


AMBIENTES DE ENGENHARIA DE SOFTWARE ORIENTADOS A
CORPORAÇÃO

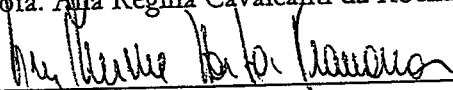
Gleison dos Santos Souza

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS
PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE
FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM
ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

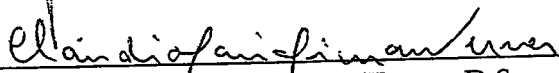
Aprovada por:



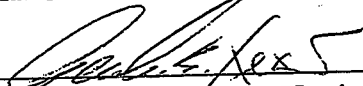
Prof. Ana Regina Cavalcanti da Rocha, D.Sc.



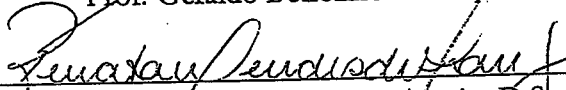
Prof. Guilherme Horta Travassos, D.Sc.



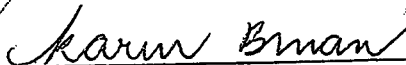
Prof. Cláudia Maria Lima Werner, D.Sc.



Prof. Geraldo Boñorino Xexéo, D.Sc.



Prof. Renata Mendes de Araujo, D.Sc.



Prof. Karin Koogan Breitman, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

JULHO DE 2008

SOUZA, GLEISON DOS SANTOS

Ambientes de Engenharia de Software
Orientados a Corporação [Rio de Janeiro] 2008

XIX, 319 p. 29,7 cm (COPPE/UFRJ, D.Sc.,
Engenharia de Sistemas e Computação, 2008)

Tese - Universidade Federal do Rio de Janeiro,
COPPE

1. Ambientes de Desenvolvimento de Software
2. Processos de Software
3. Qualidade de Software

I. COPPE/UFRJ II. Título (série)

AGRADECIMENTOS

À minha família pelo apoio e compreensão.

À Ana Regina, por acreditar em mim, pela orientação, pelas muitas oportunidades de aprendizado durante todos estes anos e pela amizade.

A Guilherme pela amizade, pela orientação, por acreditar no meu trabalho e por ter-me feito olhar com outros olhos quando preciso.

À Káthia pelo apoio e amizade desde o início do meu projeto final de curso, o que me proporcionou chegar até aqui.

A Mariano, Mariella e Carmem pelo tempo de trabalho na Embratel e por terem me incentivado a iniciar o doutorado.

A Karina pelo apoio e incentivo no final do mestrado e no início do doutorado.

A todos os amigos que passaram pelo Laboratório de Engenharia de Software (LENS) da COPPE/UFRJ neste período, pelo aprendizado e oportunidades de trabalho. Em especial Ana Candida, André, Anne, Cristina, David, Gustavo Barbosa, Gustavo Coutinho, Mariano, Paula, Peter, Reinaldo, Rômulo Coutinho, Sávio e Sômulo. Aos companheiros de doutorado Adriano, Ahilton, Andrea, Fábio, Leonardo Murta, Monalessa e Tayana.

Aos amigos que compartilharam comigo várias angústias, alegrias, ausências, descontentamento e aspirações ao longo de todo o tempo, em especial a Thadeu, Mariano Jonice e Mônica.

A Luciano e Rafael por terem trabalhado para a construção da primeira versão da Estação Taba na Web. A Cristina e Peter por terem continuado e evoluído o trabalho. A Gustavo Barbosa, Gustavo Coutinho, Rômulo Coutinho e Reinaldo por terem contribuído decididamente para a implementação das ferramentas.

Aos professores Cláudia Werner, Geraldo Xexéo, Karin Breitman e Renata Araújo, por aceitarem contribuir com este trabalho participando da banca.

A Ricardo Falbo pelo auxílio sempre que foi preciso.

À Taisa Guidini, Ângela Coppieters, Ana Paula Prata, Claudia Prata, Solange, Sônia, Mercedes, Flávia e demais funcionários do PESC pelo auxílio, sempre que necessário. Aos demais professores do programa, pelo aprendizado proporcionado ao longo das disciplinas.

Aos membros dos Centros de Computação da Aeronáutica em São José dos Campos e Brasília por contribuírem na realização do exemplo de uso deste trabalho.

Ao CNPq e ao PESC, pelo apoio financeiro.

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

AMBIENTES DE ENGENHARIA DE SOFTWARE ORIENTADOS A CORPORAÇÃO

Gleison dos Santos Souza

Julho/2008

Orientadores: Ana Regina Cavalcanti da Rocha
Guilherme Horta Travassos

Programa: Engenharia de Sistemas e Computação

Esta tese propõe os Ambientes de Engenharia de Software Orientados a Corporação (AESCorp) como sendo ambientes de engenharia de software que fornecem o apoio computacional que possibilita a uma corporação, em relação aos processos de software, gerenciar a diversidade e os estágios de maturidade de cada uma das organizações que a compõem de forma adequada às suas necessidades através da definição de ambientes que apóiam a execução, gerência e melhoria dos diversos processos de Engenharia de Software, que incluem, mas não se limitam aos de desenvolvimento e de manutenção. Esse apoio computacional, também, permite às corporações e organizações serem capazes de gerenciar e controlar os diversos processos de software de que dispõem e/ou necessitam, bem como o conhecimento organizacional envolvido através da configuração de ambientes para corporações e, a partir desses, para as organizações que as compõem. Além disso, esta tese propõe um modelo que especifica os componentes necessários para que tais ambientes contemplem os seus requisitos. Por fim, uma estratégia para a construção de AESCorps a partir de um meta-ambiente foi definida, implementada e utilizada, evidenciando o apoio fornecido pelas ferramentas desenvolvidas.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

CORPORATE-ORIENTED SOFTWARE ENGINEERING
ENVIRONMENTS

Gleison dos Santos Souza

July/2008

Advisors: Ana Regina Cavalcanti da Rocha
Guilherme Horta Travassos

Department: Computer Science and System Engineering

This thesis defines Corporate-Oriented Software Engineering Environments (COSEE) as software engineering environments that provides corporations the computational support to manage software processes diversity and maturity of its subordinated organizations. COSEE also allows the definition of environments to support the enactment, management and improvement of the software engineering processes involved. This computational support also allows corporations and organizations to manage and control their software processes and knowledge involved through the configuration of specific environments. Finally, a strategy for the construction of COSEE from a meta-environment was defined, implemented and used demonstrating the support provided by the tools developed.

ÍNDICE

Capítulo 1 - Introdução	1
1.1 Motivação	1
1.2 Cenário.....	2
1.3 Objetivo.....	4
1.4 Organização do Texto	6
Capítulo 2 - Melhoria de Processos de Software	8
2.1 Processos de Software	8
2.2 Normas e Modelos de Qualidade de Processos de Software	11
2.2.1 Norma ISO/IEC 12207	11
2.2.2 Norma ISO/IEC 15504.....	13
2.2.3 CMMI-DEV.....	14
2.2.4 MPS.BR.....	16
2.3 Melhoria de Processos de Software em Organizações	18
2.4 Melhoria de Processos de Software em Corporações.....	19
2.5 Diversidade de Processos de Software	24
2.6 Considerações Finais	25
Capítulo 3 - Infra-estrutura para Iniciativas de Melhoria de Processos de Software.....	27
3.1 Introdução.....	27
3.2 Apoio Ferramental a Iniciativas de Melhoria de Processos.....	28
3.3 Ambientes para Apoio a Iniciativas de Melhoria de Processos.....	33
3.3.1 Ambientes de Desenvolvimento da COPPE/UFRJ.....	36
3.3.1.1 Estação Taba	36
3.3.1.2 CORE-KM: Um Ambiente Customizável de Gerência de Conhecimento.....	43
3.3.2 Outros Ambientes	44
3.4 Considerações Finais	49
Capítulo 4 - Ambientes de Engenharia de Software Orientados a Corporação	50
4.1 Introdução.....	50
4.2 Evoluindo de Ambientes de Desenvolvimento de Software para Ambientes de Engenharia de Software.....	52
4.2.1 Definição de Processos.....	52

4.2.2	Execução de Processos.....	54
4.2.3	Melhoria de Processos	54
4.2.4	Gerência dos Ativos de Processos.....	55
4.2.5	Gerência de Conhecimento	55
4.2.6	Requisitos de um Ambiente de Engenharia de Software	56
4.3	Evoluindo de Ambientes Orientados a Organização para Ambientes Orientados a Corporação.....	56
4.3.1	Definição de Processos.....	57
4.3.2	Execução de Processos.....	58
4.3.3	Melhoria de Processos	58
4.3.4	Gerência dos Ativos de Processos.....	59
4.3.5	Gerência de Conhecimento	60
4.3.6	Disponibilização de Infra-estrutura	60
4.3.7	Requisitos de um Ambiente Orientado a Corporação.....	61
4.4	Modelo para Construção de Ambientes de Engenharia de Software Orientados a Corporação.....	62
4.4.1	Definição e Geração de Ambientes.....	63
4.4.1.1	Meta-Ambiente	64
4.4.1.2	Ambiente Corporativo.....	65
4.4.1.3	Ambiente Organizacional.....	65
4.4.1.4	Ambiente de Projeto	65
4.4.1.5	Ambientes de Engenharia de Software Orientados a Corporação	66
4.4.2	Definição de Processos.....	66
4.4.2.1	Definição de Processos para a Corporação	69
4.4.2.2	Definição de Processos para a Organização.....	71
4.4.2.3	Definição de Processos para os Projetos	71
4.4.3	Melhoria de Processos	72
4.4.3.1	Melhoria de Processos Determinada pela Corporação	73
4.4.3.2	Melhoria de Processos em Projetos em Execução	75
4.4.4	Execução de Processos.....	77
4.4.5	Gerência de Ativos de Processo.....	78
4.4.5.1	Controle de versão.....	78
4.4.5.2	Controle de modificação.....	79

4.4.5.3	Rastreabilidade de versões e mudanças	80
4.4.5.4	Bases de Ativos de Processo e Base de Versões de Ativos de Processo	81
4.4.6	Gerência de Conhecimento	81
4.5	Considerações Finais	82
Capítulo 5	- Definição e Construção de Ambientes de Engenharia de Software Orientados a Corporação na Estação Taba	83
5.1	Introdução.....	83
5.2	Funções e Ambientes da Estação Taba	83
5.3	Requisitos dos Ambientes da Estação Taba	85
5.4	Ontologia de Corporação	89
5.4.1	Organização e Corporação.....	90
5.4.2	Processo Padrão, Processo Especializado e Processo Definido para o Projeto.....	91
5.4.3	Ativo de Processo.....	94
5.5	Arquitetura da Estação Taba e dos Ambientes de Engenharia de Software Orientados a Corporação.....	95
5.5.1	Histórico de Evolução da Arquitetura da Estação Taba	95
5.5.1.1	A Arquitetura da Estação Taba de 1990 a 1999.....	96
5.5.1.2	Arquitetura da Estação Taba de 1999 a 2003 – Implementação dos ADSOD.....	98
5.5.1.3	Arquitetura da Estação Taba de 2003 a 2007 – Implementação dos ADSOrg.....	99
5.5.2	A Implementação Atual da Estação Taba em Plataforma Web	101
5.5.2.1	A Estação Taba na Web	102
5.5.2.2	Arquitetura Atual da Estação Taba.....	106
5.5.2.3	Integração de Gerenciamento e Controle.....	108
5.5.2.4	Integração de Apresentação e Interação	113
5.5.2.5	Integração de Dados – Mapeamento Objeto-Relacional na Estação Taba.....	114
5.5.2.6	Integração de Dados – Repositórios dos Ambientes	114
5.5.2.7	Integração de Conhecimento	115
5.5.3	Avaliação do Modelo de Integração da Estação Taba.....	116
5.6	Modelo de Desenvolvimento da Estação Taba.....	117

5.6.1	Processo do Escritório de Projetos da Estação Taba	117
5.6.2	Processo de Desenvolvimento	118
5.6.3	Estratégia de Versionamento	119
5.6.4	Ferramentas de Apoio	120
5.7	Modelo para Construção dos Ambientes de Engenharia de Software Orientados a Corporação na Estação Taba.....	120
5.7.1	Gerência de Ambientes	121
5.7.2	Gerência de Processos.....	125
5.7.3	Gerência de Ativos de Processos.....	128
5.7.4	Gerência de Execução de Processos	130
5.7.5	Gerência de Conhecimento	132
5.8	Cenário de Uso dos Ambientes de Engenharia de Software Orientados a Corporação.....	133
5.8.1	Iniciativa de Melhoria de Processos na Aeronáutica.....	134
5.8.2	Apoio Ferramental	135
5.8.3	Resultados da Iniciativa de Melhoria de Processos	144
5.9	Considerações Finais	144
Capítulo 6 - Conclusão e Perspetivas Futuras.....		145
6.1	Conclusão.....	145
6.1.1	Contribuições.....	146
6.2	Perspetivas Futuras	147
Referências		149
Anexo I - Linguagem para Modelagem de Processos Organizacionais		170
1	Introdução.....	170
2	Linguagem para Modelagem de Processos Organizacionais	172
Referências Bibliográficas		179
Anexo II - Perfil UML para Modelagem de Ontologias		180
1	Perfil UML para Modelagem de Ontologias.....	180
Referências Bibliográficas		182

Anexo III - Estudo Baseado em Revisão Sistemática – Infra-estrutura de Apoio A	
	Iniciativas de Melhoria de Processos de Software 183
1	Introdução..... 183
2	Protocolo do Estudo Baseado em Revisão Sistemática 185
2.1	Contexto..... 185
2.2	Objetivo e Questões de Pesquisa 185
2.2.1	Objetivo..... 185
2.2.2	Questões de pesquisa 186
2.3	Escopo da pesquisa 187
2.3.1	Critérios adotados para seleção das fontes..... 188
2.3.2	Restrições 188
2.4	Idiomas 188
2.5	Métodos de Busca de Publicações..... 189
2.5.1	Expressão de Busca 189
2.5.2	Busca Manual..... 190
2.6	Procedimentos de Seleção e Critérios..... 190
2.6.1	Procedimentos de Seleção 190
2.6.2	Critérios de Inclusão..... 195
2.7	Procedimentos para Extração dos Dados..... 195
2.7.1	Na seleção e catalogação preliminar dos dados coletados 195
2.7.2	Na seleção dos dados relevantes..... 195
2.7.3	Extração de Dados 195
2.7.4	Sumarização dos resultados..... 197
2.8	Procedimentos para Análise 197
2.8.1	Análise Quantitativa 197
2.8.2	Análise Qualitativa 198
3	Planejamento e Execução..... 198
3.1	Definição do Escopo e Estudos Preliminares 198
3.2	Identificação de Publicações de Controle e Palavras-Chave 199
3.2.1	Primeira Rodada..... 200
3.2.2	Segunda Rodada..... 200
3.2.3	Terceira Rodada 202
3.3	Definição das Máquinas de Busca 202
3.3.1	Expressão de Busca na Biblioteca Digital da Compendex 203

3.3.2	Expressão de Busca na Biblioteca Digital da IEEE	203
3.3.3	Expressão de Busca na Biblioteca Digital da Scopus	203
3.3.4	Instrumento para Consulta Manual.....	203
3.4	Identificação do Período de Busca.....	204
3.5	Execução do Protocolo.....	205
3.6	Considerações sobre o Resultado do Estudo	207
3.6.1	Caracterização das Organizações.....	208
3.6.2	Caracterização do Apoio Ferramental	210
4	Apoio Ferramental para Condução do Estudo	212
4.1	Análise da Expressão de Busca.....	212
4.2	Análise das Publicações.....	213
4.3	Relatórios	214
5	Dados Coletados.....	215
5.1	Dados das Publicações Dentro do Escopo do Estudo.....	216
5.2	Dados das Publicações Seleccionadas a Partir das Palavras-Chave.....	239
5.3	Dados das Publicações Seleccionadas no Primeiro Filtro	253
	Referências Bibliográficas	259
	Anexo IV - Ontologia de organização	260
1	Introdução.....	260
2	Definição e Formalização da Ontologia.....	261
2.1	Subontologia de Capital Intelectual.....	264
2.1.1	Taxonomia de Competência	264
2.1.2	Interação entre Experiência e Conhecimento	265
2.1.3	Disponibilidade de Competências.....	265
2.1.4	Decomposição de Domínio de Conhecimento.....	266
2.2	Subontologia de Estrutura.....	267
2.2.1	Decomposição da Organização	267
2.2.2	Distribuição de Autoridade e Responsabilidade entre Unidades Organizacionais.....	269
2.2.3	Decomposição de Unidade Organizacional.....	271
2.2.4	Distribuição de Autoridade e Responsabilidade entre Posições.....	271
2.2.5	Especificação de Cargos e Posições.....	272
2.2.6	Preenchimento de Vagas e Formação de Equipes.....	273

2.2.7	Definição de Objetivos	273
2.3	Subontologia de Artefatos	276
2.3.1	Decomposição de Artefato.....	277
2.3.2	Taxonomia de Artefato	277
2.4	Subontologia de Comportamento	280
2.4.1	Atividade como Ação de Transformação.....	280
2.4.2	Taxonomia de Atividade.....	281
2.4.3	Decomposição de Processo e Atividade	283
2.4.4	Subtipos de Processo.....	285
2.4.5	Adoção de Procedimento	287
2.4.6	Taxonomia de Procedimento.....	288
2.4.7	Método como Procedimento Sistemático	289
2.4.8	Automatização de Procedimento	290
2.4.9	Processos definidos na Organização e Normas Associadas.....	291
2.4.10	Projetos da Organização	291
2.4.11	Ativos de Processo	292
2.5	Subontologia de Estratégia Geral.....	293
2.5.1	Domínios de Atuação.....	294
2.5.2	Artefatos e Serviços oferecidos pela Organização	294
2.5.3	Relação com Organizações Clientes.....	294
	Referências Bibliográficas	295
	Anexo V - Especificação de Requisitos	298
1	Necessidades do Cliente	298
2	Modelo de Domínio	299
2.1	Gerência de Ambientes.....	299
2.2	Gerência de Ativos de Processos	299
2.3	Gerência de Organização.....	300
2.4	Gerência de Processos	300
2.5	Gerência de Execução de Processos.....	301
3	Requisitos do Cliente.....	301
3.1	Gerência de Ambientes.....	302

3.2	Gerência de Ativos de Processos	302
3.3	Gerência de Organização.....	303
3.4	Gerência de Processos	303
3.5	Gerência de Execução de Processos.....	303
4	Requisitos Funcionais de Software.....	304
4.1	Gerência de Ambientes.....	304
4.2	Gerência de Ativos de Processos	305
4.3	Gerência de Organização.....	306
4.4	Gerência de Processos	306
4.5	Gerência de Execução de Processos.....	307
5	Caso de Uso e Processos de Apoio.....	308
5.1	Gerência de Ambientes.....	308
5.2	Gerência de Ativos de Processos	311
5.3	Gerência de Organização.....	313
5.4	Gerência de Processos	315
6	Requisitos de Produto da Estação Taba.....	317

ÍNDICE DE TABELAS

Tabela 2.1 – Níveis de Capacidade dos Processos da Norma ISO/IEC 15504 (ISO/IEC, 2003)	14
Tabela 2.2 – Áreas de Processo do CMMI-DEV	15
Tabela 2.3 – Processos e Atributos de Processo do MPS.BR.....	17
Tabela 5.1 – Requisitos da Estação Taba e de seus Ambientes.....	86
Tabela 5.2 – Requisitos de Arquitetura do Produto da Estação Taba.....	103
Tabela 5.3 – Solução Atual para Implementação dos Principais Requisitos de Arquitetura do Produto da Estação Taba.....	104
Tabela 5.4 – Repositórios Presentes nos AESCorp	115
Anexo I	
Tabela 1 – Definição e Notação dos Objetos	172
Tabela 2 – Definição e Notação dos Adornos.....	174
Tabela 3 – Definição e Notação das Áreas.....	174
Tabela 4 – Definição e Notação das Ligações.....	175
Anexo III	
Tabela 1 – Padrão para Descrição do Uso de Apoio Ferramental em Iniciativas de Melhoria de Processos	208
Tabela 2 – Artigos Seleccionados pelas Palavras-Chave.....	239
Tabela 3 – Artigos Lidos Após Aplicação do Primeiro Filtro	253
Anexo V	
Tabela 1 – Requisitos de Cliente da Gerência de Ambientes.....	302
Tabela 2 – Requisitos de Cliente da Gerência de Ativos de Processos	302
Tabela 3 – Requisitos de Cliente da Gerência de Organização.....	303
Tabela 4 – Requisitos de Cliente da Gerência de Processos	303
Tabela 5 – Requisitos de Cliente da Gerência de Execução de Processos.....	304
Tabela 6 – Requisitos Funcionais – Gerência de Ambientes.....	304
Tabela 7 – Requisitos Funcionais – Gerência de Ativos de Processos.....	305
Tabela 8 – Requisitos Funcionais – Gerência de Organização.....	306
Tabela 9 – Requisitos Funcionais – Gerência de Processos.....	307

Tabela 10 – Requisitos Funcionais – Gerência de Execução de Processos.....	307
Tabela 11 – Casos de uso do processo Definir Ambientes Configurados.....	309
Tabela 12 – Casos de uso do processo Definir Ambientes de Projetos.....	309
Tabela 13 – Casos de uso dos processos Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais e Alterar Ambientes de Projetos	310
Tabela 14 – Casos de uso do processo Definir Base de Ativos de Processos.....	311
Tabela 15 – Casos de uso do processo Gerenciar Base de Ativos de Processos	312
Tabela 16 – Casos de uso dos processos Descrever Corporação e Descrever Organização	315
Tabela 17 – Casos de uso do processo Manter Cadastros.....	315
Tabela 18 – Casos de uso do processo Definir Processos e seus subprocessos	317
Tabela 19 – Casos de uso do processo Adaptar Processo para o Projeto	317
Tabela 20 – Requisitos de Produto da Estação Taba.....	318
Tabela 21 – Requisitos da Arquitetura do Produto da Estação Taba.....	318

ÍNDICE DE FIGURAS

Figura 2.1 – Framework para promoção das Atividades de Melhoria de Processos na Toshiba (OGASAWARA <i>et al.</i> , 2006)	21
Figura 3.2 – Esquema para Construção de ADS na Estação Taba	38
Figura 4.3 – Geração de Ambientes.....	64
Figura 4.4 – Abordagem para Definição de Processos (VILLELA, 2004)	67
Figura 4.5 – Definição de Processos para a Corporação	70
Figura 4.6 – Definição de Processos para a Organização.....	71
Figura 4.7 – Definição de Processos para Projetos	72
Figura 4.8 – Legenda para o cenário de melhoria	73
Figura 4.9 – Uso dos processos pelos projetos e melhoria de processos na Organização ...	74
Figura 4.10 – Uso do processo evoluído em novos projetos.....	74
Figura 4.11 – Avaliação e melhoria dos processos pela Corporação	75
Figura 4.12 – Evolução dos processos organizacionais a partir de decisão corporativa.....	75
Figura 4.13 – Legenda para o cenário de melhoria para processos em execução	76
Figura 4.14 – Processos em execução na organização	76
Figura 4.15 – Implantação de melhoria em projetos em execução	76
Figura 5.16 – Ambientes da Estação Taba.....	85
Figura 5.17 – Relações entre Organização e Corporação	90
Figura 5.18 – Subtipos de processos segundo a natureza do processo	92
Figura 5.19 – Conceitos derivados de Processo.....	94
Figura 5.20 – Serviços Básicos da Estação Taba.....	107
Figura 5.21 – Detalhes de Implementação da Estação Taba na Web e dos AESCorp.....	108
Figura 5.22 – Componentes do Meta-Ambiente.....	110
Figura 5.23 – Serviços dos Ambientes Corporativos e Organizacionais.....	111
Figura 5.24 – Serviços dos Ambientes de Projetos.....	112
Figura 5.25 – Padronização de Interface.....	113
Figura 5.26 – Visão Geral do Processo do Escritório de Projetos.....	118
Figura 5.27 – Notação dos Diagramas de Requisitos.....	121
Figura 5.28 – Definição de Ambientes Configurados – Requisitos e Casos de Uso	122
Figura 5.29 – Definição de Ambientes de Projeto – Requisitos e Casos de Uso.....	123
Figura 5.30 – Definição de Ambiente Configurado – Processo de Apoio	124
Figura 5.31 – Definição de Ambiente Configurado – Definir Ambiente Configurado	124

Figura 5.32 – Definição de Ambiente de Projeto – Processo de Apoio	124
Figura 5.33 – Definição de Processos Padrão e Especializados – Requisitos e Casos de Uso	126
Figura 5.34 – Definição de Processos Definidos para o Projeto – Requisitos e Casos de Uso.....	127
Figura 5.35 – Biblioteca de Ativos de Processos – Requisitos e Casos de Uso	129
Figura 5.36 – Gerência de Versão de Ativos de Processos – Requisitos e Casos de Uso .	130
Figura 5.37 – Execução de Processos nos Ambientes de Projeto – Requisitos e Casos de Uso.....	131
Figura 5.38 – Definição da Estrutura Organizacional I – Requisitos e Casos de Uso	132
Figura 5.39 – Definição da Estrutura Organizacional II – Requisitos e Casos de Uso	133
Figura 5.40 – Definição de Ambiente Configurado – Definir Ambiente Configurado.....	135
Figura 5.41 – Definição de Ambiente Configurado – Caracterizar Contexto de Configuração	136
Figura 5.42 – Definição de Ambiente Configurado – Selecionar Ativos de Processos	138
Figura 5.43 – Definição de Ambiente Configurado – Gerar Ambiente Configurado.....	138
Figura 5.44 – Definição de Processos Especializados – Implementação.....	139
Figura 5.45 – Descrição da Estrutura Organizacional da Corporação – Implementação .	140
Figura 5.46 – Biblioteca de Ativos de Processos - Implementação	141
Figura 5.47 – Gerência de Configuração de Ativos de Processos - Implementação.....	141
Figura 5.48 – Definição de Ambiente de Projeto – Implementação	142
Figura 5.49 – Execução de Processos nos Ambientes de Projeto - Implementação.....	143
Anexo I	
Figura 1 - Primitivas utilizando a Notação do Diagrama de Atividade UML.....	178
Figura 2 - Primitivas utilizando a Notação Proposta.....	179
Anexo II	
Figura 1 – Perfil UML (MIAN, 2003)	180
Anexo III	
Figura 1 – Quadro com Palavras-Chave	205
Figura 2 – Quadro para Controle de Pesquisa Manual	205
Figura 3 – Artigos Retornados pela Expressão de Busca	206

Figura 4 – Artigos Retornados que Passaram pelo Primeiro Filtro	207
Figura 5 – Artigos Retornados que Passaram pelo Segundo Filtro.....	208
Figura 6 – Tamanho das Organizações	210
Figura 7 – Agregação das Organizações na Iniciativa de Melhoria de Processos	210
Figura 8 – Influência de Modelos, Normas, Técnicas ou Métodos	211
Figura 9 – Abrangência do Apoio Ferramental.....	212
Figura 10 – Agregação das Ferramentas Utilizadas	212
Figura 11 – Uso de Apoio Ferramental.....	213
Figura 12 – Tela de Teste de Expressão de Busca.....	214
Figura 13 – Transformação de Arquivo no Formato BibTeX para BibXML.....	215
Figura 14 – Formulário de Coleta de Informações.....	216
Figura 15 – Relatório Final do Estudo Baseado em Revisão Sistemática	217
Figura 16 – Relatório Auxiliar para Análise de Abstracts	217

Anexo IV

Figura 1 – Subontologias da Ontologia de Organização	263
Figura 2 – Subontologia de Capital Intelectual	264
Figura 3 – Decomposição da Organização	268
Figura 4 – Distribuição de Autoridade e Responsabilidade	269
Figura 5 – Esquema das Estruturas adotadas nas Organizações.....	270
Figura 6 – Decomposição de Unidade Organizacional.....	271
Figura 7 – Distribuição de Autoridade e Responsabilidade	272
Figura 8 – Especificação de Cargos e Posições.....	272
Figura 9 – Preenchimento de Vagas e Formação de Equipe	273
Figura 10 – Definição de Objetivos.....	274
Figura 11 – Subontologia de Artefatos.....	277
Figura 12 – Subontologia de Comportamento - Atividade como Ação de Transformação	281
Figura 13 – Taxonomia de Atividade quanto ao Papel.....	282
Figura 14 – Taxonomia de Atividade quanto à Natureza.....	283
Figura 15 – Decomposição de Processo e de Atividade.....	284
Figura 16 – Subtipos de processos segundo a natureza do processo.....	286
Figura 17 – Conceitos derivados de Processo	288
Figura 18 – Adoção de Procedimento.....	288

Figura 19 – Taxonomia de Procedimento.....	289
Figura 20 – Método como Procedimento Sistemático.....	290
Figura 21 – Automatização de Procedimento	291
Figura 22 – Processos definidos na Organização e Normas Associadas.....	292
Figura 23 – Projetos da Organização.....	292
Figura 24 – Subontologia de Estratégia Geral.....	294

Anexo V

Figura 1 – Diagrama de domínio da Gerência de Ambientes.....	299
Figura 2 – Diagrama de domínio da Gerência de Ativos de Processos	300
Figura 3 – Diagrama de domínio da Gerência de Organização.....	300
Figura 4 – Diagrama de domínio da Gerência de Processos	301
Figura 5 – Diagrama de domínio da Gerência de Execução de Processos.....	301
Figura 6 – Processo Definir Ambientes Configurados	308
Figura 7 – Processos Definir Ambientes de Projetos	309
Figura 8 – Processos Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais e Alterar Ambientes de Projetos.....	310
Figura 9 – Processo Definir Base de Ativos de Processos	311
Figura 10 – Processo Gerenciar Base de Ativos de Processos	312
Figura 11 – Processos Descrever Corporação e Descrever Organização - 1	313
Figura 12 – Processos Descrever Corporação e Descrever Organização - 2	314
Figura 13 – Processo Manter Cadastros.....	315
Figura 14 – Processo Definir Processos e seus subprocessos	316
Figura 15 – Processo Adaptar Processo para o Projeto.....	317

CAPÍTULO 1 - INTRODUÇÃO

1.1 Motivação

Uma característica cada vez mais comum no cenário de desenvolvimento de software é a existência de corporações com organizações subordinadas que desenvolvem software de forma independente entre si. Estas organizações muitas vezes possuem diferentes necessidades e realidades em relação ao desenvolvimento de software, por exemplo, nichos de mercado com características diferentes ou diferentes níveis de maturidade; corporações em que uma das organizações tem o desenvolvimento de software aderente a um modelo de maturidade e as demais desenvolvem software de forma *ad hoc*; ou ainda, corporações cujas organizações tenham seus processos institucionalizados em níveis distintos de maturidade e capacidade.

A variação destes processos se dá geralmente em função do tamanho do produto a ser desenvolvido (por exemplo, projetos de curta, média e longa duração) ou tipo de produto (por exemplo, desenvolvimento de software, desenvolvimento de software e hardware, desenvolvimento de softwares para web etc.). Os processos também podem ser influenciados pela adoção de métodos, técnicas ou abordagens de desenvolvimento. Neste aspecto, entretanto, a adoção de modelos de maturidade como o MPS.BR (Melhoria de Processo do Software Brasileiro) (SOFTEX, 2007) e o CMMI (*Capability Maturity Model Integration*) (CHRISISSIS *et al.*, 2006) desempenha papel importante ao influenciar a implementação de diversas práticas e em diferentes níveis de rigor, algo que é, de certa forma, intrínseco à filosofia dos níveis de maturidade e capacidade. O CMMI consiste de boas práticas que tratam do desenvolvimento e manutenção de produtos e serviços cobrindo o ciclo de vida de um produto desde a concepção até a entrega e a manutenção. O MPS.BR é um programa para Melhoria de Processo do Software Brasileiro e tem como objetivo definir um modelo de melhoria e avaliação de processo de software, preferencialmente para as micro, pequenas e médias empresas, de forma a atender as suas necessidades de negócio e a ser reconhecido nacional e internacionalmente como um modelo aplicável à indústria de software.

Outro fator que tem aumentado a complexidade do desenvolvimento de software ao longo do tempo é o número de processos que são executados para a conclusão de um projeto de software. Organizações desenvolvedoras de software têm necessidade de executar diversos processos integrados ao processo de desenvolvimento para a execução de

um projeto de software (por exemplo, processos de apoio, como Garantia de Qualidade) e, também, processos executados de forma independente dos processos de desenvolvimento em si, mas também relacionados à produção de software ou gerência e estruturação do desenvolvimento (por exemplo, processos como Aquisição e Gerência de Projetos de Melhoria).

A existência de diversidade de processos não é exclusividade de organizações envolvidas com desenvolvimento de software ou com grandes departamentos relacionados a TI. A General Motors, por exemplo, recentemente fez com que seus principais fornecedores de serviços de TI (dentre eles EDS, HP e IBM, com unidades espalhadas pelo mundo) se comprometessem a utilizar 44 processos padrão desenvolvidos e utilizados por ela relacionados a várias tarefas de TI (WEIER e MURPHY, 2007). A gerência de processos em grandes organizações envolve lidar com grupos e processos com diferentes características dentro da organização. Por exemplo, a estrutura do grupo corporativo responsável pelos processos de engenharia de software da Toshiba era composto em 2005 por 20 pessoas e se relacionava hierarquicamente ou colaborativamente com cerca de 60 outros grupos de organizações subordinadas ou unidades organizacionais (OGASAWARA *et al.*, 2006).

