial que motivou a realização deste trabalho.

2.1 Introdução

A globalização do mercado e a alta competitividade entre as organizações desenvolvedoras de software tornam a melhoria contínua uma condição necessária para a sobrevivência e a sustentabilidade destas organizações, já que os avanços dos competidores levam a uma corrida contínua, onde o vencedor é aquele que oferece produtos e serviços mais rapidamente com melhor qualidade e menor custo (LÓPEZ-CORTIJO *et al.*, 2007).

É um fato amplamente aceito que a qualidade dos produtos de software é altamente determinada pela qualidade do processo utilizado para desenvolvê-lo e mantê-lo (GOLDENSON e HERBSLEB, 1995; FUGGETTA, 2000; BIRK e PFAHL, 2002; SWEBOK, 2004; KIM *et al.*, 2005). A comunidade de engenharia de software procura meios para melhorar a qualidade dos produtos provenientes do desenvolvimento de software através da melhoria dos processos utilizados pelas organizações desenvolvedoras (BOFFOLI, 2006).

Neste contexto, tornaram-se muito comuns, as iniciativas de melhorias em processos de software. Estas iniciativas são caracterizadas pela definição de um conjunto de ações que uma organização deve executar para mudar seus processos de forma a fazê-los atender aos objetivos de negócio com mais eficiência e eficácia (IBRAHIM e PYSTER, 2004). Existe uma série de normas e modelos de referência para a melhoria de processos de software como, por exemplo, a ISO/IEC 12207 (2008), a ISO/IEC 15504 (2003), o CMMI (SEI, 2006) e o MPS.BR (SOFTEX, 2007).

A melhoria de processos baseia-se fortemente no ciclo PDCA (*Plan-Do-Check-Act*), proposto por Walter Shewart e difundido por um de seus seguidores, William Edwards

Deming. Este ciclo caracteriza-se por iniciar na definição dos objetivos de melhoria e do planejamento das atividades de melhoria (*Plan*). O plano elaborado é, então, executado (*Do*) e os resultados obtidos são verificados (*Check*) e, por fim, ações de refinamento, ajuste e institucionalização são realizadas no processo (*Act*). Este ciclo é então re-executado, caracterizando a melhoria contínua da organização (DEMING, 1982).

Durante as atividades de melhoria, as experiências obtidas em projetos anteriores desempenham um importante papel. Essas experiências acumuladas auxiliam na determinação dos primeiros passos a serem executados numa iniciativa de melhoria, auxiliam na antecipação do impacto e da aceitação por parte dos membros da organização na introdução de novas práticas, auxiliam na manutenção do foco no processo de melhoria e ajudam a antecipar como restrições e estruturas organizacionais irão reagir às mudanças que acompanham as iniciativas de melhoria (KIM *et al.*, 2005).

A implementação de programas de melhoria requer experiências reais, fornecendo um mecanismo fundamental para aprender com os erros e melhorar continuamente os processos de desenvolvimento de software. A execução dessas iniciativas de melhoria permite a identificação de uma série de fatores considerados críticos para o sucesso de um programa de melhoria, sendo que, geralmente, esses fatores estão relacionados às experiências reais proporcionadas por estas iniciativas (NIAZI *et al.*, 2005b). O acúmulo de conhecimento sobre estes fatores críticos de sucesso visa aumentar a probabilidade de sucesso das iniciativas de melhoria (MONTONI *et al.*, 2008b).

Este capítulo trata da revisão da literatura sobre os principais paradigmas da melhoria contínua utilizados para melhoria de processos de software, e o estudo sobre os fatores críticos de sucesso das iniciativas de melhoria, que irão nortear a definição da abordagem proposta por este trabalho. A seção 2.2 apresenta o paradigma do PDCA. A seção 2.3 apresenta os conceitos do QIP (*Quality Improvement Paradigm*). A seção 2.4 apresenta a abordagem de Fábrica de Experiências, a seção 2.5 apresenta normas e modelos de referência para melhoria de processos de software, a seção 2.6 apresenta um estudo inicial realizado para a identificação de fatores críticos de sucesso em programas de melhorias em processos de software. E, por fim, a seção 2.7 apresenta as considerações finais deste capítulo.

2.2 O Ciclo PDCA

O ciclo PDCA (*Plan-Do-Check-Act*) foi proposto inicialmente por Walter Shewart, na década de 20. Porém, sua disseminação ocorreu por responsabilidade de um de seus discípulos, Edwards Deming, o qual difundiu a filosofia proposta na indústria japonesa logo após o fim da segunda guerra (DEMING, 1982).

Parte integrante do movimento conhecido como TQM (*Total Quality Management*), o paradigma do PDCA visa auxiliar na tomada de decisões relativas aos objetivos estratégicos estabelecidos com o intuito de garantir o alcance das metas necessárias à sobrevivência de uma organização (DEMING, 1982).

O ciclo subdivide-se em 4 fases, conforme pode ser visto na Figura 2.1 (DEMING, 1982; JARVINEN *et al.*, 1998; SPENGLER *et al.*, 1999).

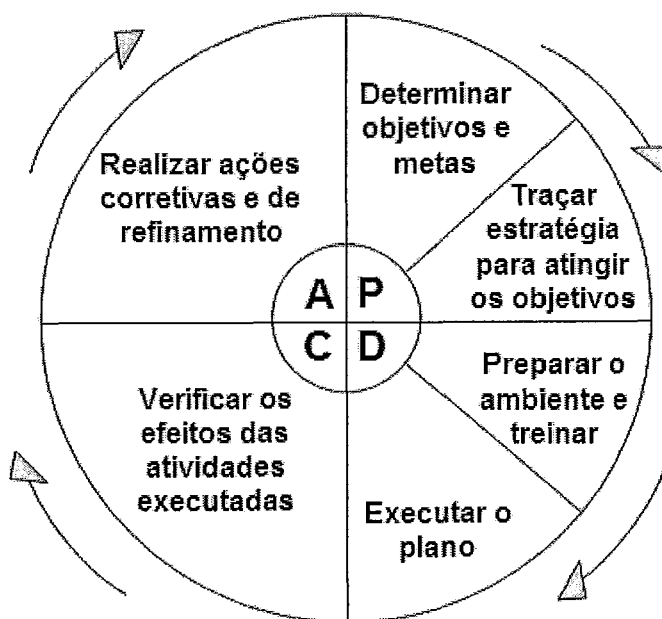


Figura 2.1 – Ciclo do PDCA (DEMING, 1982)

1. Planejar (*Plan*) – É essencial que o problema a ser solucionado esteja muito bem entendido e analisado. Isto pode fazer com que esta fase seja a mais extensa do ciclo. Devendo-se, dentre outros, identificar o(s) processo(s) a ser(em) melhorado(s), definir a equipe de implementação de melhorias, os recursos necessários e os objetivos da iniciativa de melhoria. Após a identificação da oportunidade de melhoria, esta deve ser analisada em detalhes para que as questões relevantes ao problema sejam identificadas. As causas do problema devem ser identificadas e soluções devem ser propostas.

2. Realizar (*Do*) – Uma vez que soluções para o problema foram selecionadas, realizam-se os treinamentos da equipe de desenvolvimento e o plano de melhoria estabelecido é executado.
3. Verificar (*Check*) – Nesta etapa são verificados os efeitos das mudanças, ou seja, verifica-se se elas estão obtendo os resultados esperados e se os objetivos definidos estão sendo atingidos. Caso isto não esteja ocorrendo, deve-se replanejar o projeto de melhoria. Esta fase pode ser auxiliada por atividades de medição do processo de melhoria, ou seja, deve-se comparar as medições extraídas deste processo antes e após o ciclo de melhoria, com o intuito de determinar se as melhorias implementadas estão sendo efetivas.
4. Agir (*Act*) – Nesta etapa, são tomadas ações baseadas no que foi encontrado na etapa de verificação. Se as melhorias realizadas surtiram efeito positivo, ações de institucionalização devem ser tomadas para tornar estas melhorias parte do processo. Caso as melhorias não tenham se mostrado efetivas, deve-se analisar as causas do insucesso e procurar novas soluções para o problema identificado. É possível que novas oportunidades de melhoria tenham sido identificadas durante o ciclo, o que também remete a novos ciclos de melhoria para implementação destas.

Portanto, o PDCA deve ser visto como uma roda giratória. Cada volta significa um ciclo de melhoria, que culmina em um novo ciclo, e assim sucessivamente. Todas as informações relativas às análises, tomadas de decisão, medidas coletadas, melhorias e achados em geral, auxiliarão as organizações nos ciclos de melhoria futuros, de forma que erros semelhantes não serão cometidos, lições aprendidas facilitem a implementação das ações de melhoria e os pontos passíveis de melhoria sejam identificados mais facilmente (DEMING, 1982).

2.3 QIP

O QIP é um paradigma de melhoria da qualidade desenvolvido por BASILI *et al* (1994), resultado da aplicação de método científico ao problema de melhoria da qualidade do software. Trata-se de um paradigma baseado no ciclo PDCA de SHEWART-DEMING (DEMING, 1982), utilizado para a implementação de projetos de melhoria da qualidade.

O QIP consiste das seis fases descritas a seguir:

1. Caracterização: O objetivo desta etapa é entender o ambiente da organização, tendo como base informações em forma de modelos, dados, intuição ou quaisquer outros, que estejam disponíveis na organização. A partir desses dados, estabelecem-se linhas de base dos processos de negócio existentes, caracterizando-os criticamente.
2. Definição de Objetivos: Tendo como base a caracterização inicial e os objetivos estratégicos da organização, são definidos objetivos quantificáveis de sucesso para o desempenho da organização como um todo e também para a melhoria dos seus produtos. Estes objetivos são definidos em relação à *baseline* produzida na etapa inicial de caracterização.
3. Escolha do Processo: Baseando-se na caracterização do ambiente e nos objetivos estratégicos definidos, é realizada a escolha e a priorização dos processos passíveis de melhoria juntamente com as ferramentas e métodos que darão apoio às iniciativas de melhoria.
4. Execução: Nesta etapa, os processos de construção dos produtos são executados e estes são realimentados com os dados coletados que indicam o grau de atendimento aos objetivos estratégicos definidos.
5. Análise: No final de cada projeto, os dados coletados são analisados e informações são reunidas com o intuito de avaliar as práticas atuais, determinar problemas, registrar achados e fazer recomendações para futuras iniciativas de melhoria.
6. Empacotamento: Todas as experiências adquiridas durante este e outros projetos anteriores são empacotadas em forma de novos modelos ou de modelos atualizados e refinados, e, também, em outras formas de conhecimento estruturado. Esses pacotes de informação são, então, armazenados em uma base de experiências acessível aos projetos para futura reutilização.

O QIP se baseia na noção de que a melhoria dos processos e dos produtos de software requer uma contínua acumulação de experiências avaliadas (aprendizado) em um formato que possa ser facilmente entendido e modificado (modelos baseados em experiências), e que esteja armazenado em um repositório de modelos integrados (base de experiências), que podem ser acessados e modificados de acordo com as necessidades específicas de cada projeto (BASILI *et al.*, 1994).

O QIP é implementado em dois ciclos distintos: o ciclo do projeto e o ciclo organizacional, como podem ser vistos na Figura 2.2.

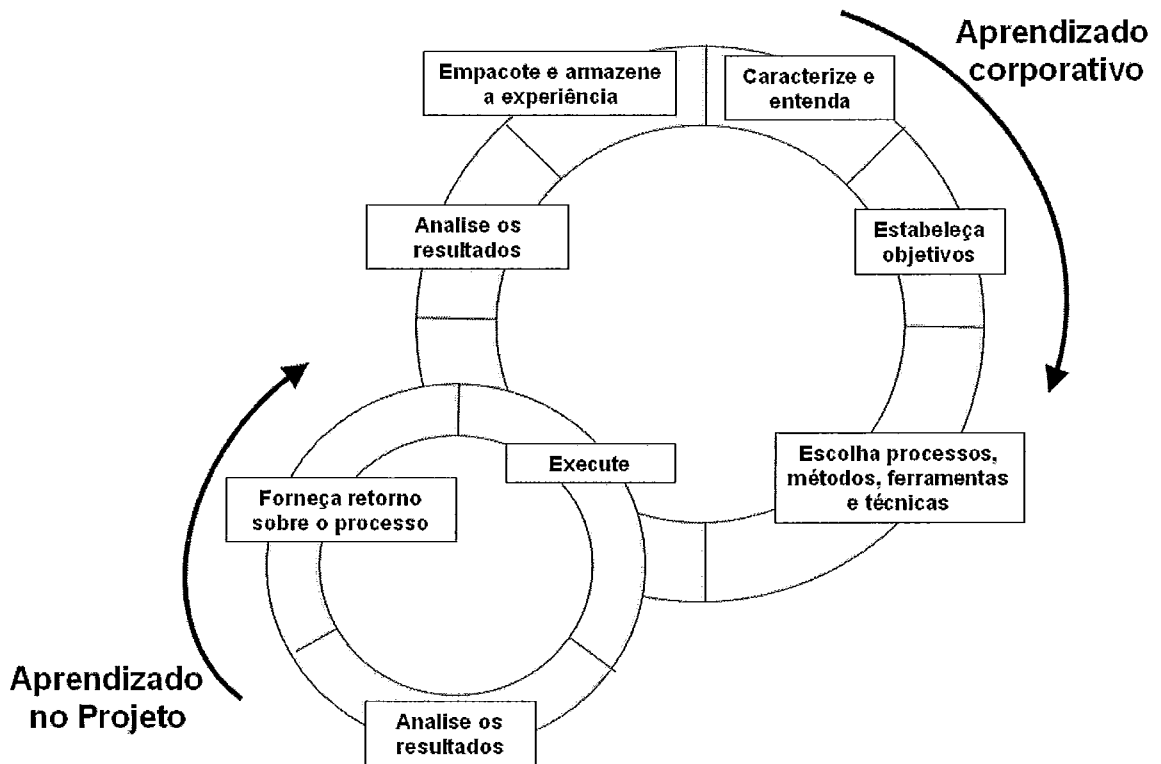


Figura 2.2 – Ciclos do QIP (MCGARRY e DECKER, 2002)

No âmbito dos projetos, a realimentação é feita durante a fase de execução. Indicadores quantitativos no nível das atividades dos projetos podem ser úteis na prevenção e solução de problemas.

Já no âmbito organizacional, a realimentação provê informações sobre o desempenho geral dos projetos, comparando seus dados com os dados provenientes de outros projetos da organização, analisando os pontos fortes e pontos fracos do projeto. A realimentação no âmbito organizacional também visa ao acúmulo de experiências em forma de artefatos que possam ser reutilizados em outros projetos. Estes artefatos são, geralmente, melhorados durante a atividade de análise, de forma que melhores práticas sejam estabelecidas e disponibilizadas para projetos futuros.

Existe uma série de características de contexto que afetam o desenvolvimento de software. Dentre elas estão fatores pessoais, como por exemplo, o número de pessoas, o nível de experiência ou a organização das equipes; fatores relacionados ao problema, como o domínio da aplicação, estado da arte, suscetibilidade a mudanças, etc.; fatores relacionados aos processos, como o modelo de ciclo de vida, os métodos, técnicas e ferramentas de apoio utilizadas ou as linguagens de programação selecionadas; fatores relacionados ao produto, como por exemplo, os entregáveis do projeto, tamanho do sistema, critérios relacionados à qualidade como portabilidade, confiabilidade, segurança; e

fatores relacionados aos recursos disponíveis, como os equipamentos, ambiente de desenvolvimento e ambiente de produção, cronograma, recursos humanos e financeiros ou outros sistemas existentes.

A correta e criteriosa caracterização do ambiente é um pré-requisito para a efetiva aplicação do paradigma QIP. Esta classificação requer que caracterizemos os projetos correntes em relação a uma variedade de características, o que permitirá classificar projetos com características e objetivos semelhantes, juntamente com os projetos em execução. Isso deverá facilitar a reutilização das experiências e artefatos produzidos em projetos anteriores e também a comparação de desempenho entre os projetos e a realização de melhores projeções para os projetos em andamento (BASILI *et al.*, 1994).

2.4 Fábrica de Experiências

O conceito de Fábrica de Experiências (BASILI *et al.*, 1994) foi introduzido para auxiliar na institucionalização do aprendizado coletivo da organização. Esta institucionalização é uma condição necessária para a melhoria contínua da organização e, conseqüentemente, para a conquista de vantagens competitivas em relação ao mercado ao qual pertence.

Semelhante ao QIP, a abordagem de Fábrica de Experiências é guiada pelo conceito de melhoria contínua de uma organização assim como em seus objetivos estratégicos definidos, em suas características específicas, e também em seu crescente conjunto de experiências adquiridas em projetos atuais e anteriores. A Fábrica de Experiências provê mecanismos que auxiliam a organização na contínua captura de informações valiosas para se atingir a melhoria e no empacotamento dessas informações para utilização em projetos futuros. O objetivo principal é a melhoria mensurável e corrente dos processos e produtos de software da organização (BASILI e MCGARRY, 1997).

Existe um conjunto de conhecimento importante dentro das organizações que não está documentado explicitamente, mas está representado de forma tácita nas mentes dos colaboradores. Isso dificulta a disseminação de conhecimento em forma de lições aprendidas, melhores práticas, idéias novas, etc., aos outros membros da organização que possam se beneficiar desse conhecimento. Sendo assim, a reutilização de experiências e o aprendizado coletivo não podem ser deixados a cargo da imaginação dos indivíduos mais talentosos que detém esses conhecimentos, pois estes acabam por se tornar uma

preocupação organizacional, assim como um portfólio ou um ativo organizacional. A saída destes profissionais pode se tornar um transtorno para a organização, já que os novos membros necessitarão de treinamento nos processos e nas técnicas utilizadas. Esses treinamentos podem ser custosos e demorados, fazendo com que o desempenho da organização seja afetado. A utilização da abordagem de Fábrica de Experiências também visa a reter esse conhecimento, de forma que este possa se tornar de fato um ativo reutilizável da organização como um todo.

A abordagem de Fábrica de Experiências foi, inicialmente, desenvolvida para organizações desenvolvedoras de software, levando em consideração as características experimentais, evolucionárias e não-repetitivas desta disciplina. Contudo, foi observado que versões adaptadas da Fábrica de Experiências podem ser benéficas na criação de organizações que desejam aprender (*learning organizations*), mesmo que a principal atividade destas organizações não seja o software. Isto se deve ao fato de que as organizações não gerenciam bem suas experiências (BASILI *et al.*, 2001b). A chave para aprender com sucesso é uma contínua e simultânea interação entre o conhecimento que a organização estabelece com o tempo e os conhecimentos dos colaboradores da organização nos seus respectivos contextos (DYBÅ, 2002).

Numa efetiva implementação desta abordagem, subdivide-se a organização em duas partes: a Organização dos Projetos, que é responsável pela implementação dos projetos específicos, e a Fábrica de Experiências. A Fábrica de Experiências é a organização que provê apoio à reutilização de experiências e ao aprendizado coletivo através do desenvolvimento, atualização e distribuição destas experiências por requisição das organizações dos projetos. Estas por sua vez, oferecem à Fábrica de Experiências seus produtos, planos e dados reunidos durante a execução dos projetos (BASILI *et al.*, 1992). A Figura 2.3 mostra a infra-estrutura do paradigma de Fábrica de Experiências.

A Fábrica de Experiências pode ser física ou lógica, mas é importante que estas atividades sejam separadas e independentes daquelas executadas no âmbito dos projetos (BASILI *et al.*, 1992).

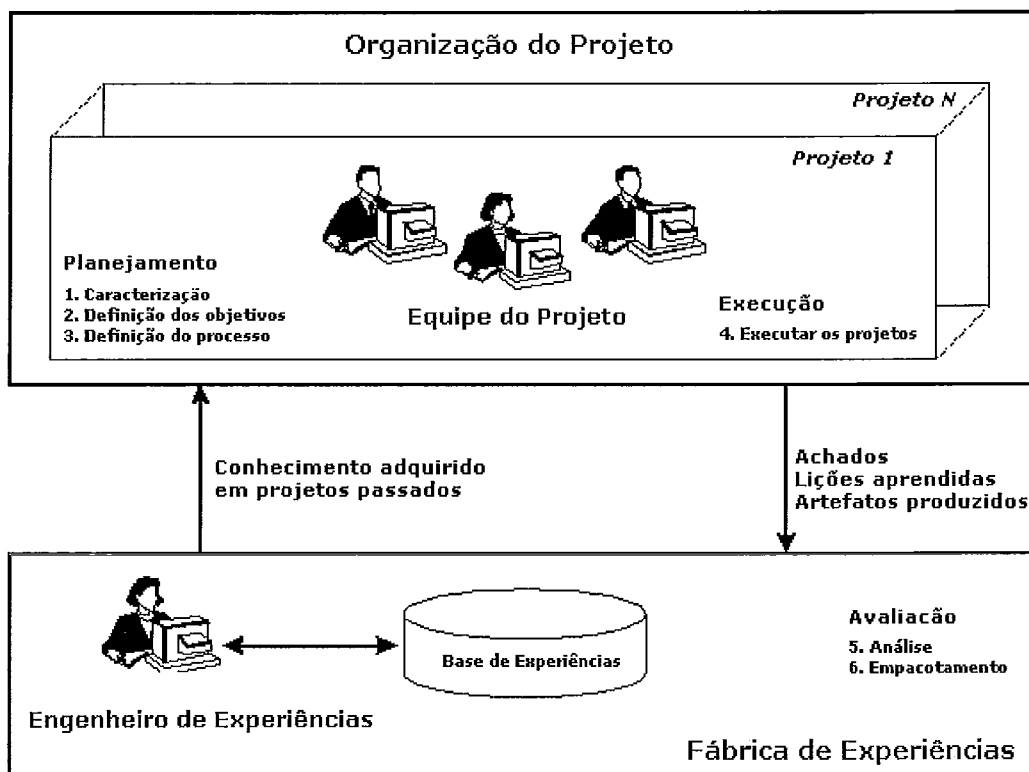


Figura 2.3 – Infra-estrutura da abordagem de Fábrica de Experiências (ALTHOFF *et al.*, 1998)

Para uma correta implementação do paradigma da Fábrica de Experiências, os projetos e a fábrica terão diferentes processos de negócio. Cada projeto deve escolher seu modelo de processo baseado nas características dos produtos a serem entregues, enquanto a Fábrica de Experiências será responsável por definir e modificar os modelos de processos baseando-se no desempenho e nas questões organizacionais. O produto principal da Fábrica de Experiências são as próprias experiências empacotadas (BASILI *et al.*, 1992).

Um problema comum que ocorre em bases de experiências se deve ao fato de que pessoas dificilmente colaboram com conhecimento para estas bases. Também é necessário que as informações contidas na base de experiências tenham qualidade, relevância e atualidade, de forma a garantir a sua efetiva utilização. Se os usuários não conseguem encontrar o que procuram ou o que encontram não é útil, eles perderão o interesse pela base. Sendo assim, é necessário analisar constantemente os dados contidos na base e remover informações desatualizadas ou inúteis, além de manter um controle das informações que são inseridas na base impedindo que dados não relevantes sejam registrados e acabem por ocultar aqueles que realmente interessam (BASILI *et al.*, 2001b).

A abordagem de Fábrica de Experiências e o QIP foram inicialmente desenvolvidos e vêm sendo aplicados com sucesso nos ambientes de desenvolvimento do SEL (*Software*

Engineering Laboratory), que compreendem o NASA/GSFC *Flight Dynamics Division*, o CSC (*Computer Science Corporation*) e a Universidade de Maryland (*Department of Computer Science*) (BASILI e MCGARRY, 1997; BASILI *et al.*, 2001a). Esses conceitos foram aplicados em mais de 100 projetos. KOENNECKER (2000) descreve o desenvolvimento de uma fábrica de experiências em uma organização australiana do campo das telecomunicações. SHNEIDER (2002) descreve a aplicação dos conceitos de Fábrica de Experiências na Daimler-Chrysler com o intuito de melhorar os processos de desenvolvimento e a aquisição de software para as unidades eletrônicas de controle de seus veículos, responsáveis por funções como assistentes inteligentes de freios, controle eletrônico de estabilidade e controles de motorização diversos. No Brasil, há relatos de utilização da abordagem na área de desenvolvimento de sistemas do Tribunal de Contas do Distrito Federal (BRASIL *et al.*).

2.5 Normas e Modelos de Referência

A melhoria de processos de software trata de um conjunto de atividades que uma organização deve empregar, com o intuito de modificar seus processos visando a atender seus objetivos de negócio de forma mais eficaz (IBRAHIM e PYSTER, 2004). Este tema é amplamente tratado por várias normas e modelos de referência, e esta seção objetiva descrevê-los de forma sucinta. Na comunidade de engenharia de software, essas normas e modelos são geralmente tomados como base para a definição das estratégias de implementação de iniciativas de melhorias em processos de software. Um problema comumente apontado diz que os modelos de referência são úteis para informar “o quê” deve ser feito numa implementação de melhorias em processos de software, mas raramente informam “como” implementar essas melhorias (MINGHUI *et al.*, 2004). Os modelos e as normas de melhoria de processos de software oferecem poucas instruções sobre a tomada de decisão relativa a ações de melhoria concretas. Contudo, essas normas e modelos são úteis para enfatizar o papel dos métodos de engenharia de software dentro de um projeto, um aspecto que métodos convencionais de gerência de projetos negligenciam. Ao invés de oferecer soluções concretas, os modelos e normas guiam as organizações na identificação de potenciais melhorias em geral (BIRK e PFAHL, 2002). Portanto, esses modelos desempenham um papel importante neste contexto e devem ser considerados. As seções a seguir descreverão algumas destas normas e modelos.

2.5.1 A Norma Internacional ISO/IEC 12207

A Norma Internacional ISO/IEC 12207 – Engenharia de Sistemas e de Software – Processos de Ciclo de Vida de Software (ISO/IEC, 2008), descreve um conjunto de ciclos de vida de processos de software. Estes processos são compostos de atividades e tarefas cujo intuito é apoiar a aquisição, o fornecimento, o desenvolvimento, a operação, a manutenção e a descontinuação de produtos de software. Esta norma foi desenvolvida visando fornecer o conhecimento necessário às organizações desenvolvedoras de software para que estas forneçam de forma eficiente e eficaz, serviços que de alguma forma envolvam software. No entanto, assim como muitos dos modelos de referência existentes na literatura, a norma não especifica detalhes de como as práticas nela contidas devem ser implementadas, deixando esta preocupação a cargo das organizações de software, sendo que estas devem, então, adaptar os processos, atividades e tarefas descritos, segundo as características específicas da organização.

A Norma teve sua primeira versão publicada em 1995 (ISO/IEC, 1995), e, por conta da evolução da engenharia de software, das necessidades identificadas por seus usuários e da necessidade de adequação à Norma ISO/IEC 15504 – Tecnologia da Informação – Avaliação de Processos (ISO/IEC, 2003), foram elaboradas as Emendas 1 (ISO/IEC, 2002) e 2 (ISO/IEC, 2004). Em 2008, uma nova versão consolidada da norma foi publicada.

As atividades e tarefas definidas na norma estão subdivididas em sete grupos de processo, sendo que cada processo contido nestes grupos é definido em termos de objetivo, resultados esperados e uma lista de atividades e tarefas necessárias para atingir estes objetivos.

Os grupos são os seguintes (ISO/IEC, 2008):

- *Processos de Estabelecimento de Acordos*: Estes processos apóiam o estabelecimento de acordos entre organizações. Estes acordos podem ser estabelecidos para a aquisição ou fornecimento de serviços de software.
- *Processos Organizacionais*: Estes processos são responsáveis por capacitar a organização no fornecimento e aquisição dos produtos ou serviços. São os processos responsáveis por controlar e fornecer os recursos e a infra-estrutura necessários para apoiar os projetos e assegurar a satisfação dos objetivos organizacionais e acordos estabelecidos.

- *Processos de Projeto:* Estes processos subdividem-se em processos de gerência de projeto e processos de suporte de projetos, sendo que os processos de gerência de projetos apóiam o estabelecimento de planos de projeto, a monitoração do progresso dos projetos e o controle e execução dos projetos fazendo uso dos planos estabelecidos, e o segundo grupo provê um conjunto de atividades de apoio à tomada de decisão, gerência de riscos, gerência de configuração, gerência de conhecimento e medição.
- *Processos Técnicos:* Estes processos apóiam a transformação dos requisitos estabelecidos em produtos de software, caminhando até a descontinuação deste último, mesmo quando não se faz mais necessário.
- *Processos de Implementação do Software:* São utilizados para apoiar a produção de cada elemento de um sistema de software, transformando comportamentos, interfaces e restrições em ações de implementação que resultam em um elemento do sistema.
- *Processos de Apoio:* Estes processos provêm apoio aos processos de implementação do software, no sentido de garantir o sucesso e qualidade do projeto. Envolve atividades de documentação, gerência de configuração, garantia da qualidade, verificação, validação e outros.
- *Processos de Reutilização:* Estes processos provêm suporte à organização na reutilização de elementos de software, ultrapassando os limites de um projeto, ou seja, fazendo com que esses elementos sejam reutilizados em diferentes projetos.

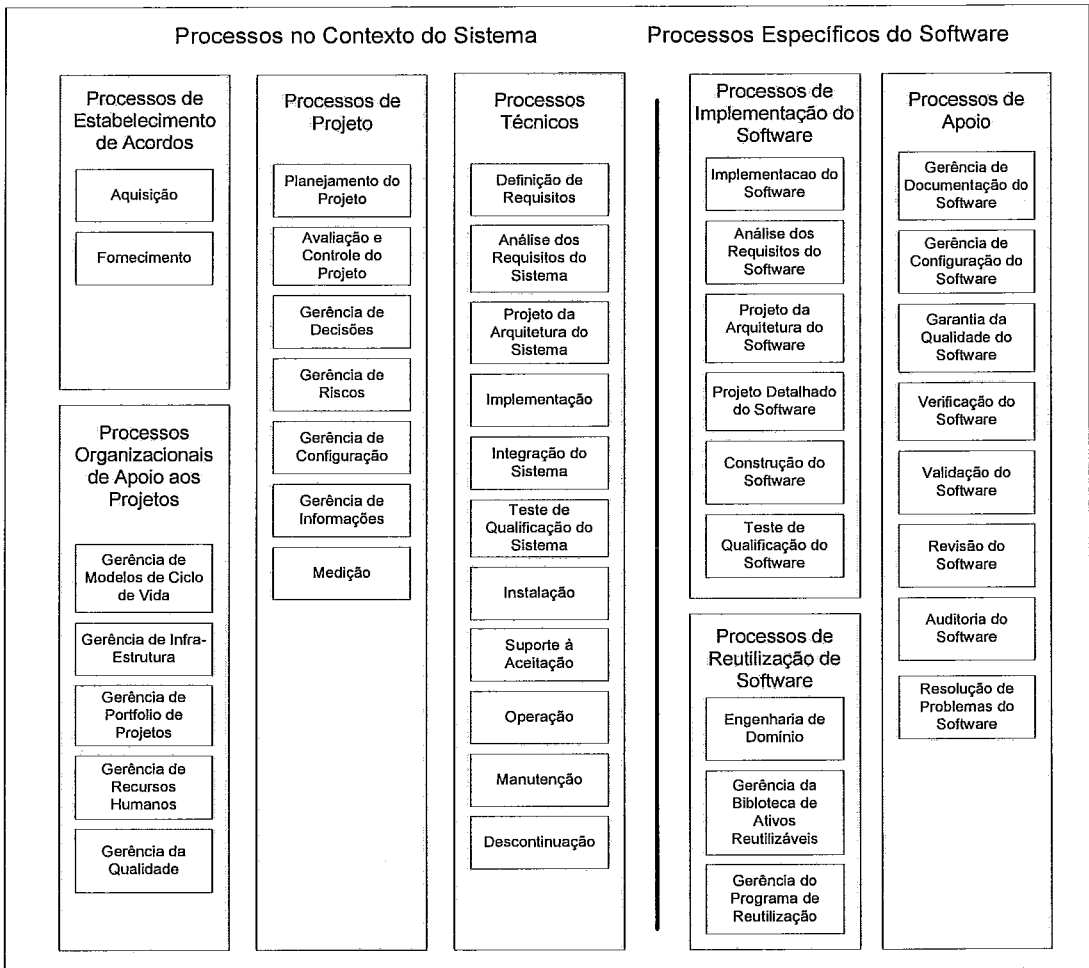


Figura 2.4 – Estrutura de processos da norma ISO/IEC 12207 (2008)

2.5.2 A Norma Internacional ISO/IEC 15504

A norma internacional ISO/IEC 15504 – Tecnologia da Informação – Avaliação de Processos (ISO/IEC, 2003) estabelece os princípios, os requisitos e as metodologias a serem aplicados na condução de avaliações de processos de organizações. Este modelo de avaliação se baseia na Norma Internacional ISO/IEC 12207 – Tecnologia da Informação – Processos de Ciclo de Vida de Software (ISO/IEC, 1995), a qual define os processos de ciclo de vida de software que devem ser avaliados segundo o modelo.