Além disso, de acordo com a importância das iniciativas de melhoria de processos para as organizações serem competitivas e, também, nas dificuldades que as organizações têm em implementar tais iniciativas devido ao custo e tempo necessários, é preciso realizar programas de melhoria mais acessíveis para a maior parte das organizações independentemente das características de cada organização (AMESCUA *et al.*, 2006). Uma forma de reduzir os custos associados com uma iniciativa de melhoria de processos de software é adotar ferramentas adequadas para automatizar certas tarefas e, assim, reduzir o esforço necessário para realizá-las. É comum em relatos sobre fatores de sucesso e dificuldades relacionadas à execução de iniciativas de melhoria de processos (NIAZI *et al.*, 2005; SANTOS *et al.*, 2007a) a identificação da necessidade de uma infra-estrutura adequada como um importante fator de sucesso. De fato, grande parte das organizações com baixos níveis de maturidade não possuem uma infra-estrutura adequada para iniciar a iniciativa de melhoria de processos de software (SANTOS *et al.*, 2007a).

1.2 Cenário

Com o crescimento, uma empresa de software tende a aumentar e mudar sua estrutura organizacional, se comportando como uma corporação, ou seja, passa a ser uma

organização composta de outras organizações (por exemplo, subsidiárias ou filiais) com certo grau de independência, mas subordinadas, de alguma forma, aos interesses e regras corporativas. O que difere uma corporação de uma organização é o fato de haver uma relação de hierarquia entre as organizações que compõem a corporação e a direção central da corporação. Esta relação de hierarquia pode estar configurada de diversas formas, por exemplo, ser uma relação apenas administrativa, sendo as organizações unidades independentes com ramos de negócio particulares, ou então ser uma relação de comando e gerência das atividades fins da organização, estando as organizações subordinadas a uma linha mestra definida pela corporação. Neste segundo caso, é comum, também, que estas organizações tenham ramos de atuação semelhantes e compartilhem alguns nichos de mercado, mas em contextos ou localidades diferentes.

Além disso, conforme a organização aumenta de tamanho, é possível que a variedade de tipos de processos também aumente de acordo com as características dos novos projetos ou com as necessidades específicas de determinados clientes ou mercados em que atua. Por exemplo, atualmente são comuns organizações com equipes diferentes em fábricas de programas (contemplando apenas a construção e, às vezes, parte do projeto de software) e/ou fábricas de software (contemplando todo o ciclo de vida de desenvolvimento de um software, desde o levantamento de requisitos até a entrega para o cliente).

Em uma corporação, a diversidade de processos se torna mais evidente à medida que as organizações subordinadas podem adotar um conjunto variado de processos. Estes processos, apesar de se basearem num modelo padrão (por exemplo, específico para uma fábrica de software) podem sofrer adaptações decorrentes de necessidades diferentes. Por exemplo, organizações que tem um modelo de desenvolvimento de fábrica de software e processos compatíveis com diferentes níveis de modelos de maturidade.

A premissa principal que auxilia no entendimento do escopo e motivação deste trabalho é que organizações independentes, corporações e organizações subordinadas a corporações e têm diferentes necessidades de infra-estrutura para a implantação de um programa de melhoria de processos de software.

Nesta configuração, a gerência e monitoração da corporação sobre os processos de software podem ultrapassar os limites físicos de uma organização. Dessa forma, em vez de gerenciar ou monitorar como uma organização está gerenciando seus processos como um todo, pode gerenciar ou monitorar, por exemplo, como todas as equipes que compõem Fábricas de Software estão tendo seus processos evoluídos para atingir e manterem-se

aderentes a um determinado nível de maturidade. Neste caso, a monitoração realizada pela corporação passa a ser sobre os processos e não sobre as organizações.

Para este trabalho interessam as corporações cujas organizações desenvolvem software e compartilham uma linha de comando delineada pela corporação no que tange seus programas de melhoria de processos de software. Também no contexto deste trabalho, uma corporação é considerada como tendo um papel fundamentalmente gerencial em relação às atividades de software desempenhadas pelas organizações. Desta forma, assume-se, por exemplo, que não há desenvolvimento de software na corporação, que estas atividades são desempenhadas apenas nas organizações. Além disso, assume-se que as organizações tenham estruturas hierárquicas internas independentes entre si e que a cadeia de comando entre a corporação e as organizações esteja restrita à elaboração e execução das iniciativas de melhoria de processos em cada organização.

1.3 Objetivo

O objetivo deste trabalho é descrever os *Ambientes de Engenharia de Software Orientados a Corporação* (AESCorp). Tais ambientes fornecem o apoio computacional que possibilita a uma corporação, em relação aos processos de software, gerenciar a diversidade e os estágios de maturidade de cada uma das organizações que a compõem de forma adequada às suas necessidades através da definição de ambientes de apoio à execução e à gerência dos processos de Engenharia de Software, que incluem, mas não se limitam aos de desenvolvimento e de manutenção. Além disso, esse apoio computacional permite às corporações e organizações serem capazes de gerenciar e controlar os diversos processos de software de que dispõem e/ou necessitam, bem como o conhecimento organizacional envolvido através da configuração de ambientes para corporações e, a partir desses, para as organizações que as compõem.

A definição e construção dos Ambientes de Desenvolvimento de Software Orientados a Organização (VILLELA, 2004) supriu a necessidade de se apoiar a utilização do conhecimento organizacional e não só o conhecimento de domínio provido por especialistas (conforme identificado por OLIVEIRA (1999)), durante o desenvolvimento de software e, também, a necessidade de se apoiar o desenvolvimento de software em uma organização em particular. Entretanto, esta abordagem também possui limitações ao não apoiar outros processos que não os de desenvolvimento e manutenção e considerar apenas a existência de organizações independentes.

Dessa forma, os Ambientes de Engenharia de Software Orientados a Corporação visam a suprir estas limitações e podem ser considerados uma evolução dos Ambientes de Desenvolvimento de Software Orientados a Organização em dois sentidos:

- (i) Evolução de Ambientes de *Desenvolvimento* de Software para Ambientes de *Engenharia* de Software, que envolve definir ambientes para outros processos além do de desenvolvimento e de manutenção, apoiando também a execução e a gerência de quaisquer processos de Engenharia de Software; e
- (ii) Evolução de Ambientes Orientados a *Organização* para Ambientes Orientados a *Corporação*, que envolve configurar ambientes para corporações e, a partir desses, para as organizações que as compõem.

Estas duas evoluções não são concorrentes, mas sim complementares. Num sentido horizontal, o espectro de processos apoiado pelos ambientes é expandido, tornando possível a existência dos mais diversos tipos de processos de software (por exemplo, fornecimento, aquisição, treinamento, avaliação e melhoria, medição etc.). Por outro lado, num sentido vertical, cria-se um novo nível de hierarquia (ou seja, um nível corporativo além do organizacional e do de projetos) para cada processo de software individualmente, além de, em decorrência disso, possibilitar novos níveis de controle para gerenciar tanto a diversidade de processos quanto os níveis hierárquicos envolvidos.

Apesar de as organizações que compõem a corporação poderem estar geograficamente distribuídas, não serão discutidos no contexto deste trabalho aspectos relacionados ao desenvolvimento de software por equipes distribuídas (por exemplo, (AUDY e PRIKLADNICKI, 2007)) nem possíveis diferenças culturais que possam afetar a forma de trabalho nas organizações (por exemplo, (SIAKAS e GEORGIADOU, 2002)).

Este trabalho também não pretende discutir os aspectos gerenciais necessários para uma adequada gerência das atividades de melhoria de processos de software em corporações e organizações, discorrendo, por exemplo, sobre lições aprendidas ou propondo métodos ou diretrizes para a implantação de iniciativas de melhoria de software neste contexto. O foco principal está na identificação de requisitos e itens que devam compor a infra-estrutura computacional necessária para a condução de uma iniciativa de melhoria de processos de software no contexto corporativo. Dessa forma, sempre que o termo *infra-estrutura* é mencionado no texto desta tese, salvo observação em contrário, ele representa a infra-estrutura computacional (composta por ferramentas, ambientes, instrumentos de apoio etc.) que é necessária para apoiar uma iniciativa de melhoria de

processos de software, isto inclui, por exemplo, necessidades específicas relacionadas a disponibilidade de pessoas capacitadas ou de recursos financeiros.

1.4 Organização do Texto

Este capítulo introdutório apresentou a motivação e o cenário de aplicação das idéias a serem apresentadas nesta tese. Estes tópicos serão refinados ao longo dos próximos capítulos. A organização do texto deste trabalho segue a estrutura abaixo:

- **Capítulo II – Melhoria de Processos de Software:** Descreve os conceitos relacionados a melhoria de processo de software que de alguma forma tem relação com o tema da tese. Esta seção utiliza o resultado de um estudo baseado em revisão sistemática para descrever necessidades relacionadas a processos de software durante a execução de uma iniciativa de melhoria de processos de software em uma organização e/ou corporação.
- **Capítulo III – Infra-estrutura para Iniciativas de Melhoria de Processos de Software:** Descreve exemplos de infra-estrutura necessária para a execução de processos de software no contexto de uma iniciativa de melhoria de processos de software. Esta seção também utiliza o resultado de um estudo baseado em revisão sistemática para descrever necessidades de infra-estrutura para a execução de uma iniciativa de melhoria de processos de software em uma organização e/ou corporação.
- **Capítulo IV - Ambientes de Engenharia de Software Orientados a Corporação:** Descreve as necessidades de uma corporação para a execução de seus processos e os requisitos necessários para que seja provido um apoio ferramental que atenda a estas necessidades.
- **Capítulo V – Definição e Construção de Ambientes de Engenharia de Software Orientados a Corporação na Estação Taba:** Descreve as mudanças realizadas na Estação Taba para tornar possível a implementação da abordagem proposta. Além disso, apresenta a descrição de um cenário de uso dos Ambientes de Engenharia de Software Orientados a Corporação.
- **Capítulo VI – Conclusão e Perspectivas Futuras:** Contém as conclusões e contribuições do trabalho, além de indicar possíveis trabalhos futuros.
- **Anexo I – Linguagem para Modelagem de Processos Organizacionais:** Apresenta a linguagem de modelagem de processos utilizada em alguns dos diagramas presentes no texto desta tese.

- **Anexo II – Perfil UML para Modelagem de Ontologias:** Apresenta a linguagem de modelagem de ontologias utilizada em alguns dos diagramas presentes no texto desta tese.
- **Anexo III – Estudo Baseado em Revisão Sistemática – Infra-estrutura de Apoio a Iniciativas de Melhoria de Processos de Software:** Apresenta detalhes do estudo baseado em revisão sistemática realizada dentro do escopo desta tese.
- **Anexo IV – Ontologia de Corporação:** Apresenta a versão completa da ontologia de corporação definida no contexto desta tese a partir da atualização da ontologia de organização anterior.
- **Anexo V – Especificação de Requisitos:** Apresenta com mais detalhes o levantamento de requisitos inicial para as principais funcionalidades dos AESCorp identificadas nesta tese.

CAPÍTULO 2 - MELHORIA DE PROCESSOS DE SOFTWARE

Neste capítulo são apresentados conceitos relacionados a melhoria de processos de software, incluindo normas e modelos de qualidade de processos de software nacionais e internacionais. São apresentados também exemplos de iniciativas de melhoria de processos de software em organizações e corporações.

2.1 Processos de Software

Desenvolvimento de software não é apenas uma questão de criar ferramentas e linguagens de programação efetivas. O desenvolvimento de software é uma tarefa coletiva, complexa e criativa. Assim sendo, a qualidade do produto de software depende em grande parte das pessoas, da organização e dos procedimentos utilizados para criá-lo e disponibilizá-lo. A noção de um processo de software implica na noção de um ciclo de vida e provê um grande e abrangente conceito para enquadrar e organizar diferentes fatores e questões relacionadas às atividades de desenvolvimento de software (FUGGETTA, 2000).

Um processo pode ser definido como um conjunto de atividades estruturadas e destinadas a resultar em um artefato ou serviço de valor para a organização ou para um determinado cliente ou mercado e implica em uma ordenação específica das atividades com começo, fim, insumos e produtos claramente identificados (VILLELA, 2004). Um processo de software, por sua vez, pode ser definido como um conjunto de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos que são necessários para conceber, desenvolver, entregar e manter um produto de software (FUGGETTA, 2000) e também produtos associados, como planos do projeto, documentos de projeto, código fonte, casos de teste e manuais do usuário (PAULK *et al.*, 1993b). Além disso, um processo de software independe da tecnologia utilizada no desenvolvimento e define o modo pelo qual o desenvolvimento de software é organizado, gerenciado, medido, apoiado e melhorado (MONTANGERO *et al.*, 1999). Em todo caso, o processo deve ajudar a guiar as pessoas em relação ao que fazer sobre a divisão e coordenação dos trabalhos e garantir uma comunicação efetiva (LINDVALL e RUS, 2000).

O trabalho em processo de software apóia a construção da qualidade no software ao manter o foco em como o software é construído e avaliado. A comunidade de processos

de software trata o software como uma informação agregada complexa, intrincada e interconectada que foi construído através da execução de um processo que deveria ser ordenado e sistemático. Um produto de software pode ser considerado de grande qualidade apenas quando se pode demonstrar que os artefatos que o constituem são consistentes entre si de diferentes formas. Infelizmente, no entanto, produtos de software são geralmente desenvolvidos utilizando processos *ad hoc* que geralmente residem na cabeça daqueles que os executam e supervisionam. Como consequência, projetos de software são geralmente mal coordenados, supervisionados inadequadamente e falham em entregar produtos que demonstrem grande qualidade. Tornar o processo de desenvolver software explícito pode ajudar a resolver muitos destes problemas e, de fato, parece que o aumento do rigor e detalhe das definições de processo de software leva um aumento na eficiência no desenvolvimento de produtos de software cuja qualidade é tanto assegurada como demonstrável (OSTERWEIL, 1996).

Segundo BOEHM (2006), ao traçar um panorama da engenharia de software, a partir do trabalho de OSTERWEIL (1996) que dizia que “processos de software também são software”, houve uma reorientação do foco dos ambientes de desenvolvimento de software além da exposição da dualidade entre práticas boas para desenvolver produtos e aquelas boas para o desenvolvimento de processos. Inicialmente, o foco foi em ferramentas e linguagens de programação, mas o conceito foi estendido para abranger *insights* úteis relacionados a requisitos de processos de software, arquitetura de processos, processo de gerência de mudança, família de processos, biblioteca de ativos de processos com componentes de processo reusáveis e passíveis de composição. Isso tudo permitiu a realização de maiores níveis de maturidade de processo de software com uma melhor relação custo-benefício (BOEHM, 2006).

Um dos primeiros modelos de maturidade a surgir, o SW-CMM (PAULK *et al.*, 1993a) foi criado a partir de necessidades do Departamento de Defesa Americano (*DoD*) em identificar empresas capazes de desenvolver software e não aquelas que eram capazes de fazer boas propostas de desenvolvimento (BOEHM, 2006). Ao longo do tempo, muitas empresas têm relatado bom retorno de investimento devido à redução de retrabalho (BOEHM, 2006). Além desse, outros modelos de maturidade e normas internacionais voltados para processos de software surgiram com o passar do tempo. Cada vez mais, organizações são forçadas a aderir a restrições regulatórias que exigem a existência de processos explícitos e a demonstração da aderência a estes processos (MUNCH e VIERIMAA, 2006).

Segundo MUNCH e VIERIMAA (2006), atualmente dois tipos de abordagens para melhoria de processos de software são mais utilizados na prática: (i) abordagens contínuas (também mencionadas como abordagens orientadas a problemas) e (ii) abordagens baseadas em modelos (também mencionadas como abordagens orientadas a soluções). O escopo dos métodos e modelos de melhoria devem ser abrangentes o suficiente para considerar todos os diferentes fatores que afetam as atividades de desenvolvimento de software e, também, reutilizar experiências adquiridas em outros domínios de negócio e em pesquisas sobre o comportamento organizacional (FUGGETTA, 2000).

As abordagens contínuas (por exemplo, o PDCA (DEMING, 1986) e QIP (BASILI *et al.*, 1994)) focam na seleção de problemas de uma organização de desenvolvimento de software e geralmente envolvem ciclos de melhoria com base na comparação com uma situação ou dados específicos. Uma importante vantagem dessa abordagem é o foco na resolução de problemas específicos ao analisar o problema em questão, implementando e observando ações de melhoria focadas em problemas e medindo os efeitos de tais ações. Entretanto, devido à sua natureza, é difícil criar uma percepção geral para as questões de qualidade em organizações muito grandes, com muitos colaboradores (MUNCH e VIERIMAA, 2006).

As abordagens baseadas em modelos (por exemplo, ISO/IEC 15504 (ISO/IEC, 2003) e CMMI (CHRISISSIS *et al.*, 2006)) comparam os processos e práticas correntes de uma organização contra um modelo de referência ou *benchmark*. Uma desvantagem desta abordagem é a de geralmente não avaliar o impacto das características do processo ou do produto e, por isso, não podem ser utilizadas para analiticamente identificar e lidar com os problemas dos processos que causam as deficiências nos produtos. Tipicamente é verificado se a prática ou processo está presente, mas seu impacto nos objetivos de negócio ou seu valor para a organização não é avaliado (MUNCH e VIERIMAA, 2006).

Segundo MUNCH e VIERIMAA (2006), no entanto, as abordagens contínuas e as baseadas em modelo podem ser vistas como complementares: as baseadas em modelos podem ser utilizadas para identificar áreas de problemas e opções potenciais de melhoria e as contínuas podem ser utilizadas para implementar e otimizar as soluções. Na prática, a questão típica não é se a melhoria de processos é necessária, mas como definir e implementar uma estratégia para introduzir melhoria avançada de processo passo a passo e como avaliar seu sucesso.

2.2 Normas e Modelos de Qualidade de Processos de Software

Modelos de Capacidade e Maturidade têm foco na melhoria de processos numa organização. Eles contêm os elementos essenciais de processos para uma ou mais disciplinas e descrevem um caminho evolucionário de melhoria partindo de processos *ad hoc* e imaturos até chegar a processos disciplinados e maduros com a melhora da qualidade e eficiência (CHRISSIS *et al.*, 2006). Apesar de serem importantes mecanismos para fomentar definição, institucionalização e melhoria de processos de software em uma organização, os modelos de maturidade e normas internacionais não descrevem de fato os processos a serem seguidos pelas organizações. Padrões de engenharia de software apenas provêem referências normativas e estas referências não são diretamente aplicáveis em um processo de engenharia de software (EMMERICH *et al.*, 1999). A definição destes processos deve ser feita respeitando as características da organização e depende de muitos fatores, incluindo o domínio da aplicação, a estrutura e o tamanho da organização (CHRISSIS *et al.*, 2006).

Há algumas razões, no entanto, para que padrões de Engenharia de Software e melhoria de processos de software tenham ganhado mais atenção ao longo dos tempos, por exemplo (EMMERICH *et al.*, 1999): (i) padrões são reconhecidos pela indústria de software como um meio de transferência de boas práticas em uso industrial; (ii) padrões garantem que o nível alvo de qualidade (associado com o padrão) foi seguido durante o desenvolvimento de software e é, por isso, refletido na qualidade do próprio software; (iii) padrões são usados como a base pela qual organizações e/ou produtos de software são certificados e/ou avaliados; finalmente, (iv) se duas organizações colaboram no desenvolvimento de um produto de software seguem os mesmos padrões, então a cooperação será consideravelmente simplificada.

Alguns exemplos de normas internacionais e modelos de maturidade utilizados pela indústria atualmente são o CMMI (*Capability Maturity Model Integration*) (CHRISSIS *et al.*, 2006), MR MPS.BR (Melhoria de Processo do Software Brasileiro) (SOFTEX, 2007), a norma ISO/IEC 12207 (ISO/IEC, 2008b) e a norma ISO/IEC 15504 (ISO/IEC, 2003) e serão descritos nas subseções a seguir.

2.2.1 Norma ISO/IEC 12207

A norma internacional ISO/IEC 12207 (ISO/IEC, 2008b) teve sua primeira versão editada em 1995 (ISO/IEC, 1995), posteriormente, duas evoluções foram elaboradas em 2002 (ISO/IEC, 2002) e 2004 (ISO/IEC, 2004) devido à evolução natural da Engenharia

de Software ao longo do tempo, ao retorno fornecido pelos seus usuários e, também, à necessidade de adequação à norma ISO/IEC 15504 (ISO/IEC, 2003) que trata da avaliação de processos de software.

A versão inicial da norma (ISO/IEC, 1995) agrupa as atividades que podem ser executadas durante o ciclo de vida de software em um conjunto de processos fundamentais, de apoio e organizacionais, além de um processo de adaptação para estes processos. Os *processos fundamentais* do ciclo de vida (Aquisição, Fornecimento, Desenvolvimento, Operação e Manutenção) atendem as partes fundamentais (ou seja, aquela que inicia ou executa o desenvolvimento, operação ou manutenção dos produtos de software) durante o ciclo de vida de software. Estas partes fundamentais são o adquirente, o fornecedor, o desenvolvedor, o operador e o mantenedor do software. Os *processos de apoio* de ciclo de vida (Documentação, Gerência de Configuração, Garantia da Qualidade, Verificação, Validação, Revisão Conjunta, Auditoria e Resolução de Problema) auxiliam outro processo como uma parte integrante, com um propósito distinto, e contribui para o sucesso e qualidade do projeto de software. Um processo de apoio é empregado e executado quando necessário por outro processo. Os *processos organizacionais* de ciclo de vida (Gerência, Infra-estrutura, Melhoria e Treinamento) são utilizados por uma organização para estabelecer e implementar uma estrutura subjacente, constituída de processos de ciclo de vida e pessoas associados, e melhorar continuamente a estrutura e os processos. Eles são tipicamente empregados fora do domínio de projetos, entretanto contribuem para a melhoria da organização.

Com as duas revisões (ISO/IEC, 2002; ISO/IEC, 2004), novos processos foram definidos (Gerência de Ativos, Gerência de Pedidos de Mudança, Engenharia de Domínio, Recursos Humanos, Avaliação de Produto, Gerência de Programa de Reuso, Usabilidade), outros foram expandidos e para cada processo foram definidos um propósito e uma lista de resultados esperados.

A versão atual da norma (ISO/IEC, 2008b) integra a versão original com suas revisões (portanto, mantendo o alinhamento com a norma ISO/IEC 15504). Além disso, devido à abrangência do software (que indica que o software e seus processos de construção não devem ser considerados de forma separada daqueles processos relacionados a sistemas, mas, sim, como uma parte deles), também mantém o alinhamento com a norma ISO/IEC 15288 (ISO/IEC, 2008a) que descreve os processos de ciclo de vida de sistema.

A Norma não especifica os detalhes de como implementar ou executar as atividades e tarefas incluídas nos processos, nem prescreve um modelo específico de ciclo de vida ou método de desenvolvimento de software. As empresas podem adaptar os processos definidos na norma ISO/IEC 12207, de acordo com as particularidades dos seus projetos de software, utilizando o processo de adaptação. Por exemplo: DEMIRORS *et al.* (2000) descrevem a adaptação da norma para a definição de um processo de desenvolvimento de software para o governo da Turquia; SANTOS *et al.* (2005) descrevem a utilização da norma durante a definição de processos de software de desenvolvimento e manutenção adequado para pequenas e médias empresas brasileiras; SIQUEIRA e SILVA (2005) utilizam o processo gerencial definido pela norma para fazer um mapeamento com papéis gerenciais baseados no *Rational Unified Process* (RUP); HOYER e CHROUST (2006) apresentam uma análise da norma visando a identificação dos processos que necessitam de alguma adaptação para adequação ao desenvolvimento de uma linha de produtos. Além disso, como a ISO/IEC 12207 é o modelo de referência primário para a aplicação da norma ISO/IEC 15504, muitas das vezes é usada indiretamente em conjunto com essa norma, como, por exemplo, descrito por JABLONSKI e FAERBER (2007).

2.2.2 Norma ISO/IEC 15504

A norma ISO/IEC 15504 (ISO/IEC, 2003) provê um *framework* para a avaliação de processos de software que pode ser usado por organizações envolvidas com o planejamento, monitoração, controle, aquisição, fornecimento, desenvolvimento, operação, evolução e suporte a software. Este *framework* não provê um método único de avaliação de processos, mas todo um conjunto de requisitos que uma avaliação deve seguir para ter conformidade com a norma. O ganho resultante dessa abordagem é flexibilidade. Dessa forma podem existir diferentes modelos/métodos de avaliação, usando diferentes modelos de processo de referência, todos aderentes à norma.

A ISO/IEC 15504 presta-se à realização de avaliações de processos de software com dois objetivos: a melhoria de processos e a determinação da capacidade de processos de uma unidade organizacional. Se o objetivo for a melhoria de processos, a unidade organizacional pode realizar uma avaliação com o objetivo de gerar um perfil dos processos que será usado para a elaboração de um plano de melhorias. A análise dos resultados identifica os pontos fortes, os pontos fracos e os riscos inerentes aos processos. No segundo caso, a organização tem o objetivo de avaliar um fornecedor em potencial, obtendo o seu perfil de capacidade. O perfil de capacidade permite ao contratante estimar o risco associado à contratação daquele fornecedor em potencial para auxiliar na tomada de

decisão de contratá-lo ou não. A norma define seis níveis de capacidade, como pode ser visto na Tabela 2.1.

Tabela 2.1 – Níveis de Capacidade dos Processos da Norma ISO/IEC 15504 (ISO/IEC, 2003)

Nível	Descrição
0 – Incompleto	Processo não existe ou geralmente falha.
1 – Executado	Processo atinge os objetivos, porém sem padrão de qualidade e sem controle de prazos e custos.
2 – Gerenciado	Processo planejado e acompanhado. Satisfaz requisitos definidos de qualidade, prazo e custos. Seus produtos de trabalho são gerenciados.
3 – Estabelecido	Processo executado e gerenciado com uma adaptação de um processo padrão, definido, eficaz e eficiente.
4 – Previsível	Processo executado dentro de limites de controle definidos e com medições detalhadas e analisadas.
5 – Em Otimização	Processo melhorado continuamente de forma disciplinada.

Exemplos de utilização da norma incluem a sua utilização como modelo para avaliação de processos em organizações e também como base para a definição de processos, como descritos, por exemplo, por CASS *et al.* (CASS *et al.*, 2002), GUZMAN *et al.* (2006), CHOI *et al.* (CHOI *et al.*, 2005), ANACLETO e von WANGENHEIM (2005), von WANGENHEIM *et al.* (2006) e COLETTA (2007).

2.2.3 CMMI-DEV

O CMMI (*Capability Maturity Model Integration*) (CHRISISS *et al.*, 2006) consiste de boas práticas que tratam do desenvolvimento e manutenção de produtos e serviços cobrindo o ciclo de vida de um produto desde a concepção até a entrega e a manutenção. O modelo é composto de 22 áreas de processo. Cada área contém práticas relacionadas que, quando implementadas, coletivamente satisfazem um conjunto de objetivos considerados importantes para fazer melhorias significativas nesta área.

Existem dois tipos de representação no CMMI: em estágios e contínua. A representação em estágios define um conjunto de áreas de processo para definir um caminho de melhoria para a unidade organizacional, descrito em termos de níveis de maturidade. Os possíveis níveis de maturidade da organização na representação por estágios são: 1 – Inicial; 2 – Gerenciado; 3 – Definido; 4 - Gerenciado Quantitativamente; 5 – Otimizado. A representação contínua (conforme é utilizada na norma ISO/IEC 15504) permite que uma organização selecione uma área de processo específica e melhore com relação a esta área. A representação contínua usa níveis de capacidade para caracterizar melhoria relacionada a uma área de processo. Os possíveis níveis de capacidade de uma área de processo são: 0 – Incompleto; 1 – Desempenhado; 2 – Gerenciado; 3 – Definido;

4 - Gerenciado Quantitativamente; 5 – Otimizado. A Tabela 2.2 apresenta a estrutura de cada nível do CMMI-DEV.

Tabela 2.2 – Áreas de Processo do CMMI-DEV

Nível de Maturidade	Área de Processo
2	Monitoração e Controle do Projeto (PMC)
	Planejamento do Projeto (PP)
	Gerência de Requisitos (REQM)
	Análise e Medição (MA)
	Garantia da Qualidade do Processo e do Produto (PPQA)
	Gerência de Configuração (CM)
	Gerência de Fornecedor Integrada (SAM)
3	Gerência de Projeto Integrada (IPM)
	Gerência de Riscos (RSKM)
	Definição do Processo Organizacional (OPD)
	Foco no Processo Organizacional (OPF)
	Treinamento Organizacional (OT)
	Desenvolvimento de Requisitos (RD)
	Integração do Produto (PI)
	Solução Técnica (TS)
	Validação (VAL)
	Verificação (VER)
	Análise de Decisão e Resolução (DAR)
4	Gerência Quantitativa do Projeto (QPM)
	Desempenho do Processo Organizacional (OPP)
5	Inovação Organizacional e Posicionamento Estratégico (OID)
	Resolução e Análise Causal (CAR)

O mapeamento entre as práticas do CMMI e os requisitos descritos na norma ISO/IEC 15504 ainda não é total, ajustes ainda precisam ser feitos (ROUT e TUFFLEY, 2007) apesar das alterações realizadas na versão 1.2 e de o SEI (órgão criador e mantenedor do CMMI) estar trabalhando para isso (SEI, 2008).

Os exemplos de utilização do CMMI (e de suas versões anteriores) são variados na literatura, por exemplo, (CHANWOO *et al.*, 2004; FALBO *et al.*, 2004; AGUIAR *et al.*, 2005; DUARTE *et al.*, 2005; MARCZAK *et al.*, 2005; MOREIRA *et al.*, 2005; PRIKLADNICKI *et al.*, 2005; EKDAHL e LARSSON, 2006; GONÇALVES *et al.*, 2006; GUERRA *et al.*, 2006; MACEDO *et al.*, 2006; MARINHO *et al.*, 2006; MONTEIRO *et al.*, 2006; OGASAWARA *et al.*, 2006; WANG e LI, 2006; MARÇAL *et al.*, 2007). Parte de sua influência pode ser creditada à longa existência se contado a partir da primeira versão do CMM, de que é derivado, e também à sua influência econômica. Essa influência pode ser ilustrada, por exemplo, pelo fato de ter sido criado a partir de uma cooperação entre o Departamento de Defesa dos EUA e a Universidade Carnegie Mellon para facilitar a identificação de bons fornecedores de software e, para citar um exemplo brasileiro, pela bonificação oferecida em licitações a empresas avaliadas em algum nível do modelo.

2.2.4 MPS.BR

O MPS.BR (SOFTEX, 2007) é um programa para Melhoria de Processo do Software Brasileiro coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) e tem como objetivo definir um modelo de melhoria e avaliação de processo de software, preferencialmente para as micro, pequenas e médias empresas, de forma a atender as suas necessidades de negócio e a ser reconhecido nacional e internacionalmente como um modelo aplicável à indústria de software. Este é o motivo pelo qual ele está aderente a modelos e normas internacionais. O MPS.BR também define regras para sua implementação e avaliação, dando sustentação e garantia de que o MPS.BR está sendo empregado de forma coerente com as suas definições.

O Modelo de Referência para Melhoria de Processo do Software Brasileiro (MR-MPS) contém os requisitos que as organizações deverão atender para estar em conformidade com o MR-MPS. Ele contém as definições dos níveis de maturidade, da capacidade de processos e dos processos em si, tendo sido baseado nas normas ISO/IEC 12207 e suas emendas 1 e 2, na ISO/IEC 15504. Além disso, existe uma equivalência entre o MPS.BR e o CMMI (SOFTEX, 2008b). Esta equivalência é total do ponto de vista do MPS.BR para o CMMI, isto é, todos os requisitos das áreas de processo do CMMI estão presentes no MPS.BR. Entretanto não existe equivalência total do ponto de vista do CMMI para o MPS.BR, pois, visando à conformidade com as normas internacionais ISO/IEC 12207 e 15504-2, as exigências do MR-MPS são maiores do que as do CMMI nos seguintes processos: Gerência de Recursos Humanos que no MR-MPS (conforme disposto nas normas ISO/IEC 12207 e 15504-2) abrange aquisição de pessoal, treinamento organizacional e gerência do conhecimento e no CMMI trata apenas de treinamento organizacional; Gerência de Reutilização e Desenvolvimento para Reutilização estão incluídos no MR-MPS (atendendo requisitos das normas ISO/IEC 12207 e 15504-2) e não existem no CMMI.

O MR-MPS define sete níveis de maturidade: G (Parcialmente Gerenciado), F (Gerenciado), E (Parcialmente Definido), D (Largamente Definido), C (Definido), B (Gerenciado Quantitativamente) e A (Em Otimização). A escala de maturidade se inicia no nível G e progride até o nível A. Para cada um destes sete níveis de maturidade foi atribuído um perfil de processos e de capacidade de processos (baseados naqueles definidos pela ISO/IEC 15504) que indicam onde a organização tem que colocar esforço para melhoria. A divisão em estágios, embora baseada nos níveis de maturidade do CMMI-DEV, tem uma graduação diferente, com o objetivo de possibilitar uma implementação e

avaliação mais gradual e adequada às pequenas e médias empresas. A possibilidade de se realizar avaliações considerando mais níveis permite uma visibilidade dos resultados de melhoria de processos com prazos mais curtos.