Segundo a norma, as avaliações de processo podem ser o objetivo de dois contextos: (i) Determinação da capacidade dos processos (ii) Melhoria contínua dos processos. O objetivo da determinação da capacidade dos processos é a identificação dos pontos fortes, oportunidades de melhoria e riscos relacionados com os processos que estão sendo avaliados. Já as avaliações executadas no contexto de iniciativas de melhorias em

processos visam à melhoria contínua da eficiência e eficácia da organização (ISO/IEC, 2003).

A norma ISO/IEC 15504 (2003) subdivide-se em cinco partes distintas, sendo que a primeira apresenta os conceitos relacionados à avaliação de processos, a segunda é a única parte normativa e apresenta os requisitos mínimos de uma avaliação, a terceira fornece um guia para a realização das avaliações, a quarta define as atividades a serem realizadas para melhoria de processos ou para determinação da capacidade dos processos, e, por fim, a quinta parte apresenta um exemplo de modelo de processo de avaliação, baseado em um modelo de referência de processo definido pela ISO/IEC 12207 (ISO/IEC, 1995).

As avaliações, segundo a norma ISO/IEC 15504 (2003), são realizadas levando-se em consideração os propósitos e as práticas de cada um dos processos descritos na norma ISO/IEC 12207 (ISO/IEC, 1995), sendo que estes processos podem atingir níveis de capacidade, que são definidos em termos de atributos mensuráveis de cada processo.

2.5.3 As Normas da Família ISO/IEC 9000: 2000

As normas da família ISO/IEC 9000: 2000 (ISO/IEC, 2000) foram desenvolvidas com o intuito de apoiar organizações de todos os tipos e tamanhos, no estabelecimento e manutenção de sistemas de gestão da qualidade. Estes sistemas de qualidade têm por objetivo melhorar continuamente o desempenho das organizações.

A família ISO/IEC 9000:2000 (ISO/IEC, 2000) é composta de:

- ISO/IEC 9000, que descreve os fundamentos de sistemas de gestão da qualidade e a terminologia para estes sistemas.
- ISO/IEC 9001, que descreve os requisitos necessários para sistemas de gestão da qualidade.
- ISO/IEC 9004, que fornece diretrizes para melhorar o desempenho das organizações e a satisfação de seus clientes.
- ISO/IEC 19001, que fornece as diretrizes para auditoria de sistemas da qualidade.

A ISO/IEC 9000:2000 (ISO/IEC, 2000) considera oito princípios da qualidade que devem ser utilizados na implementação de seus sistemas de gestão da qualidade:

1. *Foco no cliente*: as organizações dependem de seus clientes e, portanto, devem sempre atender às suas necessidades e expectativas.

2. *Liderança*: os líderes das organizações devem criar e manter um ambiente interno no qual as pessoas possam estar totalmente envolvidas com o propósito de atingir os objetivos da organização.
3. *Envolvimento de pessoas*: colaboradores de todos os níveis devem estar envolvidos com os objetivos da organização.
4. *Abordagem de processo*: um resultado desejado é alcançado mais eficientemente quando as atividades e os recursos relacionados são gerenciados como um processo.
5. *Abordagem sistêmica para gestão*: identificar, entender e gerenciar os processos inter-relacionados como um sistema contribui para a eficácia e eficiência da organização no sentido de atingir seus objetivos estratégicos.
6. *Melhoria contínua*: deve ser o objetivo permanente das organizações que desejam se manter competitivas. A melhoria contínua significa menores custos de produção, maiores lucros, prazos mais curtos de entrega, satisfação do cliente, e outros.
7. *Abordagem factual para tomada de decisão*: decisões eficazes são tomadas tendo como base a análise de dados e informações factuais.
8. *Benefícios mútuos nas relações com os fornecedores*: uma organização e seus fornecedores são interdependentes, e, portanto, uma relação de benefícios mútuos aumenta a capacidade de ambos em agregar valor.

A Figura 2.5 ilustra o modelo de um sistema de gestão da qualidade, baseado em processos, descrito nas normas da família ISO/IEC 9000:2000 (ISO/IEC, 2000). As partes interessadas fornecem insumos para a organização e sua satisfação é monitorada através da avaliação do grau de atendimento das suas necessidades e expectativas.

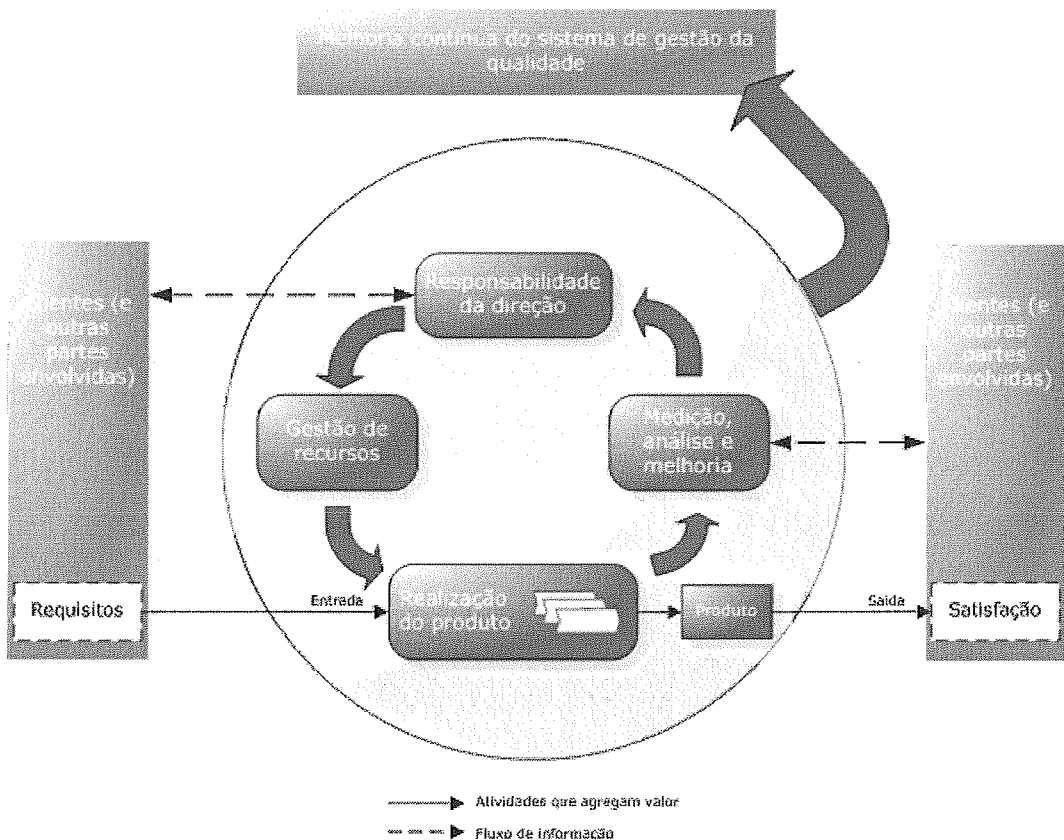


Figura 2.5 – Processo baseado no sistema de gestão da qualidade (ISO/IEC, 2000)

2.5.3 CMMI-DEV

O CMMI-DEV (*Capability Maturity Model Integration for Development*) (SEI, 2006) é um modelo de maturidade de melhoria de processos de software voltado para o desenvolvimento de produtos e serviços. Consiste em um conjunto de melhores práticas relacionadas às atividades de desenvolvimento e manutenção que constituem o ciclo de vida de um produto, desde a concepção até a entrega e manutenção.

O SEI (*Software Engineering Institute*) acolheu a premissa de que “a qualidade de um sistema ou produto é altamente influenciada pela qualidade do processo utilizado para desenvolvê-lo e mantê-lo” para definir uma série de CMMs (*Capability Maturity Models*), que incorporam essa premissa. Estes CMMs tinham como foco a melhoria de processos numa organização, através da implementação de melhores práticas em processos de uma ou mais disciplinas, com o objetivo de melhorar a qualidade e efetividade desses processos. Contudo, a utilização de múltiplos modelos de referência se tornou um problema, pois muitas organizações utilizavam vários destes modelos em diferentes grupos dentro da mesma organização, o que dificultava a disseminação e o sucesso dos esforços de melhoria

(SEI, 2006). O CMMI é uma combinação de três modelos CMM: o SW-CMM (*Capability Maturity Model for Software*) (SW-CMM, 1997), o SECM (*Systems Engineering Capability Model*) e o IPD-CMM (*Integrated Product Development Capability Maturity Model*) (SEI, 1997). A combinação destes três modelos num modelo único e integrado foi feita para atender às organizações que desejam realizar iniciativas de melhoria através de todas as áreas da organização.

O CMMI (SEI, 2006) subdivide seus processos por 22 áreas conhecidas como áreas de processo, sendo que cada área de processo é definida em termos de seu propósito, de objetivos específicos e de objetivos genéricos. Os objetivos específicos relacionam-se especificamente a uma área de processo enquanto os objetivos genéricos estão relacionados a todas as áreas de processos e também à organização, e dizem respeito ao que deve ser atendido para que haja a institucionalização das práticas na organização. Os objetivos dividem-se em um conjunto de práticas específicas e genéricas que, quando implementadas, satisfazem os objetivos definidos para uma determinada área de processo.

O CMMI (SEI, 2006) apóia duas formas de implementação de melhorias, sendo que na primeira forma, a iniciativa de melhoria executadas através de uma ou mais áreas de processo selecionadas pela organização e é chamada de “representação contínua”, e seus processos são classificados em termos de “nível de capacidade”. Na segunda forma de implementação, a organização seleciona um conjunto pré-definido de processos relacionados, e as melhorias são realizadas em todos os processos que constituem este conjunto. Este tipo de representação é conhecido como representação estagiada e os processos são classificados em termos de “nível de maturidade”. Independentemente do tipo de abordagem de melhoria selecionada, o conceito de nível é o mesmo. Para satisfazer um determinado nível, a organização deve atender a todos os objetivos das áreas de processo daquele nível. A Tabela 2.1 apresenta os níveis de capacidade e maturidade referentes às representações contínua e estagiada do CMMI (SEI, 2006).

Tabela 2.1 – Níveis de Capacidade e Maturidade do CMMI (SEI, 2006)

Nível	Nível de Capacidade (representação contínua)	Nível de Maturidade (representação estagiada)
0	Incompleto	Não existe
1	Realizado	Inicial
2	Gerenciado	Gerenciado
3	Definido	Definido
4	Gerenciado Quantitativamente	Gerenciado Quantitativamente
5	Em Otimização	Em Otimização

2.5.4 MR-MPS.BR (Modelo de Referência para Melhoria de Processo do Software Brasileiro)

O MPS.BR (SOFTEX, 2007) é um Programa para Melhoria de Processos do Software Brasileiro coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) e tem por objetivo definir e aprimorar um modelo de melhoria e avaliação de processos de software com foco nas micro, pequenas e médias empresas, que representam uma fatia considerável do mercado brasileiro de software.

O MPS.BR (SOFTEX, 2007) é composto de um modelo de referência para processos de software (MR-MPS), que contém os requisitos que os processos das unidades organizacionais devem atender para estar em conformidade com o MA-MPS, método de avaliação de processos que assegura a coerência na aplicação do MR-MPS e suas definições. O MA-MPS está em conformidade com a norma ISO/IEC 15504-2 (2003). Por último, o MPS.BR também define e é composto por um modelo de negócio que descreve as regras de negócio para a implementação do MR-MPS pelas Instituições Implementadoras (IIs), avaliação seguindo o MA-MPS pelas Instituições Avaliadoras (IAs), organização de grupos de empresas para implementação do MR-MPS e avaliação MA-MPS pelas Instituições Organizadoras de Grupos de Empresas (IOGEs).

A base técnica do modelo é formada pelas normas ISO/IEC 12207 – Processos de Ciclo de Vida de Software e suas Emendas 1 e 2 (1995; 2002; 2004), e pela norma ISO/IEC 15504-2 – Avaliação de Processos (2003). O MPS.BR também teve a preocupação de assegurar a compatibilidade com o CMMI (SEI, 2006).

O modelo está dividido em sete níveis de maturidade: A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Gerenciado Parcialmente). A escala de maturidade inicia no nível G e progride até o nível A. Estes níveis de maturidade estabelecem patamares de evolução dos processos, caracterizando os estágios de melhoria da implementação de processos em uma organização.

Os processos no MR-MPS são definidos em termos de propósito e resultados esperados, sendo que o propósito descreve o objetivo a ser atingido com a execução do processo e os resultados esperados estabelecem os resultados a serem obtidos com a efetiva implementação do processo. A capacidade do processo é representada por um conjunto de atributos descritos em termos dos resultados esperados e expressa o grau de institucionalização dos processos na organização. No MR-MPS, à medida que a

organização evolui nos níveis de maturidade, um maior nível de capacidade para desempenhar o processo deve ser atingido pela organização, sendo que estes níveis são acumulativos.

2.6 Fatores Críticos de Sucesso

Muitas organizações desenvolvedoras de software estão dedicando esforços significativos para melhorar seus processos através das iniciativas de melhoria. Contudo, essas iniciativas de melhoria nem sempre têm sucesso. Neste contexto, diversos estudos experimentais foram desenvolvidos para tentar compreender os fatores que influenciam as iniciativas de melhorias em processos de software de forma positiva ou negativa. A seguir são descritos alguns destes principais estudos.

WILSON et al. (2001) desenvolveram um *framework* para avaliação de sucesso de iniciativas de melhorias em processos de software e validaram essa abordagem com entrevistas em grupos em sete organizações do Reino Unido do ponto de vista de desenvolvedores, supervisores/líderes de equipe, alta gerência e coordenadores da melhoria. Os autores identificaram questões específicas no *framework* que aparentam ser indicadores significativos das diferenças entre organizações que apresentaram sucesso e fracasso. Este *framework* poderá auxiliar organizações a conduzirem auto-avaliações da sua prontidão para comprometer-se com iniciativas de melhorias em processos de software. O *framework* proposto pelos autores foi elaborado com base em quatro perspectivas de requisitos organizacionais:

- Contexto – ambiente no qual o programa de melhoria de processos de software é desenvolvido e operado.
- Entradas – fatores ou recursos que são aplicados para o programa de melhorias em processos de software.
- Processo – método utilizado para desenvolver, implementar e manter o programa.
- Produtos – processos melhorados, relatórios produzidos e outras saídas do programa.

A coleta de dados envolveu entrevistas em grupo aprofundadas com representantes de sete organizações voluntárias e um questionário completo pelo coordenador da melhoria de processos de software. Para cada companhia, três grupos foram formados: desenvolvedores, supervisores/equipe de líderes e alta gerência.

A partir de análises estatísticas dos dados do estudo, foram identificadas questões que representam indicadores significativos da diferença entre organizações bem sucedidas e mal sucedidas. Estas questões indicam os seguintes fatores como sendo críticos para o sucesso da melhoria de processos de software:

- Comprometimento da alta gerência;
- Respeito da equipe responsável pelo programa de melhorias em processos de software;
- Definição inicial dos processos (adequação dos processos); e
- Explicações sobre eventos associados ao programa de melhorias em processos de software (comunicação).

BADDOO e HALL (2002) apresentam achados de estudos sobre os motivadores de melhorias em processos de software envolvendo quase 200 engenheiros de software em 13 organizações do Reino Unido. Os autores sugerem que a melhoria de processos de software tem chances maiores de sucesso em organizações nas quais os praticantes possuem alta motivação para isso. Os autores também sugerem que motivações para melhoria de processos de software variam entre grupos nas organizações e que essa variação dilui a efetividade de estratégias de implementação de melhorias em processos de software que não consideram as diversidades de motivadores entre esses grupos. Os autores esperam que a identificação de similaridades e diferenças entre os motivadores dos envolvidos nas iniciativas de melhorias em processos de software permita que gerentes de melhoria em processos de software desenvolvam múltiplas estratégias de implementação de melhoria, tentando maximizar o apoio aos grupos. Além disso, é possível reduzir os custos da implementação, pois os fatores mais importantes poderão ser tratados com mais urgência e trazer resultados em menor espaço de tempo. Uma lista completa com os motivadores envolvidos em iniciativas de melhorias em processos de software pode ser encontrada em BADDOO e HALL (2002)

BADDOO e HALL (2003) relataram em outro estudo com foco nos desmotivadores para implementação de melhoria em processos de software. Neste estudo, foi observado que os desmotivadores para implementação de melhorias em processos de software representam a ausência dos motivadores apresentados em (BADDOO e HALL, 2002), sugerindo que alguns dos motivadores e desmotivadores podem ser interpretados como sendo parte do mesmo fator crítico de sucesso. Os fatores identificados pelos autores como desmotivadores para a melhoria de processos de software são os seguintes:

- Resistência a mudanças;
- Falta de evidência;
- Iniciativas impostas de melhoria de processos de software;
- Restrições de recursos; e
- Pressões comerciais.

Em EL-EMAM et al. (2001), os autores apresentaram os resultados de um estudo de fatores que influenciam o sucesso de melhoria de processos de software envolvendo organizações que conduziram avaliações nos seus processos. O estudo analisou os dados extraídos de 138 questionários de acordo com o papel dos participantes nas organizações (gerente de projetos de software, alta gerência e gerente do grupo de processos). Dentre os fatores resultantes desse estudo estão: (i) comprometimento da equipe; (ii) rotatividade dos membros da equipe; (iii) políticas organizacionais; (iv) respeito entre os membros; e (v) foco nos objetivos da iniciativa.

NIAZI et al. (2005b) descrevem um modelo de maturidade para a implementação de melhorias em processos de software desenvolvido a partir dos resultados do estudo apresentado. O objetivo deste modelo é guiar organizações na avaliação e melhoria dos seus processos de implementação de melhorias em processos de software. A base desse modelo é constituída de informações sobre fatores críticos de sucesso e barreiras críticas identificadas a partir da revisão da literatura e de um estudo experimental (*survey*) realizado com desenvolvedores de software. A estrutura do modelo de maturidade apresentado possui três dimensões:

- Dimensão de estágios de maturidade;
- Dimensão de fatores críticos de sucesso; e
- Dimensão de avaliação.

A categorização dos fatores críticos de sucesso e das barreiras críticas permitiu que fossem definidos diferentes níveis de maturidade para a implementação de melhorias em processos de software. Esta categorização foi realizada com base na revisão da literatura e nas entrevistas. Segundo este modelo, as organizações devem tratar cada fator para alcançar certo nível de maturidade. Diferentes práticas foram definidas e associadas a cada um dos fatores do modelo para guiar na avaliação e implementação do respectivo fator. O modelo apresentado no estudo faz parte de um trabalho maior que tem como objetivo desenvolver

um *framework* para guiar o desenvolvimento de estratégias efetivas de implementação para melhoria de processos de software (NIAZI *et al.*, 2005a).

NIAZI *et al.* (2006) apresentam os resultados de um estudo experimental dos fatores críticos de sucesso na implementação de melhorias em processos de software com 34 implementadores de melhoria. Os autores identificaram oito fatores críticos. Estes fatores foram confirmados por meio de comparação dos resultados com uma revisão da literatura sobre fatores críticos de sucesso que tem impacto na melhoria de processos de software.

Outros estudos importantes conduzidos para investigar fatores que afetam a melhoria de processos de software são freqüentemente incluídos na revisão da literatura dos estudos descritos acima. Por exemplo, GOLDENSON e HERBSLEB (1995) conduziram um estudo com 138 participantes que estavam envolvidos em 56 avaliações CMM, com o objetivo de identificar fatores associados a programas de melhorias em processos de software de sucesso e de fracasso. EL-EMAM *et al.* (1999) analisaram dados coletados a partir da administração de questionários em 14 organizações envolvidas nos *trials* do projeto SPICE (*Software Process Improvement and Capability dEtermination*) (ISO/IEC, 2003). A partir desse estudo, o relacionamento entre fatores a duas variáveis identificadas de sucesso foi investigado. STELZER e MELLIS (1998) revisaram relatos de experiência da literatura e estudos de caso de 56 organizações que executaram com sucesso programas de melhorias em processos de software e identificaram 10 fatores que afetam o sucesso de melhoria a partir dessa revisão.

2.6.1 Um Estudo sobre os Fatores Críticos de Sucesso em Iniciativas de Melhorias em Processos de Software

Para apoiar a condução dos estudos experimentais na área de melhoria de processos de software no contexto dos trabalhos relacionados a esta dissertação, foi desenvolvida uma metodologia que combina técnicas qualitativas e quantitativas para coletar e analisar dados relacionados a fatores que têm impacto positivo ou negativo em experiências de iniciativas de melhorias em processos de software de instituições implementadoras de melhoria de processos. Os métodos e as técnicas utilizados neste estudo foram aplicados por outros estudos com os mesmos objetivos de pesquisa, mas com foco em outros contextos. A metodologia desenvolvida será apresentada por meio da descrição dos passos

realizados na sua aplicação em um estudo específico (MONTONI, 2007; MONTONI e ROCHA, 2007).

Inicialmente, foram definidos um contexto e escopo para o estudo. O estudo foi restringido para analisar apenas experiências de um grupo específico de implementadores de melhorias em processos de software que participaram em iniciativas de melhoria baseadas em modelos e padrões de referência (por exemplo, CMMI (SEI, 2006) e MPS.BR (SOFTEX, 2007)). Neste estudo, teve-se uma preocupação de identificar fatores críticos de sucesso sob dois pontos de vista. O primeiro ponto de vista é o de implementadores que participaram como consultores em projetos de implementação de melhorias em processos de software. O segundo ponto de vista é o de membros das organizações envolvidas nos projetos de melhoria.

No início do estudo foi aplicado o método *Grounded Theory* (STRAUSS e CORBIN, 1998). O primeiro passo para aplicação desse método foi coletar os dados por meio da aplicação de dois tipos de questionários, com o objetivo de identificar fatores que influenciam a implementação de melhorias em processos de software. O primeiro tipo de questionário foi enviado para um grupo selecionado de implementadores de melhorias em processos de software que participaram como consultores em organizações nacionais de diversos tipos. O segundo tipo de questionário foi enviado para membros de organizações envolvidos em projetos de implementação de melhoria. Os questionários não continham nenhum item pré-determinado e os participantes o preencheram separadamente. Depois de retornados os questionários, foram associados códigos a cada uma das declarações nos questionários, representando tipos de achados de fatores críticos de sucesso que influenciaram iniciativas de implementação de melhorias em processos de software.

Os dados codificados (achados) foram, então, analisados e categorizados. Estas categorias foram denominadas de propriedade de fator crítico de sucesso. A associação entre os achados e uma propriedade de fator crítico de sucesso foi classificada como um achado representando a presença ou a ausência de um fator crítico de sucesso em um contexto específico da análise. A teoria começa a tomar forma à medida que mais relações são identificadas entre os achados e as propriedades de fator crítico de sucesso. Essa teoria foi refinada e mais conhecimento sobre os fatores críticos de sucesso foram integrados até que a coleta e análise de mais dados não agregavam nenhum novo conhecimento à teoria. Como resultado desse processo de refinamento e integração, chegou-se à categoria central denominada “Fatores que facilitam e dificultam a condução de iniciativas de melhorias em processos de software”.

Em seguida, foram aplicadas técnicas de análise estatística com o objetivo de derivar, agregar e correlacionar os fatores críticos de sucesso principais de forma que pudessem ser estabelecidas correlações multivariadas entre as propriedades dos fatores críticos.

O Fator 1 foi chamado de “Ambiente” e suas variáveis medem a capacidade ambiental para estabelecer e manter iniciativas de melhorias em processos de software. Estas variáveis medem se existem condições favoráveis para iniciar e manter uma iniciativa de melhoria sob dois pontos de vista: o indivíduo e a organização. As medidas do indivíduo estão relacionadas à satisfação dos membros da organização e ao relacionamento entre esses membros e a equipe de consultoria de melhorias em processos de software. As medidas da organização são relacionadas à conciliação entre os objetivos estratégicos e interesses da organização na melhoria de processos de software e à estabilidade organizacional interna.

O Fator 2 foi chamado de “Estratégia”. Este fator indica que uma estratégia de melhorias em processos de software deve garantir que os membros da organização têm consciência dos benefícios que podem ser alcançados com a implementação da melhoria de processos de software.

O Fator 3 foi chamado de “Institucionalização” e suas variáveis medem o grau de institucionalização das melhorias pela caracterização do grau de resistência da institucionalização dos processos e procedimentos a mudanças estruturais da organização, por exemplo, rotatividade de pessoal, e a dificuldades inerentes da implementação de melhorias em processos de software nos diferentes níveis organizacionais.

O Fator 4 foi chamado de “Comprometimento”. Todas as suas variáveis são consideradas indicadores de comprometimento para a melhoria de processos de software. A alta gerência comprometida para a melhoria de processos de software fornece recursos financeiros adequados desde a concepção do programa de melhoria e ao longo dos projetos de melhoria. Além disso, uma gerência sênior comprometida garante que os membros da organização têm competências adequadas e tempo disponível para executar mudanças de processos eficientemente.

O Fator 5 foi chamado de “Motivação e aceitação” e suas variáveis são uma indicação que a equipe de melhorias em processos de software é um facilitador da aceitação dos membros da organização para a institucionalização de mudanças nos processos promovidas pelas iniciativas de melhorias em processos de software.

A Tabela 2.2 apresenta o quadro com a composição dos principais componentes de fatores críticos resultantes do estudo.

Tabela 2.2 – Principais Componentes de Fatores Críticos de Sucesso (MONTONI *et al.*, 2008b)

Fator	Variáveis	Descrição
Ambiente	Competências dos membros da organização	Grau de competências em engenharia de software dos membros da organização
	Conciliação de interesses	Grau de adequação da conciliação de interesses na implementação de processos
	Estrutura da organização	Grau de estabilidade interna na organização
	Motivação e satisfação dos membros da organização	Grau de satisfação dos membros da organização
	Política de reconhecimento à colaboração na melhoria dos processos	Existência de política de reconhecimento à colaboração na melhoria dos processos
	Respeito da consultoria pelos membros da organização	Grau de relacionamento dos membros da organização com a consultoria especializada
Estratégia	Conscientização dos benefícios da implementação da melhoria dos processos	Grau de conscientização dos membros da organização quanto aos benefícios obtidos com a implantação dos processos
	Estratégia de implementação de melhorias em processos de software	Grau de adequação da gerência do projeto de implementação da melhoria dos processos
Institucionalização	Estratégia de implementação de melhorias em processos de software	Grau de adequação da relação push-pull na implementação dos processos
	Estrutura da organização	Grau de rotatividade de pessoal
	Processos	Grau de adequação dos processos/procedimentos e Grau de institucionalização das melhorias implementadas nos projetos
Comprometimento	Apoio, comprometimento e envolvimento	Grau de apoio, comprometimento e envolvimento da alta gerência
	Competências dos membros da organização	Grau de competências em engenharia de software dos membros da organização
	Recursos	Grau de disponibilidade de recursos financeiros dos membros da organização para atividades de melhoria de processos e Grau de disponibilidade de tempo dos membros da organização para atividades de melhoria de processos
Motivação e aceitação	Aceitação a mudanças	Grau de aceitação a mudanças
	Motivação e satisfação dos membros da organização	Grau de motivação dos membros da organização
	Respeito da consultoria pelos membros da organização	Grau de confiança dos membros da organização na consultoria especializada

Baseado no conhecimento adquirido sobre os fatores que influenciam o sucesso das iniciativas de melhoria, MONTONI (2008a) definiu uma abordagem para auxiliar as organizações de consultoria a gerenciar seus projetos de melhoria mais eficientemente, de forma a tratar adequadamente esses fatores de influência. Esta abordagem é apoiada por uma infra-estrutura computacional baseada em um ambiente customizado em gerência de conhecimento centrado em processo, que será descrita com mais detalhes no Capítulo 4.

2.7 Considerações finais

Para se ter sucesso nas iniciativas de melhorias em processos de software, são necessários vários entendimentos. Deve-se entender a dinâmica dos processos de desenvolvimento e as propriedades do produto desenvolvido. Deve-se, também, estabelecer objetivos estratégicos muito bem definidos para evitar a perda do foco, além de avaliar as experiências tanto de sucesso, quanto de falhas para que possam ser utilizados numa futura tomada de decisão, ou para que sejam evitados problemas passados que já possuem solução e que consomem recursos preciosos. Para isto, é necessário prover mecanismos de manutenção das experiências obtidas em projetos correntes e passados, de forma que se possa aprender com as próprias experiências e retroalimentar as iniciativas de melhoria futuras com essas lições aprendidas. Espera-se que isso leve a uma maior efetividade na condução desses tipos de iniciativa, evitando perda de recursos muitas vezes escassos, o que poderia influenciar de forma negativa o sucesso dessas iniciativas.

Este capítulo apresentou a revisão da literatura sobre os paradigmas e abordagens de melhoria bastante difundidos na comunidade de engenharia de software e que serão utilizados para nortear a definição da abordagem proposta por este trabalho e seus trabalhos conjuntos. Foram abordados os paradigmas do PDCA (*Plan-Do-Check-Act*) e do QIP (*Quality Improvement Paradigm*) e a abordagem de Fábrica de Experiências. Além destes, foram apresentados as principais normas e modelos de referência para a melhoria de processos de software e também um estudo sobre fatores que influenciam de forma positiva e negativamente o sucesso das iniciativas de melhoria realizado em um contexto de pesquisa mais amplo no qual a abordagem proposta por este trabalho está relacionada.

No próximo capítulo, será abordado o tema central desta dissertação: o *benchmarking*.

CAPÍTULO 3 – *BENCHMARKING*

Este capítulo apresenta a definição da metodologia de benchmarking, seus principais conceitos relacionados a esta técnica, além dos tipos, abordagens e as atividades comumente realizadas em estudos de benchmarking.

3.1 Introdução

Segundo o lendário general Sun Tzu, “Se você conhece seu inimigo e a si mesmo, não precisará temer o resultado de cem batalhas”. Assim como numa guerra, resolver problemas comuns de negócios, conduzir batalhas gerenciais e sobreviver no mercado atual são formas de guerra, lutadas pelas mesmas regras, e ter conhecimento sobre o domínio do negócio é de suma importância para ganhar essa guerra.

Estabelecer alvos operacionais com base nas melhores práticas da indústria é um componente crítico no sucesso de toda empresa de desenvolvimento de software. As equipes de gerência em geral estão sempre à procura de formas melhores e mais baratas de realizar as tarefas diárias (MARD *et al.*, 2004). Talvez, uma forma ligeiramente mais esclarecida seja respeitar o passado, mas modificá-lo em função de condições conhecidas: crescer em função da atividade conhecida, efetuar ajustes em função dos fatores conhecidos de planejamento e acrescentar um ajuste para a produtividade necessária (CAMP, 1993).

Isto remete a uma metodologia amplamente utilizada em várias áreas da indústria, cujo objetivo é determinar as boas práticas relacionadas a uma determinada operação: o *benchmarking*. Dentre as abordagens que podem ajudar uma empresa a melhorar seu desempenho, *benchmarking* é visto hoje como uma das mais eficientes e eficazes ferramentas de gerência (MAIRE e BUYUKOZKAN, 1997).

Este capítulo trata da revisão da literatura sobre a metodologia de *benchmarking*. A seção 3.2 apresenta o conceito de *benchmarking*, a seção 3.3 apresenta o vocabulário de termos relacionados, a seção 3.4 apresenta os tipos de *benchmarking* e as suas possíveis abordagens de utilização, a seção 3.5 apresenta as atividades comumente realizadas em estudos de *benchmarking*, a seção 3.6 apresenta alguns trabalhos correlatos e, por fim, a seção 3.7 apresenta as considerações finais deste capítulo.

3.2 O que é *Benchmarking*?

Benchmarking é uma metodologia utilizada pelas organizações para aperfeiçoar a gestão através da realização sistemática de levantamentos de dados e análises de práticas, processos, produtos e serviços prestados por outras organizações (IAPMEI). O conceito de *benchmarking* não é novo. Na indústria de serviços, os *benchmarks* têm sido aplicados de diferentes formas para avaliar o desempenho dos processos de negócio. (GRIMBLE e UDUEHI, 2001).