Para que as organizações alcancem um determinado nível de maturidade do MPS.BR, todos os propósitos e resultados esperados dos respectivos processos e dos atributos de processo estabelecidos para aquele nível e para os níveis anteriores devem ser atendidos, conforme pode ser visto na Tabela 2.3 (onde a sigla AP significa Atributo de Processo).

Tabela 2.3 – Processos e Atributos de Processo do MPS.BR

Nível	Processo	Capacidade
A	Análise de Causas de Problemas e Resolução (ACP)	AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1, AP 4.2, AP 5.1 e AP 5.2
B	Gerência de Projetos (GPR, evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1, AP 3.2, AP 4.1 e AP 4.2
C	Gerência de Riscos (GRI)	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
	Desenvolvimento para Reutilização (DRU)	
	Análise de Decisão e Resolução (ADR)	
	Gerência de Reutilização (GRU, evolução)	
D	Verificação (VER)	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
	Validação (VAL)	
	Projeto e Construção do Produto (PCP)	
	Integração do Produto (ITP)	
	Desenvolvimento de Requisitos (DRE)	
E	Gerência de Projetos (GPR, evolução)	AP 1.1, AP 2.1, AP 2.2, AP 3.1 e AP 3.2
	Gerência de Reutilização (GRU)	
	Gerência de Recursos Humanos (GRH)	
	Definição do Processo Organizacional (DFP)	
	Avaliação e Melhoria do Processo Organizacional (AMP)	
F	Medição (MED)	AP 1.1, AP 2.1 e AP 2.2
	Garantia da Qualidade (GQA)	
	Gerência de Configuração (GCO)	
	Aquisição (AQU)	
G	Gerência de Requisitos (GRE)	AP 1.1 e AP 2.1
	Gerência de Projeto (GPR)	

Exemplos de utilização do MPS.BR por empresas brasileiras na literatura especializada, apesar de ainda pequeno em relação a outros modelos e normas de maturidade, é significativo levando em consideração ao pequeno tempo de vida do modelo, como pode ser visto, por exemplo, em (ROCHA *et al.*, 2005b; FERREIRA *et al.*, 2006a; NUNES *et al.*, 2006; BORSSATTO e MORO, 2007; BRIETZKE *et al.*, 2007; FERREIRA *et al.*, 2007a; FERREIRA *et al.*, 2007b; FERREIRA *et al.*, 2007c; MONTONI *et al.*, 2007b; SANTOS *et al.*, 2007b; SANTOS *et al.*, 2008b). Além disso, mecanismo instituído pela entidade mantenedora do modelo garante que lições aprendidas e experiências relacionadas a adoção e utilização do modelo por Instituições Implementadoras (II), Avaliadoras (IA) e

Organizadoras de Grupos de Empresas (IOGE) além das próprias empresas sejam discutidas e compartilhadas através de workshops periódicos (SOFTEX, 2008a).

2.3 Melhoria de Processos de Software em Organizações

Os objetivos para adoção de iniciativas de melhoria de processos de software em organizações encontrados na literatura são os mais variados. Podemos destacar, dentre outros:

- Definir de uma base de conhecimento para apoiar a execução de um programa de melhoria de processos de software (KESS e HAAPASALO, 2002; FALBO *et al.*, 2004; GARCIA *et al.*, 2005; AMESCUA *et al.*, 2006);
- Melhorar a produtividade (SAKAMOTO *et al.*, 1998) e a duração (PRACCHIA, 2004) dos projetos;
- Aumentar a capacidade das organizações através do uso adequado de técnicas de engenharia de software para aumentar a qualidade dos produtos de software e, assim, a competitividade organizacional (FERREIRA *et al.*, 2006b);
- Implantar processos de software, seja com base em modelos de maturidade ou referência como o MPS.BR e o CMMI (ROCHA *et al.*, 2005a; MONTONI *et al.*, 2006c; MONTONI *et al.*, 2007b) em conjunto, o CMMI (WANG e LI, 2006; JUN *et al.*, 2007), SW-CMM em conjunto com o IDEAL (GONG *et al.*, 2006a) ou com base em nenhum modelo específico (MONTONI *et al.*, 2005);
- Avaliar a aderência a um modelo de referência (TAVARES *et al.*, 2002a; VARKOI, 2002; GUZMAN *et al.*, 2006);
- Atingir objetivos específicos de qualidade, custo e entregas no desenvolvimento de software e estabelecer uma cultura organizacional para tornar possível a melhoria contínua (OGASAWARA *et al.*, 2006);
- Institucionalizar um guia eletrônico de processos (SCOTT *et al.*, 2002; KURNIAWATI e JEFFERY, 2004; MOE *et al.*, 2005; KURNIAWATI e JEFFERY, 2006);
- Institucionalizar um programa de medição de processos (JARVINEN, 2000; RUIZ *et al.*, 2002; WANG e LI, 2005a; CANFORA *et al.*, 2006);
- Aplicar o QIP e PDCA para estruturar iniciativas de melhoria de processos (SCOTT *et al.*, 2001).

2.4 Melhoria de Processos de Software em Corporações

A adoção de iniciativas de melhoria de processos de software em corporações muitas vezes possui características e objetivos comuns à adoção em organizações, como podem ser vistos, por exemplo, nos trabalhos apresentados por SAKAMOTO *et al.* (1998), LIMA FILHO *et al.* (2002), TAVARES *et al.* (2002b), MALHEIROS *et al.* (2006) e GUZMAN *et al.* (2006).

GUZMAN *et al.* (2006) descrevem recomendações para realizar uma avaliação de processos de software numa grande organização espanhola que possui características de corporação, apesar de não ser tratada como tal, com seis unidades organizacionais desenvolvedoras de software. Os autores recomendam uma estratégia de melhoria de processos de software baseada no desenvolvimento evolucionário dos processos e bibliotecas de ativos de processos. Dessa forma, a estratégia implica no desenvolvimento de versões iniciais de um produto operacional e, posteriormente, de sucessivas versões, onde cada núcleo destas versões responde por requisitos mínimos e essenciais e é construído de forma flexível de forma a antecipar futuras inclusões de mais capacidades de forma simples. Duas abordagens para a implementação dessa estratégia incremental são a implantação de (i) pequenas mudanças em várias unidades organizacionais ou (ii) a implantação de grandes mudanças em poucas unidades organizacionais. As principais vantagens da primeira abordagem são o alto compromisso obtido e facilidades para atingir níveis de maturidade e projeto integral das soluções; a principal da segunda é a obtenção de resultados parciais em pouco tempo. As principais desvantagens da primeira são o maior impacto organizacional e a implantação complexa; as da segunda são o maior risco associado com a implantação, o menor comprometimento da organização e maior dificuldade de atingir níveis de maturidade.

SAKAMOTO *et al.* (1998) descrevem uma iniciativa de melhoria de processos de software da OMRON Corporation que durou 17 meses cujo objetivo foi melhorar o esforço dos projetos (medido em homem/mês). Ao final da iniciativa pode-se perceber uma redução de 30% do esforço em homem-mês no projeto alvo da iniciativa de melhoria. Além disso, foi percebida a falta de uma abordagem para gerência do conhecimento pelos membros do SEPG e não foi percebida pela alta gerência a relação entre os esforços de SPI e os benefícios obtidos. Entretanto, a análise realizada pelos envolvidos na iniciativa de SPI gerou a criação de um protótipo de ambiente para apoiar as atividades de melhoria de processos na corporação denominado PIASS (*Process Improvement Activities Support Systems*) com ferramentas para apoiar a iniciativa de melhoria de processos na avaliação de

processos, gerência de conhecimento e gerência das atividades de melhoria de processos de software. Este ambiente possui funções próprias para membros do SEPG, desenvolvedores e gerentes da iniciativa de melhoria. O protótipo apresentado não foi utilizado, entretanto, sua definição foi fortemente baseada na observação dos problemas encontrados durante a iniciativa de melhoria de processos e procurou suprir várias necessidades identificadas ao longo da iniciativa de melhoria.

Um dos poucos trabalhos que descrevem uma abordagem para iniciativa de melhoria de processos de software em corporações e citam como as organizações subordinadas foram envolvidas na iniciativa e citam a estrutura de comando entre os diferentes níveis envolvidos é apresentado por OGASAWARA *et al.* (2002; 2005; 2006).

OGASAWARA *et al.* (2002; 2005; 2006) descrevem os problemas encontrados para a promoção de atividades de melhoria de processos de software na Toshiba e as soluções que foram dadas. Os autores alertam que para uma promoção efetiva de tais atividades em uma grande organização é necessário o estabelecimento de uma estrutura organizacional e um método de implantação para a promoção e desenvolvimento de treinamentos, ferramentas de apoio e outros materiais. Além disso, alertam, é necessário o estabelecimento de uma comunidade de melhoria de processos de software envolvendo os diversos departamentos.

O Grupo de Processos de Engenharia de Software (ou SEPG, da sigla em inglês para *Software Engineering Process Group*) Cooperativo da Toshiba foi criado em 2000 sendo composto, então, por 6 pessoas e, em 2005, já contava com 20 pessoas e utiliza-se o CMM como um modelo para as atividades de melhoria de processos de software. Os principais objetivos das atividades desenvolvidas foram atingir os objetivos de qualidade, custo e entregas no desenvolvimento de software e estabelecer uma cultura organizacional para tornar possível a melhoria contínua. Para isso, foi definido um plano de longo prazo:

(i) Entre 2000 e 2003 – a meta foi o estabelecimento de um *framework* para melhoria de processos de software com: a promoção das atividades relacionadas à melhoria de processos, ao uso do CMM e ao próprio SEPG; a organização de SEPGs em cada departamento onde houvesse desenvolvimento; treinamento em assuntos relacionados a melhoria de processos de software; estabelecimento de uma hierarquia, em três níveis, de SEPGs.

(ii) Entre 2004 e 2007 – a meta era a implantação das atividades de melhoria de processos de software com: apoio a maior maturidade e qualidade de processos e produtos;

estabelecimento do relacionamento entre o CMM e gerência de projetos; estabelecimento de uma política de garantia de qualidade de software nas organizações.

(iii) Entre 2008 e 2010 – a meta consiste no estabelecimento de uma cultura de melhoria de processos de software com: introdução de novas tecnologias para processos e produtos; implantação de compartilhamento de informações; estabelecimento de padrões de competências para melhoria de processos de software nas organizações e de nomes para os papéis associados a estas habilidades (por exemplo, líder de SEPG, membro do SEPG etc.).

Os SEPGs são divididos em três diferentes classes: SEPG Corporativo, SEPG Organizacional e SEPG da Unidade de Negócio. A estrutura do SEPG é hierárquica, como pode ser visto na Figura 2.1, existindo o SEPG Corporativo e pelo menos um SEPG por organização subordinada. Os SEPGs relacionados às unidades de negócio podem estar subordinados aos SEPGs Organizacionais ou ao Corporativo. O SEPG Corporativo também tem como responsabilidades: garantia de qualidade (via testes) através de ferramentas de análise estática de programas; promoção de ferramentas automáticas de qualidade de software (por exemplo, análise estática, gerência de configuração, gerência de defeitos, estimativas de confiabilidades, geração de casos de testes etc.); estabelecimento e execução de cursos para treinamento em engenharia de software. Além das ferramentas citadas anteriormente, são utilizadas para promoção das atividades *sites*, *newsletters*, listas de discussão e relatórios de atividades de SPI.

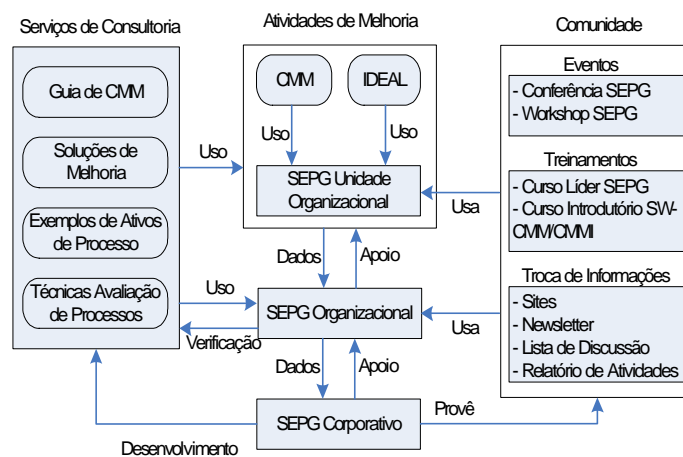


Figura 2.1 – Framework para promoção das Atividades de Melhoria de Processos na Toshiba (OGASAWARA *et al.*, 2006)

Em 2006 havia cerca de 60 SEPGs em unidades de negócio ou organizações contra 17 em 2001. Além disso, a taxa de SEPGs em unidades de negócio que eram apoiados pelo SEPGs corporativo caiu de 94% para 66%, sendo que o papel dessa liderança passou a ser

realizado pelos SEPGs organizacionais. Segundo os autores, isso é um indicativo de que a infra-estrutura organizacional necessária para um processo sistemático de melhoria foi construída e que as atividades de melhoria de processos de software foram promovidas. Foi notada, também, uma melhora significativa das unidades de negócio em relação a conhecimento e domínio de processos de software e várias unidades de negócio foram avaliadas no SW-CMM níveis 2 e 3.

A unidade da Motorola no Brasil descreve a implantação de processos visando a obtenção dos Níveis 2 e 3 do SW-CMM (LIMA FILHO *et al.*, 2002). A implantação do CMM na empresa foi feita com a ajuda de um grupo específico para a gerência das atividades envolvidas, o SEPG Local. As principais atividades do SEPG foram o gerenciamento de mudanças de processo e o acompanhamento da implantação do processo. Havia três níveis de processos de software: Processo Global (ditado pela Matriz americana), Versão Regional (para a filial brasileira) e as versões utilizadas nos projetos de fato. Adaptações nos processos podem ser realizadas na passagem de um nível para o outro. O SEPG Local é subordinado ao SEPG Global. Se uma melhoria identificada pelo SEPG Local afeta o Processo Global, essa melhoria é submetida ao SEPG Global para aprovação e institucionalização.

O SERPRO (Serviço Federal de Processamento de Dados) é uma empresa vinculada ao Ministério da Fazenda brasileiro, responsável pela execução de serviços de tratamento de informações e processamento de dados para o governo federal. A empresa está estruturada em unidades de gestão responsáveis por um segmento da administração pública. Cada segmento atende a pelo menos um órgão federal (com características e necessidades próprias), por intermédio de projeções da unidade de gestão em cada uma das unidades organizacionais. Há unidades espalhadas por dez capitais brasileiras (Rio de Janeiro, Belém, Belo Horizonte, Curitiba, Fortaleza, Porto Alegre, Recife, Salvador, São Paulo e Brasília). Desde 2000 está envolvida com a adoção dos níveis 2 e 3 SW-CMM/CMMI em suas unidades (TAVARES *et al.*, 2002b; MALHEIROS *et al.*, 2006).

A estratégia de implantação dos processos visando o nível 2 do SW-CMM utilizou uma estrutura de grupos de trabalho responsáveis pela definição e gerência dos processos composta por (TAVARES *et al.*, 2002b): GEP (Grupo Executor do Projeto) formado por pessoas de cada unidade e responsável pelas diretrizes e aprovação das decisões e definições relacionadas ao Programa de Modernização do Desenvolvimento (PMoD); GTS (Grupo de Trabalho Serpro) que são grupos corporativos com função de definir a política, processos e procedimentos de cada uma das áreas de processo do nível 2 do CMM e para o

processo de Engenharia; GTI (Grupo de Trabalho Interno) que são projeções dos GTS nas unidades de gestão formados por profissionais responsáveis pela elaboração e revisão do conteúdo do processo de desenvolvimento. Ferramentas foram utilizadas como forma de apoio à execução dos processos de gerência de projetos (apoio tarefas como cadastro de solicitações de usuários, medição de esforço e tamanho em pontos por função e controle e acompanhamento dos custos), gerência de requisitos, gerência de configuração e divulgação do processo padrão da organização. Não fica claro a partir do trabalho se alguma das unidades já tinha sido avaliada no nível 2 na época, mas lições aprendidas são descritas que comprovam a institucionalização dos processos de software em, pelo menos, parte das unidades.

Num relato posterior, MALHEIROS *et al.* (2006) descrevem uma abordagem e ferramenta apoiar a melhoria de processos de software na empresa. A abordagem apóia ao trabalhado do SEPG durante a implantação de CMMI níveis 2 e 3 na empresa e tem como base o programa de melhoria de processos que visa garantir treinamento, recursos e fundos para incorporação do PSDS (o processo de desenvolvimento da empresa) pelas unidades de desenvolvimento, e mantê-lo alinhado com as boas práticas de mercado. O programa trata o processo de desenvolvimento de forma corporativa, auferindo-lhe a independência necessária à formulação das políticas e procedimentos e à padronização das práticas executadas pelas unidades do SERPRO que desenvolvem software. A ferramenta GM-PSDS é utilizada pelas diversas unidades de gestão da organização e pelos envolvidos com os processos para a solicitação e acompanhamento de melhorias no processo padrão e tem por objetivo viabilizar a análise, controle e monitoração de oportunidades de melhoria nos processos de software, vinculando-as aos ganhos obtidos para o processo, numa abordagem de trabalho cooperativo. Segundo os autores, a ferramenta tem se mostrado efetiva para a execução e acompanhamento da melhoria nos processos e, através dela, é possível obter dados sobre as melhorias como, por exemplo, quantas melhorias foram abertas, aprovadas, implantação por regional ou unidade de gestão.

Em se tratando de relatos sobre implementação de melhoria de processos de software no contexto de corporações, há, também, relatos de ações pontuais dentro do contexto e necessidades específicas de determinadas unidades de negócio de grandes empresas que se comportam como corporações. A seguir, alguns exemplos, relacionados às empresas Alcatel, Nokia, IBM, Motorola e Ericsson:

- JOKIKYYNY (1999) discute a utilização de uma ferramenta para melhorar o processo de inspeção de software em unidades da Ericsson na Finlândia,

incluindo atividades de coleta e análise dos dados.

- KETTUNEN e LAANTI (2005) descrevem um quadro comparativo para seleção de modelos de processos de software a serem utilizados em projetos em uso na Nokia.
- KILPI (2001) descreve a implantação de um programa de medição na Nokia com base numa adaptação do GQM.
- STARK e CROCKER (2003) descrevem a adoção de processos de software ágeis em uma das unidades da Motorola com bons resultados; dentre eles o tempo e o custo necessários para o projeto piloto foram menores que o previsto em relação à estratégia padrão da empresa, além disso, a qualidade do produto também foi melhor que a esperada.
- EBERT (EBERT, 1999a; EBERT, 1999b) descreve uma abordagem para definição de um programa de medição corporativo na Alcatel, uma corporação de origem francesa.
- EBERT e DE MAN (2002) descrevem a experiência da Alcatel na gerência da diversidade de seus processos. Ao utilizarem uma ferramenta de workflow capaz de gerenciar a diversidade de processos dentro da empresa, conseguiram como benefícios: melhora da qualidade, redução do tempo de desenvolvimento, aumento da flexibilidade de engenharia, redução de esforço para tarefas, melhora na comunicação, melhora no alinhamento de processos e ferramentas e maior facilidade de geração de planos de treinamento.

2.5 Diversidade de Processos de Software

Uma corporação, por ser intrinsecamente grande e formada por várias organizações, tem tendência a possuir uma grande diversidade de processos dentro dos seus domínios e tais processos também precisam ser gerenciados e evoluídos de uma forma estruturada.

O conceito de processo de engenharia de software engloba um conjunto de práticas seqüenciais que são funcionalmente coerentes e reusáveis para organização, implementação e gerência de engenharia de software. A orientação do processo de software para um sistema de processo de engenharia de software reflete uma tendência na procura por um meio ideal para organizar e descrever o conjunto completo de framework de processo e infra-estrutura de engenharia de software (WANG e BRYANT, 2002).

Diversidade de processo pode ser definida como o uso de diferentes variantes de processos ao longo do tempo (DECK, 2001). Segundo BALDASSARRE *et al.* (2002), a diversidade de processos deriva da necessidade de os modelos de processo serem especializados em relação ao contexto em que são executados. Além disso, diferentes processos precisam ser integrados para evitar inconsistências e ineficiência causadas pela replicação do trabalho (EBERT e DE MAN, 2002). Dessa forma, a diversidade de processos é um fato, o que precisa ser feito, portanto, é saber como gerenciá-la; a única forma de evitá-la é adotar o processo perfeito e, então, nunca alterá-lo (DECK, 2001). A gestão da diversidade do processo é atingida em princípio pela instanciação automática ou manual de tais processos para um ambiente concreto e suas condições específicas (EBERT e DE MAN, 2002).

Uma possível recomendação para gerenciar a diversidade de processos é usar a análise de riscos para identificar as oportunidades de melhoria de processo e, após, identificar os riscos mais sérios, as melhorias podem ter maior foco e serem relacionadas aos problemas mais significativos em vez daquelas que requerem as ferramentas mais interessantes (DECK, 2001).

O ponto mais importante a ser considerado em relação à diversidade de processos no desenvolvimento de software é conhecer suas necessidades, os processos existentes e adotar e adaptar o que for melhor para a ocasião (LINDVALL e RUS, 2000). Além disso, os processos de engenharia de software podem ser adotados como uma infra-estrutura para a engenharia de software (WANG e BRYANT, 2002).

2.6 Considerações Finais

Este capítulo apresentou uma breve revisão da literatura relacionada a processos de software, incluindo a apresentação de alguns modelos de referência para o desenvolvimento de software. Discutiram-se também aspectos relacionados à diversidade de processos em organizações, muitas vezes influenciada pela adoção de modelos de maturidade, e exemplos de necessidades relacionadas a processos de software e iniciativas de melhoria de processos em algumas corporações (que também são afetadas, devido a suas características, pela diversidade de processos).

O aumento na padronização internacional relativa a engenharia de sistemas e de software é uma consequência do aumento contínuo da importância das tecnologias de informação e comunicação e dos sistemas, produtos e serviços da economia global baseados nestas tecnologias, assim como o aumento da maturidade das disciplinas relativas

à engenharia de software e de sistemas (COALLIER, 2003). Para facilitar aos engenheiros de software a seguirem os padrões desejados estas tecnologias de informação e comunicação podem ser automatizadas. Esta automação pode ajudar os engenheiros de software a aderirem aos padrões, verificar se os documentos são estruturados da forma prescrita pelos padrões e permitir a cooperação entre diferentes engenheiros como prescritos pelo padrão (EMMERICH *et al.*, 1999). Processos sem ferramentas de apoio adequadas tendem a permanecer teóricos (EBERT e DE MAN, 2002). A automação tem o potencial de reduzir o erro humano e tem o potencial de gerar produtos similares ou melhores que os produzidos manualmente mesmo por pessoas com poucas habilidades potencialmente reduzindo, assim, o custo (BARRY *et al.*, 2007). Da mesma forma, o apoio ferramental adequado pode influenciar no sucesso de uma iniciativa de melhoria de processos e também ajudar a reduzir custos e esforços associados a esta iniciativa. Este tópico será abordado com mais detalhes no capítulo a seguir.

CAPÍTULO 3 - INFRA-ESTRUTURA PARA INICIATIVAS DE MELHORIA DE PROCESSOS DE SOFTWARE

O uso de apoio computacional adequado em iniciativas de melhoria de processos de software pode auxiliar uma organização a ter resultados melhores e mais rápidos. Este capítulo não tem por objetivo fazer uma revisão completa da literatura sobre ferramentas CASE ou ambientes em geral, mas, sim, apresentar abordagens e relatos encontrados na literatura sobre infra-estruturas de apoio utilizadas para auxiliar tais iniciativas.

3.1 Introdução

Vários fatores influenciam a definição, gerência, implantação, execução e melhoria de seus processos de software e no controle dos diferentes estágios de maturidade e características possuídos por cada organização e por corporações. Como parte desta tese, foi realizada uma revisão disciplinada da literatura através da execução de um estudo baseado em revisão sistemática visando a identificar os elementos que compõem as diferentes abordagens e instrumentos de apoio utilizados em iniciativas de melhoria de processos de software em organizações e corporações e identificar as abordagens existentes de apoio à definição, execução e melhoria de processos de software no escopo de projetos, organizacionais ou corporativos, através de ferramentas CASE isoladas e/ou ambientes específicos. Com este estudo procurou-se reduzir o viés de uma revisão informal e, também, permitir que tal pesquisa bibliográfica possa ser atualizada com novas publicações disponibilizadas ao longo do tempo.

O objetivo do estudo foi definido, então, como sendo *analisar* relatos de experiência e publicações científicas através de um estudo baseado em revisão sistemática *com o propósito de* identificar os elementos que compõem as abordagens de apoio empregadas em iniciativas de melhoria de processos de software em organizações e corporações. A análise foi realizada *com relação aos* instrumentos de apoio empregados para a definição, implantação, execução e melhoria dos processos de software em organizações e corporações, a partir *do ponto de vista* dos pesquisadores e *no contexto* acadêmico e industrial com foco em programas de melhoria de processos de software em organizações e corporações.

A principal questão de pesquisa foi identificar quais as principais características e apoio computacional das abordagens para execução de iniciativas de melhoria de processos de software em organizações e corporações. Para capturar tais informações, buscou-se (i) descrever as características das organizações onde a iniciativa de melhoria de processos foi definida ou executada e/ou as ferramentas foram utilizadas; (ii) descrever as características da iniciativa de melhoria de processos apresentadas e (iii) descrever as características do apoio ferramental apresentado.

A partir da identificação dos elementos que compõem as diferentes abordagens e instrumentos de apoio empregados em iniciativas de melhoria de processos de software procurou-se, então, realizar uma análise mais detalhada das abordagens investigadas com o objetivo de caracterizá-las e, se possível, influenciar a definição dos requisitos e infraestrutura dos Ambientes de Engenharia de Software Orientados a Corporação (AESCorp).

Dessa forma, parte dos resultados apresentados neste capítulo é oriunda deste estudo (do qual podem ser vistos mais detalhes no Anexo III). Além disso, outras fontes bibliográficas foram consultadas para a elaboração deste capítulo de forma a complementar a descrição de apoios ferramentais encontrados na literatura.

3.2 Apoio Ferramental a Iniciativas de Melhoria de Processos

Devido à importância que as iniciativas de melhoria de processos têm para as organizações serem competitivas e, também, às dificuldades que as organizações têm em implementar tais iniciativas por razões de custo e tempo necessários, é preciso realizar programas de melhoria que sejam mais acessíveis para as organizações independentemente de suas características específicas (AMESCUA *et al.*, 2006). Uma das formas de se reduzir os custos associados com uma iniciativa de melhoria de processos de software é adotar ferramentas adequadas para automatizar certas tarefas e, assim, reduzir o esforço necessário para realizá-las.

Segundo AAEN (2003), quando a melhoria de processos de software tem muito foco em documentação e pouco em proficiência, as organizações começam a se importar muito em descrições e prescrições e pouco em competência e comportamento. Se o processo de software é visto como algo que é feito com o fruto do que está na cabeça das pessoas, há um novo sentido para a coleta e publicação de informações sobre os processos de software: a facilitação do conhecimento sobre o trabalho a ser realizado. Esta facilitação envolve o acesso às informações dos processos, ferramentas e conhecimento

organizacional. O acesso a ferramentas pode aumentar consideravelmente a capacidade das equipes de desenvolvimento de software ao apoiar e, talvez, guiar a execução de tarefas.

Apesar do grande número de publicações relatando ferramentas CASE e ambientes de desenvolvimento de software, não são muitos os relatos sobre a utilização de tal apoio ferramental em contextos de iniciativas de melhoria de processos de software ou que situam tal apoio ferramental como adequado a tais contextos¹. Entretanto, é comum em relatos sobre fatores de sucesso e dificuldades relacionadas à execução de iniciativas de melhoria de processos (por exemplo, (BADDOO e HALL, 2002; BADDOO e HALL, 2003; NIAZI *et al.*, 2005; ROCHA *et al.*, 2006; SANTOS *et al.*, 2007a)) a identificação da existência de uma infra-estrutura adequada como um importante fator de sucesso. A maior parte das organizações com baixos níveis de maturidade no desenvolvimento de software não possuem uma infra-estrutura adequada para iniciar a iniciativa de melhoria de processos de software (SANTOS *et al.*, 2007a). Segundo ZAHARAN (1998) existem dois tipos de infra-estrutura para apoiar atividades relacionadas a processos e para sustentar as atividades de melhoria de processos de software: (i) infra-estrutura para organização e gerência (relacionada aos papéis e responsabilidades) e (ii) infra-estrutura técnica (relacionada às ferramentas técnicas e outras facilidades).

LEUNG *et al.* (2007) realizaram uma pesquisa sobre soluções comerciais para o apoio a atividades de melhoria da qualidade de processos e as avaliaram segundo oito funções relacionadas a melhoria de processos de software (coleta de dados, avaliação de processos, identificação de problemas, análise de problemas, definição de processos, identificação de soluções, medição de resultados e gerência de documentos). Segundo os autores: (i) nenhuma ferramenta provia apoio para a função de identificação de problemas; (ii) a maior parte das ferramentas podem prover diversas funções relacionadas à melhoria de processos de software e acabam por apoiar outras funções de forma integrada; (iii) nenhum conjunto de ferramentas analisado apóia todas as funções identificadas, o que sugere que os responsáveis pela melhoria de processos de software devem fazer adaptações e customizações para possuir um conjunto completo de ferramentas que possa trabalhar de forma integrada e transparente.

O escopo do apoio ferramental para auxiliar na execução de iniciativas de melhoria de processos em organizações encontrados na literatura é variado. Os relatos mencionam desde a utilização de ferramentas no contexto de um processo específico (por exemplo, o

¹ O Anexo III apresenta um resumo do estudo baseado em revisão sistemática elaborado sobre infra-estruturas de apoio a iniciativas de melhoria de processos.

relacionado a desenvolvimento de requisitos (NIKULA e SAJANIEMI, 2005) ou gerência de configuração (PARK *et al.*, 2007)), divulgação dos processos (por exemplo, (HIKICHI *et al.*, 2006)), até a adoção de modelos de maturidade (por exemplo, (GONG *et al.*, 2006b)), o que envolve a execução de vários processos por pessoas de diferentes papéis (por exemplo, (KOENIG, 2003)).

A melhoria de processos de software pressupõe a existência de um processo de software sendo seguido na organização, estando ele formalmente definido ou não. Assim, a definição de processos de software é uma etapa importante em grande parte de iniciativas de melhoria. Associado a essa necessidade é comum que organizações procurem utilizar um sistema para divulgação de tais processos geralmente em conjunto com uma base de conhecimento.

Segundo MARÍNEZ *et al.* (2005), um ambiente para gerência de conhecimento relacionado a gerência de processo de software necessita utilizar resultados anteriores de pesquisa oriundos de quatro áreas: melhoria de processos de software, ferramentas de gerência de processo de software, linguagens de definição de processos de software e meta-modelos. Neste sentido, RAMALA (GARCIA *et al.*, 2005) representa uma base de conhecimento e uma ferramenta que captura o conhecimento de engenharia de software necessário para implantar um programa de melhoria de software em uma organização. A base contém um *framework* de processo de software baseado no PMBOK (PMI, 2004) e com boas práticas vindas do CMMI (CHRISISS *et al.*, 2006) e da ISO/IEC 15504 (ISO/IEC, 2003) e a ferramenta apóia a definição de processos (com apoio de uma biblioteca de ativos) e também permite a avaliação (*assessment*) do processo. O uso da base de conhecimento, segundo *survey* realizado em oito empresas, foi apontado positivamente por 85% dos respondentes.

Um trabalho extenso sobre a definição e utilização de um guia eletrônico de processos foi desenvolvido no contexto de uma pequena empresa australiana, com a criação de um guia eletrônico de processos e repositório de experiências e seu uso durante a execução de projetos de software. Os principais objetivos eram divulgar os processos da organização, aumentando o seu uso e melhorando a institucionalização, criar um repositório de experiências e permitir o aumento da memória organizacional (KURNIAWATI e JEFFERY, 2004). Dentre os benefícios identificados estão a melhora no planejamento de projetos e na documentação do projeto e o aumento da satisfação dos usuários (KURNIAWATI e JEFFERY, 2006). O maior uso do guia de processos foi em prover informação para a criação de *templates* para o desenvolvimento de documentos chave

e definir a lista de tarefas dos projetos. O guia também foi utilizado como um fórum de discussão sobre os *templates* de documentos (SCOTT *et al.*, 2001; SCOTT *et al.*, 2002). O estudo realizado serve para apoiar a evidência de que orientação das ferramentas às necessidades do usuário é um importante atributo para a eficiência de repositórios de experiências e que os processos provêm uma estrutura natural e lógica para guiar a execução do trabalho (KURNIAWATI e JEFFERY, 2006).

Uma ferramenta de gerência de conhecimento também pode ser utilizada como elemento guia da iniciativa de melhoria de processos e não meramente como elemento de apoio. Por exemplo, a ferramenta de gerência de conhecimento ProKnowHow (FALBO *et al.*, 2004) é descrita no contexto de uma organização avaliada no Nível 3 do CMMI e é baseada no uso de ontologias, age como memória organizacional, define políticas para filtragem de conhecimento, auxilia a definição de processos para projetos a partir da adaptação de processos organizacionais e também apóia a definição e coleta de métricas relacionadas aos processos de software.

Não só a definição do processo ou a gerência de conhecimento é necessária ou suficiente para executar uma iniciativa de melhoria de processo. A execução dos processos de software é, também, um fator importante em iniciativas de melhoria. É comum que haja soluções específicas para processos individuais. Geralmente esses processos demandam grandes necessidades computacionais ou que envolvam a execução de tarefas repetitivas e que demandariam grande esforço se executadas manualmente. Afinal, o apoio computacional tem o potencial de facilitar a criação de produtos similares ou melhores que os produzidos manualmente e por pessoas com menos habilidades do que seria necessário sem o uso de ferramentas e, também, ajudam a reduzir o custo de execução das tarefas (BARRY *et al.*, 2007).

Como exemplo de processo de software que requer grande esforço para execução, podemos citar o processo de Medição. Por exemplo, XU *et al.* (2006) descrevem uma infra-estrutura para apoiar a implantação de um programa de medição utilizado durante a implantação do Nível 3 do CMMI em uma grande organização. Esta infra-estrutura provê um conjunto de medidas e apoio ferramental para a coleta e análise destas medidas e utiliza uma ferramenta composta por componentes para captura, armazenamento, análise e comunicação de resultados. Os resultados de um programa de medição também podem ser utilizados como subsídio para a simulação de processos. FERNANDEZ *et al.* (2004) descrevem uma abordagem para a definição e simulação de processos de software com o auxílio de um guia eletrônico de processos. Ruiz *et al.* (2002) descrevem o uso de simulação

e dinâmica de sistemas como forma de promover melhoria de processos de software apoiada por uma ferramenta que, ao utilizar dados de projetos passados, tenta prever o comportamento de um novo projeto ao longo do tempo. A própria construção do modelo de simulação permitiu ao gerente do projeto ganhar *insights* em aspectos do projeto relacionados ao processo de desenvolvimento com grande influência no sucesso do projeto, como tempo, custo e qualidade.

O uso de medidas também pode ser útil para monitorar quantitativamente uma iniciativa de melhoria de processos de software. Por exemplo, TIANFIELD (2003) descreve um *framework* baseado em computação autonômica² com o objetivo de gerenciar e controlar via dados quantitativos uma iniciativa de melhoria de processos de software. JARVINEN (2000) propõe uma ferramenta com o objetivo de associar as medições de um plano baseado no GQM (BASILI *et al.*, 1994) com os processos e práticas básicas da ISO/IEC 15504. O objetivo é mapear e, posteriormente, avaliar (*asses*) as atividades do processo dos projetos em relação a um modelo de referência de processos de software para dar maior confiança à monitoração. Como resultado, as informações sobre avaliações contínuas permitiram a identificação de novos *insights* nas seções de *feedback* do GQM, por exemplo, ao tornar mais claras aos participantes dos projetos fatores que impactavam a execução do processo e, assim, permitir uma melhor execução do processo.

Muitas ferramentas também são desenvolvidas visando a avaliação (*assessment*) de processos ou das abordagens desenvolvidas para melhoria em processos. CHOI *et al.* (2005), por exemplo, apresentam um protótipo construído com o objetivo de apoiar a realização de “*gap analysis*” em execuções de processos com base na norma ISO/IEC 15504. Segundo os autores, os participantes do estudo ganharam confiança em sua habilidade de desempenhar os papéis no grupo do projeto. Depois a ferramenta foi evoluída para funcionar também como uma ferramenta de treinamento para os desenvolvedores. RAMACHANDRAN (2005) apresenta um *framework* para melhoria de processos para avaliar (*asses*) e melhorar as práticas de XP (*Extreme Programming*). Foi construída uma ferramenta na web com o foco em pequenas e médias empresas, um protótipo que implementa uma base de conhecimento sobre técnicas de XP buscando

² A computação autonômica (TIANFIELD, 2003) é inspirada no comportamento do sistema nervoso humano e possui duas características principais: (i) auto-mecanismos, responsáveis por controlar automaticamente o comportamento do sistema; e (ii) *complexity hiding*, responsável por prover aos usuários um ambiente que permita a eles se concentrarem apenas no que querem fazer sem se preocupar sobre como deve ser feito.

identificar que melhorias podem ser implementadas na organização em relação à aderência ao XP.

3.3 Ambientes para Apoio a Iniciativas de Melhoria de Processos

Os requisitos de modelos de maturidade e capacidade, por seu aspecto abrangente e o pressuposto de haver diferentes tipos de processos distribuídos em diferentes níveis, apresentam uma maior complexidade em relação ao apoio ferramental por gerar a necessidade de automatização de um grande número de tarefas especializadas. Dessa forma, uma abordagem comum em organizações com estes requisitos é a adoção de ambientes (ou, pelo menos, um conjunto de ferramentas integradas que se comportam, de fato, como um ambiente no sentido clássico da acepção) para apoiar as tarefas necessárias. Se os relatos de uso de ferramentas CASE como apoio efetivo a iniciativas de melhoria de processos de software não são abundantes na literatura, a exemplificação de uso de ambientes para o apoio a iniciativas de melhoria de processo são ainda mais raros. Exemplos de tais ambientes são o Future (JUN *et al.*, 2007), ProjectMan (GONG *et al.*, 2006b), Project Web (MOE *et al.*, 2005) e a Estação Taba (MONTONI *et al.*, 2005; ROCHA *et al.*, 2005a; FERREIRA *et al.*, 2006b; MONTONI *et al.*, 2006c; MONTONI *et al.*, 2007b). Estes e outros ambientes que apóiam a execução de iniciativas de melhoria de processos de software são apresentados nesta seção.

Ambientes de Desenvolvimento de Software (ADS) têm evoluído ao longo do tempo para fornecer apoio mais amplo e efetivo aos desenvolvedores de software, de modo que metas como aumento da produtividade, melhoria da qualidade, diminuição de custos e, mais recentemente, diminuição do tempo para introdução no mercado possam ser alcançadas. VILLELA (2004) apresenta um histórico da evolução dos ADS e lista um conjunto de ADS concentrando-se em ADS Centrados em Processo e, em especial, em ADS Centrados em Processo que apóiam a Gerência de Conhecimento. MONTONI *et al.* (2006b) identificam outros ADS centrados em processo.

Os *Ambientes de Desenvolvimento de Software* identificados foram:

- ASTRAL (KOLANO *et al.*, 1999) – conjunto integrado de ferramentas de análise e projeto baseado na linguagem ASTRAL de especificação formal para sistemas de tempo real que permite a especificação modular de sistemas complexos. O ADS fornece funcionalidades para reduzir erros e facilitar a manutenção e a reutilização de especificações.

- CHIME (*Columbia Hypermedia IMersion Environment*) (DOSSICK e KAISER, 1999) – ADS baseado em meta-dados cujo objetivo é apoiar, através do uso da tecnologia de ambientes virtuais, grandes projetos de desenvolvimento de software, que, freqüentemente, têm membros dispersos geográfica e temporariamente.
- *Reusable Software Engineering Environment* (GOMAA e FARRUKH, 1999) – ADS composto de um ambiente de modelagem de domínio e um ambiente de configuração e programação distribuída que apóia as atividades de engenharia de domínio (modelagem, projeto e implementação de componentes) e atividades de configuração do sistema desejado (geração da especificação, composição da arquitetura e geração da aplicação executável).
- *Odyssey* (BRAGA *et al.*, 1999) – ADS que apóia o desenvolvimento de software baseado em componentes, contemplando desde a elaboração dos modelos conceituais até a implementação dos componentes.
- TDE (*Telecom Design Environment*) (ROSSI e MARTTIN, 2000) – ambiente que possibilita gerência das informações do projeto; fornece ferramentas gráficas que apóiam o desenvolvimento de software e permite que desenvolvedores compartilhem o mesmo espaço de trabalho e trabalhar sobre os mesmos artefatos, tendo sido adaptado para fornecer modelos de processos enriquecidos com orientação, mas sem oferecer suporte à execução do processo e ao controle do seu progresso.
- VPP (*Visual Programming Platform*) (WANG *et al.*, 2002) – ADS voltado para o desenvolvimento de software para instrumentos virtuais, onde um instrumento virtual é um instrumento que contém hardware e software com algumas funcionalidades, tais como processamento e apresentação de dados e controles inteligentes. O ambiente permite que o usuário graficamente crie diagramas da aplicação a partir de uma biblioteca de componentes.

Os *Ambientes de Desenvolvimento de Software Centrados em Processo* identificados foram:

- ASEE (*Agile Software Engineering Environment*) (AOYAMA, 1998) – composto de um ADS Centrado em Processo e de um ambiente de compartilhamento de informações baseado na *Web*. Visa a rápida distribuição de produtos de software e a rápida adaptação a mudanças de requisitos, promove uma fábrica de software virtual através da Internet e tendo sido usado para desenvolver sistemas de software de comunicação.

- ADDD (*A Development environment for Dynamic Documents*) (KOBIALKA, 1998) – Integra gerência de tarefa com gerência de configuração e cuja máquina de processo é baseada no paradigma de *triggers*, possibilitando instalação incremental do suporte ao processo, adaptação deste suporte de acordo os requisitos da tarefa específica, modificação do processo durante a execução e utilização de políticas para orientar e restringir o processo.
- PROSYT (*Process Support System*) (CUGOLA e GHEZZI, 1999) – Concebido para apoiar processos de software distribuídos, adota uma arquitetura baseada em código móvel e eventos, possui uma linguagem de modelagem de processo baseada em artefato e oferece flexibilidade na execução do modelo de processo.
- APSEE (ABREU *et al.*, 2003) – uma infra-estrutura integrada para automação de processos de software, composta de componentes que armazenam informações relativas ao meta-modelo, mecanismos para gerência de processo, e componentes de visualização e interação. A partir da especificação formal APSEE, o código fonte básico do ambiente é gerado. As principais contribuições são a flexibilidade na execução dos processos, permitindo a modificação dinâmica dos mesmos, e o grau de automação das diferentes fases do ciclo de vida.
- EPOS (*Expert System for Program and ("og") System Development*) (MINH *et al.*, 1997) – ADS centrado em processo com ênfase em modelagem de processo, gerência de configuração de software e apoio ao trabalho cooperativo.
- OZ (BEN-SHAUL *et al.*, 1994; FUGGETTA, 2000) – ADS centrado em processo descentralizado baseado em um projeto para descentralização da modelagem e execução de processo.
- SPADE (*Software Process Analysis, Design and Enactment*) (BANDINELLI *et al.*, 1996) – ADS centrado em processos baseado na utilização de uma linguagem de modelagem de processos denominada SLANG. Foi evoluído para apoiar a colaboração no desenvolvimento de software.
- Artemis 7 (AISC, 2006) – Baseado na Web, foi desenvolvido para apoiar processos de negócio e papéis associados e para permitir a implantação de múltiplas soluções em uma plataforma comum.

Os *Ambientes de Desenvolvimento de Software Centrados em Processo com apoio de Gerência de Conhecimento* identificados foram:

- MILOS (*Minimally Invasive Long Term Organizational Support*) (HOLZ *et al.*, 2001) – Tem como objetivo fornecer uma infra-estrutura que integre os conceitos de ADS

Centrado em Processo e de Gerência do Conhecimento. O sistema é composto de Ambiente de Modelagem Estendida de Processos, Ambiente de Planejamento de Projeto, Ambiente de Workflow e Assistente de Informação.

- *Ontology-based software Development Environment (ODE)* (BERTOLLO *et al.*, 2002; FALBO *et al.*, 2003) – ADS Centrado em Processo e baseado em ontologias que possui uma infra-estrutura de gerência do conhecimento.

Dentre os ambientes utilizados como apoio a iniciativas de melhoria de processos de software em organizações, e corporações, podemos citar a Estação Taba (MONTONI *et al.*, 2005; ROCHA *et al.*, 2005a; FERREIRA *et al.*, 2006b; MONTONI *et al.*, 2006c; MONTONI *et al.*, 2007b), SoftPM (WANG e LI, 2005b; WANG e LI, 2006; WANG *et al.*, 2006), FMESP (CANFORA *et al.*, 2006; GARCIA *et al.*, 2006; GARCIA *et al.*, 2007), DiME (KOENIG, 2003), Future (JUN *et al.*, 2007), WAGNER EPG/ER (KURNIAWATI e JEFFERY, 2004), PIASS (SAKAMOTO *et al.*, 1998), e-R&D (EBERT e DE MAN, 2002), ProjectMan (GONG *et al.*, 2006b) e Project Web (MOE *et al.*, 2005). Além desses, o WebAPSEE (COSTA *et al.*, 2007; PAXIÚBA *et al.*, 2007), o ImPProS (OLIVEIRA e VASCONCELOS, 2006) e o CORE-KM (GALOTTA, 2003; MONTONI *et al.*, 2007a; MONTONI e ROCHA, 2007) como ambientes provenientes de projetos acadêmicos candidatos a apoiar iniciativas de melhoria de processos em organizações. Ferramentas comerciais, como o *Microsoft Team Foundation* (MICROSOFT, 2008) e as suítes da IBM Rational (IBM, 2008) e da Borland (BORLAND, 2008) também podem ser adaptadas a este contexto.

3.3.1 Ambientes de Desenvolvimento da COPPE/UFRJ

A área de Engenharia de Software da COPPE/UFRJ vem desenvolvendo, há alguns anos, ambientes de apoio à execução de processos organizacionais e de desenvolvimento de software. Estes ambientes são utilizados durante projetos de implantação e melhoria de processos de software e/ou no apoio à gerência de conhecimento nas organizações.

3.3.1.1 Estação Taba

A Estação Taba surgiu em 1990 (ROCHA *et al.*, 1990) como um meta-ambiente de desenvolvimento de software capaz de gerar, através de instanciação, ambientes de desenvolvimento de software (ADS) centrados em processos e adequados a projetos com diferentes características. A implementação da Estação Taba foi iniciada, de fato, em 1994 a partir do trabalho de TRAVASSOS (1994). No início, foi desenvolvida utilizando a

linguagem Eiffel numa estação de trabalho Sun. Ao longo do tempo, percebeu-se que o poder e a robustez da plataforma, apesar de adequadas ao ambiente de pesquisa, dificultavam a experimentação das idéias em ambientes reais de desenvolvimento de software devido à falta de portabilidade do código para plataformas mais acessíveis e comumente utilizadas nas empresas brasileiras. Na versão em Eiffel da Estação Taba, foram instanciados dois ADS: Orixás (WERNECK, 1995), com o objetivo de apoiar o desenvolvimento de sistemas baseados em conhecimento, e Memphis (WERNER *et al.*, 1997), com o objetivo de apoiar o desenvolvimento de software baseado em reutilização.

Em 1999 a Estação Taba foi estendida para a criação de Ambientes de Desenvolvimento de Software Orientados a Domínio (OLIVEIRA, 1999) através da percepção de que o conhecimento sobre o domínio da aplicação a ser desenvolvida também é relevante aos desenvolvedores na execução de suas tarefas. Com o intuito de aumentar as possibilidades de utilização prática de tais ambientes, foi realizada uma re-implementação da Estação Taba onde se optou pela plataforma de microcomputadores e pelo uso da linguagem C++ (OLIVEIRA, 1999; SANTOS e ZLOT, 1999). Nesta reimplementação foram migradas para a nova plataforma o núcleo básico da Estação Taba e algumas das ferramentas relacionadas à definição de processos necessárias para a definição de um ADS. A versão em C++ foi utilizada para a definição e instanciação de três ADSOD: Cordis (OLIVEIRA, 1999), para o domínio de cardiologia, Netuno (GALOTTA, 2000), para o domínio de Acústica Submarina, e Insecta (FOURO, 2002), para o domínio de entomologia³.

Em 2004, uma nova evolução da Estação Taba (VILLELA, 2004) deu origem aos Ambientes de Desenvolvimento de Software Orientados a Organização (ADSOrg), o que permitiu a configuração de ambientes para uma organização específica, denominados Ambientes Configurados. Estes ambientes possibilitam a instanciação do ambiente para projetos específicos e a aquisição e disseminação do conhecimento organizacional adquirido durante o desenvolvimento de software através dos ambientes instanciados para cada projeto. Com a criação dos ADSOrg, novamente o modelo foi revisto para adequá-lo ao novo esquema de configuração/instanciação de ambientes e para permitir a definição de novas ferramentas.

Os ADSOrg têm como objetivo:

- a) apoiar os desenvolvedores de software na execução de suas atividades, fornecendo todo o conhecimento que tenha sido capturado e acumulado pela organização por

³ Entomologia é uma disciplina da agronomia/zoologia que estuda os insetos.

sua importância para o desenvolvimento e a manutenção de software, e

- b) apoiar o aprendizado organizacional em Engenharia de Software a partir do aprendizado dos desenvolvedores da organização nos projetos de software específicos.

A estratégia para construção de ADSOrg é iniciada com a configuração do Meta-ambiente para uma organização. Através dessa configuração são levantadas as principais características do desenvolvimento de software e é disponibilizado um ambiente capaz de dar apoio às atividades de software. Este ambiente contém o processo padrão e processos especializados para o desenvolvimento e/ou manutenção de software, além de conhecimento para dar apoio às atividades envolvidas nestes processos. De posse deste ambiente a organização pode definir novos ambientes, denominados Ambientes do Projeto, que apóiam a execução das atividades dos processos em projetos específicos, além de apoiar o aprendizado organizacional a partir dos projetos. Estes ambientes são gerados a partir da adaptação dos processos especializados para atender às características dos projetos específicos.

A Figura 3.1 exibe o esquema atual utilizado para a construção de ADS na Estação Taba.

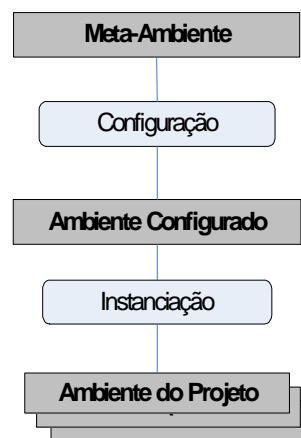


Figura 3.2 – Esquema para Construção de ADS na Estação Taba

Estes ambientes podem ser definidos da seguinte maneira:

- **Meta-Ambiente:** É utilizado em uma entidade externa à organização desenvolvedora (atualmente apenas a COPPE), sendo utilizado por profissionais especializados em engenharia de software, que fazem a manutenção e evolução dos ativos de processos e demais recursos disponíveis neste ambiente. Sua principal função é apoiar a configuração de ambientes para organizações específicas, gerando os Ambientes Configurados.

- **Ambiente Configurado:** É utilizado por profissionais de nível gerencial e pelo grupo de processos da organização. Uma de suas principais funções é apoiar o processo de Instanciação, ou seja, a geração dos ambientes que serão utilizados pelos profissionais nos projetos, os Ambientes do Projeto.
- **Ambientes de Projeto:** Ambiente de desenvolvimento de software instanciado a partir do Ambiente Configurado e específico a um projeto (também chamados de Ambientes Instanciados). É utilizado pelo gerente/líder do projeto e demais membros da equipe do projeto, dando apoio às várias atividades dos processos do ciclo de vida de um software através de ferramentas específicas, conhecimento organizacional, descrição de procedimentos e *templates*, dentre outros.

Cada um destes ambientes é composto por um conjunto de ferramentas e serviços que auxiliam os engenheiros de software na definição, execução, avaliação e melhoria dos processos de desenvolvimento e manutenção. Eles também possuem funcionalidades de gerência de conhecimento integradas às atividades dos processos apoiados de forma a preservar o conhecimento organizacional e aumentar a institucionalização destes processos.

Desde 2003, a Estação Taba vem sendo utilizada pela indústria no apoio a implantação e melhoria de processos de software com foco no MPS.BR (SOFTEX, 2007) e no CMMI (CHRISIS *et al.*, 2006). Os resultados de uso da Estação Taba incluem o papel importante no processo de obtenção de avaliações CMMI e MPS.BR em diversas empresas brasileiras (DUARTE *et al.*, 2005; ROCHA *et al.*, 2005a; ROCHA *et al.*, 2005b; FERREIRA *et al.*, 2006a; FERREIRA *et al.*, 2006b; GUERRA *et al.*, 2006; MACEDO *et al.*, 2006; MONTONI *et al.*, 2006b; MONTONI *et al.*, 2006c; NUNES *et al.*, 2006; ROCHA *et al.*, 2006; FERREIRA *et al.*, 2007a; FERREIRA *et al.*, 2007b; FERREIRA *et al.*, 2007c; MONTONI *et al.*, 2007b; SANTOS *et al.*, 2007a; SANTOS *et al.*, 2007b; SCHEID *et al.*, 2007; SANTOS *et al.*, 2008a; SANTOS *et al.*, 2008b). O principal objetivo da Estação Taba neste sentido é de prover uma infra-estrutura consistente com os requisitos (ou parte dos requisitos) destes modelos com o objetivo de superar uma das dificuldades da implantação de iniciativas de melhoria de processos de software, que é a falta de recursos financeiros para aquisição de ferramentas comerciais.

Como dados quantitativos, podem-se mencionar a redução do tempo gasto em retrabalho no projeto de 44% para 7% decorrente de iniciativa de melhoria de processo, realizada com o uso de ferramentas de apoio da Estação Taba, em uma destas empresas (FERREIRA *et al.*, 2006b; FERREIRA *et al.*, 2007b).

Como a Estação Taba foi desenvolvida pela COPPE/UFRJ no contexto acadêmico, através de várias teses de doutorado e dissertações de mestrado, ela não é comercializada e, sim, disponibilizada às empresas sem custo.

Diversas ferramentas compõem a Estação Taba e seus ambientes. As principais ferramentas em uso atualmente são utilizadas no apoio a implantação de processos em empresas e, assim, apóiam os níveis G, F, E, D e C do MR-MPS (e equivalentes do CMMI) (MONTONI *et al.*, 2007b). Atualmente, há teses de doutorado em andamento que visam à definição de ferramentas (e, possivelmente, alterações em ferramentas existentes) para apoio às exigências dos níveis A e B do MR-MPS (e equivalentes do CMMI).

Apesar de a descrição das ferramentas apresentadas seguir a estrutura dos níveis de maturidade do MPS.BR, elas podem ser utilizadas por quaisquer empresas independentemente do nível pretendido. Por exemplo, as ferramentas relacionadas a Gerência de Conhecimento estão intrinsecamente conectadas às demais ferramentas e, portanto, delas não podem ser desassociadas. Além disso, devido à natureza evolutiva dos níveis de maturidade, as ferramentas de um nível inferior (por exemplo, o nível G) são sempre utilizadas quando se deseja atingir um nível superior (por exemplo, o nível C). Note, ainda, que as ferramentas também podem ser utilizadas em diferentes combinações, independentemente de a empresa estar interessada em ser aderente a um determinado nível ou modelo de maturidade.

Apoio ao nível G do MR-MPS – as principais atividades relativas ao planejamento de projetos são implementadas através de três ferramentas que apóiam a elaboração do cronograma do projeto, a TempManager (BARCELLOS, 2003; BARCELLOS *et al.*, 2003), plano de custos, a CustManager (BARCELLOS, 2003; BARCELLOS *et al.*, 2003), e plano de alocação de recursos humanos, a RHManager (SCHNAIDER, 2003). As ferramentas também provêm apoio à monitoração do progresso do projeto em relação aos planos elaborados e permitem a avaliação do desempenho do projeto. A ferramenta ReqManager possibilita que a rastreabilidade bidirecional entre os requisitos seja identificada para cada nível de decomposição do produto e do plano do projeto. Ainda há ferramentas que apóiam a identificação dos requisitos de qualidade de produtos, como o qFuzzy (OLIVEIRA, 1999), a elaboração de planos de ação ao longo do projeto (ActionPlanManager), a identificação e acompanhamento de marcos e pontos de controle do projeto (ControlManager), a elaboração de um plano de organização para um projeto (OrgPlan), o registro das atividades do desenvolvedor no projeto (Planilha de Atividades) e o registro e comunicação da situação dos projetos sendo conduzidos pela organização

(ProjectStatus). A gerência de riscos é apoiada pela ferramenta RiscManager (FARIAS, 2002) que permite a identificação dos riscos baseada no conhecimento adquirido através da execução de projetos passados. Esta ferramenta também sugere fatos e restrições associados a cada risco, apóia a definição de planos de mitigação e contingência e, também, a monitoração dos riscos em relação ao plano definido.

Apoio ao nível F do MR-MPS – práticas de medição são apoiadas por duas ferramentas integradas, MedPlan e Metrics (SCHNAIDER *et al.*, 2004), que provêm os meios para a definição de planos de medição organizacionais e de projetos baseado no método GQM (BASILI *et al.*, 1994) e também apóiam a coleta e análise dos dados. As tarefas relacionadas ao processo de gerência de configuração são apoiadas pela ferramenta GConf (FIGUEIREDO *et al.*, 2004) que provê um mecanismo para monitorar as alterações nos requisitos ao longo do projeto, para controlar os pedidos de mudanças e para o versionamento de documentos e estabelecimento e liberação de *baselines*. A geração de laudos de aderência à execução do processo é apoiada através da integração com a ferramenta AdaptPro de definição de processos para o projeto (MONTONI *et al.*, 2006a).

Apoio ao nível E do MR-MPS – as tarefas relacionadas à definição de processos são implementadas através da ferramenta GConf (VILLELA, 2004; VILLELA *et al.*, 2004) que utiliza a norma ISO/IEC 12207 (ISO/IEC, 2008b) como base para definição de processos e também modelos de maturidade como o MR-MPS (SOFTEX, 2007) e o CMMI (CHRISISS *et al.*, 2006). Os processos padrão e especializados para um determinado paradigma (por exemplo, estruturado ou orientado a objetos) são os processos padrão organizacionais. Para que o processo padrão seja utilizado num projeto específico, um processo especializado escolhido devido à adequabilidade ao projeto deve ser instanciado considerando as características do projeto (por exemplo, tamanho, complexidade do produto e características relevantes de garantia da qualidade e da equipe de desenvolvimento).

A instanciação do processo é apoiada pela ferramenta AdaptPro (BERGER, 2003; VILLELA *et al.*, 2004) que permite a seleção do modelo de ciclo de vida e os métodos e ferramentas de desenvolvimento. A abordagem garante que os projetos são executados com base na definição de um processo adaptado a partir do conjunto de processos padrão da organização e facilitam a institucionalização de processos padrão. A ferramenta AvalPro (ANDRADE, 2005; CAMPOS *et al.*, 2006) permite a avaliação e melhoria do processo de software organizacional e dos ativos de processo com base na avaliação de medidas de monitoração do processo, relatórios à gerência de alto nível gerados durante os projetos,

checklists de aderência ao processo e análises *post mortem* e a revisão de lições aprendidas capturadas. A ferramenta Pilot (SILVA FILHO, 2006) permite a identificação, planejamento, execução e avaliação das oportunidades de melhoria do processo através de projetos piloto.

A existência de uma Biblioteca de Ativos de Processo para organização integrada à ferramenta de Gerência de Configuração possibilita o controle da evolução e divulgação destes ativos (SANTOS *et al.*, 2008a).

O apoio à Gerência de Conhecimento é feito através da integração dos ambientes e todas as suas ferramentas à ferramenta Acknowledge de gerência de conhecimento (MONTONI, 2003) que apóia a disseminação das lições aprendidas e também a evolução do repositório de conhecimento com experiências adquiridas durante os projetos de software por todos os seus participantes. Esta abordagem permite a identificação de pontos fracos e fortes dos processos e também facilita a identificação de oportunidades de melhoria (MONTONI *et al.*, 2006a). Além disso, a maioria das ferramentas provê conhecimento específico e especializado sobre a subárea de Engenharia de Software relacionada, por exemplo, a ferramenta de adaptação de processo provê conhecimento sobre diferentes modelos de ciclo de vida. Outra ferramenta, a Sapiens, permite a descrição e visualização de estruturas organizacionais, englobando os profissionais alocados e as competências requeridas e possuídas ao longo dessas estruturas (SANTOS, 2003; SANTOS *et al.*, 2005) construindo um mapa de conhecimento associado com uma “página amarela” de especialistas dentro da organização.

Apoio ao nível D do MR-MPS – o desenvolvimento, a avaliação e a seleção de alternativas de solução técnica são apoiados por duas ferramentas, MBR e TechSolution (FIGUEIREDO *et al.*, 2006), que provêm o mapeamento automático dos critérios de avaliação a produtos de trabalho específicos e permite a adaptação de tal conhecimento a necessidades específicas de clientes e sistemas e/ou requisitos de software. Outras duas ferramentas apóiam a verificação, a VerificationManager (BARRETO, 2006), e a validação de software, a ValidationManager (NATALI, 2006), ao longo do processo de desenvolvimento.

Apoio ao nível C do MR-MPS – a formalização da decisão de análise e resolução relacionada ao projeto arquitetural é provida pelas ferramentas MBR e TechSolution, que apóiam o processo de solução técnica (FIGUEIREDO *et al.*, 2006). A gerência de riscos neste nível também é apoiada pela ferramenta RiscManager (FARIAS, 2002).

3.3.1.2 CORE-KM: Um Ambiente Customizável de Gerência de Conhecimento

Organizações, mesmo as que desenvolvem software, possuem processos organizacionais que são executados por seus membros como atividade fim e, também, como forma de apoio a outros processos (por exemplo, os de software, citados anteriormente). Estes processos muitas vezes interagem entre si e precisam ser apoiados, de forma integrada, da mesma forma como os processos de software. Segundo GALOTTA (2003), citando (ABNT, 2000a) e (ABNT, 2000b), para uma boa gestão da qualidade é necessária a identificação de processos inter-relacionados e interativos, pois, freqüentemente, a saída de um processo resultará diretamente na entrada do processo seguinte. Além disso, um dos problemas mais comuns nas organizações é a dificuldade em manter dentro da organização o conhecimento necessário para executar os seus processos de negócio. Esse conhecimento constitui um verdadeiro capital intelectual e é crítico para garantir a sobrevivência de organizações de qualquer tipo, tamanho e área de atuação (MONTONI, 2007).

Pensando neste problema, e a partir da constatação de que cada organização tem características próprias e processos organizacionais diversos, surgiu a idéia de apoiar a execução destes processos através de um ambiente de gerência de conhecimento específico que possa ser customizado para diferentes organizações, de acordo com suas necessidades relacionadas a conhecimento e processos organizacionais (GALOTTA, 2003). Um modelo de ambiente foi definido, então, para apoiar a definição, customização e execução de ambientes de gerência de conhecimento específicos: o CORE-KM (*Customizable Organizational Resources Environment with Knowledge Management*). Dessa forma, este ambiente é customizável para gerência de conhecimento em diferentes organizações, capaz de apoiar seus processos organizacionais; cada customização é, portanto, um ambiente diferenciado que contempla as características particulares de cada organização. A proposta original deste ambiente prevê sua integração com os ambientes configurados a partir da Estação Taba (GALOTTA, 2003).

Para a definição e construção do ambiente CORE-KM, foram identificadas funções que devem fazer parte do ambiente customizável, além de requisitos capazes de atendê-las.

As funções identificadas foram (GALOTTA, 2003):

- (i) Auxiliar o engenheiro de conhecimento na especificação do ambiente de gerência de conhecimento mais adequado à organização;

- (ii) Auxiliar o engenheiro de conhecimento na customização do ambiente de gerência de conhecimento;
- (iii) Permitir ao engenheiro de software a integração, ao ambiente customizado, de ferramentas desenvolvidas para apoiar processos específicos da organização;
- (iv) Permitir ao gerente de conhecimento a execução do ambiente customizado independentemente do ambiente CORE-KM.

Alguns trabalhos já foram realizados no contexto do ambiente CORE-KM. MONTONI (2003) desenvolveu uma ferramenta de apoio à pesquisa médica. COSTA (2003) desenvolveu uma ferramenta de apoio à elaboração e propostas de projetos de software. MURADAS (2006) apoiou a avaliação da maturidade de processo de software por meio da definição de um processo de avaliação de maturidade apoiado por uma ferramenta de avaliação. GALOTTA *et al.* (2004) desenvolveram um trabalho base para a definição de um meta-ambiente de customização de ambientes específicos para uma organização, o qual é o próprio CORE-KM. Uma abordagem para gerência da condução de iniciativas de melhoria de processos de software foi definida e está sendo implementada utilizando a infra-estrutura do CORE-KM (MONTONI *et al.*, 2007a; MONTONI e ROCHA, 2007; CERDEIRAL, 2008).

3.3.2 Outros Ambientes

Esta subseção lista os ambientes citados na literatura utilizados em iniciativas de melhoria de processos de software em organizações. A maior parte destes ambientes é fruto de iniciativas das próprias empresas e não há subsídios para relacioná-los a abordagens surgidas do ambiente acadêmico (como aqueles ambientes citados na seção anteriormente por (VILLELA, 2004) e (MONTONI *et al.*, 2006b)). As exceções são o ImPProS (OLIVEIRA e VASCONCELOS, 2006) e o WebAPSEE (COSTA *et al.*, 2007), além da Estação Taba e do CORE-KM, já citados anteriormente. Destes ambientes, apenas o PIASS (SAKAMOTO *et al.*, 1998) e o e-R&D (EBERT e DE MAN, 2002) foram desenvolvidos no contexto de corporações.