A metodologia do *benchmarking* nasceu na década de 70 e teve um marco importante na empresa Xerox, que iniciou um processo sistemático de pesquisa sobre os processos de gestão e produção utilizados pelas suas concorrentes e por outras organizações de destaque em determinadas áreas. Durante a década de 80, este conceito foi difundido em todo mundo e na década de 90 o seu uso estendeu-se, principalmente, em organizações norte-americanas e europeias (IAPMEI).

Benchmarking é tido como um processo positivo, pró-ativo e estruturado que conduz a mudanças nas operações e finalmente, ao desempenho superior e a uma vantagem competitiva. Ele trata da pesquisa industrial ou coleta de informações que permite a um gerente comparar o desempenho da sua função com o desempenho das mesmas funções em outras empresas ou internamente (entre equipes, setores ou unidades organizacionais). É a pesquisa industrial ou coleta de informações que permite a um gerente comparar o desempenho da sua função com o desempenho das mesmas funções em outras empresas (CAMP, 1993).

CAMP (1993), define *benchmarking* como:

“Um **processo contínuo de medição de produtos, serviços e práticas** em relação aos mais fortes concorrentes, ou às empresas reconhecidas como líderes em suas indústrias.”

Algumas palavras merecem destaque nessa definição (CAMP, 1993):

- **Processo contínuo** – As práticas executadas nas organizações necessitam de constante monitoração, no intuito de garantir a descoberta daquelas que serão consideradas as melhores.

- **Medição** – O termo *benchmarking* significa medição. Portanto, é necessário o estabelecimento de um programa de medição bem definido para que as práticas possam ser quantificadas e para que se tenha uma medida do tamanho da oportunidade da melhoria associada àquela prática em relação a outras referenciadas como melhores práticas.
- **Produtos, serviços e práticas** – Nas análises de *benchmarking*, parte-se do princípio de que as atividades empresariais podem ser analisadas como processos. O estudo dos processos empresariais e seus métodos e práticas será o principal objetivo da realização de um *benchmarking*.

O ISBSG (*Internacional Software Benchmarking Standards Group*) (ROLLO *et al.*, 2006) define que o propósito do *benchmarking* é coletar, analisar e reportar dados relativos aos produtos desenvolvidos e/ou processos executados dentro de uma organização, unidade organizacional, setor ou departamento, para apoiar a gerência dos processos e demonstrar o desempenho comparativo desses processos. Dentre as possíveis razões para se realizar comparações, estão:

- Comparar unidades, setores ou departamentos de uma organização;
- Comparar uma organização com seus concorrentes;
- Comparar o desempenho de uma organização com as médias de desempenho da indústria;
- Comparar o desempenho entre os projetos de uma organização;
- Obter medidas de desempenho de projetos finalizados, para servir de entrada para a realização de estimativas de projetos futuros.

HOST (2002) relata que as organizações que participam deste tipo de iniciativa possuem uma série de objetivos em comum. Primeiramente, o objetivo é receber informações sobre como outras organizações realizam o seu trabalho no mesmo domínio de negócio. De certa forma, isso permitirá receber propostas de melhoria baseadas no trabalho que outra organização vem desempenhando. Outro objetivo comum é receber informações sobre seus processos, de especialistas no domínio do negócio.

Um *benchmarking* pode ser usado como um veículo para a comparação de desempenho, assim como para prover um ímpeto inicial para uma iniciativa de melhoria. Deve-se destacar que, à medida que uma equipe reconhece oportunidades de melhoria além de suas observações diretas, um senso de competitividade surge, motivando os membros

da organização a se empenharem por excelência, inovação e aplicação do pensamento inovador, a fim de conseguir sua própria melhoria de processos (CAMP, 1993).

Em termos de definições operacionais, um *benchmark* é um ponto de referência através do qual medições e comparações de qualquer tipo podem ser realizadas (JURAN, 1994). Explicitamente, podemos encarar o *benchmarking* como um processo de avaliação e melhoria do desempenho (MAIRE e BUYUKOZKAN, 1997).

A realização do *benchmarking* pode gerar alguns benefícios importantes (CAMP, 1993; HARRINGTON e HARRINGTON, 1996; GRIMBLE e UDUEHI, 2001) tais como:

- Atendimento mais adequado às exigências do cliente;
- Estabelecimento de metas com base em uma visão conjunta das condições internas e externas;
- Aumento da produtividade e eficiência através da determinação de medidas reais de produtividade;
- Determinação de como os objetivos estabelecidos podem ser atingidos, através do reconhecimento dos pontos que merecem maior atenção dos programas de melhoria;
- Conquista de uma posição competitiva;
- Conscientização a respeito das melhores práticas; e
- Revelação de tecnologias e práticas emergentes.

MILLER e TUCKER (1991), destacam alguns outros benefícios que estão associados à realização de *benchmarking*:

- A construção e alimentação de um repositório de dados através do qual a organização pode gerar um grande número de relatórios úteis para a realização de análises de tendência, identificação de áreas, atividades ou processos problemáticos, etc.
- A identificação de pontos fortes e pontos fracos nos projetos. Esta informação pode servir como motivação para novas iniciativas em áreas fracas em relação às demais, assim como a transferência de tecnologia e conhecimento.
- A identificação de práticas utilizadas em vários projetos e que são provavelmente eficazes, o que as torna marcos de referência e faz com que sejam recomendadas para utilização em projetos futuros.

Contudo, não é simples conduzir um estudo de *benchmarking*. Segundo CARD e ZUBROW (2001), existem três fatores cruciais para o sucesso deste tipo de estudo:

- Ter objetivos bem definidos: O que desejamos aprender a partir dos esforços de *benchmarking*?
- Planejar cuidadosamente: O processo de *benchmarking* deve estar intimamente alinhado aos objetivos organizacionais estabelecidos;
- Ter cuidado com os dados coletados e comparados: É necessária uma boa análise dos dados coletados a fim de garantir que é possível comparar os resultados. A comparação de produtividade pode ser bastante complexa, mesmo que se tenha um processo de coleta de dados de qualidade (MAXWELL, 2001). Dados randômicos geram resultados randômicos.

A caracterização do desempenho é uma das etapas iniciais na busca pela melhoria de processos e das melhores práticas. Para ajustar os alvos de desempenho, uma organização deve conhecer seu próprio desempenho e determinar que metas de desempenho são possíveis dentro de suas restrições. Um exercício de *benchmarking* serve, dentre outros, para caracterizar o desempenho de uma organização em relação a um grupo de projetos similares, baseado em alguns indicadores chave, tais como a produtividade, a qualidade, ou o tempo de entrega. Uma pergunta respondida pelo *benchmarking* é: “onde nós estamos?”. A organização irá utilizar a *baseline* estabelecida através da caracterização como base de comparação com um desempenho futuro (LOKAN *et al.*, 2001).

CAMP (1993) cita quatro passos considerados como essenciais para o estudo de *benchmarking*. Os passos, abaixo relacionados, são fundamentais para o sucesso do estudo:

- Conhecer a própria operação, pois será necessário avaliar os pontos fortes, pontos fracos e oportunidades de melhoria das operações internas.
- Conhecer os líderes da indústria ou concorrentes. Só se estará preparado para diferenciar as próprias capacidades no mercado caso se conheça as forças e as fraquezas dos concorrentes.
- Incorporar o melhor, aprendendo com os líderes da indústria e com seus concorrentes.
- Conquistar superioridade. Se investigações cuidadosas das melhores práticas tiverem sido efetuadas e as melhores entre elas tiverem sido instaladas, então você terá aproveitado as forças existentes, corrigindo as fraquezas para que se equiparem ao mercado e ido além, incorporando o melhor do melhor.

O *benchmarking* é a aplicação formalizada e disciplinada desses passos básicos para o aperfeiçoamento operacional e a conquista da superioridade (CAMP, 1993).

Podemos dividir o *benchmarking* em duas partes: (i) práticas e (ii) métricas. As práticas são definidas como os métodos que são usados; e as métricas são o efeito quantificado da instalação das práticas. Cada uma delas pode ser investigada pelo processo. A tendência é de os gerentes trabalharem na determinação dos alvos quantitativos ou métricos, para que possam compreender rapidamente o que, de bom ou de mau, a organização obterá do *benchmarking* em relação ao ambiente externo (CAMP, 1993).

3.3 Definições

Antes de continuar a revisão da literatura sobre a metodologia de *benchmarking*, é interessante o estabelecimento da terminologia a ser utilizada nesta dissertação.

O ISBSG contém em seu documento de padronização um dicionário de termos que define o *benchmarking* e seu vocabulário (ROLLO *et al.*, 2006). Alguns desses termos estarão presentes de forma constante neste trabalho, de forma que é interessante defini-los:

- **Benchmark (substantivo):** Um valor de alguma medida básica ou derivada que é indicativa do relacionamento entre um atributo organizacional e os valores daquele atributo mantido num repositório de *benchmarking*.
- **Benchmarking (verbo):** executar uma série de processos para estabelecer um valor relativo de um atributo organizacional com respeito ao repositório de dados a ser usado para fins de comparação.
- **Repositório de Benchmarking:** Uma coleção de dados organizada e persistente da qual é possível a recuperação desse conjunto de dados que são designados para serem usados como fonte de comparação para o *benchmarking*.
- **Método de Benchmarking:** Uma seqüência lógica de operações, descritas de forma genérica, usadas na quantificação de um atributo com respeito a uma escala específica.
- **Processo de Benchmarking:** O processo para identificar, definir, selecionar, aplicar e melhorar o *benchmarking* dentro de um projeto ou estrutura de *benchmarking* organizacional.
- **Usuário do Benchmarking:** Um indivíduo ou organização que utilizar as informações geradas pelo *benchmarking*.

- **Desempenho:** Uma medida derivada, que fornece uma indicação de um atributo associado a quão bom, quão rápido ou quão eficiente um processo é executado.

3.4 Tipos de *Benchmarking*

Dentre os tipos de *benchmarking* existentes, destacam-se os mais gerais, que são o *benchmarking* interno e *benchmarking* externo.

No *benchmarking* interno, a busca pelas melhores práticas ocorre dentro da própria organização. Ele utiliza medidas básicas da organização, ou as melhores unidades organizacionais (outros departamentos, sedes, etc.) ou pessoas (operário do mês, melhor vendedor, etc.). Tem como vantagens a facilidade para se obter parcerias, custos mais baixos e a valorização pessoal interna. A grande desvantagem é que não se tem uma visão de desempenho em relação ao mercado concorrente. Segundo MAIRE e BUYUKOZKAN (1997), provavelmente o erro mais freqüente na realização do *benchmarking*, é estudar os outros antes mesmo de entender nossas próprias operações. Algumas questões que devem ser respondidas por um *benchmarking* interno são (ROLLO et al., 2006):

- Nossa iniciativa de melhoria é eficaz?
- Que evidências existem para justificar as estimativas que estamos usando?
- Estamos atingindo níveis de desempenho aceitáveis?
- Todas as unidades organizacionais possuem desempenho semelhante?
- A introdução de novas tecnologias atingiu os benefícios esperados?

Benchmarking externo é o processo contínuo de comparar e medir uma organização com líderes de negócios em qualquer lugar do mundo, para obter informações que ajudem a organização a tomar ações para melhorar o seu desempenho (ROLLO et al., 2006). É também conhecido como *benchmarking* competitivo, pois envolve a comparação de práticas e resultados entre as organizações. O inconveniente desta abordagem se situa no fato de que as organizações nunca desejam compartilhar suas práticas baseadas na hipótese de que terceiras se beneficiariam de forma unilateral com isso. Porém, há um desejo geral em conhecer as metas a serem atingidas para sua competitividade frente ao mercado (MARD et al., 2004). Este tipo de *benchmarking* pode ser realizado por uma série de motivos (ROLLO et al., 2006):

- Como as atividades da organização estão em relação aos padrões da indústria?
- Nós estamos mais ou menos eficazes em comparação com nossos concorrentes?

- Nosso processo é mais eficaz ou nós precisamos de uma iniciativa de melhoria?
- Qual o retorno do investimento esperado no caso de iniciarmos um programa de melhoria?

Existe ainda uma forma de *benchmarking* externo que não apresenta tantos problemas relacionados à confidencialidade das informações. Este é o chamado *benchmarking* genérico. Neste tipo de *benchmarking*, realizam-se comparações entre operações e processos de indústrias que nem sempre estão relacionadas, mas que compartilham algumas funções em comum. Daí a menor resistência em ocultar informações, pois estas organizações não serão diretamente concorrentes e podem se beneficiar mutuamente do estudo (MARD et al., 2004).

Apesar de haver um consenso de que existem marcos externos, as organizações não reconhecem que não há nenhuma realização externa que possa refletir fatos e circunstâncias internas, ou seja, cada boa prática que surge e é realizada em organizações terceiras está impregnada e contextualizada com os fatores e circunstâncias específicas daquela organização.

Porém, isso não quer dizer que não se deva dar o real valor ao *benchmarking* externo. Os resultados obtidos em outras organizações podem ser vistos como fatores motivadores de melhoria para uma terceira organização, além de apoiar a definição de novas estratégias específicas (CAMP, 1993).

Internamente à classificação por tipo, é possível descrever dois métodos de *benchmarking*:

- Qualitativo, onde a quantificação de um atributo envolve julgamento humano ou comparação; e
- Quantitativo, que está baseado em regras numéricas e onde as comparações são feitas baseadas em valores numéricos, o que permite que se submeta as comparações a análises estatísticas. O *benchmarking* quantitativo depende da habilidade em se coletar e montar um bom conjunto de dados históricos (CARD e ZUBROW, 2001).

O *benchmarking* pode ser adaptado segundo quatro estratégias (MARD et al., 2004):

- Manter o nível atual de desempenho; ou seja, a organização deseja caracterizar suas operações e determinar seu nível de desempenho.

- Ajustar o nível de desempenho segundo um nível interno desejável; ou seja, a organização deseja comparar internamente suas operações e igualar o nível de desempenho de todas as suas unidades, melhorando assim o desempenho organizacional.
- Ajustar o nível de desempenho segundo o desempenho da indústria; ou seja, a organização deseja comparar suas operações com os tidos como melhores do setor industrial a que pertence e, assim, ajustar o seu nível de desempenho.
- Ajustar o nível de desempenho segundo o nível de desempenho de um determinado grupo semelhante. Neste tipo de estratégia, a organização compara suas operações com as operações de um determinado grupo de organizações cujas operações se assemelham àquelas que ela desempenha, mesmo que essas organizações não sejam exatamente da mesma indústria. Esta estratégia de *benchmarking* também é conhecida como *benchmarking* funcional (SPENDOLINI, 1992), pois há menos ênfase nas práticas do concorrente, mas sim uma busca das melhores práticas existentes, sem se importar com produto específico ou o serviço em que estão sendo empregados.

3.5 Modelos de Processo de *Benchmarking*

Existe uma série de abordagens de aplicação da metodologia de *benchmarking* encontradas na literatura, sendo que cada uma delas é aplicada em um contexto específico e possui, portanto, características específicas. Esta seção descreve alguns dos modelos de processo utilizados nas abordagens pesquisadas. FRIDLEY *et al.* (FRIDLEY *et al.*, 1997) concluem que, assim como ocorre com projetos de engenharia, provavelmente existem tantos modelos de processos de *benchmarking* quanto o total de autores de livros sobre o assunto. Contudo, de forma geral, a maior parte dos processos contém fases de planejamento ou preparação, análise, integração e ação (ZAIRI e LEONARD, 1994).

SPENDOLINI (1992), numa das primeiras e principais leituras sobre o assunto, descreve um processo de cinco estágios para a realização do *benchmarking*, mas sem especificar maiores detalhes:

1. Determinar o que comparar;
2. Formar uma equipe de *benchmarking*;
3. Identificar parceiros de *benchmarking*;
4. Coletar e analisar informações do *benchmarking*; e

5. Tomar ações corretivas.

CAMP (1993) define um processo básico para as atividades de *benchmarking* que consiste de cinco fases: planejamento do *benchmarking*, onde se decide em que função de negócio ou tipo de produto a iniciativa de benchmarking deve se focar, quais organizações devem participar do estudo e também detalhes de como o estudo deve ser realizado; coleta de dados, onde se coleta os dados planejados; análise dos dados, onde ocorre a efetiva comparação dos parâmetros coletados; adaptação das informações, onde se determina a forma como os achados no estudo podem ser integrados aos processos e operações correntes; e por último, implementação das melhorias, onde as mudanças são efetivamente implementadas e os resultados são monitorados. De forma mais detalhada, este processo pode ser descrito através das seguintes etapas:

1. Identificar os marcos de referência. Estes marcos de referência representam os produtos ou áreas críticas da organização, para as quais o *benchmarking* pode trazer vantagens competitivas.
2. Identificar parceiros de *benchmarking*. Nesta etapa são identificadas as organizações, concorrentes ou não, que deverão ter suas operações comparadas com as da organização realizadora do estudo de *benchmarking*.
3. Definir o método de coleta de dados e coletar os dados. Nesta etapa é planejada a coleta de dados que melhor atenda às necessidades do estudo
4. Determinar a lacuna de desempenho. A comparação das operações irá revelar uma lacuna de desempenho, positiva ou negativa, a qual deverá ser analisada e tratada de forma a garantir um bom desempenho das operações da organização.
5. Projetar níveis de desempenho futuro; considerando que as práticas da indústria estão em constante evolução.
6. Comunicar descoberta dos marcos de referência tanto para a gerência quanto aos colaboradores e obter aceitação.
7. Estabelecer metas funcionais; traduzindo as descobertas do *benchmarking* em declarações de como a organização mudará. Neste sentido, a abordagem mais eficaz tem sido a conversão das descobertas mais importantes em declarações de princípios operacionais.
8. Desenvolver plano de ação; mostrando eventos específicos que devem ser elaborados e implementados para se obter um melhor desempenho.
9. Implementar ações específicas e monitorar o progresso do plano de ação.

10. Recalibrar marcos de referência com o objetivo de mantê-los atualizados. Como as práticas dos concorrentes e da indústria mudam constantemente, é preciso implantar um processo de recalibração para que os marcos sejam reavaliados e atualizados e seja possível garantir que se baseiem nos métodos e práticas mais recentes

Um modelo de processos de oito passos, adaptado para a comparação de processos de manufatura é sugerido pela Sociedade de Engenheiros de Manufatura (ANON, 1995):

1. Determinar o processo de Benchmarking, onde o processo é selecionado e adaptado às necessidades específicas do estudo e da organização.
2. Selecionar os membros da equipe, onde todos os envolvidos na realização do estudo são selecionados e suas atribuições são definidas.
3. Desenvolver medidas dos processos, onde é desenvolvido um conjunto de medidas que serão coletadas e comparadas dos processos de manufatura.
4. Identificar parceiros de *benchmarking*. Nesta etapa, são identificados e selecionados os parceiros do estudo de *benchmarking*.
5. Identificar as práticas de interesse e medir o desempenho dos parceiros e seu próprio desempenho. Nesta etapa, as práticas de interesse são identificadas e seu desempenho nos parceiros de *benchmarking* é medido, juntamente com o desempenho dessas práticas dentro da própria organização, a fim de determinar o intervalo de desempenho entre elas.
6. Especificar programas e ações para diminuir o intervalo de desempenho. Nesta etapa, o plano de ação para a melhoria das práticas é elaborado e um programa de melhoria deverá gerenciar a execução das ações de melhoria identificadas.
7. Implementar as mudanças identificadas. Nesta etapa, as ações de melhoria identificadas são executadas.
8. Medir os resultados. Após executadas as ações de melhoria, é necessário verificar a sua efetividade em relação ao intervalo de desempenho identificado.

HARRINGTON e HARRINGTON (1996) sugerem um processo de cinco fases:

1. Planejar o processo de *benchmarking* e caracterizar os itens a serem comparados: Nesta etapa, são identificados os itens (funções, produtos, etc.) a serem comparados, é obtido o comprometimento da alta direção, é desenvolvido um

plano de medição para esses itens, e é desenvolvido um plano de coleta de dados. Finalmente, os planos são revistos com especialistas e os itens são caracterizados.

2. Coleta e análise interna dos dados: Nesta etapa, as informações disponíveis no interior da organização são coletadas e analisadas, potenciais áreas para *benchmarking* são selecionadas, informações de pesquisas internas são coletadas, entrevistas e questionários são utilizados para obter informações e um comitê é formado para auxiliar no estudo.
3. Coleta e análise externa dos dados: Nesta etapa, informações publicadas fora da organização são coletadas.
4. Melhoria do desempenho dos itens comparados: Nesta etapa, são identificadas as ações corretivas e um plano de implementação das melhorias é elaborado, novamente, é obtida a aprovação e comprometimento da alta gerência em relação ao plano desenvolvido, e este é então implementado e os resultados são medidos e monitorados.
5. Melhoria contínua: A base de dados de *benchmarking* é mantida e a melhoria contínua do desempenho é realizada através da realização cíclica do processo.

ANDERSEN e PETERSEN (1996) descrevem um processo de *benchmarking* com seis fases, da seguinte forma:

1. Planejar: Selecionar o processo a ser comparado, baseando-se na estratégia da organização, entender e documentar este processo, formar a equipe de *benchmarking* e estabelecer medidas de desempenho para o processo.
2. Pesquisar: Elaborar a lista de critérios que um parceiro de *benchmarking* deve atender, pesquisar os potenciais parceiros, comparar estes potenciais parceiros e selecionar os mais adequados, estabelecer contato e obter aceitação para participação no estudo.
3. Observar: Avaliar as necessidades de informação, selecionar os métodos e ferramentas de coleta de dados, observar e interrogar os participantes do estudo.
4. Analisar: Ordenar os dados e informações coletadas, examinar a qualidade dos dados coletados, normalizar os dados, identificar os intervalos de desempenho e identificar as causas dos intervalos de desempenho.
5. Adaptar: Comunicar os achados das análises e obter aceitação através da participação, projeto e implementação dos planos de melhoria, monitoração do

progresso e ajuste dos desvios e posterior fechamento da rodada de *benchmarking* com um relatório consolidado.

6. Reciclar: Recalibrar os *Benchmarks* ajustando-os como melhores práticas e realizar novos estudos de *benchmarking* para novas áreas de processo.

FOGLE *et al.* (2001) utilizaram em seu estudo de *benchmarking* sobre bibliotecas de ativos de processo um processo baseado em 6 fases:

1. Iniciação, onde é definido o líder do estudo, o patrocinador e os interessados e são aprovados os requisitos de negócio, objetivos e o escopo do estudo.
2. Planejamento, fase que envolve a seleção e treinamento da equipe, desenvolvimento da infra-estrutura para condução do estudo e a elaboração de um plano do projeto que deverá ser aprovado junto a patrocinados e aos interessados.
3. Identificação dos parceiros de *benchmarking*. Aqui são identificados e selecionados os potenciais parceiros do estudo de *benchmarking*.
4. Coleta de dados, onde são estabelecidos os *benchmarks* relacionados ao estudo, é desenvolvido um plano de coleta de dados e são conduzidas as reuniões com os parceiros para a coleta dos dados estabelecidos no plano.
5. Análise dos dados. Nesta fase, os dados coletados são normalizados e avaliados, são estabelecidos critérios para determinação das boas práticas e estas são, então, selecionadas e aprovadas pela equipe e, em seguida, validadas.
6. Reporte dos resultados. Nesta fase todos os achados do estudo são documentados, gerando um relatório que é destinado aos patrocinadores e interessados do estudo, para que estes determinem e apoiem as ações de melhoria.

O processo de *benchmarking* gera informações importantes para que as empresas conheçam diferentes maneiras de lidar com situações e problemas semelhantes e, desta forma, contribui para que estas possam aperfeiçoar seus processos de trabalho (CAMP, 1993). Favorece, também, a transferência de conhecimento, entre organizações ou unidades organizacionais, auxiliando na identificação de potenciais áreas a serem melhoradas.

FOGLE *et al.* (2001) afirmam que, para executar um estudo de *benchmarking*, os envolvidos devem possuir conhecimento consolidado no domínio que está sendo estudado, entender o processo de *benchmarking* e possuir habilidades para organizar, analisar e compreender os dados coletados. É recomendado também o uso de metodologias

comprovadas, a fim de evitar o desperdício de recursos e obter resultados úteis para a organização.

A Tabela 3.1 apresenta um estudo comparativo dos processos utilizados nas diversas abordagens de *benchmarking* pesquisadas e apresentadas neste capítulo. De forma geral, a maior parte desses processos possui fases de planejamento ou preparação, análise, integração, ação e continuação. Este comparativo será utilizado no próximo capítulo para nortear a definição dos processos de apoio à proposta deste trabalho.

3.6 Trabalhos Correlatos

Dentre as várias fontes de informações pesquisadas e utilizadas no contexto desta dissertação, algumas chamam a atenção por tratar de trabalhos que poderiam ser considerados como correlacionados ao tema desta dissertação. Esta seção tem por objetivo descrever alguns desses trabalhos.

MILLER e TUCKER (1991) descrevem o estabelecimento de uma base de dados corporativa que permite a análise e identificação de tendências e potenciais melhores práticas para guiar os esforços de melhoria da corporação em questão. Os dados armazenados foram coletados através da realização de avaliações dos processos. Os achados das avaliações eram comparados com modelos de referência e utilizados para guiar os esforços de melhoria.

MAXWELL (2001) cita algumas de suas experiências no estabelecimento de repositórios de dados sobre projetos de software. Essas experiências permitiram o estabelecimento de algumas regras básicas que devem ser seguidas neste tipo de estudo. É enfatizada a importância da qualidade dos dados coletados e armazenados. Vários alertas são feitos quanto à análise realizada sobre esses dados, como por exemplo, o fato de sermos tendenciosos a acreditar nos dados quando estes confirmam nossas idéias e vontades, o que pode representar um risco para o sucesso da tomada de decisão. Mesmo assumindo que temos dados de alta qualidade, devemos assegurar que os dados são comparáveis, para que se possa tirar conclusões e basear decisões.

Tabela 3.1 – Processos de *benchmarking*

Fase	SPENDOLINI (1992)	CAMP (1993)	ANON (1995)	HARRINGTON e HARRINGTON (1996)	ANDERSEN e PERTERSEN (1996)	FOGLE <i>et al.</i> (2001)
Planejamento (preparação)	1. Determinar o que comparar	1. Identificar os marcos de referência	1. Determinar o processo de <i>benchmarking</i>	1. Planejar o processo de <i>benchmarking</i> e caracterizar os itens a serem comparados	1. Planejar	1. Iniciação 2. Planejamento
	2. Formar uma equipe de <i>benchmarking</i>		2. Selecionar os membros da equipe			
	3. Identificar parceiros de <i>benchmarking</i>	2. Identificar parceiros de <i>benchmarking</i>	4. Identificar parceiros de <i>benchmarking</i>		2. Pesquisar	3. Identificação dos parceiros de <i>benchmarking</i>
	4. Coletar e analisar informações do <i>benchmarking</i>	3. Definir o método de coleta de dados e coletar os dados	3. Desenvolver medidas dos processos	2. Coleta e análise interna dos dados	3. Observar	4. Coleta de dados
5. Identificar as práticas de interesse e medir o desempenho dos parceiros e seu próprio desempenho			3. Coleta e análise externa dos dados			
Análise	4. Coletar e analisar informações do <i>benchmarking</i>	4. Determinar a lacuna de desempenho		2. Coleta e análise interna dos dados	4. Analisar	5. Análise dos dados
				3. Coleta e análise externa dos dados		
		5. Projetar níveis de desempenho futuro				

Tabela 3.1 – Processos de *benchmarking*

Fase	SPENDOLINI (1992)	CAMP (1993)	ANON (1995)	HARRINGTON e HARRINGTON (1996)	ANDERSEN e PERTERSEN (1996)	FOGLE <i>et al.</i> (2001)
Integração		6. Comunicar descobertas			5. Adaptar	6. Reporte dos resultados
		7. Estabelecer metas funcionais				
Ação	5. Tomar ações corretivas	8. Desenvolver plano de ação	6. Especificar programas e ações para diminuir o intervalo de desempenho	4. Melhoria do desempenho dos itens comparados	5. Adaptar	
		9. Implementar ações específicas e monitorar progresso do plano de ação	7. Implementar as mudanças identificadas 8. Medir os resultados			
Continuação		10. Recalibrar marcos de referência com o objetivo de mantê-los atualizados		5. Melhoria contínua	6. Reciclar	

HEIRES (2001) apresenta um estudo de *benchmarking* realizado no departamento de tecnologia da informação de uma organização produtora de soluções para dispositivos eletrônicos de aviação. O objetivo do estudo era melhorar a habilidade do departamento de desenvolvimento de soluções na implementação de seus projetos. Este estudo derivou uma *baseline* quantitativa dos custos dos projetos que permitem a realização de atividades de *benchmarking*, com o intuito de rastrear mudanças de desempenho causadas por iniciativas de melhoria de processos de software. A conclusão do estudo indica que o departamento possuía um bom desempenho em relação aos custos de desenvolvimento de suas aplicações, mas seu cronograma geralmente apresentava atrasos em relação às outras organizações. O estudo envolveu etapas de planejamento, coleta de dados, análise dos dados e comunicação dos achados às diversas instancias da organização. Apesar do estudo não ter revelado surpresas, ele auxiliou na quantificação do desempenho da organização, de forma que os esforços de melhoria puderam ser concentrados.

Já THOMAS e SMITH (2001) relatam os resultados de um estudo de *benchmarking* realizado em uma empresa para a construção de helicópteros, cujo objetivo era identificar melhores práticas que possibilitassem a aceleração da implementação do nível 3 do CMMI. A seleção dos parceiros de *benchmarking* foi realizada a partir de empresas parceiras que já haviam concluído suas iniciativas de melhoria, procurando extrair delas as melhores práticas. A coleta dos dados se deu através da distribuição de questionários com questões distribuídas em várias categorias de interesse, como por exemplo, filosofia de implementação, comprometimento da gerência, e outros. Após a análise dos questionários respondidos, iniciou-se a etapa de entrevistas para a identificação das melhores práticas. Os dados extraídos do estudo foram utilizados na definição da estratégia de implementação da iniciativa de melhorias em processos a ser realizada na organização.

LOKAN *et al.* (2001) ilustram o uso de um repositório semi-público para fins de *benchmarking* de projetos de software: o ISBSG. Este repositório é semi-público no sentido de que é mantido pelo instituto citado e pelas organizações a ele associadas. O objetivo do ISBSG é reunir um corpo de conhecimentos sobre como projetos de software são implementados, aprender lições obtidas das experiências através de pesquisa e análise das mesmas, e disseminá-las na comunidade de desenvolvimento de software. As organizações utilizam a base de dados do ISBSG para seleção de projetos a partir dos quais realizarão comparações com seus próprios projetos. São citados alguns benefícios que as organizações podem obter com a realização deste tipo de estudo, como por exemplo, a coleta de informações que permite o estabelecimento de metas futuras de desempenho, e

também quais dados são importantes para o sucesso do estudo. Dentre os atributos coletados de cada projeto, estão:

- Contexto do projeto, envolvendo informações sobre o país a que a organização pertence, tipo da organização, área de atuação e tipo de desenvolvimento;
- Características do produto, como por exemplo, domínio da aplicação e arquitetura utilizada;
- Características do desenvolvimento, como por exemplo, linguagem de programação, plataforma de desenvolvimento, técnicas e ferramentas utilizadas;
- Tamanho dos projetos em termos de esforço, duração, custo e qualidade dos entregáveis;
- Fatores qualitativos que influenciam a execução dos projetos, como por exemplo, a experiência dos desenvolvedores, estabilidade dos requisitos e as características do ambiente de desenvolvimento.

Os membros dessa organização internacional são as associações nacionais de métricas de software de países como os Estados Unidos, Austrália, Itália, Alemanha, Japão, Holanda, Espanha e Índia. O ISBSG também conta com a cooperação de organizações finlandesas e brasileiras.

FOGLE *et al.* (2001) descrevem uma abordagem para a condução de atividades de *benchmarking*, com o intuito de identificar e coletar melhores práticas nas bibliotecas de ativos de processo que são comumente criadas em iniciativas de melhorias em processos de software. Por conta da dificuldade de se encontrar uma biblioteca de ativos de processos suficientemente boa e completa para servir de guia para consolidação das boas idéias, decidiu-se pela realização de um estudo de *benchmarking*. Este estudo foi estruturado em seis fases: iniciação do projeto, planejamento, identificação dos parceiros de *benchmarking*, coleta dos dados, análise dos dados e comunicação dos resultados.

3.7 Considerações finais

O processo de *benchmarking* deve ser executado continuamente, o que implica na repetição dos passos no sentido de nunca estagnar e dar oportunidade às organizações de evoluir progressivamente, conceito esse que se assemelha bastante ao PDCA de DEMING (DEMING, 1982).

Quando o *benchmarking* é bem estruturado, se revela um poderoso meio de aprendizagem capaz de impulsionar organizações que o realizem. Contudo, caso seja feito de forma errada, exigindo resultados imediatos e fornecendo recursos escassos, desconsiderando a estratégia e a cultura corporativa, acabam por transformar as “melhores práticas” em fracassos eminentes, gerando prejuízos aos seus implementadores.

O que faz do *benchmarking* um sucesso é o uso pleno de seus processos e também o apoio da gerência. De fato, o envolvimento da gerência é essencial, porque o *benchmarking* afeta diretamente as metas com as quais a gerência se compromete. Esse envolvimento é essencial para garantir a compreensão dos requisitos para os resultados do *benchmarking*.

O próximo capítulo abordará o tema de *benchmarking* aplicado ao contexto de projetos de implementação de melhorias em processos, além de descrever o contexto de aplicação dessa abordagem.

CAPÍTULO 4 – UMA PROPOSTA PARA *BENCHMARKING* DE INICIATIVAS DE MELHORIAS EM PROCESSOS DE SOFTWARE

Este capítulo apresenta a proposta para a condução de iniciativas de melhorias em processos de software que engloba a realização de atividades de benchmarking, que é o objeto desta dissertação.

4.1 Introdução

Devido à grande competitividade do mercado de software atual, as organizações que fornecem produtos e serviços de software de forma eficaz e com a qualidade desejada pelo cliente se destacam frente àquelas que ainda desenvolvem software de maneira artesanal, *ad hoc*. Neste contexto, tornou-se muito freqüente a implementação de iniciativas de melhorias em processos de software (HUMPHREY, 1989). No entanto, nem todas as organizações obtêm sucesso nessas empreitadas. NIAZI (2006) destaca que cerca de 70% dessas iniciativas não são bem sucedidas. Uma das razões que levam a este resultado é a implementação caótica das iniciativas de melhorias em processos, sem a aplicação de métodos e técnicas já disseminadas na disciplina de gerência de projetos (BIRK e PFAHL, 2002).

Apesar da existência dos modelos de referência para a implementação de melhorias em processos de software como, por exemplo, o CMMI (SEI, 2006) e o MPS.BR (SOFTEX, 2007), esses modelos apenas indicam “o que” devem conter os processos, mas não especificam “como” implementar essas atividades, o que contribui ainda mais para a dificuldade ou mesmo o insucesso das iniciativas de melhorias (MINGHUI *et al.*, 2004).

Além disso, a implementação de programas de melhorias em processos requer profissionais com grande conhecimento especializado, sendo que esses profissionais não são facilmente encontrados dentro das organizações desenvolvedoras de software. Isto faz com que estas recorram aos serviços de organizações de consultoria em melhoria de processos de software, que possuem profissionais com as qualificações desejadas e com experiência na implementação de iniciativas de melhoria. Essas organizações de consultoria

também enfrentam dificuldades para realizar a sua tarefa devido às especificidades encontradas nas organizações de software que as contratam (MONTONI, 2007).

Neste contexto, é importante analisar formas de identificar, capturar, processar e disseminar as melhores práticas na implementação de melhorias em processos de software, com o intuito de auxiliar as organizações de consultoria em melhoria de processos de software na tomada de decisão relativa à estratégia de implementação de melhorias mais adequadas em cada contexto.

Este trabalho tem por objetivo apoiar uma organização de consultoria em melhoria de processos de software na tomada de decisão durante a definição, planejamento e monitoração dos projetos de melhoria de processos de software, através da utilização de dados de projetos passados e/ou correntes.

Nos capítulos anteriores, foi descrita a revisão da literatura relativa aos temas mais importantes para este trabalho e que formaram a base para a definição da abordagem proposta. Como citado anteriormente, a proposta nesta dissertação está contida em um trabalho mais amplo, com uma abordagem completa de apoio a condução de iniciativas de melhorias em processos de software. Este capítulo apresenta a abordagem completa, bem como a proposta desta dissertação para *benchmarking* de iniciativas de implementação de melhorias em processos de software. Como resultado desta proposta tem-se um conjunto de requisitos a serem atendidos por um apoio ferramental, descrito no próximo capítulo.

A seção 4.2 apresenta a abordagem completa para condução de projetos de melhoria de processos de software por empresas de consultoria em processos de software. A seção 4.3 apresenta a proposta deste trabalho, para apoio às atividades de *benchmarking* no contexto da abordagem descrita na seção anterior. A seção 4.4 apresenta os processos de apoio ao *benchmarking* de iniciativas de melhorias em processos de software, e, por fim, a seção 4.5 apresenta as considerações finais do capítulo.

4.2 Uma Abordagem para Condução de Iniciativas de Melhorias em Processos de Software

Com o intuito de apoiar as organizações de consultoria em melhoria de processos de software na gerência das iniciativas de melhoria sob sua coordenação, foi definida uma abordagem para a condução dessas iniciativas de melhoria, a qual será apoiada por uma infra-estrutura computacional baseada em um ambiente customizado de gestão de conhecimento centrado em processo (MONTONI, 2007).

Antes de iniciar o detalhamento da abordagem para a condução de iniciativas de melhorias em processos de software, é preciso entender alguns conceitos amplamente utilizados, e que, portanto, devem ser explicados. O primeiro conceito é o de processo padrão de melhoria de processos de software. Uma organização de consultoria em implementação de melhoria de processos de software pode possuir um ou mais processos padrão de melhoria, no qual as fases e sub-processos são definidos em alto nível. Os processos padrão de melhoria podem ser construídos com base em abordagens de implementação de melhorias em processos de software. Um exemplo seria um processo padrão definido com as fases e sub-processos encontrados no IDEAL¹ (GREMBA e MYERS, 1997) ou no SPI-KM² (SANTOS *et al.*, 2007). O processo padrão de implementação de melhorias em processos de software será utilizado para guiar as atividades de melhoria, e alimentará as bases de estratégias de implementação e de *benchmarking*. Suas atividades serão tomadas como marcos de referência para a coleta de dados para fins de comparação.

O segundo conceito é o de estratégia de implementação de melhorias em processos de software. Uma instituição de consultoria em implementação de melhorias em processos de software pode possuir uma ou mais estratégias de implementação de melhorias aplicáveis a cada um dos sub-processos de um ou mais processos padrão de melhoria de processos de software. As estratégias especializam e adaptam um sub-processo do processo padrão de melhorias, definindo as atividades a serem realizadas em cada sub-processo e gerando o processo definido para o projeto de melhorias.

Uma estratégia de implementação de melhorias em processos de software aplicada a um sub-processo do processo padrão de melhoria define: (i) as atividades realizadas no sub-processo e suas descrições, como os artefatos requeridos e produzidos pelas atividades e os perfis normalmente responsáveis por suas execuções; (ii) os contextos encontrados nas organizações nos quais a estratégia pode ser aplicada; (iii) os conhecimentos necessários para a realização das atividades definidas pela estratégia; (iv) os recursos necessários para a realização das atividades definidas pela estratégia; (v) os riscos associados com a estratégia selecionada e observados em projetos anteriores; (vi) as ações de mitigação e contingência

¹ IDEAL – Acrônimo para *Initiating* (Iniciar), *Diagnosing* (Diagnosticar), *Establishing* (Estabelecer), *Acting* (Executar) e *Learning* (Aprender). Trata-se de um modelo de programa organizacional de melhoria de processos de software que serve como guia para iniciar, planejar e implementar ações de melhoria (GREMBA e MYERS, 1997).

² SPI-KM – Acrônimo para *Software Process Improvement approach supported by Knowledge Management*. Trata-se de uma abordagem para implantação de melhorias em processos de software apoiada por gerência de conhecimento, baseada nos modelos de referência para melhoria de processos CMMI e MPS.BR (SANTOS *et al.*, 2007). Esta abordagem é utilizada pela COPPE/UFRJ em seus projetos de implementação de melhorias em processos de software.

relacionadas com os riscos associados com a estratégia selecionada e os resultados obtidos nas suas execuções em projetos anteriores; (vii) o tempo e esforço normalmente demandados pelas atividades definidas pela estratégia, com base nas suas execuções em projetos anteriores; e (viii) a comunicação necessária para o sucesso das atividades definidas pela estratégia, baseada nas suas execuções em projetos anteriores.

Como exemplo de estratégia, podemos pensar no sub-processo “Estabelecer a Infra-estrutura” definido na abordagem de implementação de melhorias em processos de software IDEAL (GREMBA e MYERS, 1997), na fase Iniciar. Dependendo do contexto encontrado nas organizações, as instituições de consultoria em implementação de melhoria de processos de software podem optar por fornecer ou indicar infra-estruturas diferentes. Caso as organizações não possuam o apoio ferramental necessário para alcançar as melhorias desejadas, por exemplo, as instituições de consultoria podem sugerir várias ferramentas, de acordo com o contexto encontrado nas organizações, como facilidade de aprendizado, ferramentas já utilizadas pelas organizações e necessidades demandadas pelas melhorias almejadas. Dependendo do apoio ferramental sugerido, as instituições de consultoria em implementação de melhoria de processos de software podem realizar atividades específicas do apoio sugerido. Um exemplo é a utilização da Estação TABA (MONTONI *et al.*, 2006; MONTONI *et al.*, 2007) fornecida pela COPPE/UFRJ para as organizações que a contratam para consultoria em processos de software. A COPPE/UFRJ precisa realizar customizações na Estação TABA para adaptá-la ao contexto das organizações que a utilizam, realizando atividades relacionadas com estas adaptações, como o cadastro do processo padrão de desenvolvimento das organizações na ferramenta.

A arquitetura da abordagem de condução de iniciativas de melhorias em processos de software (MONTONI, 2007) é composta de diferentes componentes com objetivos e responsabilidades bem definidos, como pode ser observado na Figura 4.1 explicada a seguir.

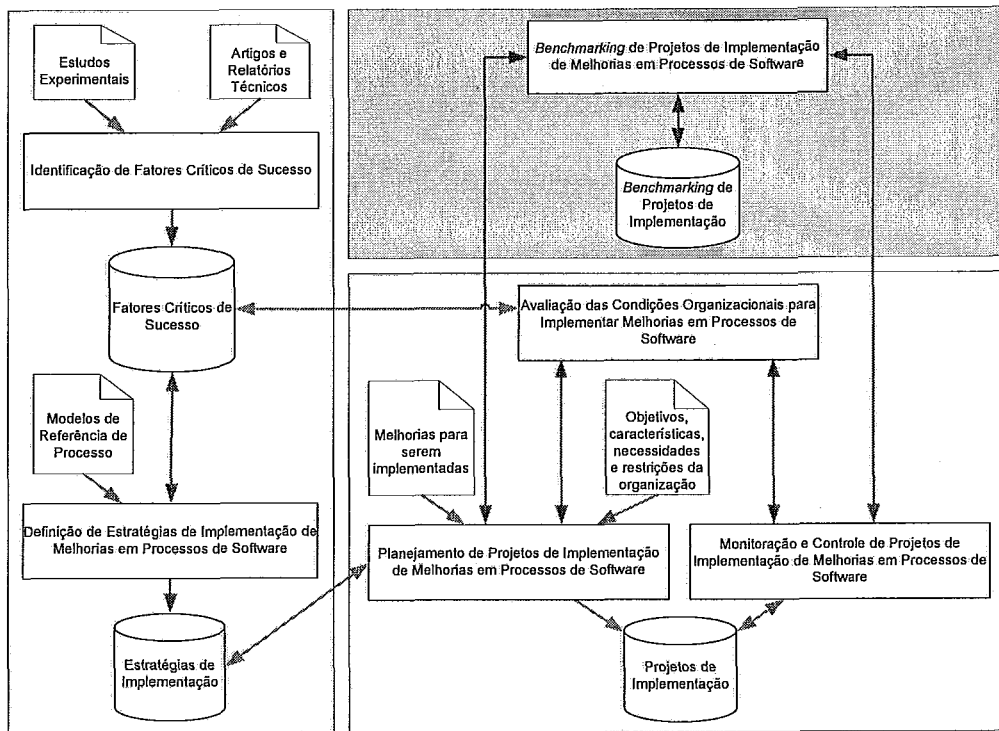


Figura 4.1 – Componentes da Abordagem para Condução de Iniciativas de Melhorias em Processos de Software (MONTONI *et al.*, 2008b).

O primeiro grupo de componentes aborda os conhecimentos sobre (i) os fatores críticos de sucesso que influenciam os resultados das iniciativas de melhoria e (ii) as estratégias de condução de implementação de melhorias de processos de software. Os componentes deste grupo são:

- **Identificação de Fatores Críticos de Sucesso:** este componente apóia a aquisição de conhecimento sobre os fatores críticos de sucesso que influenciam iniciativas de melhoria. O componente deve capturar conhecimento de diversas fontes, como artigos, relatórios técnicos e estudos experimentais e auxiliar a aplicação da metodologia de identificação de fatores críticos de sucesso.
- **Definição de Estratégias de Implementação de Melhorias em Processos de Software:** este componente apóia a aquisição de conhecimento sobre as estratégias de implementação de melhorias de processos de software em contextos específicos, considerando as características encontradas nas organizações que favorecem ou dificultam as iniciativas de melhoria. O componente deve permitir que diferentes instituições de consultoria armazenem (i) suas estratégias definidas com base em diferentes abordagens e modelos de referência de melhoria de processos, (ii) o contexto no qual estas estratégias são aplicáveis e (iii) diretrizes para adaptação das estratégias para apoiar a condução de iniciativas específicas de melhoria. Além

disso, o componente deve permitir que uma organização de consultoria em melhoria de processos mantenha as suas estratégias de implementação com base no conhecimento adquirido a partir da condução de iniciativas anteriores de melhoria.

O segundo grupo de componentes aborda a aplicação da técnica de *benchmarking* em projetos de melhoria de processos de software. Este grupo tem um componente único:

- *Benchmarking* de Projetos de Implementação de Melhorias em Processo de Software: este componente apóia a identificação de melhores práticas de uma instituição de consultoria em melhoria de processos e a aplicação destas práticas em novas iniciativas de melhoria. Além disso, este componente mantém conhecimento sobre o desempenho dos projetos de melhoria de processos de software e os contextos encontrados nas organizações favoráveis ou não às iniciativas de melhoria, auxiliando as instituições de consultoria a determinarem o desempenho esperado em projetos com contextos similares e a avaliarem o desempenho de seus projetos atuais com relação a outros projetos que possuam contextos similares.

O terceiro grupo de componentes aborda a gerência e avaliação dos projetos de implementação de melhorias. Os componentes deste grupo são:

- Planejamento de Projetos de Implementação de Melhorias em Processos de Software: este componente apóia o planejamento de projetos de implementação de melhorias em processos de software, auxiliando a seleção e adaptação de estratégias de implementação com base nas características encontradas nas organizações que favorecem ou dificultam as iniciativas de melhoria.
- Monitoração e Controle de Projetos de Implementação de Melhorias em Processos de Software: este componente apóia a monitoração das atividades executadas no contexto dos projetos de implementação de melhorias em processos de software, através de dados quantitativos e qualitativos, permitindo a identificação do desempenho dos projetos. O componente também apóia a monitoração e o tratamento, de forma contínua, dos fatores que podem influenciar significativamente o resultado das iniciativas.
- Avaliação das Condições Organizacionais para Implementar Melhorias em processo de Software: este componente apóia a identificação dos fatores que podem influenciar o sucesso de iniciativas de melhoria com base nas características

encontradas nas organizações, no início da iniciativa de melhoria e durante a sua condução.

Esta dissertação concentrará esforços no segundo grupo de componentes descrito e destacado na Figura 4.1.. O primeiro grupo de componentes, bem como a definição completa da abordagem é objeto de uma tese de doutorado (MONTONI, 2007). O terceiro grupo de componentes foi objeto de uma dissertação de mestrado (CERDEIRAL, 2008).

4.3 Uma Proposta para *Benchmarking* de Iniciativas de Melhorias em Processos de Software

A idéia básica por trás do *benchmarking* aplicado aos projetos de melhoria de processos de software é que, para cada novo projeto, há outros projetos que se assemelham àquele e que já foram realizados ou ainda estão sendo realizados, de forma que muitas das informações derivadas destes projetos podem servir de base para o planejamento ou para a análise do desempenho de outros projetos que estão em fase de iniciação ou em execução.

Esses projetos de implementação de melhorias precisam ser continuamente monitorados para garantir que estão dentro das metas estabelecidas, permitindo a identificação de desvios e a tomada de ações corretivas em tempo hábil para o correto andamento do projeto. Portanto, uma organização de consultoria em melhoria de processos de software que deseje planejar e implementar projetos de melhoria de processos da melhor forma possível pode se valer das informações de projetos similares antes e durante a execução de novos projetos, com o objetivo de aumentar as chances de sucesso desses projetos.

A intenção é que, ao se realizar as atividades de *benchmarking*, seja feita uma análise dos resultados de forma a fornecer apoio à tomada de decisão em tempo hábil, para se corrigir eventuais discrepâncias no desempenho dos projetos. Estas atividades devem ser implementadas de forma contínua, nas várias etapas do ciclo de vida do projeto de melhorias, visando a manter ou melhorar o desempenho das atividades que compõem o processo padrão de melhorias em processos de software. A análise de *benchmarking* pode ser realizada, basicamente, em dois cenários: (i) durante o planejamento do projeto de implementação de melhorias, e (ii) durante a monitoração e controle deste projeto. As

Figuras 4.2 e 4.3 descrevem estes dois cenários para execução das atividades de *benchmarking*.

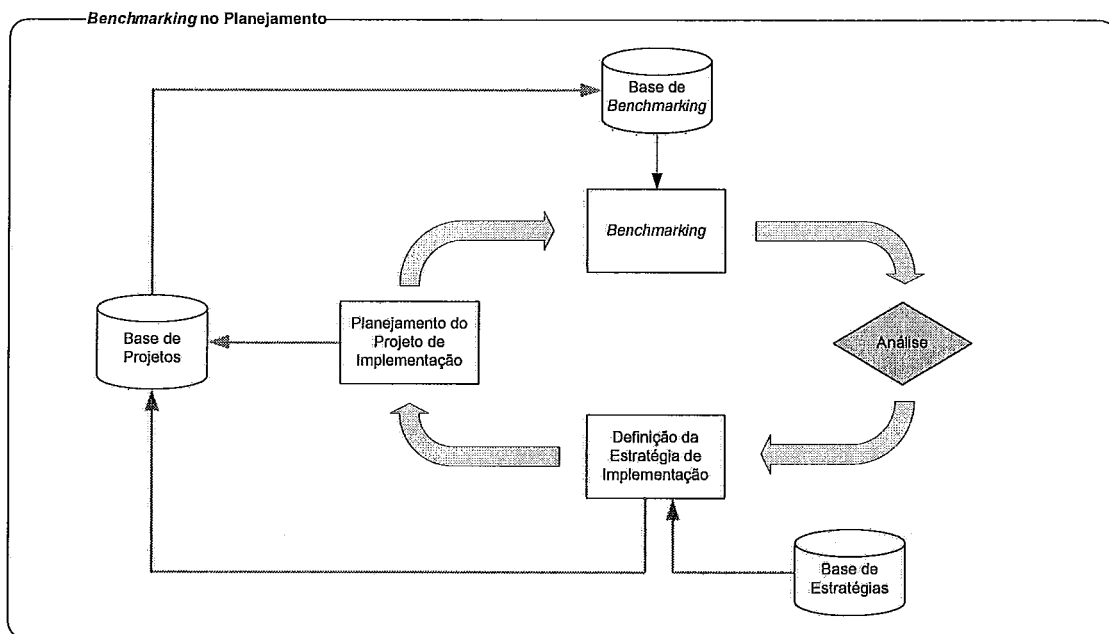


Figura 4.2 – Cenário de execução de atividades de *Benchmarking* durante o planejamento dos projetos de implementação de melhorias em processos de software.

O planejamento do projeto de melhoria se inicia na definição da estratégia de implementação a ser adotada. Nesta etapa, o processo padrão de implementação é adaptado segundo as características e restrições do projeto de melhoria e da organização onde o projeto será implementado. Em seguida, na etapa de planejamento do projeto de implementação serão planejados os recursos necessários para a execução da estratégia definida na etapa anterior. O componente Planejamento de Projetos de Implementação de Melhorias em Processos de Software identificado na seção anterior é responsável pela definição do planejamento inicial dos recursos, tempo, cronograma e custo dos projetos de melhoria, resultando no plano para gerenciamento do projeto de implementação de melhoria (CERDEIRAL, 2008). Após o planejamento, o gerente do projeto de melhoria poderá executar *benchmarking*, a fim de comparar os planos elaborados com os dados reais de projetos passados ou correntes. Essa atividade é seguida de uma etapa de análise em que decisões são tomadas baseadas na comparação com projetos reais. Essas decisões podem envolver, por exemplo, a redefinição ou re-planejamento da estratégia de implementação. Isso possibilitará que o planejamento seja o mais realista e preciso possível, já que será baseado nas experiências e no desempenho da própria organização de consultoria, sempre levando em conta os dados de contexto do projeto em questão.

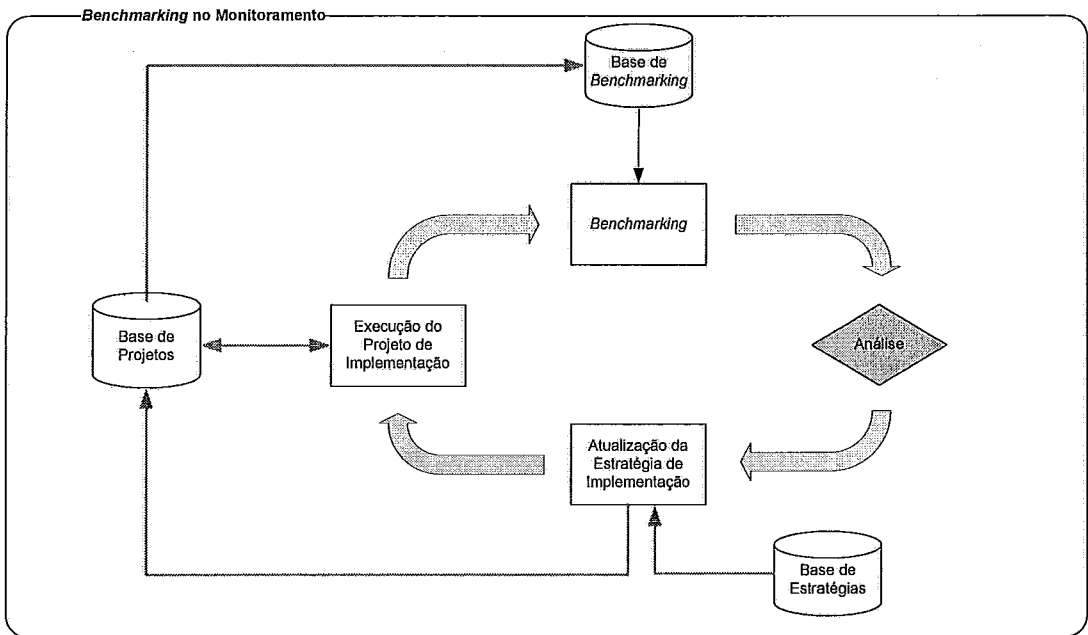


Figura 4.3 – Cenário de execução de atividades de *Benchmarking* durante o monitoramento dos projetos de implementação de melhorias em processos de software.

Durante a execução dos projetos de implementação de melhorias, a equipe de implementação executa a estratégia definida e o gerente do projeto de melhoria realiza a monitoração dos planos estabelecidos. Essa monitoração é apoiada por atividades de *benchmarking*, onde são feitas comparações dos dados de execução do projeto em questão com dados reais de outros projetos similares. Isso permitirá, ao gerente, determinar o desempenho do projeto em relação à base histórica e às *baselines* de desempenho da organização de consultoria. Esta etapa é, mais uma vez, seguida por uma etapa de análise, onde o gerente do projeto de melhoria poderá tomar as ações cabíveis para manter ou melhorar o desempenho do projeto em questão. Essas ações podem envolver, por exemplo, o replanejamento ou redefinição da estratégia de implementação.

Durante o planejamento e execução dos projetos de implementação de melhorias em processos de software, a base de dados sobre esses projetos será automaticamente alimentada e, periodicamente, esta base deverá alimentar a base de *benchmarking*, formando *baselines* de desempenho da organização de consultoria em melhoria de processos. Os exercícios de *benchmarking* da organização serão realizados tendo como referência os dados contidos nesta base de *benchmarking*.

É muito importante que os dados contidos na base de *benchmarking* sejam confiáveis e atualizados, pois, caso contrário, pode-se gerar interpretações errôneas e causar resultados desastrosos para os projetos de implementação que os utilizarem como base de

comparação. Portanto, esta base deve ser constantemente gerenciada, realizando análises dos dados e remoção de informações desatualizadas ou inúteis, além de se manter um bom controle das informações nela inseridas, impedindo que dados não relevantes sejam registrados. Essa gerência é apoiada por relatórios de avaliação post-mortem elaborados nos projetos de implementação de melhorias em processos de software. Neste relatório os gerentes devem informar o que é válido e o que não é válido para fins de *benchmarking* e a justificativa para a decisão. Estes dados de contexto serão levados em consideração nas atividades de filtragem e empacotamento dos dados da base de *benchmarking*.

Outro aspecto de igual importância diz respeito à correta e criteriosa caracterização do contexto em que esses projetos são realizados. Esta caracterização requer a descrição dos projetos de implementação de melhorias em processos de software em relação a uma variedade de critérios de caracterização, que torne possível classificar projetos com características e objetivos semelhantes. Isto deverá facilitar a reutilização dos dados de projetos anteriores e, também, a comparação do desempenho entre os projetos e a realização de melhores projeções para os projetos em andamento, evitando que sejam realizadas comparações entre itens que não são comparáveis. As comparações a serem realizadas através da base de *benchmarking* deverão contar com um fator de similaridade que determina o grau de semelhança entre os projetos a serem retornados da base e o projeto que está sendo comparado. Esse grau de similaridade deverá ser calculado com base nos diversos critérios de caracterização, determinados ainda durante as etapas de planejamento do projeto de implementação de melhoria de processos de software. Os critérios de caracterização serão determinados para a base de *benchmarking* e instanciados para cada projeto específico. Novos critérios poderão ser incluídos, a cargo da organização de consultoria de melhorias em processos de software. MONTONI (2008b) deverá tratar do preenchimento desses critérios em seu trabalho de doutorado.

Semelhante às equipes de gerência existentes em outras áreas da indústria, as equipes de implementação de melhorias em processos de software, objetivando o sucesso de suas empreitadas, devem levar em consideração os fatores de sucesso e as melhores práticas obtidas em outras iniciativas. Sendo assim, é preciso conhecer as falhas e sucessos dos projetos de implementação de melhorias em processos de software já realizados e fazer com que esse conhecimento seja reutilizado em iniciativas futuras. O *benchmarking* provê um apoio a esta necessidade. A utilização de fatores de sucesso e de outras informações úteis, extraídas de implementações já realizadas, na definição de estratégias futuras, pode aumentar a probabilidade de sucesso do programa de melhorias na medida em que se

consegue enxergar os pontos fortes e os pontos fracos existentes do projeto de implementação corrente.

A presença dos fatores de sucesso nas organizações e nos projetos de melhoria também é monitorada pelo componente de Monitoração e Controle de Projetos de Implementação de Melhorias em Processos de Software, que sugere ações corretivas para tratar os fatores que representam dificuldades para o sucesso dos projetos de melhoria (CERDEIRAL, 2008).

Sendo assim, as atividades de *benchmarking* devem apoiar, além da definição das estratégias e do planejamento dos projetos de melhoria, a monitoração desses projetos. Portanto, o componente de *benchmarking* de iniciativas de melhorias em processos de software deverá ser composto por uma base de experiências sobre os projetos de implementação de melhorias, a base de *benchmarking*, por um ferramental de apoio aos gerentes de projetos de implementação de melhorias para que estes possam realizar atividades de *benchmarking* de seus projetos em todas as etapas necessárias do ciclo de vida dos mesmos, e também por um ferramental de apoio à gerência das experiências contidas na base de *benchmarking*. Basicamente, este componente deverá atuar apoiando o acúmulo de conhecimento gerado pelos projetos em andamento, empacotando as informações e disseminando o conhecimento através da realização das atividades de *benchmarking*.

4.4 Processos de Apoio ao *Benchmarking* de Iniciativas de Melhorias em Processos de Software

Devido à falta de padronização na realização de projetos de implementação de melhorias em processos de software e ao fato de que as boas práticas que surgem em organizações terceiras estão impregnadas e contextualizadas com os fatores e as circunstâncias específicas daquela organização, torna-se muito difícil a internalização das informações que poderiam ser extraídas de um *benchmarking* externo. Por conta desta dificuldade, optou-se pelo foco no *benchmarking* interno para esta dissertação. Sendo assim, a abordagem proposta apoiará as atividades de *benchmarking* no contexto de organizações de consultoria em melhoria de processos de software, permitindo que estas realizem comparações de desempenho entre seus projetos de implementação de melhorias passados e/ou correntes.

Baseada na opção pelo *benchmarking* interno, a estratégia de *benchmarking* mais adequada ao contexto deste trabalho é ajustar o nível de desempenho segundo um nível

interno desejável. Pretende-se ajustar o nível de desempenho das implementações correntes, tendo como base a referência estabelecida por outros projetos de implementação de melhorias similares e que mereceram destaque pelo desempenho superior.

O comparativo entre os processos pesquisados, apresentado no capítulo 3, foi analisado com o foco no *benchmarking* interno e no contexto de projetos de implementação de melhorias em processos de software conduzidos por organizações de consultoria.

Em relação à fase de planejamento, a opção pelo *benchmarking* interno faz com que os marcos de referência sejam extraídos do processo padrão de implementação de melhorias em processos de software de projetos passados e/ou correntes da própria organização de consultoria. Deve-se planejar a periodicidade com que a base de projetos é analisada e a base de experiências é alimentada com dados dos projetos. Na periodicidade estabelecida, os dados dos projetos são coletados manual ou automaticamente, formando *baselines* periódicas de desempenho da organização úteis para comparar os efeitos de mudanças realizadas nas estratégias de implementação adotadas nos projetos de melhoria.

Já em relação à fase de análise, o gerente do projeto de melhoria deve, durante o ciclo de vida do projeto de melhoria, consultar a base de *benchmarking* em buscas de projetos similares. Essa busca é efetuada através da seleção dos critérios de similaridade que deverão ser comparados com os critérios de caracterização dos projetos de implementação. Os projetos cuja caracterização atender aos critérios de similaridade selecionados são então retornados para comparação. Em seguida, o gerente do projeto de melhoria deverá analisar e comparar seu projeto com as *baselines* de desempenho dos projetos retornados que já foram estabelecidas para a organização de consultoria. Esta análise deverá ser realizada sob um determinado ponto de vista de interesse, como por exemplo, tempo, esforço ou custo. Isto permitirá responder perguntas como: *o que, dentro do projeto de implementação de melhorias em processos de software, não está adequado em relação aos projetos tidos como similares?* Esses pontos de vista são aqui denominados “Perspectivas de Visualização”. As lacunas de desempenho entre os projetos de implementação de melhorias são determinadas em relação às perspectivas de visualização definidas para a base de *benchmarking* em questão. Os comparativos entre os projetos similares e o projeto corrente são determinados sob o ponto de vista das perspectivas de visualização. Essas perspectivas também são utilizadas na alimentação periódica da base de *benchmarking*, ou seja, periodicamente a base dos projetos é analisada sob o ponto de vista de cada perspectiva de visualização determinada para a base de *benchmarking*, alimentando esta base e formando as *baselines* de desempenho relativas a cada uma das perspectivas. A Figura 4.4 apresenta algumas das possíveis

perspectivas de visualização a serem apoiadas pela abordagem. Essas perspectivas podem mudar de organização para organização e novas perspectivas podem ser criadas por necessidade da organização.