SoftPM (WANG e LI, 2005b; WANG e LI, 2006; WANG *et al.*, 2006) é um sistema integrado de apoio às atividades de gerentes de projetos, alta gerência, engenheiros, testadores, membros da área de garantia da qualidade e outros membros de áreas de apoio. Ajuda a compartilhar os dados coletados, entender o cronograma, esforço e qualidade do projeto e auxilia a comunicação entre os participantes do projeto. A Plataforma para Gerência da Qualidade é composta por quatro ferramentas (gerência de projetos, biblioteca

de ativos de processo, garantia da qualidade e medição e análise) que podem ser combinadas de forma a aumentar a aderência ao CMM/CMMI. Esta plataforma é baseada num ciclo do PDCA (DEMING, 1986) e apóia organizações de diferentes maneiras de acordo com o nível de maturidade em processos pretendido. Há duas outras plataformas para engenharia de produtos e de apoio a serviços. Segundo os autores, o ambiente já foi utilizado com sucesso em mais de 100 empresas de software chinesas com uma redução de cerca de 35% do esforço referente às atividades do grupo de processos, área de qualidade e gerentes de projetos.

FMESP (*Framework for the Modelling and Measurement of Software Processes*) (CANFORA *et al.*, 2006; GARCIA *et al.*, 2006; GARCIA *et al.*, 2007) provê o apoio necessário para a representação e gerência do conhecimento relacionado a processos de software a partir das perspectivas de modelagem e medição, ao integrá-las. O ambiente possui componentes para descrição de ontologias (e é baseado no uso de ontologias de processo e de medição), modelagem de processos e apoio à medição de software. Foi utilizado numa organização espanhola dedicada ao desenvolvimento e manutenção de software que obteve, como resultado, uma certificação ISO 9000.

DiME (KOENIG, 2003) é um sistema cujo objetivo é gerenciar a definição, desenvolvimento e entrega de informação e processos de um produto. Apóia os processos relativos a diferentes classes de pessoas nos projetos (por exemplo, gerente, arquiteto, testadores etc.). Além disso, provê um repositório central para armazenamento de dados, automatiza os workflows e possui uma interface padrão para as ferramentas. Segundo os autores, a empresa que o utilizou, localizada em Israel, passou por uma evolução silenciosa na forma como define, desenvolve e entrega produtos; o conhecimento resultante foi mais rico, mais disponível, acessível, uniforme e confiável, enquanto os processos foram automatizados, padronizados e medidos em toda a organização.

Future (JUN *et al.*, 2007) é baseado na estrutura do processo definido para a organização e tem o objetivo de apoiar a execução deste processo. Foi utilizado para a implantação do CMMI níveis 2 e 3 em organizações de software chinesas com foco no desenvolvimento de software embutido.

WAGNER EPG/ER (KURNIAWATI e JEFFERY, 2004) é utilizado para a divulgação de processos numa organização através de um guia eletrônico de processos (EPG, do inglês *Electronic Process Guide*) e baseado no *framework* WAGNER com apoio de um repositório de experiências (ER). O objetivo do sistema é divulgar os processos da organização, aumentando o seu uso e melhorando a institucionalização, além da criação de

um repositório de experiências e, conseqüentemente, do aumento da memória organizacional.

PIASS (*Process Improvement Activities Support Systems*) (SAKAMOTO *et al.*, 1998) é um protótipo de ambiente com ferramentas para apoiar a iniciativa de melhoria de processos numa corporação japonesa na avaliação (*assessment*) de processos, gerência de conhecimento e gerência das atividades relacionadas a melhoria de processos de software. Este ambiente possui funções próprias para membros do grupo de processos, desenvolvedores e gerentes da iniciativa de melhoria de processos de software. A utilização do protótipo não é descrita pelos autores, entretanto, sua definição foi fortemente fundamentada na observação dos problemas encontrados durante a iniciativa de melhoria de processos e visa a suprir várias necessidades identificadas ao longo da iniciativa de melhoria de processos que durou 17 meses. Como resultado, houve a redução de 30% do esforço em homem-mês no projeto alvo da iniciativa de melhoria.

e-R&D (EBERT e DE MAN, 2002) é um sistema de gerência de *workflow* e adaptação de processos com uma biblioteca de ativos com apoio de gerência de configuração utilizado para gerenciar a diversidade de processos dentro de uma corporação francesa. Segundo os autores, os benefícios conseguidos incluíram: melhora da qualidade, redução do tempo de desenvolvimento, aumento da flexibilidade de engenharia, redução de esforço para tarefas, melhora na comunicação, melhora no alinhamento de processos e ferramentas, maior facilidade de geração de planos de treinamento.

ProjectMan (GONG *et al.*, 2006b) foi construído para ser utilizado no apoio à implantação dos níveis 2 e 3 do SW-CMM com base no IDEAL de forma adequada às pequenas empresas de software chinesas. É composto de 4 unidades: ferramenta de garantia da qualidade (relacionada a testes) baseada no GQM, ferramentas para apoiar as práticas do nível 2 e 3 do CMM, apoio ao trabalho colaborativo e integração com um *framework* de melhoria de processos de software. Este *framework* possui três etapas: definição de processos (definição do programa de melhoria incluindo infra-estrutura, responsabilidades e comprometerimentos); controle do processo (refinamento da estratégia, definição dos processos de desenvolvimento, integração das melhorias nos processos em projetos já em andamento, apoio às organizações no uso dos novos processos); estabilidade do processo (estabelecimento de revisões formais, garantia da qualidade, engenharia de produto, testes formais, revisão por pares, monitoração dos processos via medição, análise de lições aprendidas e desenvolvimento de plano para guiar o programa de melhoria).

Project Web (MOE *et al.*, 2005) é um guia eletrônico de processos para apoiar a melhoria de processos de software e fazer a conexão dos processos com as ferramentas de apoio, relacionadas a planejamento e gerência, disponíveis na empresa. Estas ferramentas de apoio incluem: planejamento de orçamento e estimativas (com relatórios de progresso; rastreamento de ações); planejamento de riscos; planejamento de pagamentos; documentações iniciais e finais do projeto; inventário; planejamento do projeto; lista de itens a serem entregues com o projeto; arquivamento de documentos e estatísticas relacionadas à execução do projeto. A abordagem evolutiva durante a implantação do ambiente numa empresa norueguesa resultou num foco contínuo na melhoria do processo de software na organização. O alto grau de envolvimento foi provavelmente a razão pelo qual o Project Web foi considerado um sucesso. A estratégia de dar foco nas ferramentas e a integração entre elas tornou o sistema como um todo mais útil. O uso do guia eletrônico de processos do Project Web também tornou possível para a empresa a manutenção da certificação ISO 9001:2000.

ImPProS (Ambiente de Implementação Progressiva de Processo de Software) (OLIVEIRA e VASCONCELOS, 2006) é um ambiente para melhoria de processos que apóia a adoção do IDEAL. Seu conjunto de funcionalidades inclui definição, simulação, execução, avaliação, melhoria e reuso de processos, análise e tomada de decisão com base nas avaliações dos processos, gerência de conhecimento relativa a processos e conversão da estrutura de processos com base em normas e modelos de qualidade. No entanto, não foram encontrados na literatura relatos da aplicação no ambiente na indústria (IMPPROS, 2008).

WebAPSEE (COSTA *et al.*, 2007) fornece apoio automatizado para a gestão de processos de software, sendo projetado para permitir a integração de vários serviços relacionados com uma visão bastante ampla do meta-processo de software. A ferramenta apóia desde a concepção e levantamento de requisitos de processo até a geração de análise *post mortem* dos processos, passando pelo controle da execução dos processos de forma flexibilizada. O WebAPSEE hoje é utilizado em projetos (PAXIÚBA *et al.*, 2007), porém não foram encontrados na literatura relatos da aplicação prática na indústria.

Dentre as abordagens estritamente comerciais, pode-se citar o *Microsoft Solutions Framework for CMMI Process Improvement* que encapsula uma abordagem para o desenvolvimento de software e melhoria contínua do processo (HUNDHAUSEN, 2006). Este *framework* é composto por um guia com processos básicos associados com papéis comuns no desenvolvimento de software (por exemplo, gerentes, testadores,

desenvolvedores etc.) e *templates* que auxiliam a executar as tarefas destes processos. As ferramentas disponíveis no *Visual Studio Team System* (MICROSOFT, 2008) podem ser utilizadas em conjunto com este *framework*. Por exemplo, o *Enterprise Project Management* (EPM) provê apoio à gerência e monitoração do projeto, principalmente na elaboração do cronograma (com o auxílio do MS-Project), atribuição e gestão de equipes, conciliação de recursos, gestão de custos etc. *Visual Studio Team System* (VSTS) provê apoio ao ciclo de vida de engenharia de software, a gestão de riscos (estratégia e planejamento) e a definição de escopo do projeto (cenários). Tanto o VSTS quanto o EPM permitem gerar relatórios e consultas importantes para obtenção de base histórica a ser usada nas estimativas. A ferramenta *Microsoft Performance Point* (MPP) permite o planejamento da estratégia, definição e monitoramento dos indicadores. A geração e acompanhamento de indicadores táticos e operacionais podem ficar a cargo das ferramentas VSTS (engenharia) e solução EPM (gestão de projetos e portfólio). Em relação à gerência de configuração, a ferramenta VSTS provê sistema de controle de versão, criação de *baselines* diretamente relacionados a código e artefatos de engenharia de software e gerenciamento do fluxo de mudança nos itens de configuração; o *SharePoint* provê versionamento mais simples e armazenamento de artefatos (documentos principalmente) com controle de acesso. O VSTS operacionaliza e automatiza o uso do *MSF for CMMI Process Improvement*, principalmente em relação aos processos relacionados à engenharia de desenvolvimento de aplicações (elicitação e gerência de requisitos, solução técnica, arquitetura, codificação, integração, testes, configuração, etc.) (VASQUES e GOMES, 2007).

Pode-se citar, ainda, como exemplos de ferramentas comerciais a suíte Rational da IBM (IBM, 2008) e o conjunto de ferramentas da Borland (BORLAND, 2008). A Rational começou como uma empresa especializada em ferramentas e IDEs para o desenvolvimento de software. A partir de 1995 investiu na construção de um conjunto de aplicativos para dar apoio à utilização de um processo padrão adaptável às empresas (o *Rational Unified Process*, ou, simplesmente RUP) e à modelagem de software baseada em UML (*Unified Modelling Language*, padrão que ajudou a criar). A Rational, atualmente uma subsidiária da IBM, possui um conjunto de ferramentas que auxiliam a modelagem de dados baseada em UML (*Rational Rose*, *Software Architect*), gerência de configuração e mudanças (*ClearCase* e *ClearQuest*), gerência de requisitos (*RequisitePro*), qualidade de código com apoio a testes e gerência de portfólio de projetos. A Borland, antes uma tradicional fabricante de compiladores e IDEs para desenvolvimento de software, atualmente tem seu foco direcionado a produtos de apoio ao ciclo de vida de desenvolvimento de software. Seus

produtos incluem ferramentas para modelagem de dados (*Together*), gerência de requisitos (*CaliberRM* e *Caliber DefineIT*), gerência de configuração e mudanças (*StarTeam*), apoio a testes e gerência de portfólio de projetos.

Tanto as ferramentas da Microsoft, quanto as da Borland e da IBM Rational, no entanto, precisam ser configuradas adequadamente para uso e devem ser adaptadas ao processo em uso e ao fluxo de vida dos produtos de trabalho em cada organização individualmente.

3.4 Considerações Finais

Este capítulo teve como objetivo mostrar um panorama dos diversos tipos de apoio ferramental para iniciativas de melhoria de processos de software. O apoio ferramental é tão diverso quanto o escopo destas iniciativas. É interessante notar, no entanto, que são poucos os relatos que relacionam o uso efetivo de apoio ferramental no contexto de iniciativas de melhoria de processos, o que pode ser visto pelo resumo do estudo baseado em revisão sistemática presente no Anexo III. Mais raros são os relatos de uso de ambientes de desenvolvimento de software neste contexto. Também raros são os relatos de iniciativas de melhoria de processos em corporações.

Uma exceção sobre o uso de ambientes em organizações têm sido os relatos de uso da Estação Taba a partir de 2003, quando começou a ser utilizada em empresas. Mais do que discutir a adequação de sua infra-estrutura aos paradigmas associados com a literatura referente a ambientes (por exemplo, os requisitos e características citados, por exemplo, por (GARG e JAZAYERI, 1996; KOBIALKA, 1998; ARBAOUI *et al.*, 2000; FUGGETTA, 2000; HARRISON *et al.*, 2000)), o que tem sido garantido desde a sua definição, é interessante notar o apoio ferramental abrangente para várias atividades comuns em iniciativas de melhoria de processos em organizações (como pode ser visto na seção 3.3.1). Este apoio, no entanto, é hoje adequado à utilização em organizações independentes.

O próximo capítulo descreve os requisitos e objetivos de uma nova família de ambientes, os Ambientes de Engenharia de Software Orientados a Corporação, cujo objetivo é apoiar iniciativas de melhoria de processos em corporações e abranger todos os processos relacionados a software e não apenas os relacionados ao desenvolvimento e manutenção dos produtos. Uma infra-estrutura baseada nestes requisitos e objetivos será construída com base na Estação Taba e seus ambientes atuais, assunto dos capítulos IV e V.

CAPÍTULO 4 - AMBIENTES DE ENGENHARIA DE SOFTWARE ORIENTADOS A CORPORAÇÃO

Neste capítulo se descrevem as necessidades de uma corporação para a execução e melhoria de seus processos e os requisitos necessários para que seja provido um apoio ferramental que atenda a estas necessidades. A resposta a estes requisitos é a definição e construção dos Ambientes de Engenharia de Software Orientados a Corporação.

4.1 Introdução

Com o passar do tempo as pesquisas em ambientes de desenvolvimento de software passaram a ter o foco na orientação a processos de software com o surgimento dos ambientes de desenvolvimento de software centrados em processos (FUGGETTA, 2000). Estes ambientes têm sua infra-estrutura condicionada à existência de um processo para guiar o desenvolvimento de software. As ferramentas que compõem os ambientes, por sua vez, apóiam a execução de processos de software e, também, geralmente utilizam um processo como um guia e orientador para auxiliar e facilitar a execução das tarefas necessárias.

O desenvolvimento de software também tem ganhado complexidade ao longo do tempo, seja pelas características das equipes, pelos desafios das tecnologias a serem utilizadas ou pelo conjunto de métodos e técnicas que devem ser integrados e utilizados para o desenvolvimento efetivo. Outro fator relevante é o número de processos que devem ser executados, para a conclusão de um projeto de software. Organizações desenvolvedoras de software têm necessidade de executar (i) diversos processos integrados ao processo de desenvolvimento para a execução de um projeto de software e, também, (ii) processos que são executados de forma independente dos processos de desenvolvimento em si, mas que também estão relacionados à produção de software ou na gerência e estruturação do desenvolvimento. Exemplos do primeiro caso são os processos de apoio cuja execução é realizada por papéis externos à estrutura do projeto (como o de Garantia de Qualidade e Medição) ou, então, processos cujas atividades estejam integradas ao fluxo de atividades do projeto, porém são executadas por equipes externas a ele (como no caso de existir uma equipe de testes independente). Exemplos do segundo caso são processos como o de

Aquisição, Definição de Processos, Gerência de Projetos de Melhoria e Gerência de Portfólio de Projetos.

Outra característica comum no cenário de desenvolvimento de software é a existência de corporações com organizações subordinadas que desenvolvem software de forma independente entre si. Estas organizações muitas vezes possuem diferentes necessidades e realidades em relação ao desenvolvimento de software, por exemplo, ao possuírem nichos de mercado com características diferentes ou possuírem diferentes níveis de maturidade no desenvolvimento de software. Por exemplo, corporações cujas organizações apenas realizam manutenção de sistemas embarcados e outras que desenvolvem sistemas de informação para automatização de atividades de apoio e outras que desenvolvem software para terceiros; corporações em que uma das organizações têm o desenvolvimento de software aderente a um nível de maturidade do MPS.BR (SOFTEX, 2007) ou do CMMI (CHRISISS *et al.*, 2006) e as demais desenvolvem software de forma *ad hoc*; ou ainda, corporações cujas organizações mesmo estando aderentes a algum modelo de maturidade tenham seus processos institucionalizados em níveis distintos de maturidade e capacidade.

A definição e construção dos Ambientes de Desenvolvimento de Software Orientados a Organização (VILLELA, 2004) supriu a necessidade de se apoiar a utilização do conhecimento organizacional, e não só o conhecimento de domínio provido por especialistas (conforme identificado por OLIVEIRA (1999)), durante o desenvolvimento de software e, também, a necessidade de se apoiar o desenvolvimento de software em uma organização em particular. Entretanto, esta abordagem tem limitações ao:

- (i) Apoiar apenas a definição e execução de processos de desenvolvimento ou manutenção de software enquanto vários outros processos precisam ser definidos e executados no contexto geral de produção de software em uma organização;
- (ii) Considerar apenas a existência de organizações independentes enquanto há corporações que precisam gerenciar os processos de software nas suas organizações subordinadas.

Os Ambientes de Engenharia de Software Orientados a Corporação, propostos nesta tese, visam a suprir estas limitações e podem ser considerados uma evolução dos Ambientes de Desenvolvimento de Software Orientados a Organização em dois sentidos:

- Evolução de Ambientes de *Desenvolvimento* de Software para Ambientes de *Engenharia* de Software, o que significa estender a definição dos ambientes

para outros processos além do de desenvolvimento e de manutenção, apoiando a execução e a gerência dos processos de Engenharia de Software (por exemplo, fornecimento, aquisição, treinamento, avaliação e melhoria etc.); e

- Evolução de Ambientes Orientados a *Organização* para Ambientes Orientados a *Corporação*, o que significa permitir a configuração de ambientes para corporações e, a partir desses, para as organizações que as compõem.

Dessa forma, **Ambientes de Engenharia de Software Orientados a Corporação** (AESCorp) devem prover o apoio computacional que possibilite a uma corporação, em relação aos processos de software, gerenciar a diversidade e os estágios de maturidade de cada uma das organizações que a compõem de forma adequada às suas necessidades. Além disso, esse apoio computacional deve permitir às corporações e organizações serem capazes de gerenciar e controlar os diversos processos de software de que dispõem e/ou necessitem, bem como o conhecimento organizacional envolvido.

4.2 Evoluindo de Ambientes de Desenvolvimento de Software para Ambientes de Engenharia de Software

A principal questão relacionada à evolução dos Ambientes de *Desenvolvimento* de Software para os Ambientes de *Engenharia* de Software está na existência de uma grande diversidade de processos existentes no contexto de organizações que produzem software. Cada um destes diferentes tipos de processos de software deve poder ser definido, executado e melhorado. A execução e gerência de tais processos devem ser apoiadas através da utilização de ambientes centrados em processo específicos a cada um destes processos. Um Ambiente de Desenvolvimento de Software é, dessa forma, apenas um dentre os possíveis Ambientes de Engenharia de Software.

A evolução dos Ambientes de Desenvolvimento de Software para os Ambientes de Engenharia de Software deve levar em consideração aspectos relacionados à definição, execução e melhoria de processos, gerência de ativos de processos e gerência de conhecimento.

4.2.1 Definição de Processos

A definição de um processo de software, ou de um conjunto de processos inter-relacionados, geralmente é feita com base no contexto em que este processo está inserido e, também, no conhecimento já existente sobre processos em geral. Exemplos de

conhecimento especializado sobre processos de software incluem modelos de maturidade e capacidade e normas internacionais relacionadas a processos de software, boas práticas e lições aprendidas, conhecimento sobre engenharia de software, conhecimento de especialistas em processos ou no contexto em que o processo será utilizado etc. O contexto em que o processo está inserido inclui o contexto de execução dos processos, aspectos culturais que podem influenciar a utilização dos processos, experiências anteriores relacionadas à execução (*ad hoc* ou não) dos processos, versões anteriores dos processos, necessidades e objetivos definidos para os processos.

O fato de um processo geralmente ser apresentado como um fluxo contínuo não significa, entretanto, que ele seja autocontido e não possua interseções com outros processos. Na verdade, a estrutura de um processo pode ser bastante dinâmica e os resultados esperados da execução de uma atividade (ou grupo de atividades) podem depender da execução de outros processos em contextos diferentes ou complementares. Um possível exemplo de processo autocontido é um processo para um projeto de desenvolvimento de software onde todas as atividades a serem desenvolvidas sejam de responsabilidade da equipe interna do projeto. Se houvesse neste processo atividades referentes a garantia da qualidade de processo e produto pode-se imaginar a possibilidade de estas atividades comporem, de certa forma, um processo específico para a equipe de garantia da qualidade da organização. Assim, o processo de garantia da qualidade seria formado por atividades presentes em um macro-processo específico para a área de garantia da qualidade e também nas atividades de garantia da qualidade presentes nos processos dos diversos projetos de desenvolvimento de software na organização. O processo de garantia da qualidade teria, então, uma interface de comunicação e consumo e geração de produtos intermediários com vários outros processos.

Dessa forma, para Ambientes de Engenharia de Software é importante que outros processos além do desenvolvimento e manutenção possam ser definidos e, caso os processos se relacionem entre si, deve ser possível a identificação desse relacionamento ou dependência entre suas atividades. Para a definição de outros processos que não desenvolvimento e manutenção, uma importante base pode ser a utilização dos processos e áreas de processos descritos em normas internacionais e modelos de maturidade como a ISO/IEC 12207 (ISO/IEC, 2008b), ISO/IEC 15504 (ISO/IEC, 2003), MPS.BR (SOFTEX, 2007) e CMMI (CHRISISSIS *et al.*, 2006).

4.2.2 Execução de Processos

Uma vez que um processo seja definido é importante prover apoio para sua execução. Esta execução é tipicamente feita através de um ambiente específico centrado em processo e composto por um conjunto de ferramentas adequado ao propósito do ambiente e ao apoio necessário para a execução do processo no qual o ambiente é baseado. Estes ambientes devem possuir, também, uma base de dados onde informações sobre a execução do processo e das ferramentas são armazenadas. Além disso, podem acessar elementos externos como outras ferramentas, ambientes e bases de dados para dar apoio à execução do processo.

Os Ambientes de Engenharia de Software devem poder ser utilizados para apoiar a execução de quaisquer dos processos de software que tenham sido definidos no contexto de uma organização e não apenas os de desenvolvimento e manutenção.

4.2.3 Melhoria de Processos

Uma das principais fontes para a melhoria de processos de software é a análise do comportamento dos processos ao longo de uma ou várias execuções. Esta análise pode levar em consideração, por exemplo, informações coletadas durante a execução do processo (por exemplo, atividades que não foram executadas corretamente), necessidades do negócio (por exemplo, evoluir a definição do processo para atingir um determinado nível de um modelo de maturidade) e conhecimento adquirido externamente (por exemplo, lições aprendidas e boas práticas). Após a melhoria ter sido identificada, uma nova versão melhorada do processo é definida e disponibilizada para uso e a versão anterior é descontinuada.

Com a definição de processos de software relacionados entre si é provável que uma possível melhoria de processos deva ser analisada não de forma isolada em relação a um único processo, mas, sim, em relação a um conjunto de processos de software cujas execuções estão relacionadas. Este padrão de melhoria (analisar vários processos e não apenas um) também comum em melhorias verticais, que envolvem a definição de novos processos de software, para, por exemplo, tornar possível a uma organização a evolução para outro nível de um modelo de maturidade.

Ambientes de Engenharia de Software devem possuir mecanismos que permitam a identificação de melhorias e disponibilização de novas versões dos diferentes processos de software em uso. Além disso, visto que pode ser necessário implantar melhorias em processos ainda em execução, deve ser possível a alteração de processos em execução sem

a perda de informações e dados uma vez que uma necessidade de melhoria tenha sido identificada.

4.2.4 Gerência dos Ativos de Processos

A definição, a execução e a melhoria de processos podem se beneficiar da existência e gerência adequada de ativos de processo, ou seja, artefatos que sejam considerados úteis para atender as necessidades de negócio da organização. Processos de software são, obviamente, parte do conjunto de ativos de processos de uma organização. Outros exemplos incluem artefatos, roteiros de documentação, itens de conhecimento (como diretrizes, lições aprendidas, melhores práticas etc.), ferramentas etc.

Uma base responsável pela gerência de tais artefatos, geralmente denominada biblioteca de ativos de processos, deve ser capaz de armazenar os documentos considerados relevantes, garantir que eles sejam disponibilizados aos interessados e garantir que suas evoluções sejam controladas através de métodos apropriados de gerência de configuração e de garantia da qualidade.

Em Ambientes de Engenharia de Software esta biblioteca deve conter mecanismos para apoiar todos os processos em uso e estar integrada às funcionalidades existentes para a definição, execução e melhoria de processos de software. Estes ambientes devem, também, possuir mecanismos que garantam a gerência de configuração dos ativos de processo, principalmente os relacionados à definição e melhoria dos processos de software para garantir um controle efetivo da evolução de tais ativos.

4.2.5 Gerência de Conhecimento

A gerência de conhecimento é um importante mecanismo para apoiar a execução de processos de software e pode, também, dar subsídios à melhoria de processos. Através da gerência de conhecimento pode-se, por exemplo, disponibilizar conhecimento disponível sobre processo de software e abordagens de desenvolvimento para os engenheiros de software responsáveis pela execução dos processos, divulgar as regras para a gerência dos ativos de processo ou, ainda, identificar pontos de melhoria na definição de determinadas atividades contidas nos processos.

A institucionalização da gerência de conhecimento pode garantir uma maior competitividade da organização e melhor capacitação de seus colaboradores. Através da institucionalização de uma rede de especialistas e de um mecanismo de apoio à troca de informações pode-se garantir que o conhecimento seja prontamente disponibilizado e compartilhado na organização de forma a aumentar a eficiência na realização das tarefas.

Em Ambientes de Engenharia de Software o ferramental disponível deve apoiar a execução das atividades presentes nos diversos processos de software existentes. Para isso é necessário prover um mecanismo que permita a aquisição de itens de conhecimento ao longo das atividades do processo sendo executado, empacotá-los, disponibilizá-los e mantê-los. Além disso, o acesso ao conhecimento acumulado ao longo do tempo deve ser disponibilizado através de mecanismos e ferramentas específicos, de acordo com a atividade do processo sendo executada. Durante as ações realizadas para a melhoria de processos tais itens de conhecimento devem estar disponíveis para que se investiguem possíveis pontos de melhoria.

4.2.6 Requisitos de um Ambiente de Engenharia de Software

Após analisar os cenários descritos acima envolvidos com a definição, execução e melhoria de processos de software, gerência de ativos de processos e gerência de conhecimento, é possível identificar alguns requisitos que Ambientes de Engenharia de Software devem atender:

- Apoiar a definição de quaisquer processos relacionados a software pertinentes a uma organização;
- Apoiar os engenheiros de software na execução de suas atividades relacionadas aos diferentes processos de software presentes numa organização;
- Apoiar a melhoria dos diferentes processos de engenharia de software utilizados por uma organização;
- Prover mecanismos que permitam alterar os processos em execução a partir de uma melhoria nestes processos sem perda de informações e dados;
- Apoiar a gerência dos ativos de processo no contexto das atividades relacionadas a software;
- Apoiar a gerência de conhecimento de forma a facilitar a execução das diferentes atividades dos processos de software de uma organização.

4.3 Evoluindo de Ambientes Orientados a Organização para Ambientes Orientados a Corporação

A evolução dos Ambientes Orientados a *Organização* para os Ambientes Orientados a *Corporação* está, como o próprio nome diz, na criação de um ambiente adaptado às necessidades de uma Corporação na gerência de suas atividades relacionadas a software e,

também, à gerência de suas organizações que desenvolvem software. O aspecto mais importante dessa gerência, no entanto, é a diversidade de estágios de maturidade em desenvolvimento de software em que estas organizações podem estar. O controle das atividades desempenhadas pelas organizações deve permitir à corporação identificar e prover mecanismos necessários para que elas convirjam para possuírem maturidade equivalente na execução de suas atividades relacionadas a software. Para que isto ocorra, devem ser observados aspectos relacionados à definição, execução e melhoria de processos, gerência de ativos de processos, gerência de conhecimento e à infra-estrutura de apoio à execução destas atividades.

4.3.1 Definição de Processos

Deve ser de interesse da Corporação a definição de regras e requisitos mínimos para a definição dos processos a serem utilizados pelas organizações que a compõem. Devido isso, pode haver algum grau de ingerência em relação a um conjunto mínimo de procedimentos ou atividades a serem executadas em cada organização sob determinado contexto.

Como a diversidade de processos nas organizações pode ser grande, a corporação pode definir processos, com o nível de detalhe que achar adequado, que possam ser incorporados nos diversos processos definidos nas organizações. Por exemplo, poderia ser definido um processo básico de Garantia da Qualidade com o conjunto mínimo de atividades a compor tal processo e o momento, ao longo do processo de desenvolvimento, em que deveriam ser executadas. Este processo também poderia ter diferenças de acordo com o tamanho do projeto ou o nível de maturidade da organização em questão.

Por outro lado a corporação pode ser a responsável por uma definição mais completa dos processos a serem executados em cada organização, limitando a estas apenas adaptações simples para deixá-los mais aderentes à sua cultura organizacional específica. Por exemplo, com o intuito de institucionalizar a adoção de um determinado nível de modelo de maturidade, a corporação pode obrigar a todas as suas organizações a utilizarem um mesmo processo padrão contendo as atividades necessárias e permitir apenas a troca do nome dos papéis responsáveis por executar cada atividade ou as ferramentas de apoio a serem utilizadas.

Uma Organização que esteja subordinada a uma Corporação deve respeitar os limites estabelecidos para a definição de seu conjunto de processos-padrão. Além disso, deve garantir que todos os procedimentos e atividades estabelecidos tenham sido adequadamente incorporados a estes processos. Regras para a adaptação dos processos-

padrão para uso nos projetos devem ser desenvolvidas pela organização em concordância às regras e limitações impostas pela corporação.

Em Ambientes Orientados a Corporação deve ser possível a definição de processos padrão pela Corporação e sua posterior adaptação por parte das Organizações que as compõem para definir seus próprios processos padrão ou quando for necessário executar tais processos. O escopo e abrangência das adaptações que podem ser realizadas devem ser definidos pela Corporação de acordo com seus objetivos e necessidades.

4.3.2 Execução de Processos

A execução de processos em corporações não acontece apenas no contexto da produção efetiva (por exemplo, desenvolvimento ou manutenção) de software. Processos podem ser definidos para um grande escopo de atividades relacionadas a software, por exemplo, atendendo a necessidades relativas à existência de programas de melhoria de processos, programa de medição ou um escritório de projetos. Estes processos, geralmente gerenciais, também devem poder ser executados e, para tal, necessitam de apoio ferramental adequado.

Dessa forma, os Ambientes Orientados a Corporação também devem ser capazes de utilizarem Ambientes de Engenharia de Software de forma a possibilitar a execução de processos de software seja no contexto Corporativo, seja no contexto Organizacional.

4.3.3 Melhoria de Processos

Uma Corporação, devido à maior diversidade de processos e à necessidade de lidar com organizações diferentes, com diferentes versões de processos e com diferentes níveis de maturidade, precisa ter maior controle sobre o ciclo de evolução de seus ativos de processos e, principalmente, de seus processos-padrão. Uma corporação deve ser capaz de identificar melhorias nos processos com base nas informações coletadas a partir da utilização dos processos nas diversas organizações. Além disso, a corporação deve ser capaz de disponibilizar novas versões de processos (e ativos de processo) às suas organizações, influenciando, dessa forma, iniciativas de melhoria de processos das organizações em andamento ou forçando, na prática, o início de uma melhoria de processos nas organizações. Apesar de uma Organização subordinada a uma Corporação poder desenvolver suas próprias estratégias para melhoria de processos de software (como uma organização independente o faria) deve sempre respeitar as regras definidas pela corporação.

Durante a análise das oportunidades de melhorias pela corporação é importante a coleta de dados e informações gerados a partir da execução dos processos nas organizações, assim como do *feedback* fornecido por aqueles que executaram os processos ou membros das equipes responsáveis por analisar e avaliar as oportunidades de melhoria nas organizações ou do acesso a dados presentes em base de dados externas que sejam consideradas úteis. Dessa forma, é necessário que a corporação possa ter acesso de forma eficiente a estes tipos de dados para auxiliar na melhoria de seus processos. A utilização de um vocabulário comum relacionado a processos de software na corporação também auxiliaria a diminuir o viés causado por falhas na comunicação e entendimento dos conceitos relacionados à execução dos processos e, dessa forma, facilitaria a identificação de melhorias no contexto corporativo.

Os Ambientes Orientados a Corporação devem, portanto, possibilitar que a Corporação identifique melhorias nos processos de software a partir de informações relacionadas ao uso dos processos por suas organizações e, também, possibilitar que a Corporação defina melhorias que devem obrigatoriamente ser implantadas nas organizações. Além disso, devem possuir mecanismos de acesso e integração de dados que permitam à corporação consultar os repositórios de dados das organizações e dos projetos para possibilitar a análise das informações disponibilizadas em cada um destes contextos visando à análise da institucionalização e da melhoria dos processos em uso.