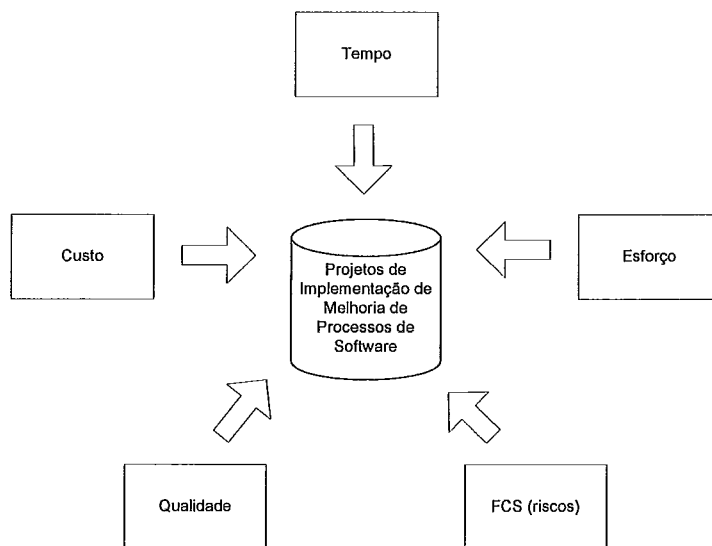


Figura 4.4 – Perspectivas de visualização

Não é foco deste trabalho apoiar as fases de Integração e Ação, apresentadas na tabela comparativa dos processo de *benchmarking*, no Capítulo 3. Toda a parte de tomada de decisão relativa aos resultados da análise dos dados do *benchmarking* está fora do escopo deste trabalho. Esta tomada de decisão será baseada no planejamento estratégico do programa de melhorias e está no escopo do principal trabalho relacionado à abordagem completa (MONTONI, 2007). A realização das modificações nos projetos pode ser acompanhada por novas etapas de *benchmarking*, cujo objetivo é quantificar o desempenho do projeto de melhoria em relação às novas estimativas e a eficácia dos planos de ação estabelecidos.

É importante observar que a fase de Continuação foi incluída no comparativo entre os processos descritos no Capítulo 3, para indicar que as atividades de *benchmarking* devem ser executadas continuamente, visando melhorar constantemente o desempenho dos projetos de implementação de melhorias em processos de software. As etapas cíclicas de *benchmarking* permitirão calibrar cada vez mais *baselines* de desempenho, fazendo com que futuras empreitadas possam se beneficiar de estimativas mais precisas, aumentando assim suas chances de sucesso.

A partir da análise comparativa dos processos e das necessidades e características da abordagem proposta, foram definidos dois processos de apoio ao *benchmarking* de projeto

de implementação de melhorias em processos de software: o Processo de Gerência da Base de *Benchmarking* e o Processo de Realização do *Benchmarking*. Devido à necessidade de gerência da própria base de *benchmarking* e dos dados que devem ser inseridos nesta base, foram definidas atividades de gerência, que visam: (i) à determinação da estrutura da base de *benchmarking*, através da inclusão/exclusão de perspectivas de visualização e itens de caracterização dos projetos, (ii) ao controle da alimentação da base de *benchmarking* através da coleta de dados da base de projetos e formação das *baselines* de desempenho, (iii) à gerência das informações contidas nestas *baselines* de desempenho, de forma a permitir a remoção de informações desatualizadas ou inúteis e manter um bom controle das informações que são inseridas na base, e (iv) à gerência das próprias *baselines* de desempenho, através da inclusão, exclusão e liberação dessas *baselines* para consulta. A Figura 4.5 apresenta o processo de apoio à gerência da base de *benchmarking*. Já a Figura 4.6 apresenta o processo de realização de *benchmarking*. A notação utilizada para representar os processos é de AGUIAR (2004). Os processos serão descritos nas seções subseqüentes.

4.4.1 Processo de Gerência da Base de *Benchmarking*

Este processo apóia a criação e a gerência da base de *benchmarking* da organização de consultoria em melhoria de processos de software. Essa gerência envolve a criação e configuração da base de *benchmarking*, bem como a manutenção dos dados nela inseridos.

Atividade	Gerenciar Base de <i>Benchmarking</i>
Tarefa	Criar Base de <i>Benchmarking</i>
Descrição	O gerente do <i>benchmarking</i> cria uma base de <i>benchmarking</i> para apoiar as atividades de <i>benchmarking</i> de projetos de implementação de melhorias em processos de software. A definição da base de <i>benchmarking</i> envolve a definição das perspectivas de visualização e dos itens de caracterização dos projetos que farão parte da mesma. As perspectivas referem-se aos tipos de dados coletados dos projetos e servirão de base para apoiar a análise de desempenho dos projetos. Os itens de caracterização serão utilizados para caracterizar os projetos de forma uniforme e permitir a comparação de similaridades.
Critério de Entrada	-
Critério de Saída	Ter-se criado a base de <i>benchmarking</i>
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	-
Pré-tarefa	-
Pós-tarefa	Especificar a Coleta de Dados
Artefatos Requeridos	-
Artefatos Produzidos	Base de <i>Benchmarking</i>

Processo de Gerência da Base de *Benchmarking*

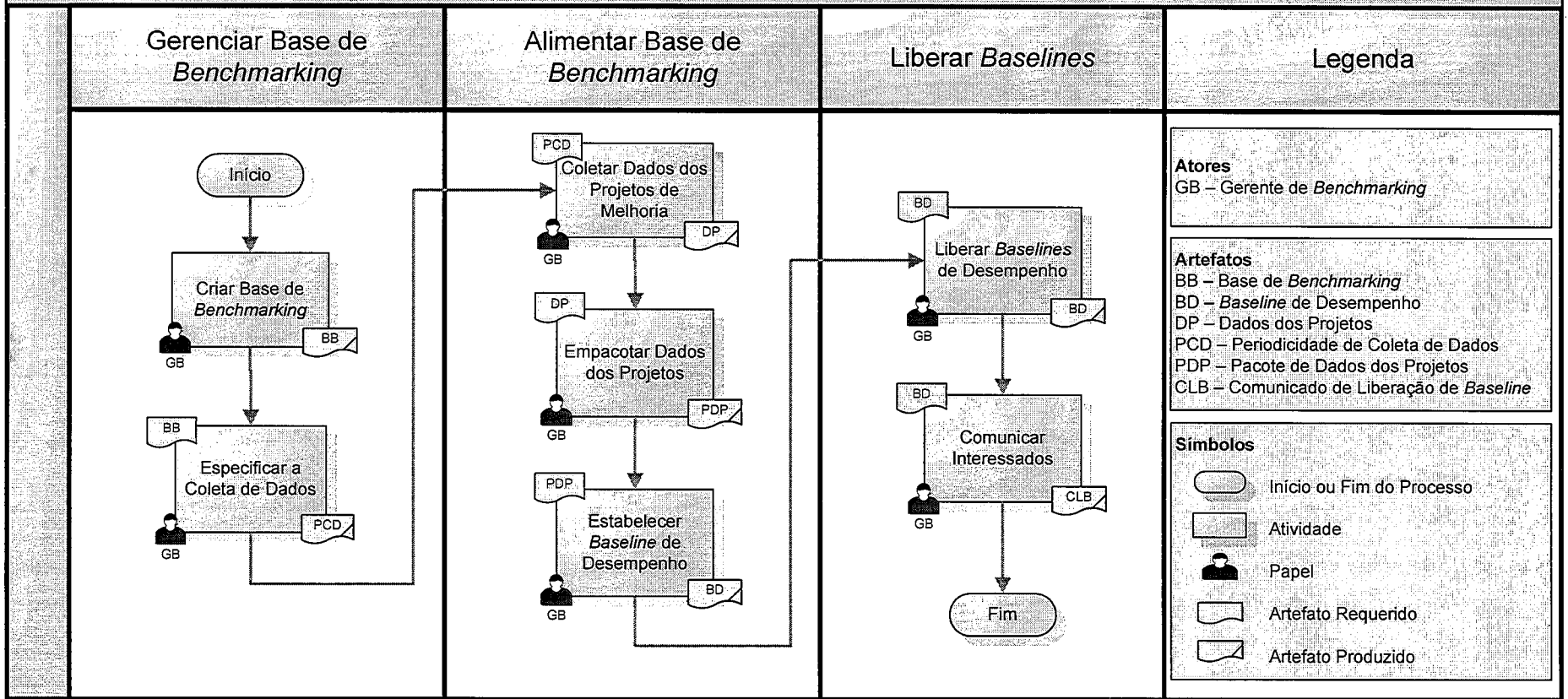


Figura 4.5 – Processo de Gerência da Base de *Benchmarking*

Processo de Realização de *Benchmarking*

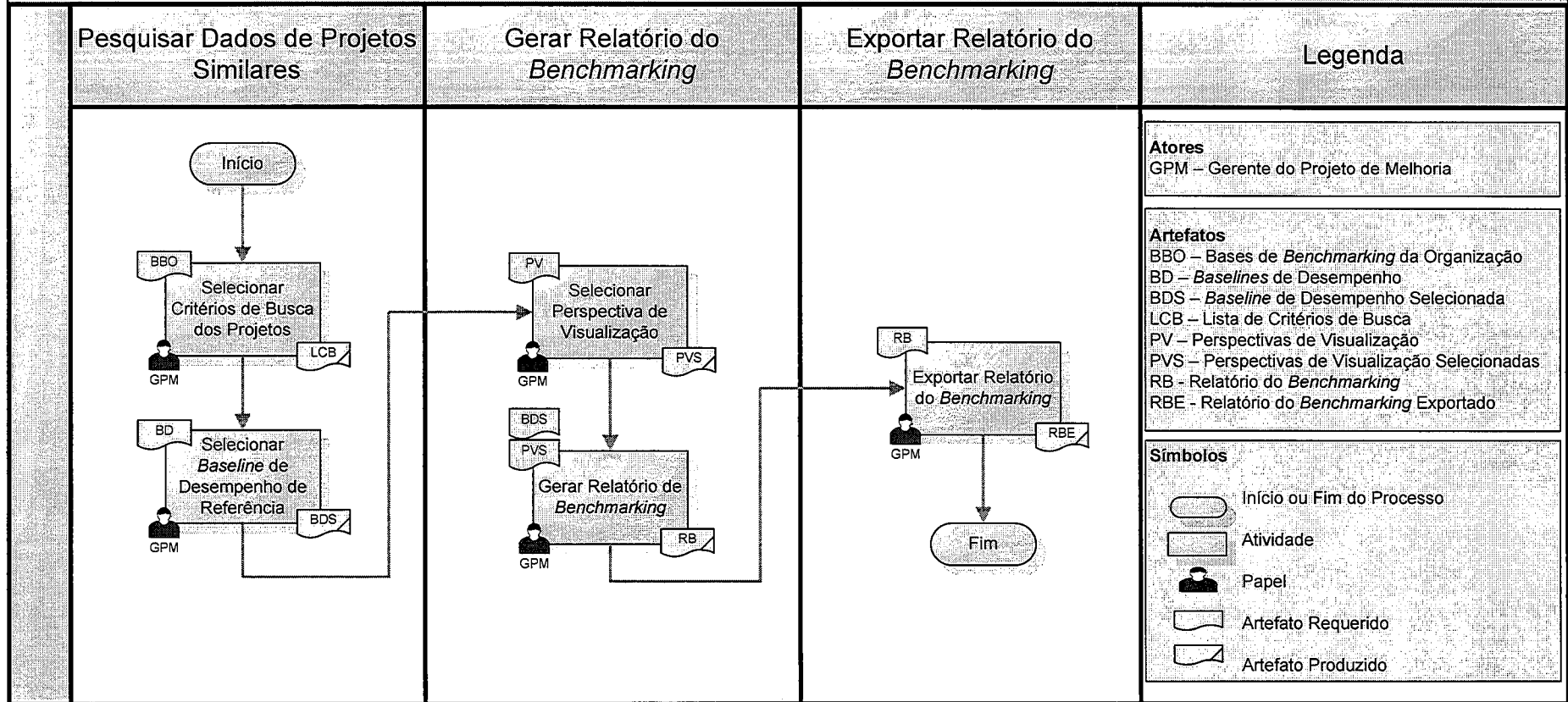


Figura 4.6 – Processo de Apoio à Realização de *Benchmarking*

Tarefa	Especificar a Coleta de Dados
Descrição	Nesta atividade, o gerente de benchmarking determina a periodicidade e a forma de coleta de dados dos projetos de implementação de melhorias para a formação das baselines de desempenho da organização.
Critério de Entrada	Ter-se criado a base de benchmarking
Critério de Saída	Ter-se especificado a forma de coleta dos dados
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	-
Pré-tarefa	Criar Base de <i>Benchmarking</i>
Pós-tarefa	Coletar Dados dos Projetos de Melhoria
Artefatos Requeridos	Base de <i>Benchmarking</i>
Artefatos Produzidos	Base de <i>Benchmarking</i> (configurada)

Atividade	Alimentar Base de <i>Benchmarking</i>
Tarefa	Coletar Dados dos Projetos de Melhoria
Descrição	Na periodicidade estabelecida, os dados dos projetos de implementação de melhorias em processos de software são coletados e alimentam a base de <i>benchmarking</i> da organização. Os dados coletados são referentes a cada perspectiva de visualização selecionada para compor a base de <i>benchmarking</i> .
Critério de Entrada	Ter chegado o momento da coleta de dados determinada na periodicidade estabelecida.
Critério de Saída	Ter-se coletado os dados dos projetos
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	-
Pré-tarefa	Determinar Periodicidade de Coleta de Dados
Pós-tarefa	Empacotar Dados dos Projetos
Artefatos Requeridos	Periodicidade de Coleta de Dados e Projetos de Implementação de Melhorias
Artefatos Produzidos	Dados dos Projetos
Tarefa	Empacotar Dados dos Projetos
Descrição	Após a coleta dos dados dos projetos, o gerente de <i>benchmarking</i> empacota os dados coletados. Este empacotamento envolve: (i) a avaliação dos dados dos projetos, (ii) os ajustes nos dados e a eliminação de dados não-relevantes ou que não caracterizam de forma adequada os projetos de melhoria, e (iii) o registro de informações de contexto dos dados e dos projetos.
Critério de Entrada	Ter-se coletado os dados dos projetos de melhoria
Critério de Saída	Ter-se a avaliação, o ajuste, a eliminação e o registro de contexto dos dados coletados.
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	Gerentes dos Projetos de Melhoria
Pré-tarefa	Coletar Dados dos Projetos de Melhoria
Pós-tarefa	Estabelecer <i>Baseline</i> de Desempenho
Artefatos Requeridos	Dados dos Projetos
Artefatos Produzidos	Pacote de Dados dos Projetos
Tarefa	Estabelecer <i>Baseline</i> de Desempenho
Descrição	Após o empacotamento dos dados dos projetos, o gerente de <i>benchmarking</i> estabelece e/ou atualiza uma <i>baseline</i> de desempenho

	com os dados dos projetos. As <i>baselines</i> são criadas com base nas características dos projetos, por exemplo, podem ser criadas <i>baselines</i> específicas com dados de projetos que utilizaram estratégias similares de implementação de melhorias ou <i>baselines</i> específicas para projetos de melhoria executados com base em modelo cooperado ou específico ³ de implementação. Os dados empacotados de um projeto podem pertencer a uma ou mais <i>baselines</i> de desempenho.
Critério de Entrada	Ter-se empacotado os dados dos projetos de melhoria
Critério de Saída	Ter-se estabelecida a <i>baseline</i> de desempenho referente aos dados coletados.
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	Equipes dos Projetos
Pré-tarefa	Empacotar Dados dos Projetos
Pós-tarefa	Liberar <i>Baseline</i> de Desempenho
Artefatos Requeridos	Pacote de Dados dos Projetos
Artefatos Produzidos	<i>Baselines</i> de Desempenho criadas e/ou atualizadas

Atividade	Liberar <i>Baselines</i>
Tarefa	Liberar <i>Baseline</i> de Desempenho
Descrição	O gerente de <i>benchmarking</i> libera a <i>baseline</i> de desempenho para utilização nas atividades de <i>benchmarking</i> de projetos de implementação de melhorias em processos de software.
Critério de Entrada	Ter-se estabelecido a <i>baseline</i> de desempenho dos projetos de melhoria
Critério de Saída	<i>Baseline</i> de desempenho liberada para utilização pelos gerentes de projetos de melhoria
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	-
Pré-tarefa	Estabelecer <i>Baseline</i> de Desempenho
Pós-tarefa	Comunicar Interessados
Artefatos Requeridos	<i>Baseline</i> de Desempenho
Artefatos Produzidos	<i>Baseline</i> de Desempenho liberada
Tarefa	Comunicar Interessados
Descrição	Os gerentes de projetos de melhoria são comunicados da liberação de uma nova <i>baseline</i> de desempenho.
Critério de Entrada	Ter-se liberado a <i>baseline</i> de desempenho
Critério de Saída	Comunicado de liberação de <i>baseline</i> de desempenho enviado aos gerentes de projetos de implementação de melhorias
Responsáveis	Gerente de <i>Benchmarking</i>
Participantes	Gerentes dos Projetos de Melhoria
Pré-tarefa	Liberar <i>Baseline</i> de Desempenho
Pós-tarefa	-
Artefatos Requeridos	<i>Baseline</i> de Desempenho
Artefatos Produzidos	Comunicado de Liberação de <i>Baseline</i>

³ Os modelos Cooperado e Específico são formas de implementação do Modelo MPS.BR. O modelo cooperado é próprio para pequenas e médias empresas que desejam compartilhar custos e esforços na implementação. No modelo Específico, cada empresa negocia com uma instituição implementadora e deve arcar com os custos da implementação (SOFTEX, 2007).

4.4.2 Processo de Realização de *Benchmarking*

Este processo apóia a execução do *benchmarking* durante o ciclo de vida dos projetos. Esta execução pode ocorrer a qualquer momento, a critério do gerente do projeto de melhoria.

Atividade	Pesquisar Dados de Projetos Similares
Tarefa	Selecionar Critérios de Busca dos Projetos
Descrição	O gerente do projeto de melhoria seleciona critérios de busca desejados que deverão ser levados em consideração para seleção dos projetos similares.
Critério de Entrada	Ter-se optado por executar <i>benchmarking</i> do projeto de melhoria.
Critério de Saída	Ter-se selecionado os critérios de similaridade
Responsáveis	Gerente do Projeto de Melhoria
Participantes	-
Pré-tarefa	-
Pós-tarefa	Selecionar <i>Baselines</i> de Desempenho de Referência
Artefatos Requeridos	Base de <i>Benchmarking</i> da Organização.
Artefatos Produzidos	Lista de Critérios de Busca.
Tarefa	Selecionar <i>Baseline</i> de Desempenho de Referência
Descrição	A partir das <i>baselines</i> de desempenho da base de <i>benchmarking</i> da organização, o gerente do projeto de melhoria seleciona a <i>baseline</i> de desempenho que deve ser utilizadas como referência para comparação do desempenho do projeto corrente.
Critério de Entrada	Ter-se selecionado os critérios de similaridade.
Critério de Saída	Ter-se selecionado a <i>baseline</i> de desempenho a ser utilizada para comparação
Responsáveis	Gerente do Projeto de Melhoria
Participantes	-
Pré-tarefa	Selecionar Base de <i>Benchmarking</i>
Artefatos Requeridos	<i>Baselines</i> de desempenho da base de <i>benchmarking</i> selecionada.
Artefatos Produzidos	<i>Baselines</i> de Desempenho de Referência selecionadas.
Pós-tarefa	Selecionar Perspectiva de Visualização

Atividade	Gerar Relatório do <i>Benchmarking</i>
Tarefa	Selecionar Perspectiva de Visualização
Descrição	O gerente do projeto de implementação seleciona uma ou mais perspectivas de visualização da base de <i>benchmarking</i> .
Critério de Entrada	Ter-se selecionado a <i>baseline</i> de desempenho a ser utilizada para comparação
Critério de Saída	Perspectiva de visualização selecionada
Responsáveis	Gerente do Projeto de Melhoria
Participantes	-
Pré-tarefa	Selecionar <i>Baselines</i> de Desempenho de Referência
Pós-tarefa	Gerar Relatório do <i>Benchmarking</i> .
Artefatos Requeridos	Perspectivas de visualização da base de <i>benchmarking</i> selecionada
Artefatos Produzidos	Perspectivas de visualização selecionadas

Tarefa	Gerar Relatório do <i>Benchmarking</i>
Descrição	O gerente de melhoria gera o relatório do <i>benchmarking</i> com base nas <i>baselines</i> de desempenho de referência e nas perspectivas de visualização selecionadas.
Critério de Entrada	Ter-se selecionado a Perspectiva de Visualização
Critério de Saída	Relatório do <i>Benchmarking</i> gerado.
Responsáveis	Gerente do Projeto de Melhoria
Participantes	-
Pré-tarefa	Selecionar Perspectiva de Visualização
Pós-tarefa	Exportar Relatório de <i>Benchmarking</i>
Artefatos Requeridos	<i>Baselines</i> de desempenho de referência e Perspectivas de visualização selecionadas
Artefatos Produzidos	Relatório do <i>Benchmarking</i>

Atividade	Exportar Relatório de <i>Benchmarking</i>
Descrição	O gerente do projeto de implementação exporta os dados do relatório para formatos de documento ou planilha eletrônica.
Critério de Entrada	Relatório do <i>Benchmarking</i> gerado
Critério de Saída	Relatório do <i>Benchmarking</i> exportado
Responsáveis	Gerente do Projeto de Melhoria
Participantes	-
Pré-tarefa	Gerar Relatório do <i>Benchmarking</i>
Pós-tarefa	-
Artefatos Requeridos	Relatório do <i>Benchmarking</i>
Artefatos Produzidos	Relatório de <i>Benchmarking</i> Exportado

Os processos descritos acima guiarão os esforços de *benchmarking* no contexto de projetos de implementação de melhorias em processos de software, sob o ponto de vista das organizações de consultoria em melhoria de processos.

4.5 Considerações finais

Neste capítulo, foi apresentada a proposta deste trabalho para apoiar as atividades de *benchmarking* de projetos de implementação de melhorias em processos de software, bem como o contexto maior no qual essa abordagem está inserida. Os processos de apoio à abordagem de *benchmarking* proposta também foram descritos.

O ferramental de apoio à abordagem proposta de *benchmarking* e o ambiente no qual este ferramental está inserido serão abordados no próximo capítulo.

CAPÍTULO 5 – UMA FERRAMENTA DE APOIO ÀS ATIVIDADES DE *BENCHMARKING* DE INICIATIVAS DE MELHORIAS EM PROCESSOS DE SOFTWARE

Este capítulo apresenta o ferramental de software implementado para apoiar a execução das atividades de benchmarking em iniciativas de melhorias em processos de software, bem como seus requisitos de nível mais alto e o ambiente customizável de gestão de conhecimento no qual o apoio ferramental está inserido.

5.1 Introdução

No capítulo anterior, foi apresentada a abordagem proposta para *benchmarking* de iniciativas de melhorias em processos de software. Esta abordagem é composta por dois processos que definem as atividades a serem executadas durante a realização de exercícios de *benchmarking* nestas iniciativas. A partir da análise da abordagem e das atividades que compõem os processos definidos, foram identificadas as necessidades e os requisitos que devem ser atendidos pelo ferramental de software que apoiará a execução dessas atividades. O ferramental de apoio às atividades de *benchmarking* de iniciativas de melhorias em processos de software foi desenvolvido utilizando o ambiente de gestão do conhecimento CORE-KM (GALOTTA *et al.*, 2004).

Este capítulo apresenta o apoio ferramental para as atividades de *benchmarking* de iniciativas de melhoria de processos de software, seus requisitos e o ambiente de gestão do conhecimento utilizado na implementação. A seção 5.2 apresenta o CORE-KM, ambiente de gestão do conhecimento no qual o apoio ferramental foi implementado. A seção 5.3 apresenta os requisitos elicitados e refinados para o ferramental proposto. A seção 5.4 apresenta o apoio ferramental propriamente dito, e, por fim, a seção 5.5 apresenta as considerações finais deste capítulo.

5.2 CORE-KM: Um ambiente de Gestão de Conhecimento

O CORE-KM (*Customizable Organizational Resources Environment with Knowledge Management*) é um ambiente customizável de gestão de conhecimento para diferentes organizações que apóia a execução de seus processos de negócio (GALOTTA *et al.*, 2004). Cada customização constitui um ambiente diferenciado contendo as características particulares de uma organização.

O ambiente CORE-KM tem como objetivos: (i) auxiliar o engenheiro de conhecimento na especificação do ambiente de gestão de conhecimento mais adequado à organização; (ii) auxiliar o engenheiro de conhecimento na customização do ambiente de gestão de conhecimento; (iii) permitir ao engenheiro de software a integração, ao ambiente customizado, de ferramentas desenvolvidas para apoiar processos específicos da organização; e (iv) permitir às organizações a execução do ambiente customizado independentemente do ambiente CORE-KM. Estas funcionalidades apóiam a definição, customização e execução de ambientes de gerência de conhecimento.

Para atender aos objetivos, foi definida uma infra-estrutura composta de um conjunto de atividades típicas de gestão de conhecimento, que representam as funcionalidades do ambiente customizado e se comunicam através de uma memória organizacional central. Juntamente com as ferramentas do ambiente, as interfaces das atividades formam uma rede de conhecimento baseada na intranet da organização. A Figura 5.1 apresenta a infra-estrutura de atividades de gestão de conhecimento do ambiente CORE-KM.

Cada ambiente customizado tem sua própria memória organizacional que será formada pelas bases selecionadas do CORE-KM e as bases pertencentes às ferramentas específicas desenvolvidas para a organização. As bases organizacionais recebem o conteúdo específico da organização após a customização.

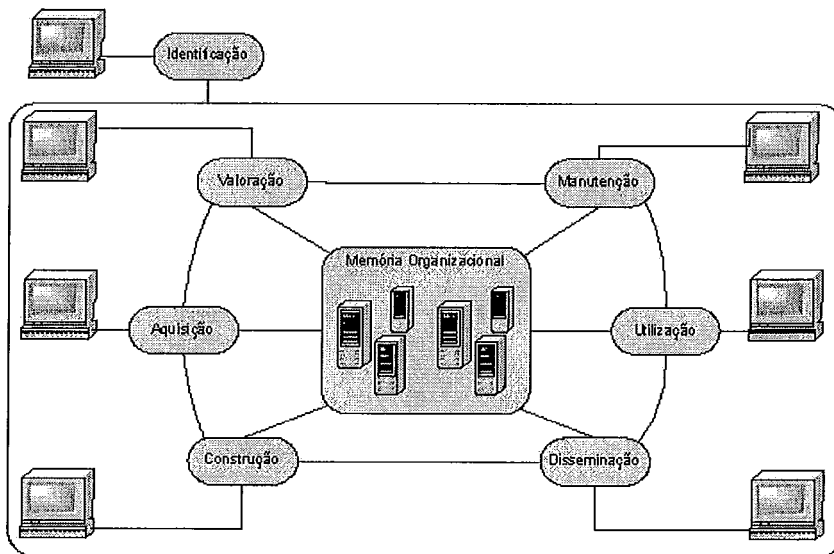


Figura 5.1 – Representação esquemática da infra-estrutura do ambiente CORE-KM (GALOTTA *et al.*, 2004).

A Figura 5.2 apresenta outra visão, com a representação dos ambientes customizados a partir do ambiente CORE-KM. O núcleo representa a infra-estrutura base que contém as ferramentas genéricas e as ferramentas que apóiam a execução de atividades típicas do processo de gerência do conhecimento. Ao customizar o ambiente, estas ferramentas podem, opcionalmente, ser incorporadas e disponibilizadas para as organizações sem conteúdo de conhecimentos, para que as empresas preencham com conteúdo próprio. Os ambientes customizados são representados pelo núcleo somado a um segmento lateral, sendo este último formado pelas ferramentas desenvolvidas para apoiar a execução de processos específicos da organização. Tanto o núcleo quanto os ambientes são formados por três camadas:

- **Interface** – é a primeira camada e contém a interface das ferramentas. No caso do núcleo, contém a interface das ferramentas genéricas e das ferramentas que apóiam a execução do processo de gerência de conhecimento. No caso dos ambientes customizados, contém a interface das ferramentas específicas.
- **Objeto** – se refere ao assunto principal do qual está se tratando. No caso do núcleo, o objeto é a memória organizacional, e no caso dos ambientes customizados varia de acordo com o objetivo da gerência do conhecimento na organização.
- **Banco de Dados** – contém os repositórios e é responsável pela persistência dos dados, informações e conhecimento.

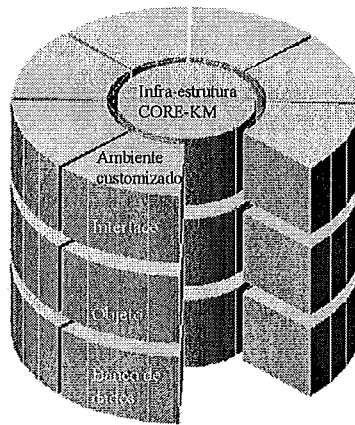


Figura 5.2 – Infra-estrutura em camadas do ambiente CORE-KM (GALOTTA *et al.*, 2004).

A arquitetura do ambiente é composta por quatro camadas, a saber:

- **Camada cliente** – compreende o programa de visualização da aplicação através de uma rede de Internet ou Intranet. O programa de visualização padrão utilizado é o Microsoft Internet Explorer.
- **Camada de apresentação** – compreende o conjunto de páginas para gerar conteúdo para a camada cliente. Esta camada foi implementada utilizando a linguagem HTML e C#. O servidor da camada de apresentação é o Microsoft Internet Information Server e o *framework* .NET.
- **Camada da lógica da aplicação** – compreende as regras do negócio, as quais determinam de que maneira os dados serão utilizados. Essa camada foi implementada utilizando a linguagem de programação C# e o paradigma orientado a objetos.
- **Camada de dados** – compreende a base de armazenamento de dados e os procedimentos de manipulação desses dados. O servidor da camada de dados é o Microsoft SQL Server Express e o NHibernate como *framework* de mapeamento objeto-relacional.

O ferramental de apoio às atividades de *benchmarking* de iniciativas de melhorias em processos de software foi desenvolvido na infra-estrutura apresentada acima. Na próxima seção, serão apresentados os requisitos deste apoio ferramental.

5.3 Requisitos do Apoio Ferramental às Atividades de *Benchmarking* em Iniciativas de Melhorias em Processos de Software

A partir da análise das necessidades e restrições da abordagem proposta para *benchmarking* de projetos de implementação de melhorias em processos de software, foi estabelecido um conjunto de requisitos que deverão ser atendidos pelo ferramental de apoio desenvolvido. O conjunto de requisitos de mais alto nível é listado a seguir.

- Permitir a comparação dos projetos de implementação de melhorias em processos de software, utilizando a abordagem de *benchmarking*, de forma a permitir a análise dos pontos fortes e pontos fracos destes projetos.
- Permitir aos gerentes dos projetos de melhorias em processos de software monitorar a evolução do desempenho/estado dos projetos e das mudanças neste desempenho durante a implementação do projeto de melhoria, de forma a permitir a tomada de ação em tempo hábil para garantir um andamento eficiente e eficaz dos projetos, mantendo ou melhorando o seu desempenho.
- Identificar atributos internos e externos relevantes aos projetos de melhoria de processos de software que possibilitem monitorar efetivamente estes projetos, minimizando os riscos, e permitindo a identificação de relações causais entre suas características.
- Permitir a especificação dos parâmetros de comparação dos projetos de implementação de melhorias em processos de software que devem ser considerados relevantes para fins de *benchmarking* destes projetos.
- Permitir a manutenção dos dados referentes ao desempenho dos projetos de implementação de melhorias em processos de software e a formação de uma base histórica destes projetos.

Estes requisitos de mais alto nível (Requisitos de Cliente) foram refinados no nível de requisitos funcionais, de forma a permitir identificar cada funcionalidade que deve compor o apoio ferramental para as atividades de *benchmarking*. O refinamento dos requisitos do cliente em requisitos funcionais é apresentado a seguir:

- *Manter a base de benchmarking de iniciativas de melhoria de processos de software:* Possibilitar a criação, configuração e manutenção de um repositório para o armazenamento dos dados sobre o desempenho e melhores práticas relacionadas aos projetos de

implementação de melhorias em processos de software realizados por organizações de consultoria em melhoria de processos.

- *Coletar os dados dos projetos de implementação de melhorias em processos de software:* Permitir a coleta dos dados de desempenho dos projetos de implementação de melhorias em processos de software com uma periodicidade estabelecida.
- *Manter as perspectivas de visualização da base de benchmarking:* Permitir a manutenção de atributos internos e externos relevantes para as comparações a serem realizadas entre os projetos de implementação de melhorias em processos de software, em forma de perspectivas de visualização da base de *benchmarking*.
- *Manter os critérios de caracterização dos projetos de implementação de melhorias em processos de software:* Permitir a manutenção dos itens de caracterização dos projetos de implementação de melhorias em processos de software.
- *Manter os projetos de implementação de melhorias em processos de software da base de benchmarking:* Permitir a manutenção dos dados sobre os projetos de implementação de melhorias em processos de software que fazem parte da base de *benchmarking* da organização de consultoria em melhoria de processos de software.
- *Manter as baselines de desempenho dos projetos de implementação de melhorias em processos de software:* Permitir a criação e manutenção de *baselines* de desempenho dos projetos de implementação de melhorias em processos de software, tendo como base os atributos internos e externos relevantes para esses projetos.
- *Manter a caracterização dos projetos de implementação de melhorias em processos de software:* Permitir a caracterização dos projetos de implementação de melhorias em processos de software em relação aos itens de caracterização definidos para a base de *benchmarking*.
- *Selecionar os critérios de caracterização relevantes para a busca de projetos de implementação de melhorias em processos de software:* Permitir a seleção de itens de caracterização que devem estar presentes nos projetos de implementação de melhorias em processos de software a serem pesquisados para fins de *benchmarking*.
- *Pesquisar dados dos projetos similares:* Permitir a consulta dos dados referentes ao desempenho dos projetos de implementação de melhorias em processos de software sob uma determinada perspectiva de visualização, levando em consideração os itens de caracterização selecionados e que devem estar presentes nos projetos pesquisados, a fim de garantir a similaridade entre esses projetos.

- *Gerar relatórios detalhados de benchmarking dos projetos de implementação de melhorias em processos de software:* Possibilitar a geração de relatórios detalhados de cada exercício de *benchmarking* executado sobre os projetos que constam da base de *benchmarking*.
- *Exportar os dados do benchmarking:* Permitir a exportação dos dados extraídos do *benchmarking* para posterior utilização.

A Figura 5.3 apresenta os requisitos do cliente e requisitos funcionais descritos acima, bem como os relacionamentos entre eles.

As descrições detalhadas dos requisitos do cliente e dos requisitos funcionais podem ser consultadas no ANEXO I, que também contém o refinamento destes requisitos em casos de uso.

5.4 Uma Ferramenta de Apoio às Atividades de *Benchmarking* de Iniciativas de Melhorias em Processos de Software

Para apoiar as organizações de consultoria na execução das atividades de *benchmarking* de iniciativas de melhoria foi desenvolvido um apoio ferramental de software. Este apoio foi desenvolvido com base nos requisitos definidos na seção 5.3, de forma integrada ao *framework* do ambiente CORE-KM. A implementação da ferramenta foi realizada seguindo um processo definido, adaptado do processo padrão da Área de Qualidade de Software do LENS (Laboratório de Engenharia de Software da COPPE/UFRJ), compatível com o nível E do MPS.BR. A modelagem da ferramenta de apoio contendo o refinamento desde os requisitos de cliente até os casos de uso encontra-se no ANEXO I.

As funcionalidades derivadas dos requisitos funcionais estabelecidos para a ferramenta foram subdivididas levando-se em consideração os processos definidos no capítulo 4, ou seja, em um módulo de gerência da base de *benchmarking* e em outro módulo de realização do *benchmarking* de projetos de melhorias em processos de software propriamente dito. Estas funcionalidades visam a apoiar as atividades dos processos definidos.

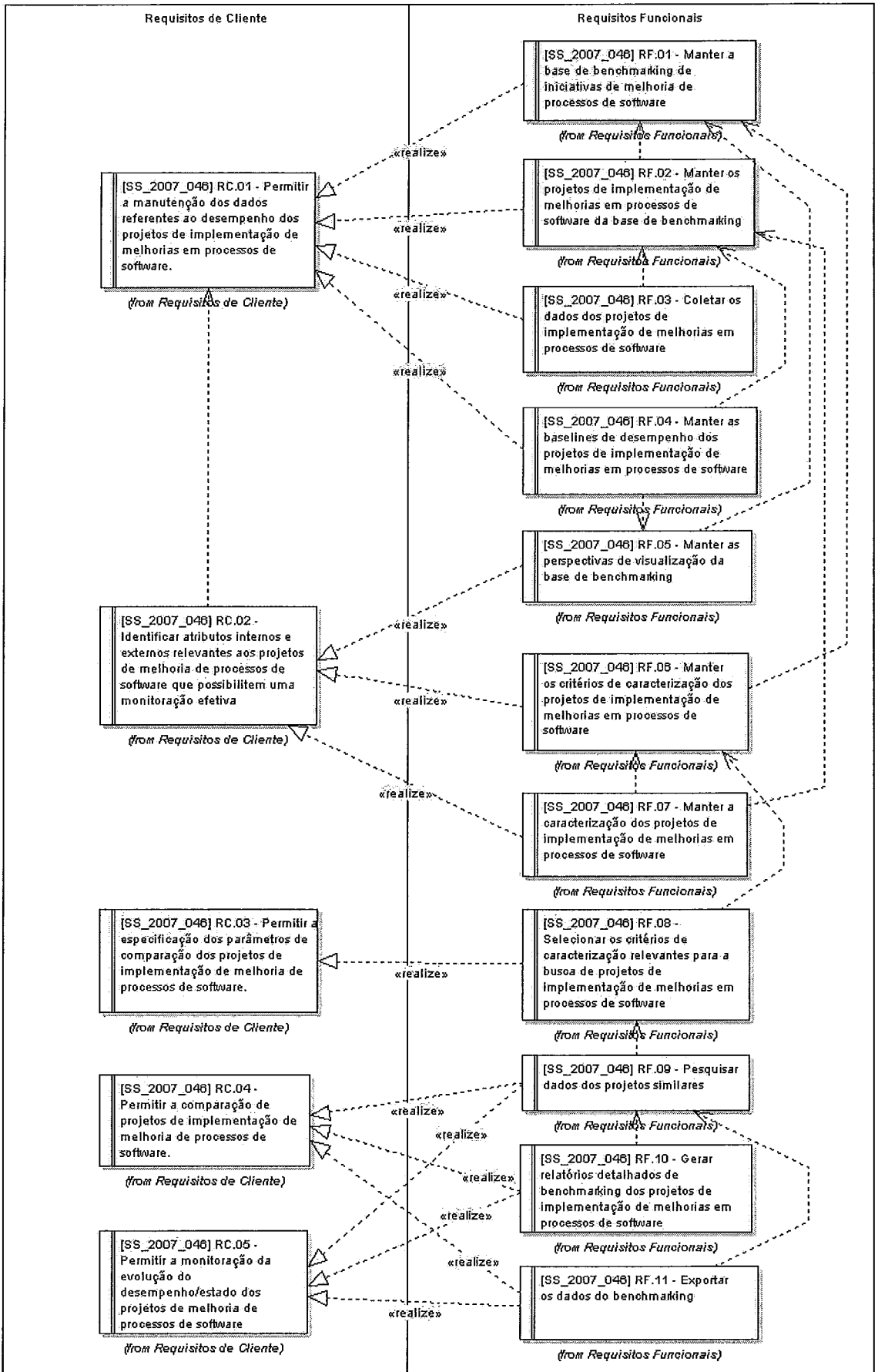


Figura 5.3 – Requisitos de Cliente X Requisitos Funcionais.

5.4.1 Gerência da Base de *Benchmarking*

Antes de realizar as atividades de execução de *benchmarking* das iniciativas de implementação de melhorias em processos de software, é necessário que a organização de consultoria defina a base onde armazenará os dados sobre o desempenho dos projetos de implementação. Para apoiar a organização de consultoria, um ferramental de apoio ao processo de gerência da base de *benchmarking*, descrito no capítulo 4, foi implementado. Durante a execução da atividade inicial deste processo, “Criar Base de *Benchmarking*”, a organização de consultoria cria uma base de armazenamento dos dados referentes ao desempenho dos seus projetos de implementação de melhoria. Esta atividade envolve ainda a criação das perspectivas de visualização e dos itens de caracterização dos projetos de implementação que serão inseridos na base de *benchmarking*. A Figura 5.4 mostra um exemplo da funcionalidade de apoio a estas atividades iniciais do processo de gerência da base de *benchmarking*. Este exemplo mostra a criação de uma base de *benchmarking* para a II (Instituição Implementadora) COPPE/UFRJ.

The screenshot shows a web application interface for creating a benchmarking base. At the top, it identifies the user as 'Ana Regina da Rocha' and the environment as 'Ambiente de Apoio à Instituição Implementadora COPPE/UFRJ'. The main heading is 'Base de Benchmarking'. Below this, there are navigation links: 'Gerenciar Base de Benchmarking' and 'Executar Benchmarking'. A note states: 'Os campos marcados com (*) são de preenchimento obrigatório'. The main content area is titled 'Manter Base de Benchmarking' and contains a form with several tabs: 'Geral', 'Itens de Caracterização', 'Perspectivas de Visualização', 'Projetos', and 'Baselines de Desempenho'. The 'Geral' tab is selected. The form fields are: 'Nome:' with the value 'Base de Benchmarking da II COPPE/UFRJ'; 'Descrição:' with a text area containing 'Esta é a Base de Benchmarking da Instituição Implementadora COPPE/UFRJ. Esta base contém os dados dos projetos de implementação de melhorias em processos que utilizaram o processo padrão de implementação SPI-KM.'; 'Coleta dos Dados' section with 'Tipo de Coleta:' set to 'Manual' and 'Períodicidade da Coleta dos Dados:' set to 'Mensal'. At the bottom of the form are 'Confirmar' and 'Cancelar' buttons.

Figura 5.4 – Apoio à atividade “Criar Base de *Benchmarking*”

Para caracterizar de forma uniforme os projetos que devem fazer parte da base de *benchmarking*, é necessário descrever os projetos de implementação de melhorias em processos de software com relação a uma variedade de características de contexto, de forma que torne possível classificar projetos com características e objetivos semelhantes. Isto deverá facilitar a reutilização dos dados de projetos anteriores e, também, a comparação do desempenho entre os projetos e a realização de melhores projeções para os projetos em andamento, evitando que sejam realizadas comparações entre itens que não

são comparáveis. Durante a atividade de criação da base de *benchmarking*, devem-se definir os itens de caracterização dos projetos a serem coletados. Estes itens de caracterização serão utilizados também como parâmetros de comparação entre os projetos. A Figura 5.5 mostra exemplos de itens de caracterização de projetos que foram inseridos na base de *benchmarking* da II COPPE/UFRJ e também a criação de um novo item de caracterização “Infra-Estrutura de Apoio à Implementação”. Este item servirá para caracterizar os projetos em termos da infra-estrutura utilizada na implementação, como por exemplo, “Somente TABA”, “TABA + MS Project”, “TABA + PWA”, “Team Foundation”. Estas opções são definidas como itens da escala utilizada para caracterização do item. Sendo assim, torna-se possível a criação e configuração completa de outros itens de caracterização de projetos. A Figura 5.6 apresenta a criação da escala de caracterização do item “Infra-Estrutura de Apoio à Implementação”.

The screenshot displays the COPPE/UFRJ Benchmarking System interface. At the top, it shows the logo and name of the system, the user name 'Usuário: Ana Regina da Rocha', and navigation links 'Início', 'Suporte', and 'Sair'. Below this, there are sections for 'Base de Benchmarking' and 'Manter Base de Benchmarking'. A navigation bar includes 'Geral', 'Itens de Caracterização', 'Perspectivas de Visualização', 'Projetos', and 'Bases de Desempenho'. The main area contains a table of existing items:

	Nome	Escala
<input type="checkbox"/>	Área de Implementação	Área
<input type="checkbox"/>	Certificação COPPE	Participação
<input type="checkbox"/>	Certificação ISO	Participação
<input type="checkbox"/>	Certificação ITIL	Participação
<input type="checkbox"/>	Classificação da Instituição	Classe de Instituição
<input type="checkbox"/>	Complexidade dos Projetos	Grav
<input type="checkbox"/>	Idade da Unidade Organizacional Avaliada	Idade
<input type="checkbox"/>	Infra-estrutura de apoio à implementação	Infra-Estrutura de Apoio

Below the table, there is a form for creating a new item. The form fields are:

- Nome:** Infra-estrutura de apoio à implementação
- Descrição:** Infra-estrutura de apoio utilizada na implementação
- Escala:** Infra-Estrutura de Apoio

At the bottom of the form, there are 'Confirmar' and 'Cancelar' buttons.

Figura 5.5 – Itens de Caracterização dos Projetos de *Benchmarking*

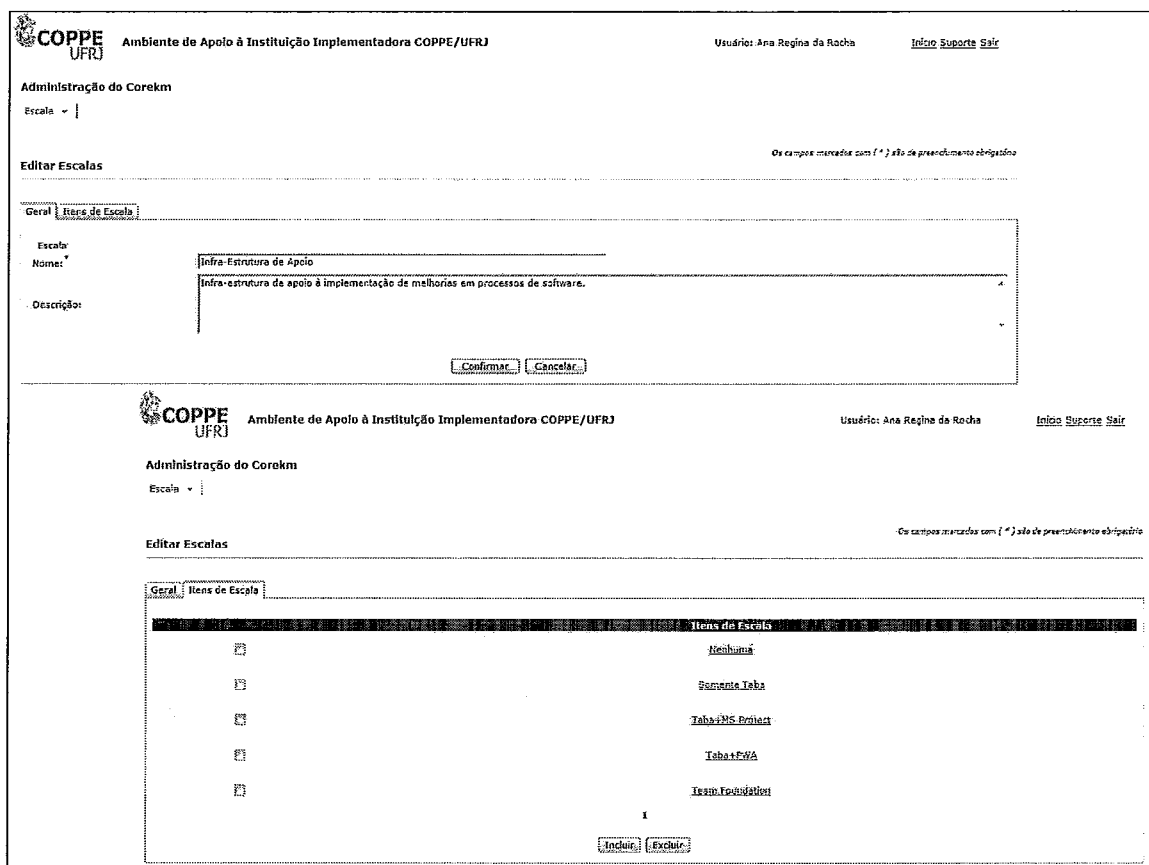


Figura 5.6 – Escalas de Itens de Caracterização

Ainda durante a atividade de criação da base de *benchmarking*, é necessário especificar as perspectivas de visualização da base. Estas perspectivas se referem aos tipos de dados coletados dos projetos e que servirão de base para apoiar a análise de desempenho. A Figura 5.7 mostra as perspectivas de visualização definidas para a base de *benchmarking* da II COPPE/UFRJ e também um exemplo da criação da perspectiva de visualização “Esforço (HH) de Acompanhamento com Perfil ‘Mestrando’”. Esta perspectiva indica o esforço, em homens-hora, gastos na implementação dos projetos de melhoria, com um recurso que se encaixe no perfil “Mestrando” da Instituição Implementadora. As perspectivas podem ser associadas a um determinado marco de execução de um processo específico, representado por uma atividade deste processo. Isto permite explicitar que o cálculo do valor relativo àquela perspectiva no contexto dos projetos é feito até aquele marco especificado.

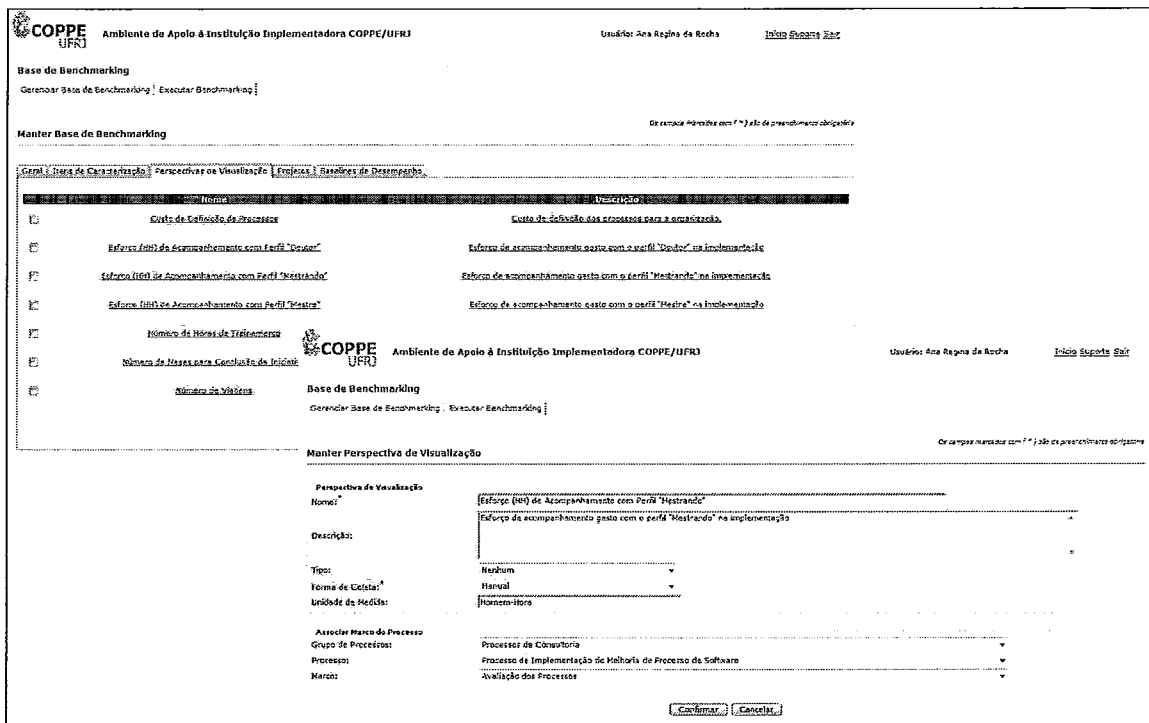


Figura 5.7 – Perspectivas de Visualização da Base de *Benchmarking*

Após criar a base de *benchmarking*, a organização de consultoria deve “Especificar a Coleta dos Dados”. Nesta atividade, é definido o intervalo de tempo no qual os dados dos projetos serão coletados e inseridos na base de *benchmarking*, além da forma de coleta destes dados, ou seja, “Manual” ou “Automática”. Devido às restrições de tempo e esforço, a coleta automática não foi implementada, mas o apoio ferramental já possui apoio à coleta automática dos dados na periodicidade estabelecida, necessitando apenas da especificação dos métodos de coleta. A configuração dos parâmetros de periodicidade e forma de coleta dos dados podem ser visualizados na Figura 5.4.

Na atividade seguinte, a organização de consultoria deve “Coletar Dados dos Projetos de Melhoria”. A coleta de dados é feita para os projetos de implementação de melhoria existentes na base de projetos da organização. Sendo assim, é necessário que esses projetos sejam inseridos também na base de *benchmarking*. A Figura 5.8 exemplifica o registro do projeto “CCA-SJ Nível E” na base de *benchmarking* da II COPPE/UFRJ. É possível associar o processo utilizado na implementação da iniciativa de melhoria. Esta associação permitirá que ao modificar o processo padrão de implementação de melhoria, sejam separados os projetos que utilizaram o mesmo processo padrão, de forma que se possa evitar comparações entre projetos implementados através de diferentes processos. É também possível associar manualmente o projeto da base de *benchmarking* com um projeto de implementação da base de projetos da organização de consultoria em melhoria de

processos de software. Esta associação permitirá a coleta automática dos dados, pois é criada uma rastreabilidade direta com o projeto real onde as medidas serão coletadas.

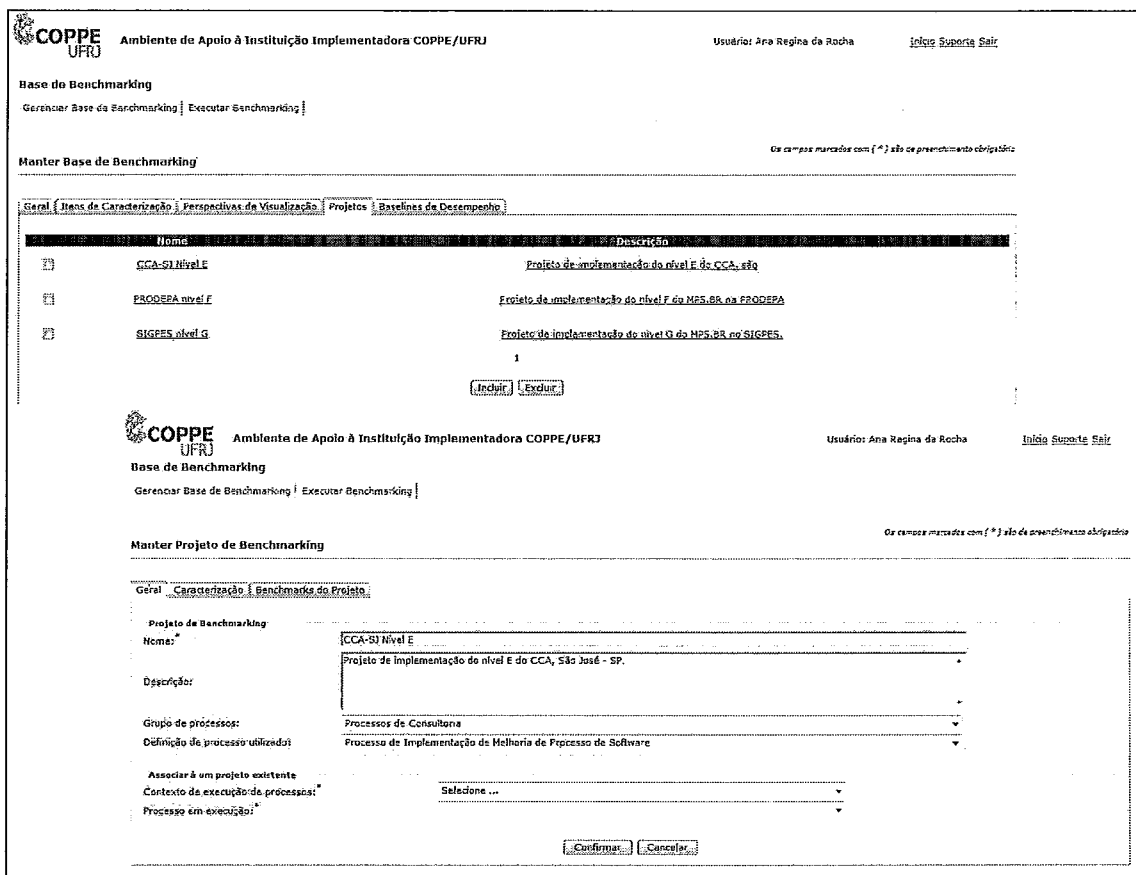


Figura 5.8 – Projeto de Implementação de Melhorias em Processos de Software

Além de inserir os projetos na base de *benchmarking*, é necessário caracterizar de forma correta e criteriosa o contexto em que estes projetos são executados para evitar que sejam realizadas comparações entre projetos que não são semelhantes. Esta caracterização é feita levando-se em consideração os itens de caracterização de contexto de projetos, registrados ainda na atividade de criação da base de *benchmarking*. A Figura 5.9 mostra a caracterização do contexto do projeto de melhoria “CCA-SJ Nível E” que está sendo registrado na base de *benchmarking*. Esta caracterização será levada em consideração durante os exercícios de *benchmarking* realizados.

Base de Benchmarking

[Gerenciar Base de Benchmarking](#) | [Executar Benchmarking](#) |

Os campos marcados com () s o de preenchimento obrigat rio*

Manter Projeto de Benchmarking

[Geral](#) | [Caracteriza o](#) | [Benchmarks do Projeto](#)

Caracteriza o do Projeto

Ano da Implementa�o	2007
Certifica�o COBIT	N�o
Certifica�o ISO	N�o
Certifica�o ITIL	N�o
Classifica�o da Institui�o	Privada
Complexidade dos Projetos	M�dio
Estado da Unidade Organizacional onde ocorreu a implementa�o.	Outros
Infra-estrutura de apoio utilizado na implementa�o	Somente Taba
Modelo de Neg�cio utilizado na implementa�o	Individual
N�vel CMHI anterior � implementa�o	Nenhum
N�vel CMHI desejado	Nenhum
N�vel NPS anterior � implementa�o	Nenhum
N�vel NPS desejado	F
N�mero de Consultores	1 � 3
N�mero de Gerentes de Projeto que participaram da iniciativa de melhoria	1 � 3
N�mero de Gerentes de Projeto que possuem certifica�o PMP e participaram da implementa�o	1 � 3
Tamanho da Unidade Organizacional onde ocorreu a implementa�o.	Pequena

Figura 5.9 – Caracteriza o do Projeto “CCA-SJ N vel E”

Na periodicidade estabelecida, os dados sobre o desempenho dos projetos s o coletados e inseridos na base de *benchmarking*. Estes dados representam os *benchmarks* associados aos projetos. Os *benchmarks* s o os valores coletados para as perspectivas de visualiza o cadastradas na base de *benchmarking*, em um determinado per odo de tempo, para um determinado projeto. A Figura 5.10 mostra os *benchmarks* coletados para o projeto “CCA-SJ N vel E” e tamb m o registro manual de um *benchmark* do projeto em quest o.

COPPE UFRJ Ambiente de Apoio à Instituição Implementadora COPPE/UFRJ Usuário: Ana Regina de Rocha [Início](#) [Suporte](#) [Sair](#)

Base de Benchmarking
 Gerenciar Base de Benchmarking | Executar Benchmarking

Os campos marcados com (*) são de preenchimento obrigatório

Manter Projeto de Benchmarking

Gerenciar | Cadastrar | Benchmark de Projeto

Benchmark	Valor
Esforço (HH) de Acompanhamento com Perfil "Mestrando"	136
Esforço (HH) de Acompanhamento com Perfil "Doutor"	56
Número de Viagens	1

COPPE UFRJ Ambiente de Apoio à Instituição Implementadora COPPE/UFRJ Usuário: Ana Regina de Rocha [Início](#) [Suporte](#) [Sair](#)

Base de Benchmarking
 Gerenciar Base de Benchmarking | Executar Benchmarking

Os campos marcados com (*) são de preenchimento obrigatório

Manter Benchmarks do Projeto

Dados do Benchmark

Esforço (HH) de Acompanhamento com Perfil "Mestrando"

Valor:

Observação:

Figura 5.10 – Benchmarks do Projeto “CCA-SJ Nível E”

Na próxima atividade do processo de gestão da base de *benchmarking*, “Empacotar Dados dos Projetos”, a organização de consultoria avalia os dados referentes aos *benchmarks* coletados dos projetos de implementação. Esta avaliação visa a permitir a realização dos ajustes necessários ou mesmo a eliminação dos dados que não são relevantes ou que não caracterizam de forma adequada o projeto de melhoria para o qual o *benchmark* foi coletado. A organização também deve registrar informações sobre o contexto de coleta do *benchmark*, o que permitirá uma melhor análise futura. O empacotamento deve ser realizado para cada *benchmark* de cada projeto de implementação registrado na base de *benchmarking*. A Figura 5.11 mostra o empacotamento dos *benchmarks* do projeto “CCA-SJ Nível E”.

COPPE UFRJ Ambiente de Apoio à Instituição Implementadora COPPE/UFRJ Usuário: Ana Regina de Rocha [Início](#) [Suporte](#) [Sair](#)

Base de Benchmarking
 Gerenciar Base de Benchmarking | Executar Benchmarking

Os campos marcados com (*) são de preenchimento obrigatório

Empacotar Dados

Item de Base de Benchmarking

Projeto: CCA-SJ Nível E

Período: Válido

Justificativa: Os dados do projeto são válidos para fins de benchmarking.

Benchmarks inválidos:

Esforço (HH) de Acompanhamento com Perfil "Doutor"	<input type="button" value="OK"/> <input type="button" value="Cancelar"/>	Número de Viagens
Esforço (HH) de Acompanhamento com Perfil "Mestrando"	<input type="button" value="OK"/> <input type="button" value="Cancelar"/>	

Justificativa para as exclusões: O número de viagens não é relevante pois as mesmas foram incluídas no orçamento do projeto de implementação.

Figura 5.11 – Empacotamento dos Benchmarks do Projeto “CCA-SJ Nível E”

Após o empacotamento dos *benchmarks* dos projetos de implementação de melhorias em processos de software é realizada a atividade “Estabelecer Baseline de Desempenho”. O empacotamento é realizado no contexto de uma *baseline*. As *baselines* são estabelecidas periodicamente, após a coleta e o empacotamento de todos os dados dos projetos de implementação da organização de consultoria. Depois de aprovadas, estas *baselines* são então liberadas e utilizadas durante os esforços de *benchmarking* da organização de consultoria em melhoria de processos de software. A Figura 5.12 exemplifica o estabelecimento de uma *baseline* de desempenho dos projetos de implementação da II COPPE/UFRJ.

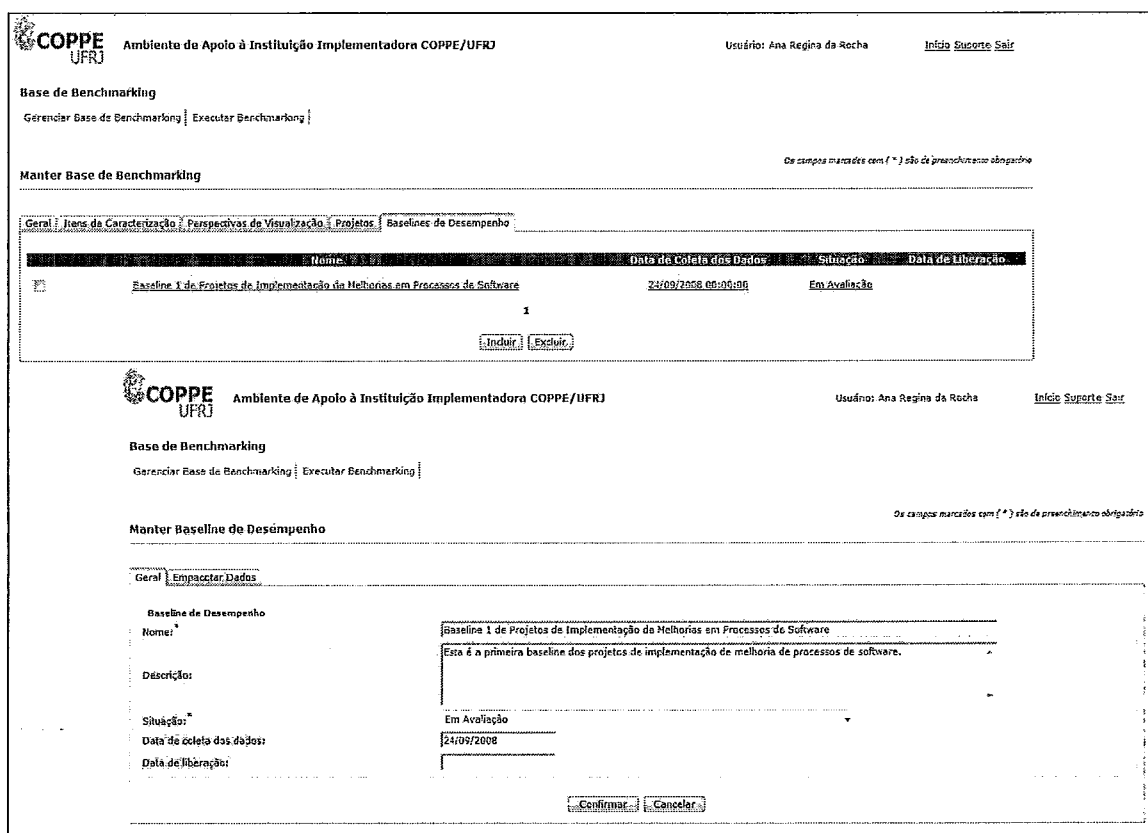


Figura 5.12 – *Baselines* de Desempenho dos projetos

Após o estabelecimento da *baseline* de desempenho, executa-se a atividade “Liberar *Baselines* de Desempenho”. Assim que todos os dados estão empacotados, a organização de consultoria determina a liberação da *baseline* realizando o cadastramento da data de liberação. Após a liberação executa-se a atividade “Comunicar Interessados”, onde todos os gerentes dos projetos de implementação, que fazem parte da base de *benchmarking*, são comunicados e podem consultar as *baselines* liberadas durante a execução das atividades de planejamento e monitoração dos projetos.