4.3.4 Gerência dos Ativos de Processos

Do mesmo modo que a diversidade de processos no contexto corporativo é maior do que em uma organização isolada, a quantidade de ativos de processo relacionados a estes processos também é maior e também devem ser gerenciados e controlados. Apesar de os ativos de processo nas organizações que compõem a corporação poderem variar enormemente devido às características e particularidades destas organizações (por exemplo, organizações com níveis de maturidade diferentes ou atuando em nichos de mercado diferentes), o alinhamento às diretrizes corporativas deve ser respeitado. Dessa forma, assim como acontece na definição de processos pelas organizações, a gerência dos ativos de processo nas organizações também pode sofrer influência das regras e necessidades da corporação.

Assim, uma Corporação deve garantir que a criação e a evolução dos ativos de processo por parte de suas organizações sejam realizadas seguindo as diretrizes gerais definidas para tal. Pode, também, prover ativos de processo às organizações sem permitir que sejam alterados. De qualquer forma, níveis de mudanças permitidas devem ser

estabelecidos de forma a garantir que evoluções de um ativo não o tornem incompatível com os objetivos e necessidades corporativos. Uma Organização subordinada a uma Corporação deve respeitar as diretrizes definidas pela corporação para a criação e evolução dos seus ativos de processo.

Os Ambientes Orientados a Corporação devem, portanto, possuir uma biblioteca de ativos de processos que garanta à corporação a gerência dos seus ativos de processos e, além disso, permitir à corporação um maior controle sobre a gerência e evolução do conteúdo das bibliotecas de ativos de processos de suas organizações subordinadas. O uso desta biblioteca deve permitir, também, às corporações analisar os dados obtidos pela organização na utilização de tais ativos de forma a contribuir para a melhoria dos ativos corporativos e, possivelmente, a partir disso, uma melhoria dos ativos em todas as organizações.

4.3.5 Gerência de Conhecimento

Uma Corporação deve possuir mecanismos eficientes para que as organizações que a compõem compartilhem conhecimento entre si e contribuam para a base de conhecimento corporativa. Mecanismos eficientes para a disseminação de conhecimento devem ser instituídos para potencializar o uso e abrangência da base de conhecimento corporativo. Assim, além de construir sua própria base de conhecimento, tipicamente com conteúdo adquirido durante a execução de seus processos, uma Organização subordinada a uma Corporação pode se beneficiar do conhecimento corporativo consolidado na base corporativa ou presente apenas em bases de outras organizações.

Dessa forma, em Ambientes Orientados a Corporação é importante garantir que o conhecimento acumulado pelas organizações durante a execução de seus processos possa ser analisado pela corporação e incorporado à sua base de conhecimento. Também deve ser possível para a corporação disponibilizar novos itens de conhecimento a organizações que possam vir a se beneficiar deles e utilizar estes itens de conhecimento durante a identificação de oportunidades de melhorias nos processos.

4.3.6 Disponibilização de Infra-estrutura

A infra-estrutura provida pelos Ambientes de Engenharia de Software possibilita apenas a execução de processos de software específicos. Entretanto, a gerência das atividades relacionadas a software nas corporações envolve mais do que apenas executar processos específicos e, portanto, requer outro tipo de infra-estrutura de apoio. Os Ambientes Orientados a Organização, conforme definidos por VILLELA (2004), por sua

vez, apóiam a gerência das atividades de organizações que desenvolvem software. Além disso, possibilitam a geração de ambientes de apoio à execução de processos de desenvolvimento e manutenção. Estes ambientes são adequados a organizações independentes (ou seja, não subordinadas a uma corporação). Entretanto, não são adequados a corporações ou a suas organizações. Dentre suas limitações está a não existência de mecanismos que permitam a uma organização evoluir de forma eficiente seus processos sem a configuração de um novo ambiente organizacional com novas versões dos processos. Além disso, por não levarem em consideração a existência da organização num contexto corporativo, não consideram a existência de organizações com diferentes níveis de maturidade no desenvolvimento de software e a necessidade de se prover mecanismos que permitam a evolução destas organizações visando ao ganho de maturidade, possivelmente, para um nível único definido pela corporação.

Os Ambientes Orientados a Corporação devem, dessa forma, prover uma infraestrutura que permita contemplar todos os itens identificados anteriormente e também a monitoração das organizações para assegurar que as diretrizes corporativas relativas a definição, execução e melhoria dos processos estejam sendo seguidas. Além disso, esta infra-estrutura deve estar adequada à gerência da diversidade de processos de software existente no âmbito da corporação e também à gerência dos diferentes níveis de maturidade das organizações que a compõem. Os Ambientes Orientados a Corporação também devem ser capazes de prover o apoio ferramental necessário para a gerência das atividades das organizações e para a execução dos diferentes processos existentes no contexto corporativo e organizacional.

4.3.7 Requisitos de um Ambiente Orientado a Corporação

Após analisar os cenários descritos acima envolvidos com a definição, execução e melhoria de processos de software, gerência de ativos de processos, gerência de conhecimento e infra-estrutura é possível identificar alguns requisitos que Ambientes Orientados a Corporação devem atender:

- Possibilitar a definição de um ambiente específico para a Corporação de forma a apoiar a gerência das atividades relacionadas a Engenharia de Software;
- Possuir mecanismos que apóiem uma Corporação na gerência e controle das atividades relativas a software das organizações que a compõem;

- Possibilitar a definição de ambientes específicos para as Organizações que compõem a Corporação de forma a apoiar a gerência das atividades relacionadas a Engenharia de Software;
- Apoiar a corporação na gerência da diversidade de processos e de níveis de maturidade possuídos pelas organizações provendo mecanismos que permitam a evolução destas organizações visando ao ganho de maturidade, possivelmente, para um nível único definido pela corporação;
- Apoiar a gerência de conhecimento nas corporações, incluindo o aprendizado corporativo em Engenharia de Software a partir do aprendizado adquirido pelas organizações nos projetos de software;
- Prover uma infra-estrutura que permita a gerência de configuração dos ativos de processo da corporação visando à adoção de abordagens de melhoria de processos;
- Prover mecanismos que permitam a uma organização evoluir de forma eficiente seus processos disponibilizando versões com as melhorias identificadas, sem exigir a geração de novos ambientes ou acarretar a perda de informações ou dados;
- Apoiar a avaliação e melhoria dos diferentes processos de engenharia de software na Corporação como um todo;
- Possibilitar a utilização de um vocabulário comum relativo à definição e execução de processos no contexto corporativo de forma a facilitar a institucionalização dos processos;
- Possuir mecanismo de integração de dados e acesso a repositórios de informações sobre a corporação, suas organizações, seus colaboradores e os processos executados e em execução de forma a fornecer informações úteis para a melhoria dos processos, monitoração da execução e institucionalização dos processos nas organizações e na corporação.

4.4 Modelo para Construção de Ambientes de Engenharia de Software Orientados a Corporação

Com base nos cenários de evolução dos Ambientes de Desenvolvimento de Software para Ambientes de Engenharia de Software e de Ambientes Orientados a Organização para Ambientes Orientados a Corporação descritos nas seções anteriores deste capítulo e fiel aos objetivos de estender a definição dos ambientes para outros

processos além do de desenvolvimento e de manutenção, apoiando a execução e a gerência dos demais processos de Engenharia de Software e permitir a configuração de ambientes para corporações e, a partir desses, para as organizações que as compõem, os objetivos dos Ambientes de Engenharia de Software Orientados a Corporação podem ser sumarizados como:

- (i) Apoiar os engenheiros de software na execução de suas atividades;
- (ii) Apoiar a definição, execução e melhoria dos diferentes processos de engenharia de software das organizações segundo suas características específicas e as diretrizes definidas pela corporação;
- (iii) Apoiar a corporação na gerência da diversidade de processos e de níveis de maturidade possuídos pelas organizações;
- (iv) Apoiar a gerência de conhecimento nas corporações, incluindo o aprendizado corporativo em Engenharia de Software a partir do aprendizado adquirido pelas organizações nos projetos de software;
- (v) Prover uma infra-estrutura que permita a gerência de configuração dos ativos de processo da corporação visando à adoção de abordagens de melhorias de processos.

De forma a atender a estes objetivos e também aos requisitos identificados nas seções 4.2.6 e 4.3.7, foi definido um modelo para construção de Ambientes de Engenharia de Software Orientados a Corporação. Este modelo foi dividido em definição e geração de ambientes, definição de processos, melhoria de processos, execução de processos, gerência de ativos de processo e gerência de conhecimento.

4.4.1 Definição e Geração de Ambientes

Para compor toda a infra-estrutura relacionada aos Ambientes de Engenharia de Software Orientados a Corporação (AESCorp) é preciso a definição de quatro tipos diferentes de ambientes: Meta-Ambiente, Ambiente Corporativo, Ambiente Organizacional e Ambiente do Projeto. Esta hierarquia e o esquema de geração de ambientes dentro deste contexto pode ser vista na Figura 2.1.

A hierarquia dos ambientes começa com o Meta-Ambiente, capaz de gerar os Ambientes Corporativos, isto é, ambientes adequados às necessidades de diferentes corporações. Estes Ambientes Corporativos são gerados a partir do processo-padrão corporativo. De posse de seu Ambiente Corporativo, uma Corporação pode configurar Ambientes Organizacionais para cada uma de suas organizações ou unidades

organizacionais. Os Ambientes Organizacionais são instâncias dos Ambientes Corporativos e são configurados customizando-se o processo-padrão corporativo para a organização/unidade organizacional. Da mesma forma, cada organização/unidade organizacional pode instanciar Ambientes de Projeto através da adaptação de um dos processos-padrão da organização/unidade organizacional para a execução em um projeto específico. De posse do seu Ambiente, cada equipe de projeto pode, então, executar o processo definido para o projeto presente neste ambiente.

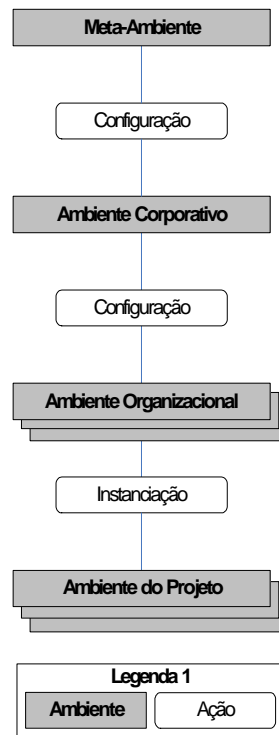


Figura 4.3 – Geração de Ambientes

4.4.1.1 Meta-Ambiente

O Meta-Ambiente, como o próprio nome diz, é um ambiente capaz de gerar outros ambientes. É utilizado em uma entidade externa à corporação ou às organizações, sendo utilizado por profissionais especializados em engenharia de software, que fazem a manutenção e evolução dos ativos de processos e demais recursos disponíveis neste ambiente. Sua principal função é apoiar a configuração de ambientes para corporações ou organizações específicas e, também, prover as funcionalidades e controlar as características e recursos disponíveis nos demais ambientes. Dessa forma, a principal funcionalidade deste ambiente é configurar os Ambientes Corporativos (no caso de configuração para Corporações) e os Ambientes Organizacionais (no caso de configuração para Organizações Independentes).

4.4.1.2 Ambiente Corporativo

Este ambiente é configurado no Meta-Ambiente a partir do processo-padrão corporativo. Sua função é auxiliar os engenheiros de software responsáveis na gerência das atividades relacionadas aos processos de software existentes na Corporação, como os grupos de processos, métricas e gerência do conhecimento corporativos. A partir deste ambiente pode ser feita a disponibilização de apoio ferramental e ativos de processo para as Organizações que a compõem a Corporação através dos Ambientes Organizacionais.

4.4.1.3 Ambiente Organizacional

Este ambiente é configurado a partir do Ambiente Corporativo e é utilizado por diferentes profissionais da organização que respondem por atividades fora do âmbito dos ambientes de projetos. Estes profissionais incluem, por exemplo, engenheiros de software e membros do grupo de processos responsáveis pela construção e disponibilização do apoio (computacional ou relativo a ativos de processo) necessário para controlar as atividades realizadas pela Organização relacionadas aos seus processos de software, gerentes de projetos que necessitam iniciar novos projetos, responsáveis pelo programa de medição organizacional, gerentes de conhecimento etc.

Para organizações que não estejam subordinadas a uma Corporação, estes ambientes são configurados no Meta-Ambiente.

4.4.1.4 Ambiente de Projeto

Ambientes de Projeto são ambientes centrados em processo e caracterizam-se pela existência de um processo a ser executado, independentemente do seu tipo ou finalidade, e são gerados com o objetivo de executar e controlar as atividades de um projeto específico. Estes ambientes são utilizados pelo gerente/líder do projeto e demais membros da equipe do projeto, dando apoio às várias atividades dos processos do ciclo de vida de um software através de ferramentas específicas, descrição de procedimentos e *templates*, dentre outros.

Apesar de o senso mais comum relacionado a um Ambiente de Projeto se referir a um Ambiente de Desenvolvimento de Software, a utilização de processos para desenvolvimento (ou manutenção) de software não é a única possibilidade para tais ambientes. Um novo ambiente pode ser gerado, por exemplo, para controlar as atividades relacionadas a Garantia da Qualidade em uma organização. Dessa forma, tendo controle sobre a execução do processo para gerenciar as atividades do Grupo de Qualidade da organização possivelmente com acesso aos insumos produzidos por outros projetos (por

exemplo, para desenvolvimento de um novo software ou para gerenciar as atividades do Grupo de Medição da organização).

A necessidade de execução de processos não é exclusiva dos Ambientes Organizacionais. Uma Corporação pode, por exemplo, necessitar de apoio para executar um processo para a gerência do seu programa de melhoria a partir do Ambiente Corporativo. Da mesma forma, a análise de melhorias nos ativos de processo utilizados no Meta-Ambiente durante a configuração dos Ambientes Corporativos também pode ser dependente da execução de um processo de melhoria. Assim, os Ambientes de Projeto também devem poder ser gerados a partir dos Ambientes Corporativos e do Meta-Ambiente. A definição de um Ambiente de Projeto está, portanto, subordinada às regras e processos-padrão definidos no ambiente do qual é originado.

4.4.1.5 Ambientes de Engenharia de Software Orientados a Corporação

As subseções anteriores apresentaram os diferentes tipos de ambientes dentro do contexto deste trabalho. Entretanto, algumas vezes será necessário mencionar coletivamente esta família de ambientes e o conjunto de toda a infra-estrutura que permite que ela seja definida e disponibilizada. Desta forma, para evitar confusões de nomes, sempre que houver referência a este conjunto de ambientes de forma integrada será utilizado o termo Ambientes de Engenharia de Software Orientados a Corporação (AESCorp); caso se deseje explicitar as características de apenas um tipo de ambiente será utilizada a nomenclatura específica.

Assim, podemos definir os Ambientes de Engenharia de Software Orientados a Corporação como o conjunto, ou família, de ambientes que possibilitam a gerência, execução e monitoração e melhoria de processos de uma corporação e das organizações que compõem esta corporação, além de possibilitar a evolução controlada dos ativos de processos envolvidos nestas iniciativas.

4.4.2 Definição de Processos

Um item chave na geração dos diferentes tipos de ambientes é a escolha dos processos a serem disponibilizados em cada um deles. De forma geral, cada ambiente contém um conjunto de processos que são adaptados para definir os processos do ambiente do próximo nível (conforme estrutura apresentada na Figura 2.1). Esta adaptação segue regras específicas próprias de cada nível de ambiente e das características de cada processo.

A abordagem para definição de processos apresentada, proposta inicialmente por OLIVEIRA (1999) e adaptada por VILLELA (2004), pode ser vista na Figura 4.4. Segundo a proposta do modelo, que estabelece etapas e produtos intermediários, a norma ISO/IEC 12207 (ISO/IEC, 2008b) é a base para a definição de qualquer processo-padrão. Para a definição do processo-padrão, são, ainda, consideradas as características do desenvolvimento de software na organização, que são relacionadas ao ambiente de trabalho, conhecimento e experiência das equipes envolvidas e à própria cultura e experiência da organização no desenvolvimento de software. Também podem ser considerados modelos de maturidade como o MR-MPS (SOFTTEX, 2007) e o CMMI (CHRISISS *et al.*, 2006).

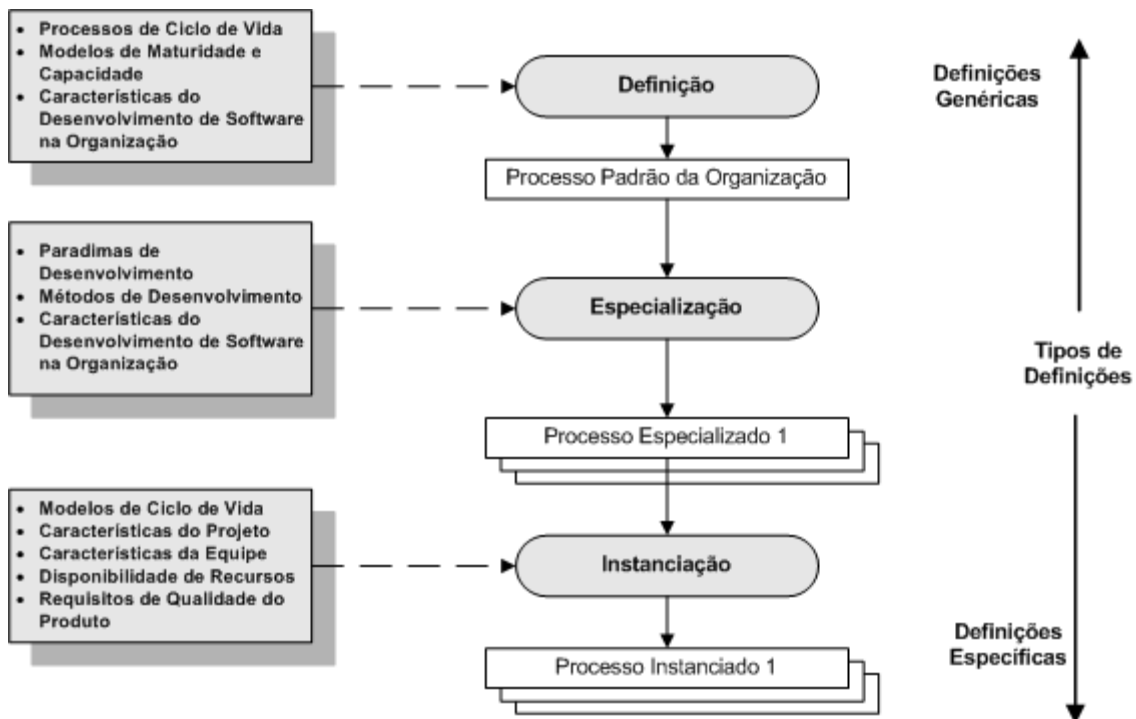


Figura 4.4 – Abordagem para Definição de Processos (VILLELA, 2004)

O objetivo da definição do processo padrão é padronizar as atividades a serem realizadas nos projetos da corporação e suas organizações. O objetivo da especialização é adaptar o processo padrão a diferentes situações e características dos tipos de produtos e projetos desenvolvidos. Por sua vez, o objetivo da instanciação é permitir adaptar um dos processos especializados às características de um projeto específico. Este esquema pode ser utilizado para a definição de cada um dos diferentes processos de software a serem utilizados nos AESCorp, por exemplo, o processo de desenvolvimento que engloba todas as atividades necessárias para a construção de um software ou processos necessários à gerência de um programa melhoria de processos.

Antes do início da definição do processo-padrão é preciso identificar as características desejadas para ele, por exemplo, que atividades, ou grupo de atividades, são ou não obrigatórias no contexto corporativo ou organizacional, atividades necessárias para aderência a um modelo de maturidade, atividades comuns a um determinado tipo de organização (por exemplo, organizações que fazem pesquisa ou agem como fábricas de software). Também devem ser levantados os itens de conhecimento de especialistas, externos ou internos à Organização/Corporação, que podem influenciar na definição dos processos (por exemplo, lições aprendidas e melhores práticas aplicáveis ao contexto corporativo). De posse destas informações, é possível definir um processo padrão mais genérico para a corporação a ser utilizado por todas as organizações, que pode ou não conter os elementos básicos definidos anteriormente. A definição do processo padrão deve levar em consideração também os diferentes processos de ciclo de vida existentes, normas e modelos de maturidade de software e características da corporação/organização. Esta etapa é realizada, inicialmente, no Meta-Ambiente e deve poder ser modificada no Ambiente Corporativo, caso necessário.

A definição dos processos padrões para cada uma das organizações que compõem a corporação deve levar em consideração os diferentes ativos de processos existentes na organização, normas e modelos de maturidade e capacidade, características da organização. Estes processos padrões devem respeitar as características obrigatórias definidas pela Corporação e podem conter outros elementos não contemplados anteriormente e que sejam considerados adequados ao novo contexto. Esta etapa da definição dos processos é realizada, inicialmente, também no Meta-Ambiente e pode ser modificada no Ambiente Corporativo ou no Ambiente Organizacional, caso necessário.

Cada processo padrão pode dar origem a vários processos especializados de acordo com os diferentes paradigmas e métodos de desenvolvimento utilizados comumente pela organização, critérios definidos pela Organização para caracterização de projetos (por exemplo, tamanho de projeto, tipo de cliente, ramo de negócio relacionado ao projeto) e características do desenvolvimento de software na organização, como os tipos de software desenvolvidos na organização (por exemplo: sistemas especialistas, sistemas de informação e sistemas de controle de processos) e dos paradigmas de desenvolvimento adotados (por exemplo: orientado a objetos e estruturado). Neste momento, novas atividades podem ser definidas e incluídas nos processos especializados e a descrição de atividades já definidas no processo-padrão pode ser adequada. Estes processos especializados, no entanto, devem respeitar as características obrigatórias definidas pela Corporação e pela Organização e

podem, também, conter outros elementos não contemplados anteriormente e que sejam considerados adequados ao novo contexto. É importante observar que um processo especializado também se comporta como um processo padrão e pode ser utilizado como tal (por exemplo, como base para uma nova especialização), pois possui apenas um novo refinamento para situações específicas.

Para ser utilizado em um projeto, o processo especializado mais adequado para um determinado tipo de software deve ser instanciado para atender às características do projeto específico e dar origem ao processo definido para o projeto. Cada processo especializado pode dar origem a vários processos instanciados de acordo com os diferentes modelos de ciclo de vida existentes e em uso pelas organizações, características específicas do projeto e da equipe, disponibilidade de recursos e requisitos de qualidade do produto ou do processo. Estes processos instanciados são os processos executados de fato e podem ser gerados tanto nos Ambientes Corporativos (geralmente associados com processos gerenciais no contexto corporativo, como o processo de gerência do programa de melhoria de processos da corporação) quanto nos Ambientes Organizacionais (neste caso, tanto para processos gerenciais, como a execução de um programa de medição, quanto para os relacionados à construção de software, como os processos de desenvolvimento ou manutenção).

A definição dos processos instanciados deve, portanto, ser realizada antes do início de um projeto e deve respeitar as características obrigatórias definidas pela Corporação e pela Organização, além de poder conter adaptações pertinentes ao contexto dos projetos. Devem ser considerados neste momento, portanto, o tamanho e a complexidade do produto bem como as características de qualidade desejadas, a expectativa de vida útil, as características da equipe de desenvolvimento e demais características do projeto. Neste momento, também devem ser selecionados o modelo de ciclo de vida, os métodos e ferramentas, conforme proposto por BERGER (2003), além de atividades relacionadas ao conhecimento de domínio específico a ser utilizado nos projetos, conforme proposto por OLIVEIRA (1999).

4.4.2.1 Definição de Processos para a Corporação

O modelo de definição de processos para Corporações é uma adaptação do modelo proposto por VILLELA (2004) para organizações. Os três níveis de processo da abordagem (processo-padrão, processo especializado e processo instanciado) são pertinentes, porém devem ser feitas adaptações nos momentos em que estes processos são definidos em adequação à estrutura de ambientes apresentada na Figura 4.5. No lado

esquerdo da figura é possível ver os ambientes relacionados à etapa em questão da definição de processos e a ação responsável pela geração do próximo ambiente a partir do anterior. Esta ação geralmente também é responsável pelo desencadeamento da criação do próximo nível de processos. No lado direito da figura é exibido o esquema, de fato, de definição de processos⁴.

A definição de processos para uma Corporação acontece no Meta-Ambiente durante a configuração do Ambiente Corporativo. Neste momento, é definido o conjunto de processos padrão da Corporação e para cada um destes processos são definidas as especializações pertinentes de acordo com o contexto específico e conforme previsto na abordagem para definição de processos descrita anteriormente.

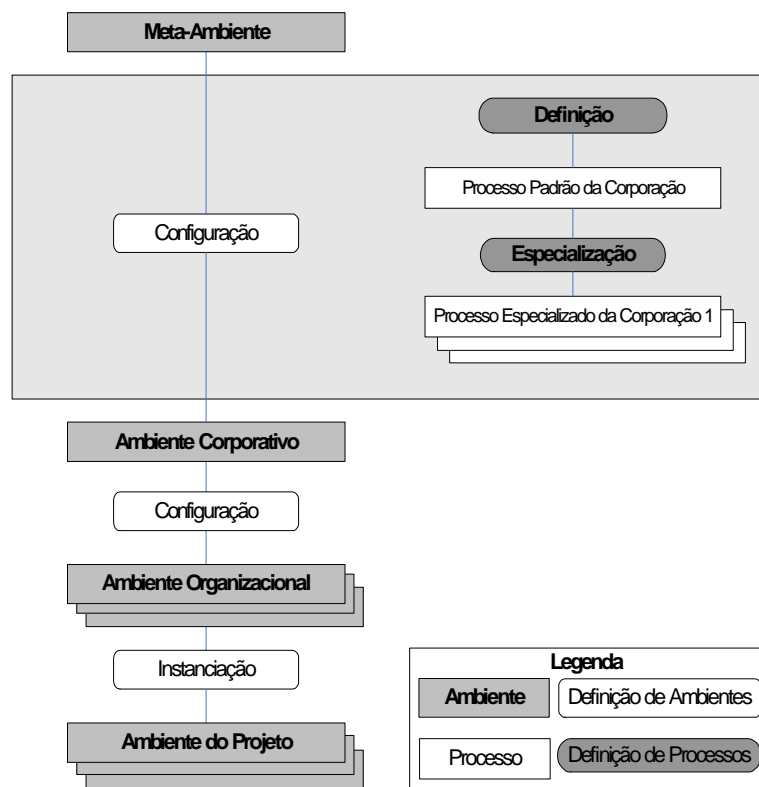


Figura 4.5 – Definição de Processos para a Corporação

De posse de seu Ambiente Corporativo, a Corporação pode fazer adaptações ao conjunto de processos tanto para fazer novas especializações quanto definir novos processos-padrão de acordo com a necessidade seguindo, também, a abordagem para definição de processos descrita anteriormente.

⁴ Este mesmo esquema é seguido na Figura 4.6 e na Figura 4.7.

4.4.2.2 Definição de Processos para a Organização

Para a disponibilização de um Ambiente Organizacional a uma de suas organizações é necessário que a Corporação escolha um conjunto de processos-padrão para compor a base de ativos deste ambiente dentre os processos disponíveis no Ambiente Corporativo, conforme pode ser visto na Figura 4.6.

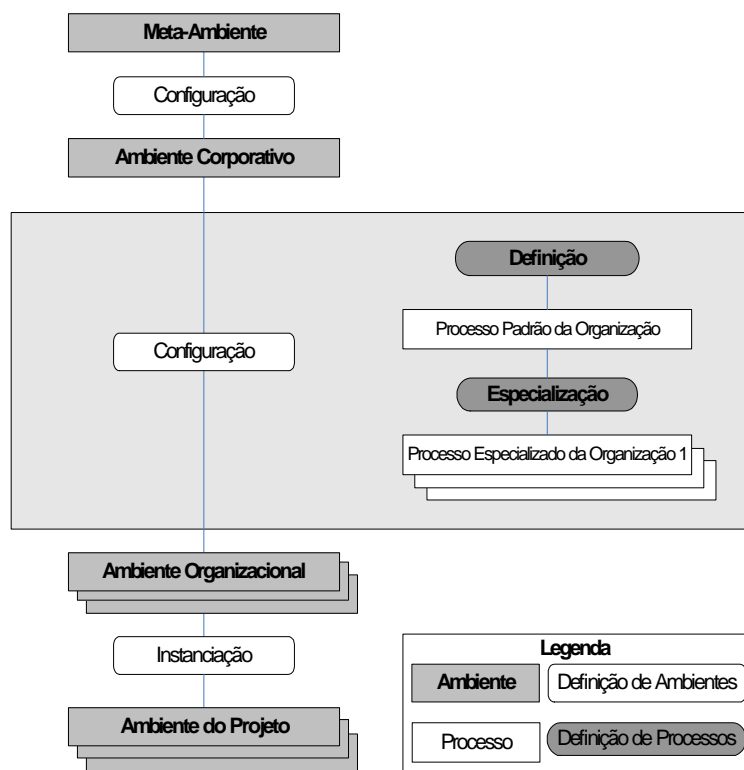


Figura 4.6 – Definição de Processos para a Organização

Uma organização de posse do seu conjunto de processos-padrão também pode evoluir os processos de acordo com as suas necessidades e as regras estabelecidas pela corporação. Dessa forma, no Ambiente Organizacional a organização pode definir novas especializações de seus processos adequando-os a sua necessidade. Caso seja permitido pela Corporação, uma Organização também pode definir seus próprios processos-padrão seguindo a abordagem de definição de processos.

4.4.2.3 Definição de Processos para os Projetos

O último estágio da definição de processos ocorre durante a definição e início de um novo projeto. Neste momento é preciso definir um processo definido para o projeto com base em um dos processos especializados disponíveis, como pode ser visto pela Figura 4.7. Adaptações também podem ser feitas, conforme previsto na abordagem, desde que sejam compatíveis com as diretrizes corporativas e organizacionais.

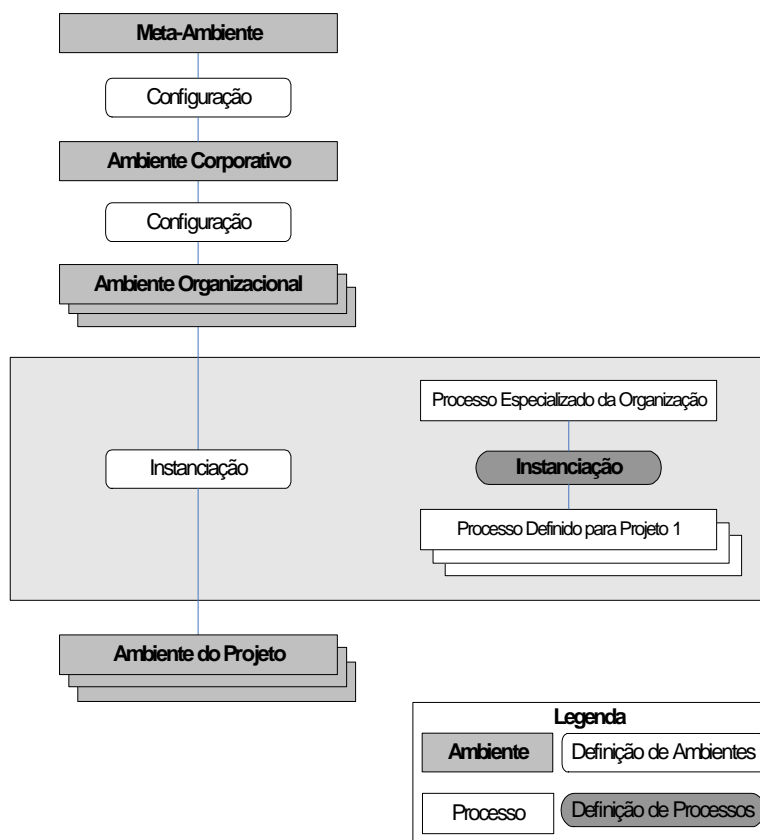


Figura 4.7 – Definição de Processos para Projetos

4.4.3 Melhoria de Processos

A etapa de definição não é o final do ciclo de vida de um processo. Após a sua definição o processo deve ser executado e, a partir dos dados coletados sobre essa execução, deve ser melhorado. A estratégia para melhoria de processos deve estar adequada aos objetivos organizacionais e corporativos e ao cenário de melhoria de processos adequado à estrutura de definição de processos em uso. Dessa forma, os AESCorp não devem apoiar apenas a definição e execução dos processos, mas, também, a contínua melhoria destes processos. Dois cenários de melhoria são investigados:

- (i) A Corporação identifica uma melhoria num dos processos em uso pelas Organizações e disponibiliza para uso uma nova versão de processos padrão corporativos que deve, então, ser utilizados nos novos projetos pelas Organizações a partir deste momento; e
- (ii) A Organização decide aplicar uma melhoria em processos ainda em execução em projetos para resolver questões pertinentes mais rapidamente ou reduzir o número de versões diferentes dos processos em execução.

Apesar de os processos serem parte do conjunto de ativos de processo e de a melhoria de processo, dessa forma, poder ser vista como parte da gerência de ativos de processo, estes dois itens são tratados de forma independente neste trabalho. Note-se, ainda, que, apesar de o esquema presente nas figuras que descrevem a definição de processos prever a existência de processos-padrão, especializados e definidos para o projeto, serão representados nas figuras a seguir apenas processos-padrão e definidos para o projeto. Isto foi feito por questões de simplificação, dado que processos especializados também podem ser considerados como processos-padrão.