5.4.2 Execução do *Benchmarking*

Na seção anterior, foram descritas as funcionalidades de gerência da base de *benchmarking*, necessárias para a execução dos exercícios de *benchmarking* de projetos de implementação de melhorias. Sem a criação e configuração da base de *benchmarking*, não é possível a execução do Processo Realização de *Benchmarking*, descrito no capítulo 4, para a consulta dos projetos cadastrados. Nesta seção, serão descritas as atividades referentes à execução do *benchmarking*.

O *benchmarking* pode ser executado em diferentes momentos da execução dos projetos de implementação de melhorias. Ele pode ser executado durante o planejamento destes projetos, com o intuito de melhor estimar os recursos necessários ao projeto, e também pode ser executado durante a monitoração, a fim de definir, em tempo hábil a tomada de ações corretivas, desvios no desempenho dos projetos. Nestes dois contextos, o *benchmarking* sempre leva em consideração o comparativo direto com o projeto em execução. CERDEIRAL (2008) definiu em seu trabalho de mestrado, um ferramental de apoio ao planejamento e a monitoração dos projetos de implementação de melhorias em processos de software que possui integração com o trabalho descrito nesta dissertação. A Figura 5.13 exemplifica uma consulta à base de *benchmarking* executada a partir do planejamento do projeto de melhoria.

The screenshot shows the 'Planejamento de Projetos de Melhoria' (Project Improvement Planning) interface. At the top, it identifies the user as 'Ana Regina da Rocha' and provides a 'Sair' (Logout) link. A navigation bar includes 'Diagnóstico', 'Planejamento Inicial', 'Planejamento Detalhado', 'Monitoração e Controle', 'Encerramento', and 'Execução'. The main section is titled 'Realizar Benchmarking' and contains a table of search criteria. A note states: 'Os campos marcados com (*) são de preenchimento obrigatório'. The table lists various criteria with their current values and dropdown menus for selection.

Critérios de busca	Valor
Ano da Implementação	2007
Certificação COBIT	Selecione...
Certificação ISO	Selecione...
Certificação ITIL	Selecione...
Classificação da Instituição	Selecione...
Complexidade dos Projetos	Selecione...
Estado da Unidade Organizacional Avaliada	Selecione...
Infra-estrutura de apoio à implementação	Semente Tabla
Modelo de Negócio	Cooperado
Nível CNMI anterior	Selecione...
Nível CNMI desejado	Selecione...
Nível MPS anterior	Nenhum
Nível MPS desejado	F
Número de Consultores	Selecione...
Número de Garantes de Projeto	Selecione...
Número de Garantes PMP	Selecione...
Tamanho da Unidade Organizacional	Selecione...

Buscar Projetos

Figura 5.13 – Seleção dos Critérios de Busca dos Projetos

Na primeira atividade “Selecionar Critérios de Busca dos Projetos”, o gerente do projeto de melhoria seleciona os critérios de busca que serão tomados como base para a busca dos projetos similares na base de *benchmarking*. Serão retornados os projetos cuja caracterização contemple os critérios selecionados. O exemplo apresentado na Figura 5.13 realiza a busca de projetos de implementação de melhoria cuja implementação ocorreu no ano de 2007, utilizando apenas a infra-estrutura da Estação TABA para apoiar a implementação no modelo cooperado para a implementação do nível F do MPS.BR, em organizações que não possuem nível MPS inicial.

Após a seleção dos critérios, é executada a atividade “Selecionar *Baseline* de Desempenho de Referência”. Nesta atividade, o gerente do projeto de implementação seleciona a *baseline* a partir da qual deseja extrair os dados sobre o desempenho dos projetos. O ideal é que as comparações sejam realizadas sempre tendo como base a última *baseline* liberada, já que esta conterà os dados mais atuais e acurados dos projetos de implementação da organização de consultoria. Em seguida, executa-se a atividade “Selecionar Perspectiva de Visualização”, onde é selecionada a perspectiva para a qual se deseja consultar os resultados do *benchmarking*. A Figura 5.14 mostra a consulta ao *benchmarking* com o refinamento realizado.

Classificação da Instituição	Seleção...
Complexidade dos Projetos	Seleção...
Estado da Unidade Organizacional Avaliada	Seleção...
Infra-estrutura de apoio à implementação	Somente Taba
Modelo de Negócio	Cooperado
Nível CMHI anterior	Seleção...
Nível CMHI desejado	Seleção...
Nível MPS anterior	Nenhum
Nível MPS desejado	F
Número de Consultores	Seleção...
Número de Gerentes de Projeto	Seleção...
Número de Gerentes PNP	Seleção...
Tamanho da Unidade Organizacional	Seleção...

Sumário	
Nº de projetos retornados:	8
Referência	
Baseline de Desempenho:	Baseline 1 de Projetos de Implementação de Melhorias em Processos de Software
Perspectiva de Visualização:	Projeto
Resultados	
Projeto corrente:	136 HH
Limite Superior:	194 HH
Limite Inferior:	97 HH
Média:	148 HH

Figura 5.14 – Resultados do *Benchmarking*

Na atividade “Gerar Relatório Detalhado”, o gerente do projeto de implementação gera o relatório do *benchmarking* contendo o detalhamento dos resultados do *benchmarking*

para todas as perspectivas que constam da base de *benchmarking*. A Figura 5.15 apresenta o relatório detalhado do *benchmarking* para a primeira *baseline* de desempenho da II COPPE/UFRJ.

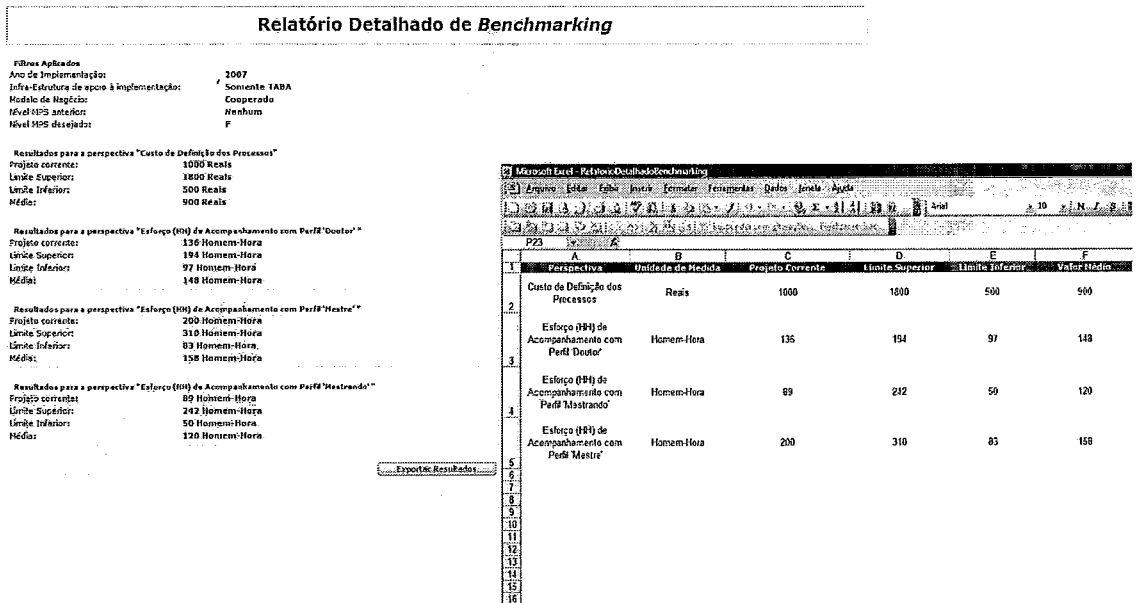


Figura 5.15 – Relatório detalhado do *Benchmarking*

A partir do relatório detalhado do *benchmarking*, é possível utilizar os dados dos projetos passados e correntes para apoiar a tomada de decisão durante o planejamento dos projetos de melhoria, bem como durante a monitoração e controle destes projetos. A atividade “Exportar Relatório do *Benchmarking*” é executada em seguida, permitindo extrair os dados do relatório detalhado do *benchmarking* para utilização em outras ferramentas de apoio à tomada de decisão. O relatório extraído é mostrado também na Figura 5.15.

5.5 Considerações finais

Este capítulo apresentou os requisitos e a implementação do apoio ferramental construído para auxiliar as instituições de consultoria em melhoria de processos de software na realização das atividades de *benchmarking* destes projetos com base na abordagem apresentada no capítulo 4. O ferramental foi desenvolvido no contexto de um ambiente de gestão de conhecimento customizado para a organização de consultoria em melhoria de processos de software COPPE/UFRJ, gerado a partir do ambiente CORE-KM (GALOTTA *et al.*, 2004). No próximo capítulo, serão apresentadas as conclusões e considerações finais deste trabalho, além de suas contribuições e perspectivas futuras.

CAPÍTULO 6 – CONCLUSÃO

Este capítulo apresenta as considerações finais desta dissertação. São também abordadas as limitações e contribuições referentes à abordagem proposta e ao apoio ferramental implementado, além das perspectivas futuras de evolução do trabalho.

6.1 Considerações finais

As pressões mercadológicas por produtos de software mais complexos, a preços cada vez menores e desenvolvidos mais rapidamente motivam as organizações de software a melhorar os procedimentos de construção destes produtos. Neste sentido, torna-se necessária a implementação de iniciativas de melhorias em processos de software, pois é conhecido que a qualidade dos produtos está fortemente associada à qualidade dos processos utilizados em seu desenvolvimento.

Contudo, a literatura relata exemplos de insucesso destas iniciativas de melhoria, sugerindo que elas não são tratadas como projetos reais que necessitam da aplicação de técnicas de gerência (PAULK *et al.*, 2000; BADDOO e HALL, 2002; NIAZI *et al.*, 2005b; NIAZI *et al.*, 2006). Mesmo com a existência de diversos modelos de referência para melhoria de processos de software, estes indicam apenas o que deve ser implementado e não como implementar as melhorias nos processos da organização (MINGHUI *et al.*, 2004).

Foi apresentada nestes trabalho, a abordagem para condução de iniciativas de melhorias em processos de software na qual está inserido este trabalho, e que é composta por três componentes. O primeiro, o componente de definição de estratégias de implementação de melhorias em processos de software, é objetivo de uma tese de doutorado (MONTONI, 2007); o segundo, o componente de gerência e avaliação dos projetos de implementação de melhorias, foi desenvolvido no contexto de uma dissertação de mestrado (CERDEIRAL, 2008); e o terceiro, o componente de *benchmarking* de projetos de implementação de melhorias em processos de software, foi desenvolvido no contexto desta dissertação. Este último componente baseia-se na aplicação de melhores práticas de iniciativas de melhoria passadas e correntes, de forma que estas informações possam ser utilizadas em iniciativas de melhoria futuras. Para atingir este objetivo, o trabalho propôs a

utilização de uma técnica bastante difundida em diversas áreas da indústria: o *benchmarking*. Além desta técnica, também foram considerados diversos paradigmas da melhoria contínua, abordados no Capítulo 3 desta dissertação.

A abordagem proposta nesta dissertação é composta por dois processos de apoio à gerência e execução das atividades de *benchmarking* no contexto de iniciativas de melhorias em processos de software e de um ferramental de software para apoiar estes processos.

6.2 Contribuições

Entre as contribuições deste trabalho, destacam-se:

- A pesquisa e comparação de diversos processos de apoio às atividades de *benchmarking*.
- A definição de processos de apoio à execução do *benchmarking* no contexto de iniciativas de melhorias em processos de software, levando em consideração a qualidade dos dados coletados.
- A definição, modelagem e implementação de um ferramental de software para apoiar o planejamento e a execução das atividades de *benchmarking* em iniciativas de melhoria.

Diversos benefícios podem ser derivados a partir do uso da abordagem proposta.

Dentre estes, destacam-se:

- A formação de uma base de dados históricos para armazenamento das informações referentes ao desempenho dos projetos de implementação de melhorias em processos de software, fazendo com que essas informações possam ser utilizadas de diferentes formas em longo prazo.
- O acúmulo de um corpo de conhecimento sobre a implementação de melhorias em processos de software que possibilite a disseminação deste conhecimento para os membros que fazem parte da equipe de consultoria em melhoria de processos de software. Desta forma, será possível realizar as tarefas da maneira mais eficiente e eficaz possível, reutilizando o conhecimento em novas iniciativas de melhoria, fazendo com que os erros cometidos em projetos anteriores não sejam cometidos novamente e as melhores práticas destacadas possam ser utilizadas para aumentar as chances de sucesso dessas iniciativas.

- A visualização do progresso efetivo destas iniciativas de forma que desvios possam ser identificados rapidamente e ações corretivas sejam executadas para manter ou melhorar o desempenho dos projetos.
- A análise da melhoria histórica da organização de consultoria em melhoria de processos de software através da análise da melhoria no desempenho dos projetos de implementação de melhorias em processos de software executados por esta organização.
- A melhoria da qualidade dos serviços prestados por organizações de consultoria em melhoria de processos de software, visto que a análise do desempenho pode servir de base para melhorias também no processo executado nessas iniciativas de melhoria.
- O aumento das chances de sucesso dessas iniciativas, pois um melhor planejamento poderá ser realizado, evitando o desperdício de recursos, ou a concentração dos mesmos em atividades menos prioritárias.
- E, por fim, a melhoria contínua dos processos executados pelas organizações de consultoria em melhorias de processos de software

6.3 Limitações

Algumas limitações foram identificadas durante o desenvolvimento da abordagem de *benchmarking* de iniciativas de melhorias em processos de software, bem como durante a construção do apoio computacional do *benchmarking*. Destas limitações, destacam-se:

- O ferramental de apoio às atividades de *benchmarking* implementado não foi ainda utilizado em projetos de melhoria da Instituição Implementadora COPPE/UFRJ. Estas experiências são necessárias para avaliar a efetividade da abordagem proposta, bem como de seu apoio ferramental.
- A automatização da coleta dos dados do *benchmarking* é de suma importância para tornar os processos do *benchmarking* o menos intrusivo possível, o que resultará em menor resistência dos usuários. Esta automatização não foi contemplada apesar de o apoio ferramental permitir que ela seja integrada. Para isto, é necessária a especificação de métodos de coleta e da implementação de todos os componentes da abordagem de condução de iniciativas de melhoria de processos de software.
- Na abordagem proposta, optou-se por considerar apenas o *benchmarking* interno. A escolha deste tipo de *benchmarking* se deve à falta de padronização na execução

destas iniciativas, o que impede a internalização de práticas que não surgem dentro de uma determinada organização de consultoria em melhoria de processos de software. No entanto, é reconhecido que a busca externa de melhores práticas pode auxiliar na disseminação dessas informações ocasionando a melhoria do setor de software como um todo.

6.4 Perspectivas futuras

A partir da utilização da abordagem proposta em organizações de consultoria em melhoria de processos de software, será possível analisar a sua efetividade. Esta é a maior perspectiva futura relacionada à esta dissertação. A utilização em iniciativas reais fará com que a abordagem proposta seja melhorada continuamente. Também espera-se que a abordagem seja expandida para contemplar o *benchmarking* externo destas iniciativas, de forma que o conhecimento relativo às melhores práticas em implementação de melhorias em processos de software seja disseminado na comunidade de software e beneficie a indústria de software como um todo.

Atualmente, os gerentes dos projetos de melhoria necessitam acessar o ferramental de *benchmarking* para realizar a monitoração de seus projetos com base em comparativos entre projetos similares. Uma possível evolução da abordagem poderá fazer com que os comparativos sejam mostrados de forma mais pró-ativa, aumentando a sua eficácia para os gerentes.

O ferramental de apoio desenvolvido neste trabalho não apóia as fases de Integração e Ação, descritas no comparativo do Capítulo 3. Espera-se que estas fases sejam contempladas no trabalho de MONTONI (2007).

Outra perspectiva de evolução relativa ao apoio ferramental diz respeito à coleta dos dados. Espera-se que este ferramental evolua de forma a permitir a coleta automática dos dados dos projetos de implementação de melhoria, o que aumentará a qualidade dos dados coletados e a usabilidade do apoio ferramental.

Este trabalho é parte de um projeto mais amplo do grupo de qualidade de software da COPPE/UFRJ, envolvendo outros trabalhos de mestrado e doutorado (MONTONI, 2007; CERDEIRAL, 2008). Para o efetivo funcionamento do apoio ferramental construído no contexto desta dissertação, é necessário que todo o apoio ferramental dos trabalhos relacionados esteja também finalizados. Portanto, espera-se que o apoio ferramental seja avaliado em conjunto com os apoios ferramentais dos demais componentes da abordagem

de condução de iniciativas de melhorias em processos de software, e que esta sirva de apoio para as organizações de consultoria na execução de seus projetos de melhoria.

REFERÊNCIAS BIBLIOGRÁFICAS

- AGUIAR, H.V., 2004, *PEPP: Processo de Software para Empresas de Pequeno Porte Baseado no Modelo CMMI*, Monografia de B.Sc., Universidade Federal de Lavras, Lavras, Minas Gerais.
- ALTHOFF, K.-D., BOMARIUS, F., TAUTZ, C., 1998, "Using Case-Based Reasoning Technology to Build Learning Software Organizations". In: *Proceedings of the 1st Workshop on Building, Maintaining, and Using Organizational Memories, OM-98*, v. 14, pp. 1-20, Brighton, UK, Aug.
- ANDERSEN, B., PETTERSEN, P., 1996, *The benchmarking handbook : step-by-step instructions*, London, Chapman & Hall.
- ANON, 1995, *Benchmarking reveals new design trends*, Machine-Design.
- BADDOO, N., HALL, T., 2002, "Motivators of Software Process Improvement: An Analysis Of Practitioners' Views", *Journal Of Systems & Software* v. 62, n. 2 (Mai), pp. 85-96.
- BADDOO, N., HALL, T., 2003, "De-motivators for software process improvement: an analysis of practitioners' views", *Journal Of Systems & Software*, v. 66, n. 1 (Abr), pp. 23-33.
- BASILI, V., CALDIERA, G., MCGARRY, F., *et al.*, 1992, "The Software Engineering Laboratory: an Operational Software Experience Factory". In: *Proceedings of the 14th International Conference on Software Engineering* pp. 370-381, Melbourne, Australia, May.
- BASILI, V., LINDVALL, M., COSTA, P., 2001a, "An Experience Management System for a Software Engineering Research Organization". In: *Proceedings of 13 th International Conference on Software Engineering & Knowledge Engineering* pp. 102--109, San Francisco, USA, Jul.

- BASILI, V., MCGARRY, F., 1997, "The Experience Factory: How to Build and Run One". In: *Proceedings of the 19th International Conference on Software Engineering*, pp. 643-644, Boston, USA, May.
- BASILI, V.R., CALDIERA, G., ROMBACH, H.D., 1994, "The Experience Factory". In: *Encyclopedia of Software Engineering*, John Wiley & Sons.
- BASILI, V.R., LINDVALL, M., COSTA, P., 2001b, "Implementing the Experience Factory concepts as a set of Experience Bases". In: *13th International Conference on Software Engineering & Knowledge Engineering*, pp. 102-109, Buenos Aires, Argentina, Jun.
- BIRK, A., PFAHL, D., 2002, "A Systems Perspective on Software Process Improvement". In: *Proceedings of the 4th International Conference on Product Focused Software Process Improvement*, v. 2559, pp. 4-18, Dec.
- BOFFOLI, N., 2006, "Non-intrusive monitoring of software quality". In: *Proceedings of the 10th European Conference on Software Maintenance and Reengineering, CSMR 2006.*, v. 0, pp. 4-7, Bari, Italy, Mar.
- BRASIL, C., SOUZA, F.J.F.D., MELO, W.L., 2008, "A Fábrica de Experiência do Tribunal de Contas do Distrito Federal: Primeiros Resultados". In: www.geocities.com/walcelio_melo/cits-tcdf.pdf, Acessado em: Março/2008.
- CAMP, R., 1993, *Benchmarking: identificando, analisando e adaptando as melhores práticas da administração que levam à maximização da performance empresarial: o caminho da qualidade total*, 3ª ed., São Paulo, SP, Pioneira.
- CARD, D., ZUBROW, D., 2001, "Benchmarking Software Organizations". In: *IEEE Software*, v. 18, IEEE Computer Society, pp. 16-17.
- CERDEIRAL, C., 2008, *Uma Abordagem para Gerência e Avaliação de Projetos de Melhoria de Processos de Software do Ponto de Vista da Instituição de Consultoria*, Dissertação de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

DEMING, W.E., 1982, *Out of the Crisis*, MIT Press.

DYBÅ, T., 2002, "Enabling Software Process Improvement: An Investigation of the Importance of Organizational Issues", *Empirical Software Engineering*, v. 7, n. 4 (Dec), pp. 387-390.

EL-EMAM, K., FUSARO, P., SMITH, B., 1999, "Success factors and barriers for software process improvement". In: *Better Software Practice for Business Benefit: Principles and Experience*, IEEE Computer Society.

EL-EMAM, K., GOLDENSON, D., MCCURLEY, J., *et al.*, 2001, "Modelling the Likelihood of Software Process Improvement: An Exploratory Study", *Empirical Software Engineering*, v. 6, n. 3 (Set), pp. 207-229.

ESTOLANO, M.H., 2005, *Base de Métricas para a Estação TABA*, Dissertação de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

FOGLE, S., LOULIS, C., NEUENDORF, B., 2001, "The Benchmarking Process: one team's experience", *IEEE Software*, v. 18, n. 5 (Sep), pp. 40-47.

FRIDLEY, J.L., JORGENSEN, J.E., LAMANCUSA, J.S., 1997, "Benchmarking: a process basis for teaching design". In: *Proceedings of the 27th Annual Conference on Frontiers in Education Conference, 'Teaching and Learning in an Era of Change'*. v. 2, pp. 960-967, Pittsburgh, USA, Nov.

FUGGETTA, A., 2000, "Software process: a roadmap". In: *Proceedings of the Conference on The Future of Software Engineering*, pp. 25-34, Limerick, Ireland, Jun.

GALOTTA, C., ZANETTI, D., ROCHA, A.R., 2004, "Organizational Learning Based on a Customizable Environment for Knowledge Management Using Intranet". In: *World Conference on e-Learning in Corporate, Government, Healthcare & Higher Education, E-LEARN 2004* v. 2, pp. 2626-2633, Washington, EUA.

- GOLDENSON, D.R., HERBSLEB, J.D., 1995, *After the Appraisal: A Systematic Survey of Process Improvement, its Benefits and Factors that Influence Success*, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA. Disponível em: <http://www.sei.cmu.edu/>.
- GREMBA, J., MYERS, C., 1997, *The IDEAL Model: A Practical Guide for Improvement*. Disponível em: <http://www.sei.cmu.edu/ideal/ideal.bridge.html>.
- GRIMBLE, M.J., UDUEHI, D., 2001, "Process control loop benchmarking and revenue optimization". In: *Proceedings of the 2001 American Control Conference*, v. 6, pp. 4313-4327, Arlington, Virginia, USA, Jun.
- HARRINGTON, H.J., HARRINGTON, J.S., 1996, *High performance benchmarking : 20 steps to success*, New York, McGraw-Hill.
- HEIRES, J.T., 2001, "What I did last summer: a software development benchmarking case study", *IEEE Software*, v. 18, n. 5 (Sep), pp. 33-39.
- HOST, M., JOHANSSON, E., NOREN, A., *et al.*, 2002, "Benchmarking of processes for managing product platforms: a case study", *IEE Proceedings - Software*, v. 149, n. 5 (Oct), pp. 137-142.
- HUMPHREY, W.S., 1989, *Managing the Software Process*, 1 ed., Reading, MA, Addison-Wesley Professional.
- IAPMEI, "Benchmarking nas empresas fornecedoras de serviços de logística". In: <http://www.iapmei.pt/iapmei-bmkartigo-01.php>, Acessado em: Março de 2008.
- IBRAHIM, L., PYSTER, A., 2004, "A Single Model for Process Improvement", *IT Professional*, v. 6, n. 3 (May), pp. 43-49.
- ISO/IEC, 1995, "ISO/IEC 12207: Information Technology – Software Life Cycle Processes", *The International Organization for the Standardization and the International Electrotechnical Commission*.

- ISO/IEC, 2000, "ISO/IEC 9000:2000: Quality Management Systems – Fundamental and Vocabulary", *The International Organization for the Standardization and the International Electrotechnical Commission*.
- ISO/IEC, 2002, "ISO/IEC PDAM 12207: Information Technology – Amendment to ISO/IEC 12207", *The International Organization for the Standardization and the International Electrotechnical Commission*.
- ISO/IEC, 2003, "15504: Information Technology – Process Assessment. Part 1 – Concepts and vocabulary; part 2 – Performing an assessment; part 3 – Guidance on performing an assessment; part 4 – Guidance on use for process improvement and process capability determination; and part 5 – An exemplar process assessment model." *The International Organization for the Standardization and the International Electrotechnical Commission*.
- ISO/IEC, 2004, "ISO/IEC PDAM 12207: Information Technology – Amendment 2 to ISO/IEC 12207", *The International Organization for the Standardization and the International Electrotechnical Commission*.
- ISO/IEC, 2008, "ISO/IEC 12207: System and software engineering – Software life cycle processes", *The International Organization for the Standardization and the International Electrotechnical Commission*.
- JARVINEN, J., PERKLEN, E., KAILA-STENBERG, S., *et al.*, 1998, "PDCA-cycle in implementing design for environment in an R&D unit of Nokia Telecommunications". In: *Proceedings of the 1998 IEEE International Symposium on Electronics and the Environment, ISEE-1998.*, pp. 237-242, Oak Brook, IL, USA, May.
- JURAN, J.M., 1994, *Benchmarking for world class leadership*, Juran Institute.
- KIM, G., LEE, M., LEE, J., *et al.*, 2005, "Design of SPICE experience factory model for accumulation and utilization of process assessment experience". In: *Third ACIS International Conference on Software Engineering Research, Management and Applications*, pp. 368-374, Michigan, USA, Aug.

- KOENNECKER, A., JEFFERY, R., LOW, G., 2000, "Implementing an experience factory based on existing organisational knowledge". In: *Proceedings of the Australian Software Engineering Conference*, pp. 53-62, Canberra, ACT, Australia, Apr.
- LOKAN, C., WRIGHT, T., HILL, P.R., *et al.*, 2001, "Organizational Benchmarking Using the ISBSG Data Repository", *IEEE Software*, v. 18, n. 5 (Sep), pp. 26-32.
- LÓPEZ-CORTIJO, R., GUZMÁN, J., SECO, A., 2007, "iCharts: Charts for Software Process Improvement Value Management". In: *Proceedings of the 14th European Conference on Software Process Improvement, EuroSPI 2007*, pp. 124-135, Potsdam, Germany, Aug.
- MAIRE, J.L., BUYUKOZKAN, G., 1997, "Methods and tools for first five steps of benchmarking process". In: *Portland International Conference on Management of Engineering and Technology, PICMET '97 - Innovation in Technology Management: The Key to Global Leadership*, pp. 798, Portland, OR, USA, Jul.
- MARD, M.J., DUNNE, R.R., OSBOURNE, E., *et al.*, 2004, *Driving Your Company's Value - Strategic Benchmarking for Value*, 1 ed., New Jersey, John Wiley & Sons.
- MAXWELL, K.D., 2001, "Collecting data for comparability: benchmarking software development productivity", *IEEE Software*, v. 18, n. 5 (Sep), pp. 22-25.
- MCGARRY, F., DECKER, B., 2002, "Attaining Level 5 in CMM process maturity", *IEEE Software*, v. 19, n. 6 (Nov), pp. 87-96.
- MILLER, S.E., TUCKER, G.T., 1991, "Software development process benchmarking". In: *Global Telecommunications Conference, GLOBECOM '91. 'Countdown to the New Millennium. Featuring a Mini-Theme on: Personal Communications Services'*, v. 1, pp. 153-157, Phoenix, AZ, USA, Dec.

- MINGHUI, W., JING, Y., CHUNYAN, Y., 2004, "A methodology and its support environment for benchmark-based adaptable software process improvement". In: *IEEE International Conference on Systems, Man and Cybernetics*, v. 6, pp. 5183-5188, Hague, Netherlands, Oct.
- MONTONI, M., 2007, *Uma Abordagem para Condução de Iniciativas de Melhoria de Processos de Software*, Exame de Qualificação, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- MONTONI, M., CERDEIRAL, C., ZANETTI, D., *et al.*, 2008a, "A Knowledge Management Approach to Support Software Process Improvement Implementation Initiatives". In: *Software Process Improvement*, v. 16, *Communications in Computer and Information Science*, Springer Berlin Heidelberg, pp. 164-175.
- MONTONI, M., CERDEIRAL, C., ZANETTI, D., *et al.*, 2008b, "Uma Abordagem para Condução de Iniciativas de Melhoria de Processos de Software". In: *VII Simpósio Brasileiro de Qualidade de Software (SBQS)*, pp. 263-277, Florianópolis, SC, Jun.
- MONTONI, M., ROCHA, A.R., 2007, "A Methodology for Identifying Critical Success Factors that Influence Software Process Improvement Initiatives: An Application in the Brazilian Software Industry". In: *Software Process Improvement*, v. 4764, *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 175-186.
- MONTONI, M., SANTOS, G., ROCHA, A.R., *et al.*, 2006, "Taba workstation: Supporting software process deployment based on CMMI and MR-MPS.BR". In: *7th International Conference on Product-Focused Software Process Improvement, PROFES 2006*, pp. 249-262, Amsterdam, Netherlands, Jun.
- MONTONI, M., SANTOS, G., ROCHA, A.R., *et al.*, 2007, "MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings". In: *Fifth International Workshop on Software Quality, WoSQ'07*, pp. 4-4, Amsterdam, Netherlands.

- NIAZI, M., WILSON, D., ZOWGHI, D., 2005a, "A framework for assisting the design of effective software process improvement implementation strategies", *Journal of Systems and Software* v. 78, n. 2 (Nov), pp. 204-222.
- NIAZI, M., WILSON, D., ZOWGHI, D., 2005b, "A maturity model for the implementation of software process improvement: An empirical study", *Journal of Systems and Software*, v. 74, n. 2 (Jan), pp. 155-172.
- NIAZI, M., WILSON, D., ZOWGHI, D., 2006, "Critical success factors for software process improvement implementation: an empirical study", *Software Process: Improvement and Practice*, v. 11, n. 2 (Mar), pp. 193-211.
- PAULK, M.C., GOLDENSON, D., WHITE, D.M., 2000, *The 1999 Survey of High Maturity Organizations*, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA. Disponível em: <http://www.sei.cmu.edu/>.
- ROLLO, A., MORRIS, P., WASYLKOWSKI, E., 2006, *Internacional Software Benchmarking Standards Group*, ISBSG. Disponível em: <http://www.isbsg.org>.
- SAKAMOTO, K., NAKAKOJI, K., TAKAGI, Y., *et al.*, 1998, "Toward computational support for software process improvement activities". In: *Proceedings of the 20th International Conference on Software Engineering*, pp. 22-31, Kyoto, Japan, Apr.
- SANTOS, G., MONTONI, M., VASCONCELLOS, J., *et al.*, 2007, "Implementing Software Process Improvement Initiatives in Small and Medium-Size Enterprises in Brazil". In: *6th International Conference on the Quality of Information and Communications Technology, QUATIC 2007*, pp. 187-198, Lisbon, Portugal.
- SCHNEIDER, K., VON HUNNIUS, J.P., BASILI, V.R., 2002, "Experience in implementing a learning software organization", *IEEE Software*, v. 19, n. 3 (May), pp. 46-49.

- SEI, 1997, *Integrated Product Development Capability Maturity Model*, Enterprise Process Improvement Collaboration and Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA. Disponível em: <http://www.sei.cmu.edu/>.
- SEI, 2006, *CMMI for Development, Version 1.2*, Pittsburgh, PA, Software Engineering Institute.
- SOFTEX, 2007, *MPS.BR: Melhoria de Processo do Software Brasileiro, Guia Geral Versão 1.2*, Campinas, SP, SOFTEX.
- SPENDOLINI, M., 1992, *The benchmarking book*, New York, Amacom Books.
- SPENGLER, A., STANTON, M., ROWLANDS, H., 1999, "Expert systems and quality tools for quality improvement". In: *Proceedings of the 7th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA '99*, v. 2, pp. 955-962, Barcelona, Spain, Oct.
- STELZER, D., MELLIS, W., 1998, "Success factors of organizational change in software process improvement". In: *Software Process: Improvement and Practice*, v. 4, John Wiley & Sons, pp. 227-250.
- STRAUSS, A., CORBIN, J., 1998, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, 2 ed., Sage Publications.
- SW-CMM, 1997, *Capability Maturity Model for Software*, Software Engineering Institute, Carnegie Mellon University. Disponível em: <http://www.sei.cmu.edu/>.
- SWEBOK, 2004, *Software Engineering Body of Knowledge*, IEEE Computer Society.
- THOMAS, G.C., SMITH, H.R., 2001, "Using structured benchmarking to fast-track CMM process improvement", *IEEE Software*, v. 18, n. 5 (Set), pp. 48-52.

WILSON, D.N., HALL, T., BADDOO, N., 2001, "A framework for evaluation and prediction of software process improvement success", *Journal of Systems & Software*, v. 59, n. 2 (Nov), pp. 135-142.

ZAHRAN, S., 1998, *Software process improvement: practical guidelines for business success*, Addison-Wesley Professional.

ZAIRI, M., LEONARD, P., 1994, *Practical Benchmarking - A complete guide*, 1 ed., London, UK, Chapman & Hall.

ANEXO I - ESPECIFICAÇÃO DE REQUISITOS

Este anexo apresenta com mais detalhes o levantamento de requisitos inicial realizado. O objetivo deste levantamento é analisar as necessidades do cliente, os processos de apoio e as necessidades de implementação referentes à criação do ferramental de apoio às atividades de benchmarking de iniciativas de melhoria de processos de software. A partir dessa análise foram identificados requisitos do cliente, requisitos funcionais e não-funcionais de uma solução de software capaz de atender as necessidades do cliente segundo os processos de apoio. Parte deste levantamento foi descrito no Capítulo V desta dissertação. O conteúdo deste anexo não representa toda a especificação e modelagem elaborada para a implementação. Existem outros diagramas e documentos que foram gerados a partir da execução do processo de desenvolvimento do Laboratório de Engenharia de Software da COPPE/UFRJ.

I.1 Necessidades do Cliente

Necessidade - Apoiar a definição, planejamento e execução de estratégias mais adequadas de implementação de melhoria de processo empregando conhecimento de melhores práticas e fatores críticos de sucesso de iniciativas anteriores ou concorrentes no contexto das organizações que implementam a melhoria.

Afeta - Organizações de Consultoria de Melhoria de Processos de Software.

Impacto - Os projetos de melhoria não são gerenciados de forma correta e as estratégias utilizadas na execução desses projetos não são definidas de forma adequada, o que pode ser uma das razões do insucesso desses programas. A ferramenta auxiliará na condução dos projetos de implementação de melhoria de processos de software.

Benefícios com a solução - Um apoio ferramental para o *benchmarking* de projetos de implementação de melhoria auxiliará o acúmulo de conhecimento sobre as melhores práticas desse tipo de projeto. Esse acúmulo de conhecimento serve de mecanismo para aumentar as chances de sucesso de programas de melhoria, pois possibilita que estratégias de implementação de melhoria de processos de software sejam definidas e adaptadas considerando o desempenho e os fatores de sucesso de implementações anteriores e/ou concorrentes.

I.2 Modelo de Domínio

Para auxiliar o entendimento dos itens que compõem este levantamento de requisitos, foi gerado um modelo de domínio para o apoio ferramental. O modelo de domínio é exibido na Figura I.1.

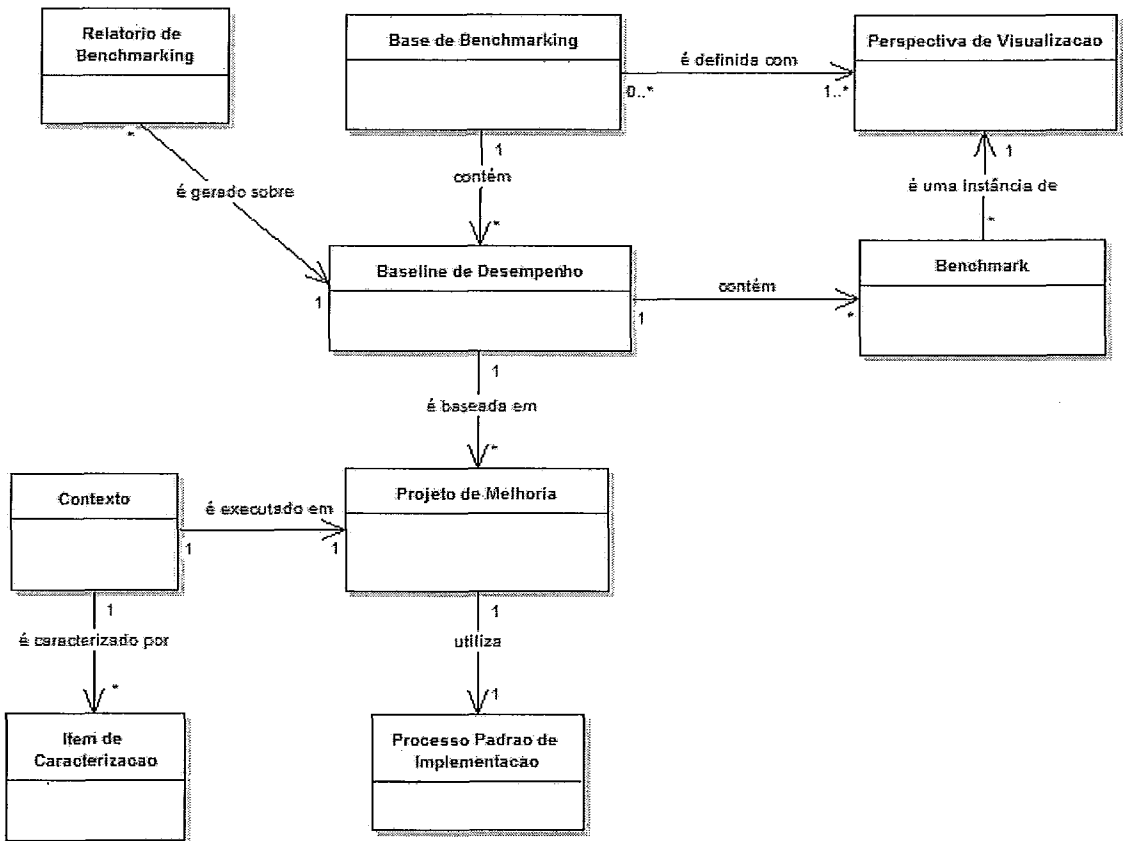


Figura I.1 – Diagrama de domínio

I.3 Requisitos do Cliente

A partir da análise do texto preliminar da tese e discussões com os orientadores, apresentação nas aulas de seminário de teses e análise de documentação do CORE-KM, foram levantados requisitos do cliente descritos a seguir. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela I.1. Os Requisitos de Produto estão identificados na seção 7 deste anexo.

Os requisitos foram classificados segundo o seguinte critério de prioridade: 'A' significa que a prioridade do requisito é alta, 'M' significa que a prioridade do requisito é média e 'B' significa que a prioridade do requisito é baixa.

Tabela I.1 – Requisitos de Cliente

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
RC.01	RP.04 RP.11	Permitir a comparação dos projetos de implementação de melhorias em processos de software, utilizando a abordagem de <i>benchmarking</i> , de forma a permitir a análise dos pontos fortes e pontos fracos desses projetos	X		
RC.02	RP.04 RP.11	Permitir aos gerentes dos projetos de implementação de melhorias em processos de software monitorar a evolução do desempenho/estado dos projetos e das mudanças neste desempenho durante a implementação do projeto de melhoria, de forma a permitir a tomada de ação em tempo hábil para garantir um andamento eficiente e eficaz desses projetos, mantendo ou melhorando o desempenho dos mesmos	X		
RC.03	RP.04 RP.06 RP.11	Identificar atributos internos e externos relevantes aos projetos de melhoria de processos de software que possibilitem monitorar efetivamente esses projetos, minimizando seus riscos, e permitindo a identificação de relações causais entre as características desses projetos	X		
RC.04	RP.04 RP.07 RP.11	Permitir a especificação dos parâmetros de comparação dos projetos de implementação de melhorias em processos de software que devem ser considerados relevantes para fins de <i>benchmarking</i> desses projetos	X		
RC.05	RP.04 RP.11	Permitir a manutenção dos dados referentes ao desempenho dos projetos de implementação de melhorias em processos de software e a formação de uma base histórica desses projetos	X		

I.4 Requisitos Funcionais de Software

O GEPS (Grupo de Engenharia de Processos de Software) desenvolve e mantém um ambiente customizado de gerência de conhecimento chamado CORE-KM composto de ferramentas de apoio a definição, planejamento e execução de processos organizacionais, dentre outras.

Para atender às necessidades do cliente levantadas, foram identificados os requisitos funcionais que devem ser implementados no ambiente CORE-KM e na ferramenta identificada. Esses requisitos estão descritos a seguir.

Os requisitos foram classificados segundo o seguinte critério de prioridade: ‘A’ significa que a prioridade do requisito é alta, ‘M’ significa que a prioridade do requisito é média e ‘B’ significa que a prioridade do requisito é baixa.

A Tabela I.2 apresenta os requisitos funcionais da ferramenta de apoio às atividades de *benchmarking* de iniciativas de melhoria de processos de software. Para cada Requisito de

Funcional (ID RF) foram identificados os Requisitos de Cliente (ID RC) afetados ou atendidos.

Tabela I.2 – Requisitos Funcionais

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
RF.01	RC.01	Possibilitar a criação e configuração de um repositório de armazenamento de melhores práticas relativas a projetos de implementação de melhoria de processos de software no contexto de organizações de consultoria em melhoria de processos.	X		
RF.02	RC.01	Permitir a manutenção dos dados sobre os projetos de implementação de melhorias em processos de software que fazem parte da base de <i>benchmarking</i> da organização de consultoria em melhoria de processos de software.	X		
RF.03	RC.01	Permitir a coleta dos dados de desempenho dos projetos de implementação de melhorias em processos de software com uma periodicidade estabelecida	X		
RF.04	RC.01	Permitir a criação e manutenção de <i>baselines</i> de desempenho dos projetos de implementação de melhorias em processos de software, tendo como base os atributos internos e externos relevantes para esses projetos.	X		
RF.05	RC.02	Permitir a manutenção de atributos internos e externos relevantes para as comparações a serem realizadas entre os projetos de implementação de melhorias em processos de software, em forma de perspectivas de visualização da base de <i>benchmarking</i> .	X		
RF.06	RC.02	Permitir a manutenção dos itens de caracterização dos projetos de implementação de melhorias em processos de software.	X		
RF.07	RC.02	Permitir a caracterização dos projetos de implementação de melhorias em processos de software em relação aos itens de caracterização definidos para a base de <i>benchmarking</i> .	X		
RF.08	RC.03	Permitir a seleção de itens de caracterização que devem estar presentes nos projetos de implementação de melhorias em processos de software a serem pesquisados para fins de <i>benchmarking</i> .	X		
RF.09	RC.04 RC.05	Permitir a consulta dos dados referentes ao desempenho dos projetos de implementação de melhorias em processos de software sob uma determinada perspectiva de visualização, levando em consideração os itens de caracterização selecionados e que devem estar presentes nos projetos pesquisados, a fim de garantir.	X		
RF.10	RC.04 RC.05	Possibilitar a geração de relatórios detalhados de cada exercício de <i>benchmarking</i> executado sobre os projetos que constam da base de <i>benchmarking</i> .	X		
RF.11	RC.04 RC.05	Permitir a exportação dos dados extraídos do <i>benchmarking</i> para posterior utilização.	X		

I.5 Requisitos Não-Funcionais de Software

Os requisitos não-funcionais representam características de qualidade de software ou restrições que devem ser consideradas na implementação dos requisitos funcionais. Os requisitos não-funcionais são aplicáveis a todos os requisitos funcionais dos módulos descritos para o ferramental de apoio as atividades de *benchmarking* de iniciativas de melhorias em processos de software. Os requisitos não-funcionais definidos para o projeto conhecer estão descritos na Tabela I.3.

Os requisitos foram classificados segundo o seguinte critério de prioridade: ‘A’ significa que a prioridade do requisito é alta, ‘M’ significa que a prioridade do requisito é média e ‘B’ significa que a prioridade do requisito é baixa. Essa classificação foi determinada de acordo com os interesses específicos do projeto em questão.

Tabela I.3 – Requisitos Não-Funcionais

ID RNF	ID RF	Descrição	Prioridade		
			A	M	B
RNF.01	Todos	<i>Adequação</i> - Capacidade do produto de software de fornecer um conjunto de funções adequado para as tarefas especificadas e os objetivos dos usuários.	X		
RNF.02	Todos	<i>Segurança</i> - Capacidade do produto de software proteger a informação e os dados de forma que pessoas ou sistemas não autorizados não possam lê-los ou modificá-los e pessoas e sistemas autorizados não tenham acesso negado aos mesmos. Isto deverá ser contemplado via um controle de seção onde os usuários do sistema deverão possuir um login e uma senha		X	
RNF.03	Todos	<i>Apreensibilidade</i> - Capacidade do produto de software permitir ao usuário aprender a sua aplicação. O sistema deverá ser implementados com os padrões estabelecidos para a web, com menus e links	X		
RNF.04	Todos	<i>Atratividade</i> - Capacidade do produto de software ser atraente ao usuário. O sistema deverá conter os padrões de projeto utilizados no projeto Conhecer.		X	
RNF.05	Todos	<i>Comportamento em relação ao tempo</i> - Capacidade do produto de software fornecer um tempo de resposta e de processamento e taxas de throughput aceitáveis ao realizar suas funções sob condições estabelecidas.		X	
RNF.06	Todos	<i>Testabilidade</i> - Capacidade do produto de software permitir que o software modificado seja validado.	X		
RNF.07	Todos	<i>Capacidade para ser Instalado</i> - Capacidade do produto de software ser instalado em um ambiente especificado.		X	
RNF.08	Todos	<i>Modificabilidade</i> - Capacidade do produto de software permitir que uma modificação especificada seja implementada.			X

I.6 Caso de Uso e Processos de Apoio

Foram identificados dois módulos distintos de apoio às atividades de *benchmarking* de Iniciativas de Melhorias em Processos de Software: Módulo de Gerência da Base de *Benchmarking* e Módulo de Realização de *Benchmarking* de Iniciativas de Melhorias em Processos de Software. Para cada um dos módulos foram identificados casos de uso e definido um processo de apoio para guiar a execução dos módulos a serem construídos. Os casos de uso e os processos de apoio atendem às necessidades identificadas através dos requisitos de cliente, funcionais e não-funcionais. Esses casos de uso e processos de apoio estão descritos nas subseções a seguir.

Os casos de uso foram classificados segundo o seguinte critério de complexidade: ‘A’ significa que a complexidade do requisito é alta, ‘M’ significa que a complexidade do requisito é média e ‘S’ significa que a complexidade do requisito é baixa.

I.6.1 Gerência da Base de *Benchmarking*

A Gerência da Base de *Benchmarking* será apoiada pelo processo definido na Figura

I.2.

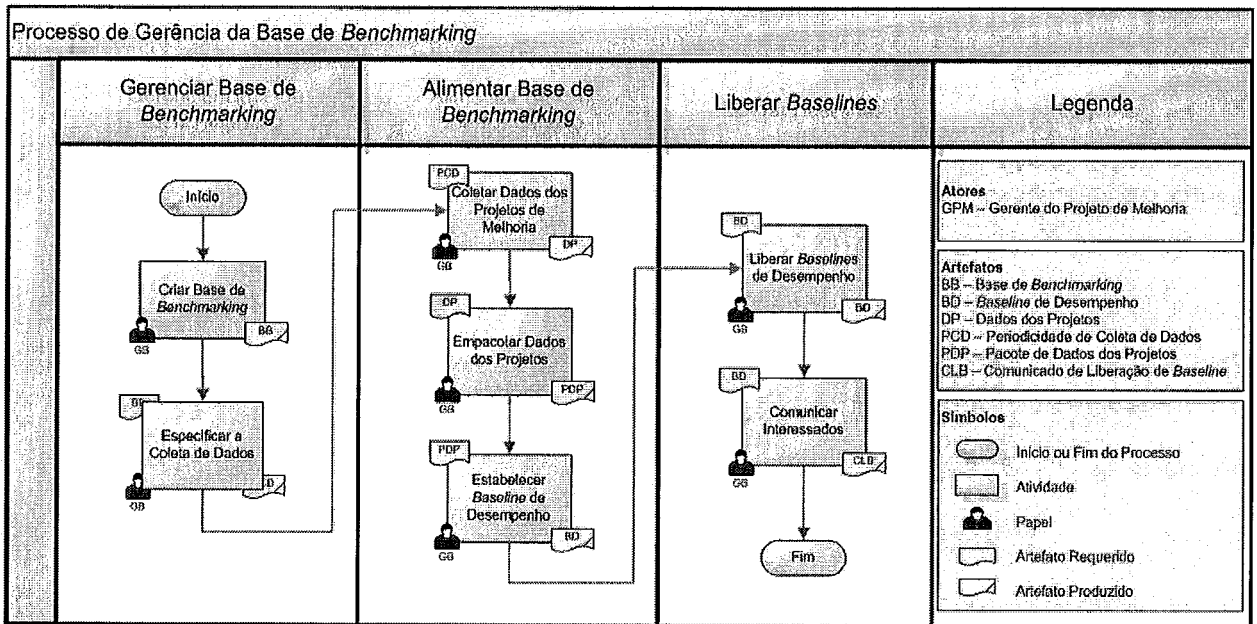


Figura I.2 – Processo de Gerência da Base de *Benchmarking*

As atividades do processo de Gerência da Base de *Benchmarking* serão atendidas pelos casos de uso identificados Tabela I.4. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela I.4 – Casos de uso do processo de Gerência da Base de *Benchmarking*

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.02	RF.01	Manter Base de <i>Benchmarking</i>		X	
UC.03	RF.04	Manter <i>Baseline</i> de Desempenho		X	
UC.04	RF.05	Manter Perspectivas de Visualização			X
UC.05	RF.04	Empacotar Dados do <i>Benchmarking</i>		X	
UC.06	RF.02	Manter Projetos de <i>Benchmarking</i>		X	
UC.07	RF.03	Manter <i>Benchmarks</i>			X
UC.08	RF.08	Manter Itens de Caracterização			X
UC.09	RF.07	Caracterizar Projetos de <i>Benchmarking</i>		X	

A Figura I.3 apresenta o relacionamento entre os atores e casos de uso derivados da análise do processo de Gerência da Base de *Benchmarking*.

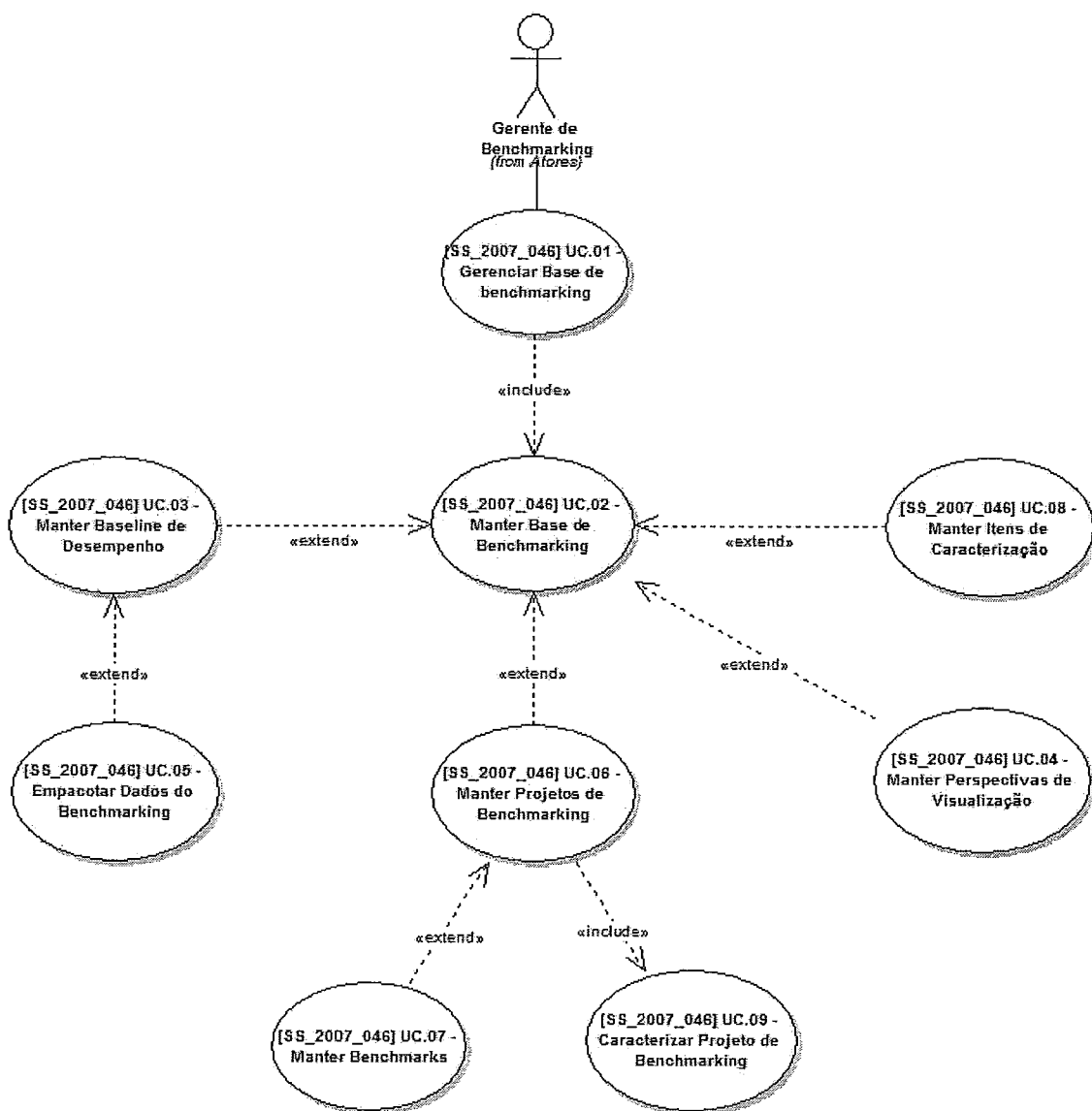


Figura I.3 – Atores e Casos de Uso do processo de Gerência da Base de *Benchmarking*

I.6.2 Realização de *Benchmarking*

A Figura I.4 apresenta o processo de apoio à Realização de *Benchmarking* de iniciativas de melhorias em processos de software.

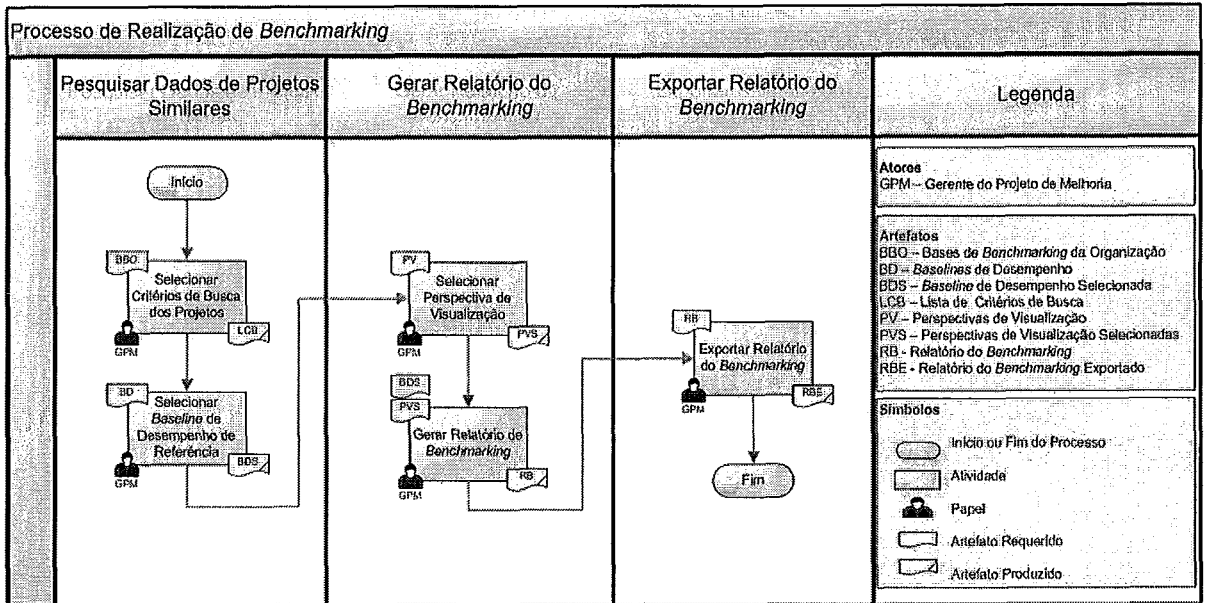


Figura I.4 – Processo de Realização de *Benchmarking*

As atividades do processo de Realização de *Benchmarking* serão atendidas pelos casos de uso identificados na Tabela I.5. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela I.5 – Casos de uso do processo de Realização de *Benchmarking*

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.12	RF.08	Realizar <i>Benchmarking</i>	X		
UC.13	RF.09	Pesquisar Projetos Similares	X		
UC.14	RF.10	Gerar Relatório de <i>Benchmarking</i>		X	
UC.15	RF.11	Exportar Resultados			X

A Figura I.5 apresenta o relacionamento entre os atores e casos de uso derivados da análise do processo de Realização de *Benchmarking*.

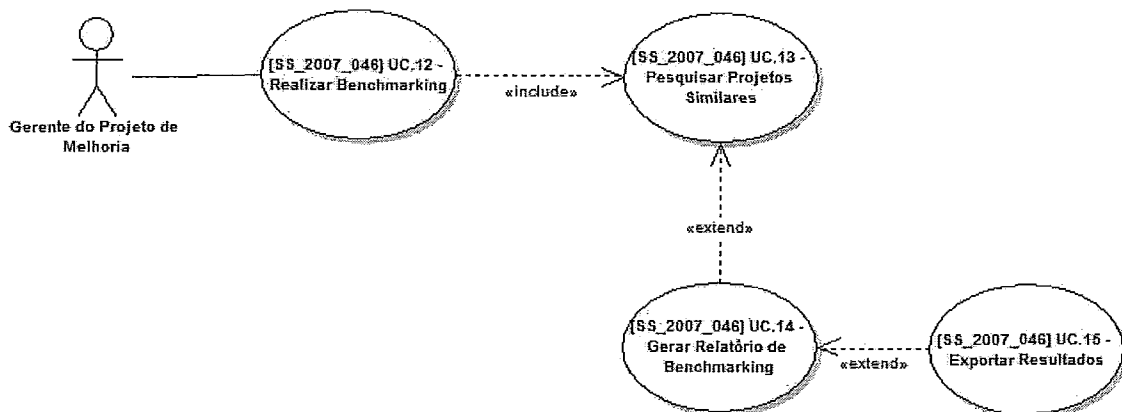


Figura I.5 – Atores e Casos de Uso do processo de Realização de *Benchmarking*

A Figura I.6 apresenta a rastreabilidade entre os requisitos de cliente, requisitos funcionais e casos de uso do ferramental de apoio às atividades de *benchmarking* de iniciativas de melhorias em processos de software.

I.7 Requisitos de Produto do CORE-KM

A Tabela I.6 apresenta os requisitos de produto do CORE-KM com os quais estão associados os requisitos funcionais definidos para o apoio ferramental.

Tabela I.6 – Requisitos de Produto do CORE-KM

ID RP	Descrição
RP.01	Ser customizável
RP.02	Possuir interface consistente
RP.03	Possuir um modelo de armazenamento de dados comum
RP.04	Fornecer assistência inteligente ao usuário
RP.05	Coletar e organizar sistematicamente o conhecimento de diversas fontes
RP.06	Ser capaz de construir novo conhecimento
RP.07	Preservar o conhecimento
RP.08	Apresentar o conhecimento de forma passiva e ativa
RP.09	Ser capaz de valorar o conhecimento
RP.10	Explorar a realimentação do usuário para manter e evoluir o ambiente
RP.11	Ser integrável ao(s) sistema(s) de trabalho já existente

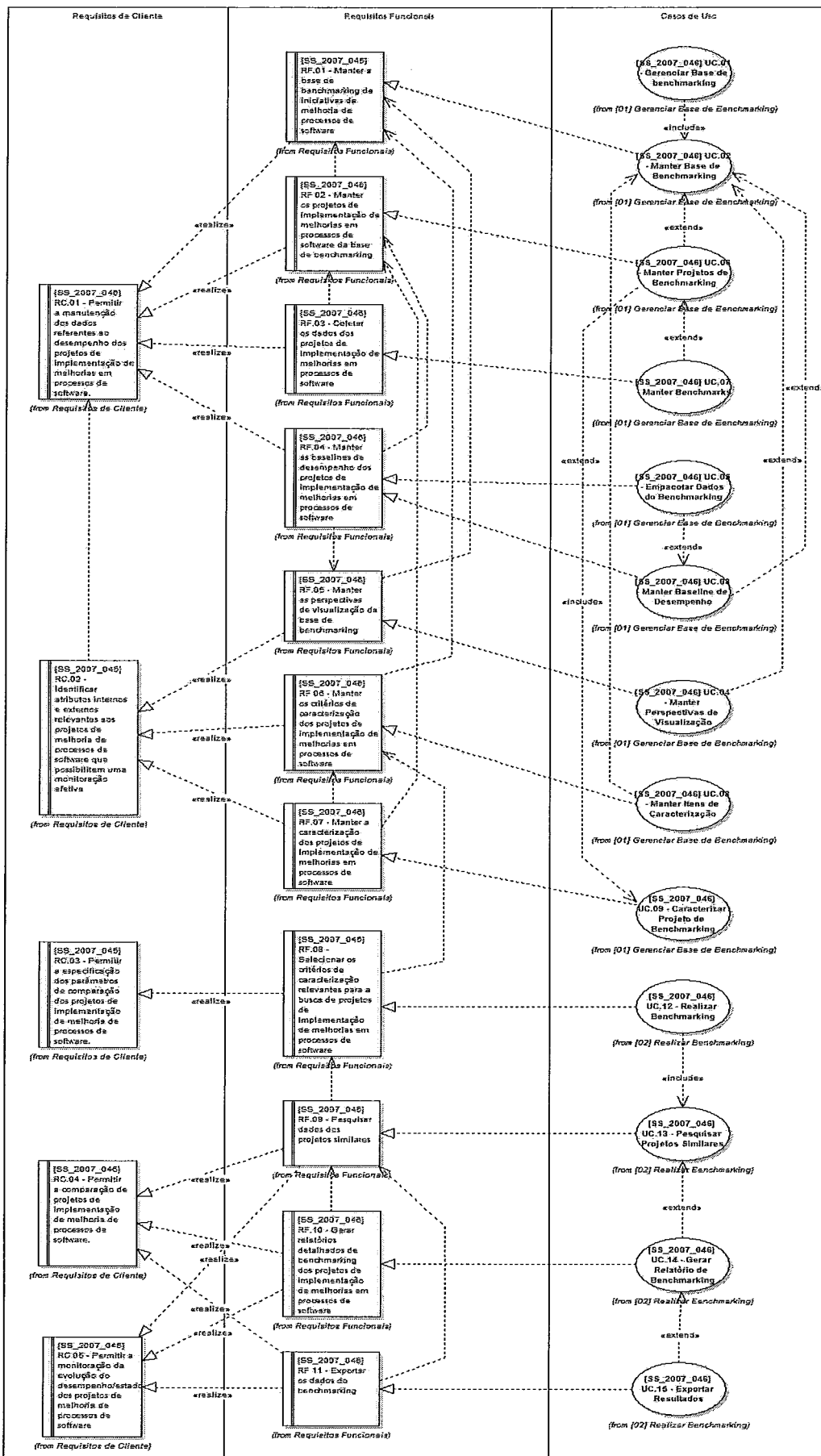


Figura I.6 – Rastreabilidade entre os Requisitos e Casos de Uso