4.4.3.1 Melhoria de Processos Determinada pela Corporação

Este cenário de melhoria de processos numa corporação é baseado na estrutura proposta para definição de processos nos três diferentes níveis de ambientes onde isto é possível, ou seja, Meta-Ambiente, Ambiente Corporativo e Ambiente Organizacional. As figuras que descrevem tal cenário utilizam a legenda presente na Figura 4.8. Nesta legenda podem-se ver os quatro elementos principais: (i) Ambiente relacionado ao cenário em questão; (ii) processo investigado no cenário; (iii) ação executada nos ambientes para geração de outros ambientes ou melhoria nos processos; e (iv) processo cujo uso foi descontinuado em vista de uma evolução identificada e disponibilização de uma nova versão de processo.

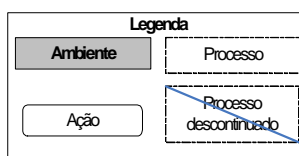


Figura 4.8 – Legenda para o cenário de melhoria

O momento inicial deste cenário é a evolução, de forma independente, dos processos por parte das organizações a partir dos processos disponibilizados pela Corporação. Neste cenário, portanto, as organizações possuem autonomia, de acordo com as regras definidas pela corporação, para evoluir de forma independente seus processos.

No exemplo da Figura 4.9, os dados da execução do processo padrão *A1* em dois projetos são analisados através da abordagem de avaliação e melhoria de processos adota pela organização e dá origem ao processo *A1'*. Dessa forma, a partir da estratégia (e das fontes) adotada para a melhoria de processos uma nova versão do processo originalmente disponibilizado pela corporação é definida e disponibilizada para uso nos novos projetos da organização, conforme exibido na (note que para simplificar a figura, é mostrado o cenário para uma das organizações e assumiu-se que o durante a configuração dos Ambientes

Organizacionais não houve modificações nos processos padrão adotados). Novos projetos são, então, iniciados (conforme pode ser visto na Figura 4.10) utilizando os processos evoluídos e, assim, novas oportunidades de melhoria podem ser novamente identificadas.

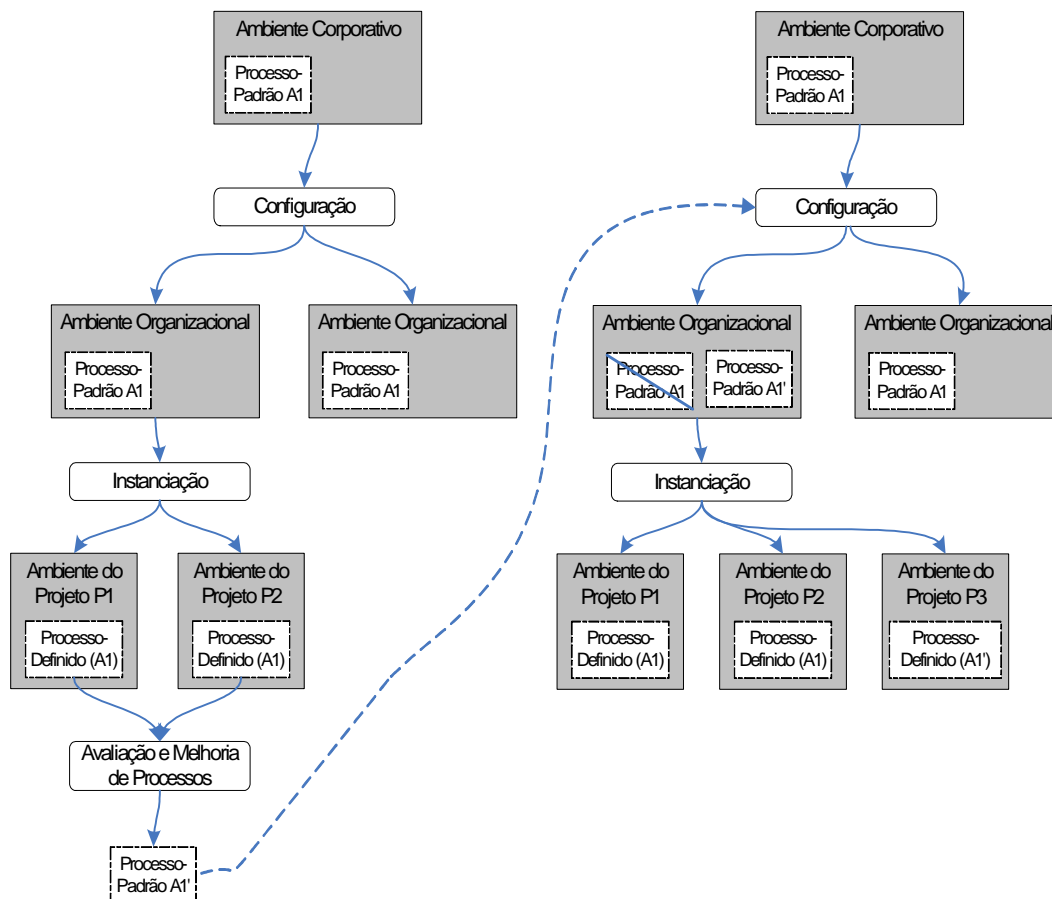


Figura 4.9 – Uso dos processos pelos projetos e melhoria de processos na Organização

Figura 4.10 – Uso do processo evoluído em novos projetos

O processo originalmente definido e disponibilizado pela corporação é melhorado, de forma semelhante aos processos em uso pelas organizações. Informações disponibilizadas pelas iniciativas de melhoria das organizações podem ser levadas em consideração para esta evolução. Dessa forma, a corporação evolui os processos utilizando como uma das fontes as evoluções feitas de forma independente pelas organizações.

A partir da evolução dos processos-padrão pela corporação, os Ambientes Organizacionais devem, então, ser atualizados, conforme mostram a Figura 4.11 e a Figura 4.12. Dessa forma, os novos projetos a serem iniciados nas organizações devem utilizar apenas a nova versão do processo.

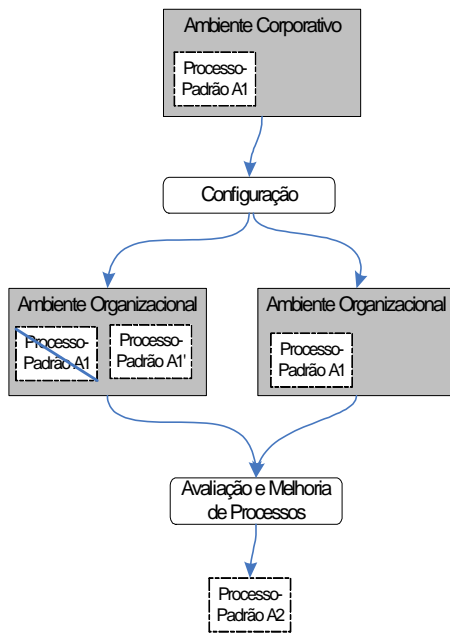


Figura 4.11 – Avaliação e melhoria dos processos pela Corporação

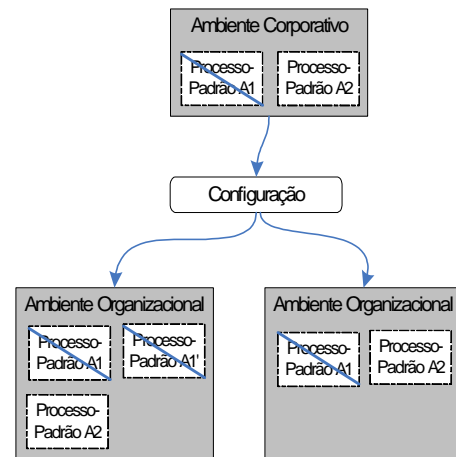


Figura 4.12 – Evolução dos processos organizacionais a partir de decisão corporativa

Este ciclo pode se repetir inúmeras vezes com a utilização dos processos pelos projetos e melhoria dos processos conforme descrito anteriormente até que um processo seja, enfim, descontinuado.

4.4.3.2 Melhoria de Processos em Projetos em Execução

Uma possível variação do cenário anterior está no momento de disponibilização das alterações nos processos. Naquele cenário a melhoria no processo organizacional só é disponibilizada para novos projetos, entretanto, para projetos de longa duração ou então quando uma melhoria é considerada crítica pode ser necessário implantá-la imediatamente em projetos já em andamento. Por exemplo, se um projeto ainda está na fase de levantamento de requisitos e uma melhoria no processo é identificada para a etapa de testes e homologação, provavelmente, a implantação desta melhoria não traria impactos nas atividades em andamento e poderia ser implantada quando o projeto fosse executar as atividades de testes e homologação. Ou, ainda, a alteração nos processos pode ter tido origem numa necessidade do projeto e não ser melhoria de fato, mas apenas uma adequação específica a uma nova necessidade do projeto. Dessa forma, é necessário prever um mecanismo para a disponibilização de melhorias em projetos em andamento.

As figuras que descrevem este cenário utilizam a legenda presente na Figura 4.13.

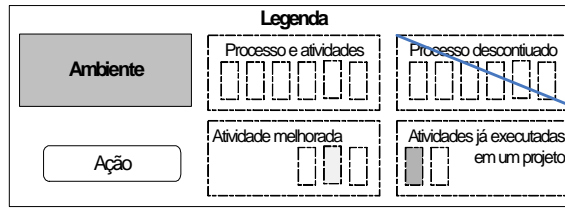


Figura 4.13 – Legenda para o cenário de melhoria para processos em execução

Neste cenário, considere uma organização com um projeto sendo executado com a versão inicial do processo-padrão enquanto uma melhoria neste processo-padrão é identificada e disponibilizada para novos projetos. Considerando que o processo-padrão é composto por diferentes atividades (ou quaisquer outros elementos de processos independentes como subprocessos ou fases) e que apenas uma destas atividades tenha, de fato, sofrido modificações entre uma versão e outra do processo e que o projeto em execução ainda não começou a executar a atividade que sofreu modificações, este projeto ainda pode se beneficiar da melhoria identificada desde que ela seja implantada a tempo. Para isso, a organização deve decidir, então, que seria mais adequado implantar a modificação do processo-padrão também no projeto em execução. Dessa forma, o processo para o projeto deveria ser alterado "no vôo", ou seja, durante sua execução, conforme pode ser visto na Figura 4.14 e na Figura 4.15.

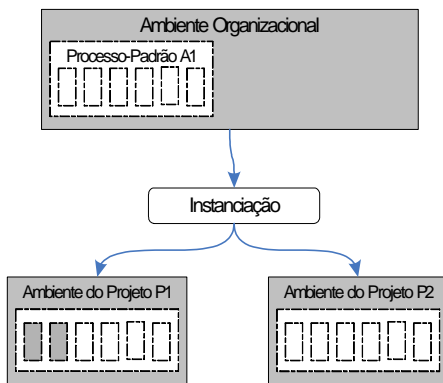


Figura 4.14 – Processos em execução na organização

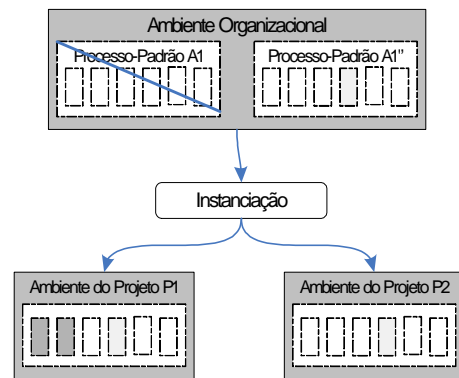


Figura 4.15 – Implantação de melhoria em projetos em execução

Com esta abordagem, melhorias podem ser implantadas mais rapidamente aumentando seus efeitos positivos e, também, reduzindo a quantidade de variações de processo existentes no contexto organizacional. Dessa forma também se pode reduzir o esforço do grupo responsável pela gerência dos processos organizacionais em manter concomitantemente muitas versões de processos em execução.

4.4.4 Execução de Processos

A execução dos processos, como explicada na seção 4.4.1, é realizada apenas em Ambientes de Projetos, pois são os únicos de fato centrados em processo. Os Ambientes de Projeto devem possibilitar o acesso a um conjunto de ferramentas, internas e/ou externas, adequado ao propósito do ambiente e ao apoio necessário para a execução do processo para o qual o ambiente foi gerado. Os Ambientes de Projeto devem possuir, também, uma base de dados onde informações sobre a execução do processo e de suas ferramentas são armazenadas. Além disso, devem poder acessar elementos externos como outras ferramentas, ambientes e bases de dados para dar apoio à execução das atividades previstas.

Um processo tipicamente possui três etapas que envolvem o planejamento, que possibilita a sua execução, a execução do processo de fato e uma etapa de finalização ou transição.

A fase de planejamento inclui uma identificação prévia das necessidades relativas a execução do processo (por exemplo, o escopo do projeto, uma lista inicial e de alto nível de requisitos para o desenvolvimento de software, a documentação gerada anteriormente que pode ser consultada para a conclusão do projeto, o conjunto de ferramentas e ambientes de apoio necessários, requisitos de equipe etc.). Também durante essa fase preliminar deve-se identificar qual o processo (ou processos) mais adequado para a execução do projeto dentre as opções disponíveis. Feita essa identificação é necessário adaptá-lo para o contexto da execução em questão segundo critérios estabelecidos. A partir desse levantamento inicial é possível, então, iniciar o planejamento de fato da execução do processo.

Uma vez planejado, o processo pode ser executado. Ao longo do tempo é necessário efetuar monitorações periódicas para garantir uma aderência ao planejamento atual e, se for o caso, fazer as adaptações necessárias ao planejamento original.

Quando o produto final do processo é concluído, é preciso finalizar a sua execução (por exemplo, realizando avaliações *post mortem* para garantir que lições aprendidas sejam registradas e disponibilizadas para futuro uso, atualizar a biblioteca de ativos organizacionais com produtos gerados a partir da execução do processo) e, caso necessário, efetuar a transição para um novo projeto (por exemplo, garantindo que a documentação esteja disponível para possíveis evoluções do projeto).

Dessa forma, o conjunto de ferramentas e Ambientes de Projeto disponibilizados dentro do contexto dos AESCorp para apoiar a execução de processos de software, de forma genérica, contemplando as fases de planejamento, apoio à execução e finalização do

processo. Além disso, deve fornecer mecanismos para que ferramentas adequadas à execução de tarefas relacionadas a um processo específico (por exemplo, geração de informações sobre rastreabilidade entre requisitos no caso de um processo de desenvolvimento de software ou de gerência de requisitos) sejam definidas, criadas ou acessadas externamente.

4.4.5 Gerência de Ativos de Processo

Uma questão não discutida na seção 0 referente a melhoria de processos de software foi como garantir um controle adequado sobre as diferentes versões dos processos e como prover algum mecanismo eficiente para garantir a rastreabilidade das mudanças ocorridas nestes processos. A identificação das mudanças ocorridas nos processos e dos motivos que levaram a estas mudanças pode ser útil, por exemplo, para que a corporação mantenha o controle sobre a evolução dos processos e também para identificar novas oportunidades de melhoria nos processos corporativos.

Como foi dito anteriormente nas seções 4.2.4 e 4.3.4, os AESCorp devem possuir uma biblioteca de ativos de processo responsável pela gerência e controle destes ativos. Parte deste controle está relacionado a mecanismos associados à gerência de configuração. Esta gerência de configuração segue o mesmo padrão qualquer que seja o tipo de ativo de processo sendo gerenciado (por exemplo, roteiros, processos, item de conhecimento etc.), porém, pode haver particularidades específicas para determinado tipo e adaptações devem ser realizadas conforme pertinente.

Além da ligação com melhoria de processos de software, a gerência de ativos de processos está diretamente relacionada a outros elementos deste trabalho, por exemplo, à definição de processos (sejam eles padrão, especializados ou definidos para o projeto), definição de ambientes corporativos, organizacionais e de projetos e gerência de conhecimento. Um AESCorp deve, dessa forma, prover mecanismos para o controle de versão, controle de modificação e rastreabilidade de versões e mudanças, além de possuir bases independentes de ativos de processo e de versões de ativos de processo.

4.4.5.1 Controle de versão

As diferentes versões dos ativos de processo geradas por uma corporação devem ser armazenadas e gerenciadas de forma a garantir um nível de controle adequado para cada tipo de ativo de processo. É preciso, também, manter um histórico das modificações realizadas em determinado ativo de processo para garantir uma rastreabilidade das alterações realizadas. Também é preciso garantir que versões concorrentes de diferentes

ativos existam e sejam utilizadas por diferentes organizações e/ou projetos ao mesmo tempo. Por exemplo, o processo padrão de uma organização pode ter sido alterado devido a uma melhoria, mas os projetos já em andamento devem continuar a utilizar a versão anterior para que não haja um impacto que possa ser prejudicial ao andamento dos projetos. Entretanto, novos projetos a serem iniciados devem ser criados já com base no novo processo padrão. Outro exemplo de versões concorrentes é a existência de variantes de um procedimento para realização de uma tarefa de acordo com a maturidade da organização que vai utilizá-lo ou com características e desejos da organização.

O controle de versões é crítico num ambiente corporativo, pois diferentes versões de um mesmo ativo de processo podem coexistir nos ambientes corporativo, organizacional e de projeto. Outra realidade que pode ser constante neste cenário é a possibilidade de que as evoluções nos ativos de processo possam ser efetuadas de forma independente e em paralelo em várias organizações e, num determinado momento, a corporação deseje ter acesso a estas versões para analisar oportunidades de melhoria dentro do contexto corporativo e, então, disponibilizar novas versões dos ativos às organizações.

Uma questão importante quando se fala do controle de versão dos ativos de processo, assim como de quaisquer outros tipos de itens de configuração, é a granularidade do item a ser gerenciado. Artefatos e roteiros de artefatos geralmente são controlados como um único item de configuração. Por outro lado, processos de software podem ter sua granularidade aumentada ao identificar como itens de configuração não o processo como um todo, mas cada atividade. Ao diminuir o tamanho do grão a itens elementares ganha-se precisão para determinar o ponto específico em que a mudança ocorreu, porém, aumenta-se a complexidade do versionamento e nem sempre a relação custo-benefício é adequada. Deve-se, portanto, analisar o custo-benefício da escolha dos itens de configuração a serem tratados para manter o controle adequado da evolução dos itens sem aumentar desnecessariamente a burocracia relacionada à sua gerência.

4.4.5.2 Controle de modificação

Controlar as versões, no entanto, não é suficiente para garantir o controle das alterações realizadas. A corporação e suas organizações devem ser capazes de registrar todas as suas necessidades de alteração/evolução dos ativos de processo existentes. Por exemplo, uma organização pode solicitar ao grupo de qualidade da corporação a alteração do conteúdo de um procedimento interno que regule a construção de um determinado artefato para adequá-lo a novos padrões internacionais que surjam ou que deveriam ser adotados. Todas as informações geradas por esta solicitação (por exemplo, justificativas do

pedido, pareceres do grupo de qualidade do processo, relatório de modificações realizadas) devem ser registradas de alguma forma e tornadas disponíveis a todos os possíveis interessados.

Por outro lado, pode ser desejo da corporação limitar as alterações num ativo de processo organizacional ou então saber das mudanças com antecedência para autorizá-las ou não. Nestes casos, o controle de versão é mantido, porém deve ser implementado um controle de alteração nos processos de forma que um comitê específico autorize as alterações e evoluções de um determinado ativo de processo. Através da ação deste comitê todas as alterações têm que ser formalmente solicitadas, aprovadas, implementadas, verificadas e aprovadas para que, só então, sejam disponibilizadas aos interessados. Isto pode ser útil caso se deseje assegurar mais facilmente que versões de diferentes ativos de processo não fiquem inconsistentes entre si, comprometendo possíveis melhorias de processos em andamento ou garantir que as alterações realizadas estejam de acordo com as diretrizes corporativas.

Da mesma forma que a adoção de um controle formal de modificação aumenta o controle sobre o ativo de processo, também aumenta a burocracia relacionada a suas evoluções. Dessa forma, é necessário estabelecer regras que identifiquem que tipos de ativos de processos devem estar sob um controle formal de alteração e aqueles cujo controle necessário pode ser somente através de controle de versões.

4.4.5.3 Rastreabilidade de versões e mudanças

Através da adoção de controle de versão ou de controle de mudanças nos ativos de processo a corporação e suas organizações podem manter registros das evoluções ocorridas nos ativos de processo. Esta rastreabilidade pode ser útil para determinar fontes de erros ou falhas em definições assim como permitir que abordagens para melhoria de processos (e ativos de processos) ocorram em ambientes controlados. Também pode ser útil durante análises dos processos visando a melhoria dos ativos de processo corporativos ao facilitar a análise das modificações implementadas nos ativos por parte das organizações.

A definição de linhas base deve ser gerenciada de forma a garantir que a liberação de um conjunto de ativos de processo para uso de uma organização seja feita de forma controlada e que os itens que componham essa linha base estejam consistentes entre si. Por exemplo, os ativos de processo utilizados por uma organização com certo nível de maturidade devem ser consistentes e adequados àquele nível.

4.4.5.4 Bases de Ativos de Processo e Base de Versões de Ativos de Processo

Para garantir um menor grau de intrusividade da solução adotada para o controle de configuração dos ativos de processo, deve haver uma diferença entre os ativos disponibilizados para uso (que compõem a base de ativos de processos) e as diferentes versões pelas quais estes ativos já tiveram (que compõem a base de versões de ativos de processos). Dessa forma, a base de ativos de processo é, na prática, um subconjunto da base de versões de ativos.

Da mesma forma que as bases de ativos de processo da corporação e das organizações devem ser independentes, as bases de versões de ativos de processo também devem manter sua independência:

- O Meta-ambiente deve ter uma base de versões de ativos de processo que pode ser utilizada durante a definição do Ambiente Corporativo;

- O Ambiente Corporativo deve ter uma base de versões de ativos de processo que pode ser utilizada durante a definição do Ambiente Organizacional;

- O Ambiente Organizacional deve ter uma base de versões de ativos de processo que pode ser utilizada durante a definição do Ambiente de Projetos.

Os itens da base de versão de ativos de processo do Meta-ambiente devem ser disponibilizados para a corporação durante a configuração do Ambiente Corporativo. Da mesma forma, os itens da base corporativa devem ser disponibilizados às organizações durante a configuração do Ambiente Organizacional, que utilizará tais ativos para a definição dos ambientes de projetos.

Algum mecanismo deve ser disponibilizado para realimentar as bases originais com possíveis evoluções dos itens utilizados ao longo do tempo. Dessa forma, deve haver uma análise da evolução dos ativos de processos nas organizações para determinar sua reincorporação à base de ativos corporativa. Há, portanto, três possibilidades de atualização de um item na base corporativa após disponibilização e alteração na base organizacional. Após uma análise da alteração realizada (e de acordo com o cenário de melhoria), pode-se decidir refletir a mudança na base corporativa, aceitar apenas parte das alterações realizadas ou, então, ignorar todas as alterações realizadas.

4.4.6 Gerência de Conhecimento

Conhecimentos adquiridos nas organizações devem ser armazenados no repositório de conhecimento organizacional e devem ficar associados a informações de contexto de

forma a avaliar a sua adequação à realidade dos projetos, das organizações e da corporação ao longo do tempo, além de permitir que possam ser reavaliados e reclassificados. Estas informações de contexto incluem, mas não se resumem a, atividades onde a coleta foi realizada ou a que se refere e conceitos e instâncias de teorias de domínio que possam servir como classificadores.

Uma vez que a base de conhecimento, pelo menos em sua versão inicial, tiver sido construída, deve-se garantir que os itens de conhecimento sejam prontamente compartilhados e disponibilizados aos membros da corporação e das organizações e dos executantes dos processos. Esta disponibilização de conhecimento será mais eficiente se acontecer de forma pró-ativa inserida no contexto de uso e aplicabilidade do conhecimento. Deve-se, também, no entanto, garantir um acesso à base de conhecimento por meio de buscas e pesquisas específicas. Possuir mecanismos que permitam incorporar conhecimentos e experiências registrados nos demais ambientes como, por exemplo, evoluções em lições aprendidas e novas melhores práticas.

Outra importante fonte de conhecimento dentro da estrutura corporativa são os especialistas em áreas específicas, como, por exemplo, o próprio domínio de processos de software. Dessa forma, deve-se garantir que haja a identificação de tais especialistas e que eles possam ser encontrados e contatados visando a uma maior disseminação do conhecimento. Uma possibilidade para a disseminação desse conhecimento detido por especialistas é a criação de um fórum específico em uma comunidade de prática de forma a disseminar mais facilmente as informações sobre a evolução os ativos de processo e, até mesmo, estimular a discussão sobre eles entre os membros da organização.

4.5 Considerações Finais

Este capítulo apresentou as principais necessidades que motivaram a definição dos Ambientes de Engenharia de Software Orientados a Corporações e também os seus principais requisitos e características de infra-estrutura, englobando apoio a definição, execução e melhoria de processos, definição de ambientes, gerência de ativos de processos e gerência de conhecimento. No próximo capítulo serão discutidas as alterações realizadas na Estação Taba para a construção de tais ambientes.

CAPÍTULO 5 - DEFINIÇÃO E CONSTRUÇÃO DE AMBIENTES DE ENGENHARIA DE SOFTWARE ORIENTADOS A CORPORAÇÃO NA ESTAÇÃO TABA

Este capítulo descreve as mudanças realizadas na Estação Taba para que a abordagem de construção dos Ambientes de Engenharia de Software Orientados a Corporação, detalhada através dos requisitos apresentados no capítulo anterior, fosse implementada.

5.1 Introdução

A criação de Ambientes de Engenharia de Software Orientados a Corporação (AESCorp) envolve a definição de um conjunto de ambientes, e ferramentas de apoio integradas a estes ambientes, adequado aos requisitos e funcionalidades propostos no Capítulo IV. O primeiro passo foi a definição da estratégia para a construção dos AESCorp. Depois foram revistas e evoluídas as funções e os requisitos em uso da Estação Taba e realizada uma revisão completa da arquitetura e dos modelos de construção dos ambientes existentes com o objetivo de adequá-la à definição e à construção dos AESCorp considerando os aspectos relacionados anteriormente. Após esta revisão foi iniciada a construção de uma nova infra-estrutura para a Estação Taba utilizando a tecnologia Web, mais adequada aos AESCorp e ao crescente uso dos ambientes pela indústria. Posteriormente, novas ferramentas e funcionalidades previstas para possibilitar a criação dos AESCorp foram definidas e construídas.

Este capítulo apresenta, então, a revisão realizada na Estação Taba para definição e construção dos Ambientes de Engenharia de Software Orientados a Corporação. Além disso, apresenta um cenário de uso dos AESCorp no contexto de uma corporação, a Aeronáutica, como uma prova de conceito da infra-estrutura desenvolvida, evidenciando, dessa forma, o apoio fornecido a uma iniciativa de melhoria de processos de software.

5.2 Funções e Ambientes da Estação Taba

Devido às características e requisitos apresentados no capítulo anterior referentes à definição dos Ambientes de Engenharia de Software Orientados a Corporação, as funções da Estação Taba, conforme atualização promovida por VILLELA (2004), sofreram modificações e se tornaram:

- a) Auxiliar o engenheiro de software na configuração do ambiente mais adequado para apoiar uma corporação específica na gerência da definição e evolução dos processos de software das organizações subordinadas, considerando o conhecimento corporativo relevante neste contexto;
- b) Auxiliar o engenheiro de software na configuração do ambiente mais adequado para apoiar as atividades de engenharia de software em uma organização específica, considerando seus processos de software e o conhecimento organizacional relevante neste contexto;
- c) Auxiliar o engenheiro de software na configuração do ambiente mais adequado para apoiar as atividades de engenharia de software em uma organização, vinculada ou não a uma corporação, considerando seus processos de software e o conhecimento organizacional relevante neste contexto;
- d) Auxiliar os gerentes de projeto na instanciação de um ambiente de engenharia de software centrado em processo para projetos específicos da organização;
- e) Apoiar, através de um dos ambientes de engenharia de software, a execução do processo definido para o ambiente assim como a gerência dessas atividades.
- f) Apoiar a avaliação e melhoria dos processos de software nos diferentes tipos de ambientes.

Duas famílias de ambientes distintas compõem os AESCorp:

- *Ambientes Configurados* – Adequados ao uso de Corporações e Organizações na gerência e controle das atividades relacionadas a software. Estes ambientes são denominados *Ambientes Corporativos* quando são configurados para apoiar Corporações e *Ambientes Organizacionais* quando são configurados para apoiar Organizações. Estes ambientes são responsáveis pela evolução dos *Ambientes Orientados a Organização* para os *Ambientes Orientados a Corporação*.
- *Ambientes de Projeto* – Adequados ao apoio da execução de quaisquer processos de software dentro do escopo de um projeto específico. Estes ambientes são responsáveis pela evolução dos *Ambientes de Desenvolvimento de Software* para os *Ambientes de Engenharia de Software*.

Devido a isso, a Estação Taba passou a ser composta por quatro diferentes tipos de ambientes, conforme pode ser visto na Figura 2.1.

O Meta-Ambiente da Estação Taba já possuía funcionalidades para a configuração de um ambiente adequado às particularidades de uma organização. A partir deste Ambiente Organizacional, o engenheiro de software podia especificar e instanciar o Ambiente de

Desenvolvimento ou Manutenção de Software, convencional ou orientado a um domínio, mais adequado a um projeto específico integrando ferramentas internas e/ou externas.

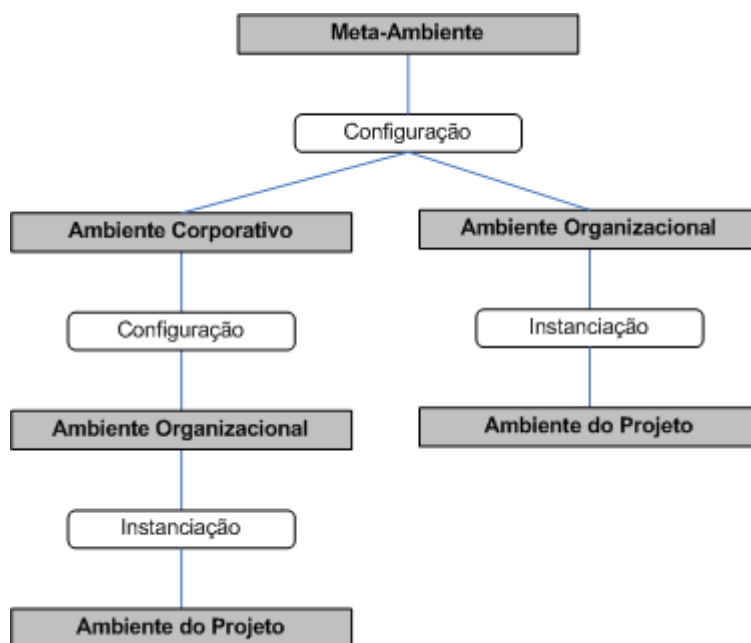


Figura 5.16 – Ambientes da Estação Taba

Com a implementação da estratégia para construção dos Ambientes de Engenharia de Software Orientados a Corporação, proposta neste trabalho, a configuração de ambientes passa a ser possível também para Corporações. Neste caso, a configuração dos Ambientes Organizacionais é feita também a partir dos Ambientes Corporativos. A partir de um Ambiente Configurado (que pode ser tanto o Ambiente Corporativo quanto o Ambiente Organizacional) é possível gerar através de instanciação Ambientes de Projeto para quaisquer dos processos de software que precise ser executado no contexto corporativo ou organizacional, e não apenas processos de desenvolvimento ou manutenção. Os Ambientes Corporativos, Organizacionais e de Projetos contemplam, juntos, os requisitos e componentes propostos para um AESCorp.

5.3 Requisitos dos Ambientes da Estação Taba

Além da revisão das funções e dos ambientes existentes, a lista de requisitos da Estação Taba, atualizada pela última vez por VILLELA (2004), foi evoluída e refinada para incluir as novas características necessárias a partir da definição dos AESCorp. Estes requisitos podem ser vistos na Tabela 2.3. Esta tabela também exhibe qual o escopo de aplicabilidade dos requisitos e a situação da implementação de cada um deles.

A coluna "Escopo" tem a seguinte legenda:

- **MA** – Requisito que afeta o *Meta-Ambiente*;
- **AC** – Requisito que afeta o *Ambiente Corporativo*;
- **AO** – Requisito que afeta o *Ambiente Organizacional*;
- **AP** – Requisito que afeta o *Ambiente de Projeto*;

A coluna "Situação" tem a seguinte legenda:

- **IU** – Requisito *implementado* e em *uso* atualmente na Estação Taba;
- **ID** – Requisito *implementado* em versões anteriores da Estação Taba, mas atualmente em *desuso* na versão industrial. Estes requisitos foram propostos e implementados ao longo do tempo de duração do Projeto da Estação Taba, porém a implementação deste requisito (através de uma ferramenta ou funcionalidade) atualmente não está disponível na versão em uso pelas empresas, ficando restrito a versões acadêmicas.
- **PI** – Requisito *parcialmente* implementado na versão atual da Estação Taba.
- **NI** – Requisito proposto, porém *não-implementado*, deve ser tratado em versões futuras.

Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes

#	Requisito	Descrição	Escopo	Situação
1	Possuir interface consistente	Possuir mecanismos de interface que permitam a utilização consistente de seus recursos e ferramentas.	MA AC AO AP	IU
2	Possuir um modelo comum de armazenamento de dados	A forma de representação das informações deve possibilitar que as ferramentas compartilhem e utilizem estas informações de forma natural e consistente.	MA AC AO AP	IU
3	Apoiar o controle de versões e a gerência de configuração	Controlar as modificações feitas nos componentes de conhecimento, itens de software e ativos de processo, mantendo-os disponíveis em suas diferentes versões e gerenciando onde essas versões estão sendo utilizadas.	MA AC AO AP	IU
4	Possuir conhecimento sobre processo de software e abordagens de desenvolvimento	Possuir conhecimento sobre processo de software, incluindo normas e modelos de processo, e as várias alternativas de modelos de ciclo de vida, paradigmas de desenvolvimento e métodos possíveis de serem utilizados, bem como sobre a adequabilidade da aplicação de cada uma dessas alternativas em diferentes contextos.	MA AC AO AP	IU
5	Possuir mecanismo de integração de ferramentas	Permitir e facilitar a integração de ferramentas internas (desenvolvidas como parte da Estação Taba).	MA	IU
		Permitir e facilitar a integração de ferramentas externas.	MA AC AO AP	IU

Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes

#	Requisito	Descrição	Escopo	Situação
6	Permitir a descrição de tarefas	Possuir mecanismos que facilitem a descrição de tarefas genéricas, que independem de um domínio de aplicação.	MA AC AO	ID
7	Apoiar a utilização de Teorias de Domínio	Possuir mecanismos que facilitem a definição de Teorias de Domínio para diferentes domínios de aplicação e para o domínio de Engenharia de Software, identificando as tarefas genéricas que são executadas nesses domínios. Além de possibilitar a evolução das Teorias de Domínio que fazem parte do ambiente permitindo que novos conceitos e relações, bem como novas instâncias de conceitos e relações, sejam incluídos nas Teorias dos Domínios de aplicação e de Engenharia de Software que fazem parte do ambiente.	MA AC AO	ID
		Oferecer mecanismos e ferramentas de acesso ao conhecimento sobre o domínio da aplicação.	AC AO AP	ID
		Permitir que novas instâncias de conceitos e relações sejam incluídas na Teoria do Domínio referente à aplicação.	AC AO AP	ID
8	Apoiar a definição de processos de software	Apoiar a definição de processos-padrão e especializados de acordo com as características e regras estabelecidas para esta definição.	MA AC AO	IU
		Apoiar a definição de um processo para um projeto de software específico a partir de um dos processos especializados, considerando, para isto, as características do projeto.	AC AO	IU
		Apoiar a evolução dos modelos de processo seguindo as orientações definidas.	AC AO	IU
9	Gerar Ambientes Configurados para o contexto desejado	Gerar um Ambiente Configurado para uma corporação, considerando os processos padrão e especializados definidos e as características da corporação e o contexto de uso dos processos.	MA AC	IU
		Gerar um Ambiente Configurado para uma organização, considerando os processos padrão e especializados definidos e as características da organização e o contexto de uso dos processos.	AC AO	IU
10	Apoiar a gerência de conhecimento nos ambientes	Apoiar a aquisição, filtragem e empacotamento de conhecimento adquirido ao longo da execução dos processos e seu armazenamento no repositório de conhecimento, o que inclui a associação do mesmo com as atividades dos processos às quais se refere e com os conceitos e instâncias de conceitos que o descrevem.	MA AC AO	IU
		Oferecer, de acordo com a atividade do processo sendo executada, mecanismos e ferramentas de acesso ao conhecimento acumulado ao longo do tempo além de possibilitar a reutilização de qualquer espécie de conhecimento em contexto diferente do contexto para o qual foi criado.	MA AC AO AP	IU
		Possuir mecanismos que permitam incorporar conhecimentos e experiências registrados nos demais ambientes, por exemplo, evoluções em lições aprendidas e novas melhores práticas, e que possibilitem a reutilização de qualquer espécie de conhecimento em contexto diferente do contexto para o qual foi criado.	MA AC AO	IU

Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes

#	Requisito	Descrição	Escopo	Situação
		Possuir mecanismos para impedir a perda de conhecimento acumulado em versões anteriores em caso de nova geração dos ambientes.	MA AC AO	IU
11	Permitir a descrição da estrutura organizacional	Permitir a descrição da estrutura da corporação e das organizações envolvidas, internas ou externas à corporação, e a definição das competências desejadas para cada posição definida pela estrutura. Além disso, permitir que estas informações sejam atualizadas e expandidas de acordo com as informações obtidas nos projetos correntes.	AC AO AP	IU
12	Permitir a descrição dos profissionais envolvidos	Permitir a descrição do perfil dos profissionais da corporação ou das organizações envolvidas e a alocação destes profissionais à estrutura corporativa ou organizacional. Além de apoiar a localização dos profissionais mais adequados para auxiliar na execução de uma atividade ou na solução de um problema.	AC AO AP	IU
13	Permitir a descrição dos processos organizacionais	Permitir a descrição textual e gráfica dos processos que não sejam de software, e dos elementos destes processos de forma a apoiar o entendimento dos processos e permitir a visualização e a navegação através dos diferentes níveis de abstração, fornecendo, sob solicitação, detalhes sobre os elementos representados e permitindo acesso às informações e conhecimentos disponíveis no ambiente. Além disso, permitir que estes os processos sejam atualizados e expandidos de acordo com as informações obtidas ao longo do tempo.	AC AO AP	ID
14	Gerar ambiente para projetos específicos	Gerar um ambiente de engenharia de software (AES) para um projeto de software específico a partir do processo definido para o projeto.	AC AO	IU
15	Apoiar a execução e a gerência de processos de software	Apoiar a execução do processo para o qual o ambiente foi instanciado, e de suas atividades, e a gerência do mesmo, através de orientação, automação e/ou monitoração do processo e, quando necessário, do apoio a sua modificação.	AP	IU
16	Possuir apoio para a avaliação do produto	Apoiar a medição dos produtos gerados ao longo do processo de forma a permitir a garantia da qualidade do produto final.	AP	PI
17	Possuir apoio para a avaliação do processo	Apoiar a medição do processo de utilizado para gerar o ambiente de forma a permitir a sua avaliação e a sugestão de melhorias.	AP	IU
18	Apoiar a definição de arquiteturas de referência	Ter acesso a uma base de conhecimento sobre arquiteturas de software que sirvam de referência no desenvolvimento de diferentes produtos de um mesmo tipo e/ou domínio de aplicação.	AP	NI
19	Apoiar o trabalho cooperativo	Definir protocolos de coordenação, colaboração e comunicação que facilitem o trabalho em equipe, o que é especialmente importante para projetos de desenvolvimento ou manutenção de software em larga escala.	AC AO AP	NI
20	Possuir apoio para a melhoria contínua do processo	Possuir mecanismos que possibilitem a avaliação dos processos em execução nos ambientes dos projetos e apoiem a melhoria contínua dando subsídios a um programa de melhoria de processos.	MA AC AO	IU
21	Apoiar a Gerência dos Ativos de Processo	Possibilitar a gerência (incluindo definição, evolução, execução, armazenamento etc.) dos ativos de processo produzido e/ou utilizado pela Estação Taba.	MA AC AO AP	PI

5.4 Ontologia de Corporação

No contexto da Estação Taba e de seus ambientes, é fundamental dispor de um vocabulário comum para a descrição de qualquer organização (ou corporação) e que possa ser utilizado para representar conhecimento útil para os desenvolvedores de software sobre as organizações envolvidas em um projeto de software (VILLELA *et al.*, 2004).

Ontologias⁵ são ferramentas importantes para a organização e contextualização de conhecimento, particularmente em contextos bem definidos, como pesquisa científica ou em organizações individuais (BREWSTER e O'HARA, 2007). Atualmente, são utilizados como base para a maior parte da infra-estrutura da Estação Taba os conceitos definidos na ontologia de Organização proposta por VILLELA (2004), que contém evoluções sobre a ontologia de Processo de Software definida por FALBO (1998)⁶. Esta ontologia descreve conceitos, propriedades, relações e restrições que contemplam os diferentes aspectos de um modelo organizacional.

A Estação Taba tem parte de sua infra-estrutura baseada no uso de ontologias, o que ajuda a mapear, disseminar e utilizar o conhecimento sobre diferentes domínios (SANTOS *et al.*, 2004; SANTOS *et al.*, 2005; VILLELA *et al.*, 2005). Este conhecimento é utilizado ao longo de sua estrutura para modelagem de dados e como guia para o uso e construção de ferramentas relacionadas a processos de software e à modelagem da estrutura de organizações. Esta ontologia descreve conceitos como Organização e Processo que, dentro do contexto desta tese, foram adaptados e evoluídos devido à identificação de novos conceitos ou à necessidade de detalhá-los melhor.

Dessa forma, (i) foi definido um novo conceito Corporação (relacionado ao conceito existente Organização); (ii) foi detalhado o conceito Processo na forma de Processos Padrão, Especializados e Definidos para o Projeto; e (iii) foi definido o conceito

⁵ Ontologia é uma representação de vocabulário, geralmente especializada para algum domínio ou assunto. Não é o vocabulário que se qualifica como uma ontologia, mas a conceitualização que seus termos no vocabulário tentam capturar. O termo ontologia algumas vezes é utilizado para se referir a um corpo de conhecimento, tipicamente um senso comum sobre um determinado domínio, utilizando um vocabulário como representação. Em outras palavras, a representação textual provê um conjunto de termos com os quais são descritos os fatos em algum domínio, enquanto o corpo do conhecimento utilizando aquele vocabulário é uma coleção de fatos sobre um domínio (CHANDRASEKARAN *et al.*, 1999).

⁶ Há, no entanto, ferramentas disponíveis no contexto da Estação TABA que possibilitam a definição de ontologias, na forma de teorias de domínio, para quaisquer domínios desejados. Neste sentido já foram definidas, por exemplo, ontologias para os domínios de Cardiologia (OLIVEIRA, 1999), Acústica Submarina (GALOTTA, 2000) e Entomologia (FOURO, 2002).

Ativo de Processo. A descrição e a discussão sobre estes conceitos são feitas nas subseções a seguir. Com esta evolução a ontologia passa a estar adequada à descrição de uma Corporação e de seus processos, passando a ser denominada, então Ontologia de Corporação. A descrição da ontologia em sua forma completa pode ser vista no Anexo IV.

As evoluções identificadas na ontologia foram definidas com base nas ontologias já citadas anteriormente, na experiência do grupo de Engenharia de Software da COPPE/UFRJ, em duas ontologias desenvolvidas por (BERTOLLO, 2006) e (RUY, 2006) sobre processo de software e organização, respectivamente, e em definições comumente aceitas pela comunidade relacionadas a processos de software (por exemplo, (ISO/IEC, 2003), (CHRISISS *et al.*, 2006) e (SOFTEX, 2007)).

A notação utilizada para a representação dos conceitos ontológicos usa o perfil UML definido por MIAN (2003) e que pode ser visto no Anexo II desta tese. Note que os axiomas que podem ser derivados automaticamente a partir do modelo apresentado não são descritos formalmente.

5.4.1 Organização e Corporação

Os conceitos de Organização e Corporação estão relacionados à questão de competência “Quais são as relações de hierarquia entre duas organizações?”, que é um detalhamento da questão mais geral “Como a organização está estruturada?”. As relações entre estes conceitos, que pertencem à subontologia de Estrutura, podem ser vistas na Figura 5.17.

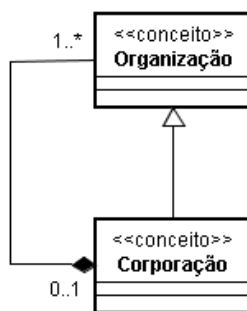


Figura 5.17 – Relações entre Organização e Corporação

Organizações, conforme definido por VILLELA (2004), são grupos de pessoas trabalhando em conjunto para o cumprimento de uma missão, onde a missão de uma organização define o seu propósito dentro do sistema econômico ou social.

Uma **corporação** pode ser definida como um tipo de organização composta de outras organizações (por exemplo, subsidiárias ou filiais) com certo grau de independência,

mas subordinadas, de alguma forma, aos interesses e regras da corporação. O que difere uma corporação de uma organização comum é o fato de haver uma relação de hierarquia entre as organizações que compõem a corporação e o comando central da corporação. Esta relação de hierarquia pode estar configurada de diversas formas, por exemplo, ser uma relação apenas administrativa, sendo as organizações unidades independentes com ramo de negócio particulares, ou então ser uma relação de comando e gerência das atividades fins da organização, estando as organizações subordinadas a uma linha mestra definida pela corporação. Neste segundo caso, é comum, também, que estas organizações tenham ramos de atuação semelhantes e compartilhem alguns nichos de mercado, mas em contextos/localidades diferentes.

Os axiomas não serão descritos formalmente por serem derivados diretamente do modelo. De forma geral, *toda corporação é uma organização* e *toda corporação é composta de organizações*. Todas as demais relações entre **organização** e os outros conceitos da ontologia continuam válidos para **corporação**. Além disso, uma *organização é uma organização independente se não está associada a uma corporação* e, de forma similar, uma *organização é subordinada a uma corporação quando está associada a uma corporação*.

5.4.2 Processo Padrão, Processo Especializado e Processo Definido para o Projeto

O conceito de Processo está relacionado à questão de competência “Quais são os processos executados na organização?”, que é um detalhamento da questão mais geral “Como a organização se comporta?”. Derivados deste conceito foram identificados três outros conceitos relacionados que visam a identificar seus diferentes tipos de abstração, respondendo, então à questão de competência “Quais são os diferentes graus de abstração de um processo?”. Um **processo**, conforme definido por VILLELA (2004), é um conjunto de atividades estruturadas e destinadas a resultar em um artefato ou serviço de valor para a organização ou para um determinado cliente ou mercado. Implica em uma ordenação específica das atividades com começo, fim, insumos e produtos claramente identificados (VILLELA, 2004). Uma importante classe de processo a ser identificada e definida é a de *processos organizacionais* (ou *processos de negócio*), que são aqueles processos utilizados por uma organização para formalizar e sistematizar as atividades que devem ser executadas para garantir que a organização se mantenha econômica e eficiente. Dentre estes processos, destacam-se os *processos de software*, que são aqueles processos utilizados por uma organização ou projeto para planejar, gerenciar, executar, controlar e monitorar suas

atividades relacionadas a software (ISO/IEC, 2003). A relação entre *processo* e seus subtipos pode ser vista através da Figura 5.18:

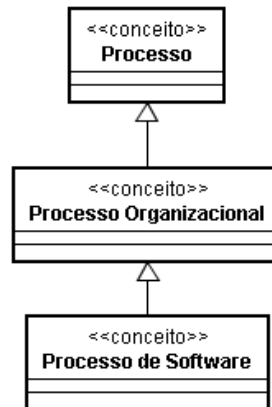


Figura 5.18 – Subtipos de processos segundo a natureza do processo

De acordo com o seu nível de abstração um *processo* pode ser um *processo padrão*, *processo especializado* ou *processo definido para o projeto*.

Um *processo padrão* é um conjunto de definições de processos básicos que guiam todos os processos na organização (ISO/IEC, 2003), geralmente são definidos num nível organizacional e contém os ativos de processo básicos que devem fazer parte de qualquer processo de projeto de uma organização (BERTOLLO, 2006). Essas definições de processos cobrem os elementos de processo fundamentais (e os seus inter-relacionamentos) que devem ser incorporados dentro dos processos definidos que são implementados nos projetos pela organização. Um processo padrão estabelece consistência entre as atividades através da organização e é desejável para estabilidade e melhoria de longo prazo. O conjunto de processos-padrão da organização descreve os elementos de processo fundamentais que será parte dos processos definidos para o projeto. Também descreve os relacionamentos (por exemplo: seqüência e interfaces) entre esses elementos do processo (SOFTEX, 2007).

Um *processo especializado* é um processo padrão adaptado a partir de outro processo padrão ao detalhar algumas das atividades do processo base com o objetivo de detalhar aspectos específicos seguindo um critério pré-determinado. Tais critérios podem incluir, mas não se limitam a detalhes específicos do paradigma, métodos de desenvolvimento, características do desenvolvimento de software na organização, como experiência dos desenvolvedores e tipos de software desenvolvidos, ou outras particularidades a critério da organização, como características de tamanho ou distribuição geográfica dos projetos a que se aplicam (adaptado de (VILLELA *et al.*, 2004) e

(BERTOLLO, 2006)). Todos os ativos de processo definidos para o processo padrão tornam-se parte do processo especializado, contudo, novos ativos podem ser incluídos (BERTOLLO, 2006)⁷.

Exemplos de paradigma são o estruturado, o orientado a objetos e o baseado em conhecimento. Os processos especializados são importantes para apoiar o desenvolvimento e a manutenção de software na organização, refletindo o conhecimento e a experiência obtidos sobre o uso do paradigma e dos métodos. A depender do tipo de software, pode haver a necessidade de incluir atividades específicas para o tipo de software e/ou de especializar atividades já previstas (VILLELA *et al.*, 2004).

Um *processo definido para o projeto* (também chamado de *processo instanciado* ou, simplesmente, *processo definido*) é um processo que é gerenciado (planejado, monitorado e ajustado) e é adaptado de um conjunto de processos padrão de acordo com os guias de adaptação da organização (ISO/IEC, 2003) e considerando as particularidades do projeto. Pode ou não ser definido a partir de uma adaptação de um **processo padrão**, neste caso será composto pelos ativos do processo padrão e novos ativos podem ser incluídos considerando características do projeto, como complexidade, tamanho, experiência da equipe etc. (BERTOLLO, 2006).

Outros axiomas não serão descritos formalmente por serem derivados diretamente do modelo que pode ser visto na Figura 5.19. De forma geral *todo processo padrão é um processo*, *todo processo especializado é um processo*, *todo processo definido é um processo* e *todo processo especializado é um processo padrão*. Além disso, *todo processo especializado é originado de um processo padrão ou de outro processo especializado* e *um processo definido pode ser originado de um processo especializado ou de um processo padrão*. Todas as demais relações entre **processo** e os outros conceitos da ontologia continuam válidos, exceto a relação original entre **processo** e **projeto** que foi transformada numa relação específica entre *processo definido para o projeto* e **projeto**. Dessa forma, *um projeto pode ser executado com base num processo definido para o projeto*.

⁷ BERTOLLO (2006) define conceitos na ontologia de Processo de Software específicos a critérios utilizados durante a especialização de processos. Nesta revisão da ontologia tais conceitos não foram incluídos, pois a definição utilizada deixa em aberto os tipos de critérios a serem utilizados para a especialização.

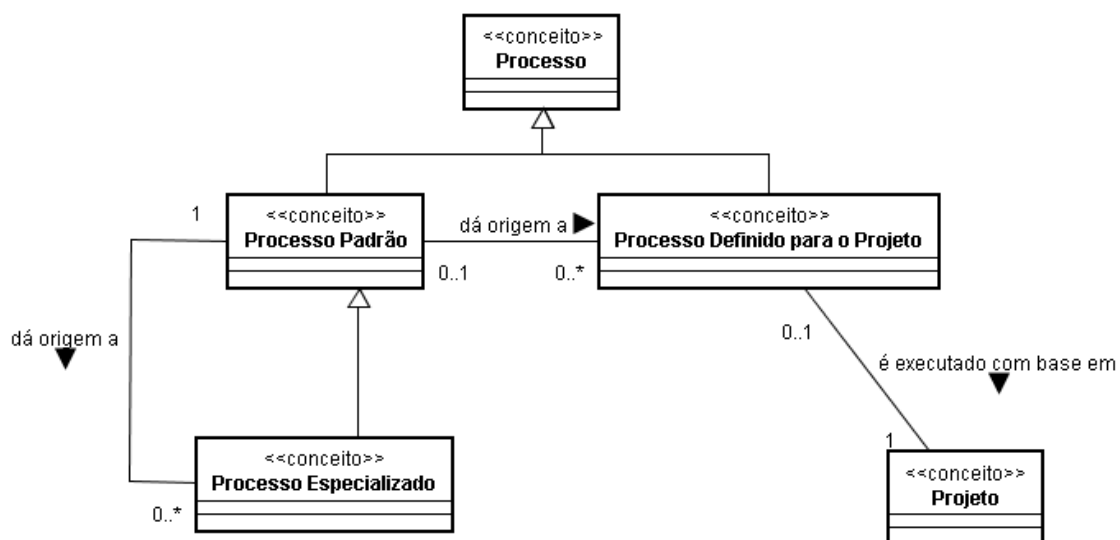


Figura 5.19 – Conceitos derivados de Processo

5.4.3 Ativo de Processo

Uma nova adição à ontologia foi o conceito *ativo de processo*, que pode ser definido como algo que a organização considere útil para atingir os objetivos do processo, por exemplo, políticas, processos definidos, lições aprendidas, roteiros (*templates*) de documentos, padrões, material de treinamento (CHRISSEIS *et al.*, 2006). A identificação de um ativo de processo é dependente do contexto em que está inserido, pois um mesmo item (por exemplo, um plano de projeto ou manual) pode ser considerado ativo de processo para uma organização e não para outra. Em geral, a definição de um ativo de processo se refere a um item material relacionado a um processo, dessa forma, uma pessoa não poderia ser considerada um ativo de processo, mas uma vez que um conhecimento desta pessoa estiver externalizado e explícito tal conhecimento poderia ser considerado um ativo de processo. Este termo foi definido na ontologia na forma de um conceito derivado expresso por meio de axiomas que indicam quando determinadas entidades da ontologia devem ser consideradas ativos de processo⁸. Esta definição ajuda a responder com mais detalhes à questão de competência “Quais são os processos executados na organização?”, citada anteriormente.

⁸ Outra possibilidade para a definição de *ativo de processo* seria modelá-lo como um papel desempenhado por outros conceitos da ontologia. GUIZZARDI (2005) apresenta um padrão para esta modelagem, entretanto, considerou-se que a modelagem dessa forma tornaria a formalização da ontologia mais complexa. Dessa forma, visando à simplificação da modelagem, e esperando que com isso haja uma maior facilidade de disseminação de seu conhecimento, a modelagem de *ativo de processo* como conceito derivado foi escolhida.

Dentre os conceitos definidos na ontologia foram identificados cinco que podem ser considerados *ativos de processo*: *processo*, *documento*, *ferramenta de software*, *atividade* e *procedimento*. Assim, são considerados ativos de processo *todo processo de uma organização, toda atividade que faça parte de um processo, toda ferramenta que seja recurso material para a execução de uma atividade que faça parte de um processo, todo documento que seja insumo de uma atividade que faça parte de um processo* e, por fim, *todo procedimento que seja alocado a uma atividade que faça parte de um processo*. É importante observar, no entanto, que outros itens não previstos na ontologia também podem ser considerados ativos de processo. Dessa forma, o conjunto de axiomas apresentado abaixo não é completo e outros ativos de processo podem vir a ser considerados na medida em que outros conceitos sejam adicionados à ontologia.

$$\forall p (\text{processo}(p) \rightarrow \text{ativo_processo}(p)) \quad (\text{A1})$$

$$\forall a, p (\text{atividade}(a) \wedge \text{processo}(p) \wedge \text{parte_de}(a, p) \rightarrow \text{ativo_processo}(a)) \quad (\text{A2})$$

$$\forall a, p, f (\text{atividade}(a) \wedge \text{processo}(p) \wedge \text{ferramenta_software}(f) \wedge \text{recurso_material}(f, a) \wedge \text{parte_de}(a, p) \rightarrow \text{ativo_processo}(f)) \quad (\text{A3})$$

$$\forall a, p, d (\text{atividade}(a) \wedge \text{processo}(p) \wedge \text{documento}(d) \wedge \text{insumo}(d, a) \wedge \text{parte_de}(a, p) \rightarrow \text{ativo_processo}(d)) \quad (\text{A4})$$

$$\forall a, p, r (\text{atividade}(a) \wedge \text{processo}(p) \wedge \text{procedimento}(r) \wedge \text{possivel_alocacao}(r, a) \wedge \text{parte_de}(a, p) \rightarrow \text{ativo_processo}(r)) \quad (\text{A5})$$

5.5 Arquitetura da Estação Taba e dos Ambientes de Engenharia de Software Orientados a Corporação

Esta seção apresenta um histórico da evolução da arquitetura da Estação Taba ao longo do tempo e detalhes da arquitetura definida para a criação da infra-estrutura na Web da Estação Taba para viabilizar os Ambientes de Engenharia de Software Orientados a Corporação.

5.5.1 Histórico de Evolução da Arquitetura da Estação Taba

Apesar de ter sido criado em 1990 (ROCHA *et al.*, 1990), o Projeto Taba teve sua primeira implementação em 1994 (TRAVASSOS, 1994). Neste trabalho foram definidos para os Ambientes de Desenvolvimento de Software (ADS) da Estação Taba os requisitos, funções, requisitos e estratégias de integração que têm sido utilizados, revistos e atualizados

desde então. Em dois outros momentos a implementação da Estação Taba foi revista: com a definição dos Ambientes de Desenvolvimento de Software Orientados a Domínio (ADSOD) em 1999 (OLIVEIRA, 1999) e com a revisão desta infra-estrutura para a definição dos Ambientes de Desenvolvimento de Software Orientados a Organização (ADSOrg) em 2004 (VILLELA, 2004). Estas duas implementações evoluíram a lista de requisitos e funções da Estação Taba e também modificaram a arquitetura vigente dos ambientes. Esta seção descreve as principais características destas três implementações e serve como ponto de partida para entender as modificações realizadas na arquitetura da Estação Taba para a construção dos Ambientes de Engenharia de Software Orientados a Corporação (AESCorp).

5.5.1.1 A Arquitetura da Estação Taba de 1990 a 1999

A partir de 1990, quando foi definida a Estação Taba (ROCHA *et al.*, 1990), muitos trabalhos foram realizados no contexto do projeto abordando diversos aspectos desde a definição e implementação de aspectos relacionados à infra-estrutura da Estação (TRAVASSOS, 1994; ARAÚJO, 1998; FALBO, 1998; FATEEV, 1998) até a definição e construção de ambientes específicos tais como os ambientes ORIXÁS (WERNECK, 1995) e MEMPHIS (WERNER *et al.*, 1997). Foram, também, realizados vários estudos teóricos e de campo (WERNECK, 1990; CRISPIM, 1991; ASSIS, 1992; MOURA, 1992; MASSOLAR, 1993) e foram implementadas ferramentas específicas aos ambientes (AGUIAR, 1992; BARROS, 1996; CIMA, 1996; COELHO, 1997; NIELBOCK, 1997; VASCONCELOS, 1997).

Dentre estes trabalhos, destaca-se o realizado por TRAVASSOS (1994) que definiu e implementou o modelo de integração de ferramentas da Estação TABA, a partir do qual os diversos ambientes e ferramentas foram construídos. A estrutura da Estação Taba foi definida contendo um conjunto de componentes integrados que possuíam controle sobre suas existências, informações, estados e funcionalidades básicas associados. Esta definição baseou-se na idéia de que um ADS está inserido no contexto de um produto de software e é o responsável, com seu conjunto de funcionalidades e informações associadas, por traduzir, de acordo com a necessidade do usuário, uma representação do mundo real para o computacional. Foram definidos, então, nove componentes nesta estrutura:

- *Sistema Computacional* - representado por uma dimensão de hardware e outra do sistema operacional;

- *Componente de Interface com o Usuário* - responsável pela gerência e controle da interação do usuário com o ADS, garantindo a consistência da apresentação das ferramentas;
- *Componente de Cooperação* - responsável pela comunicação entre os usuários e suas necessidades de interação com outros membros da equipe, possuindo definições de protocolos de comunicação que devem ser utilizados no ADS e auxiliando o componente de Interface com o Usuário para prover funcionalidades e recursos que o estendam de forma a suportar a interface de grupo;
- *Componente de Controle de Processos* - responsável pelo controle e gerenciamento do processo de desenvolvimento através da identificação de atividades, gerência da execução das ferramentas e controle dos papéis e atividades dos usuários;
- *Componente de Reutilização* - provê o ADS de mecanismos que possibilitam a reutilização de trabalhos anteriores, a nível da especificação do ADS, de componentes para a construção de ferramentas, bem como de componentes de todo tipo para a construção da aplicação;
- *Componente de Suporte Inteligente* - fornece inteligência global ao ambiente, assistindo o usuário na utilização do meta-ambiente e dos ADS, através de assistentes inteligentes;
- *Componente de Conhecimento* - incorpora mecanismos para o armazenamento e a utilização de conhecimento, que podem ser utilizados tanto pelo meta-ambiente, quanto pelos ADSs instanciados;
- *Componente de Repositório Comum* - responsável pelo controle, gerenciamento e armazenamento dos objetos manuseados pelo ambiente e meta-ambiente, e pela consistência e integridade das informações; e
- *Componente de Controle de Versões* - responsável pelo controle e gerência de versões de documentos e itens de software produzidos.

A utilização, em conjunto, destes componentes permite a integração de ferramentas ao ambiente e provê recursos para a incorporação de ferramentas externas. A utilização desses componentes define, ainda, a filosofia de integração da Estação Taba e dos seus ambientes instanciados através de quatro tipos de integração (TRAVASSOS, 1994):

- *Integração de Apresentação e Interação* - possibilita a homogeneização das formas de apresentação das informações e das técnicas de interação com o usuário;

- *Integração de Gerenciamento e Controle* - provê, para ferramentas e para os ambientes, serviços e funcionalidades que permitem o funcionamento de forma organizada ao longo do processo de desenvolvimento;
- *Integração de Dados* - estabelece a forma como as ferramentas da Estação realizarão o armazenamento e recuperação das informações; e,
- *Integração de Conhecimento* - torna possível os serviços básicos de armazenamento, gerenciamento e utilização do conhecimento descrito e adquirido ao longo do processo de desenvolvimento.

Esse modelo, proposto por Travassos, foi implementado em Eiffel, na plataforma Unix da *Sun Workstation*, por representar uma tecnologia robusta e poderosa. Os diversos trabalhos realizados posteriormente na Estação Taba procuraram manter o aspecto de integração entre os diferentes componentes. No entanto, apesar de ser uma plataforma adequada para trabalhos de pesquisa, as idéias implementadas na Estação Taba não puderam, na maioria dos casos, ser realmente utilizadas para experimentações em ambientes reais pela falta de portabilidade do código para plataformas mais acessíveis e comumente utilizadas (como a plataforma de microcomputadores). Além disso, a linguagem Eiffel tem suporte técnico de difícil acesso e não é amplamente conhecida (o que dificulta, por exemplo, esclarecimentos de problemas de implementação em listas de *newsgroup*), e apresentou vários problemas com relação à interface com o usuário (OLIVEIRA, 1999).

5.5.1.2 Arquitetura da Estação Taba de 1999 a 2003 – Implementação dos ADSOD

Considerando os problemas relatados acima e a filosofia de desenvolvimento de ambientes padronizados para organizações e para domínios específicos definidos por OLIVEIRA (1999), decidiu-se re-implementar a Estação Taba em uma plataforma que, além de permitir o desenvolvimento de pesquisas e de ADS, facilitasse a realização de experiências de uso desses ambientes nas organizações para os quais são concebidos. Essas considerações levaram à escolha da plataforma de microcomputadores e ao uso da linguagem C++ (OLIVEIRA, 1999). Decidiu-se, neste momento, a re-implementação apenas das funcionalidades de especificação, instanciação e execução de ambientes, por serem estas as fundamentais para se ter a infra-estrutura necessária para construção de ADSOD.

As principais funcionalidades providas pela Estação Taba a partir desta implementação foram:

- *Apoiar a definição de conhecimento de processo* – responsável por apoiar introdução de descrições gerais de atividades, processos, métodos, técnicas e demais aspectos necessários para auxiliar na definição de um processo de software.
- *Apoiar a descrição de tarefas* – responsável por apoiar a descrição de tarefas a serem utilizadas no ambiente instanciado.
- *Apoiar a definição da Teoria do Domínio* – responsável pela disponibilização de uma ferramenta para permitir a introdução da Teoria do Domínio específico para possibilitar a instanciação de ADSOD.
- *Gerenciar ferramentas* – responsável pelas facilidades da Estação Taba para integrar ferramentas, permitindo o seu uso na Estação e em seus ambientes instanciados (conforme definido por TRAVASSOS (1994)).
- *Instanciar ADS* – responsável pela geração do ambiente instanciado, propriamente dito, de acordo com as características do processo previamente definidas.

As três primeiras funcionalidades (definição de conhecimento de processo, descrição de tarefas e definição da Teoria do Domínio) referem-se ao *Componente de Conhecimento* e à *Integração de Conhecimento*, propostos pela primeira vez por TRAVASSOS (1994). As demais funcionalidades (gerência de ferramentas e instanciação de ADS) referem-se ao *Componente de Controle de Processo* e à *Integração de Gerenciamento e Controle*, também propostos por TRAVASSOS (1994).

Com essa infra-estrutura foram definidas novas ferramentas (GALOTTA, 2000) (MACHADO, 2000; GOMES, 2001; ZLOT, 2002) e os ambientes Cordis (OLIVEIRA, 1999), Netuno (GALOTTA, 2000) e Insecta (FOURO, 2002).

5.5.1.3 Arquitetura da Estação Taba de 2003 a 2007 – Implementação dos ADSOrg

Com a definição dos ADSOrg (VILLELA, 2004) as funcionalidades existentes na Estação Taba foram novamente revistas e também mantiveram a aderência ao modelo de integração proposto por TRAVASSOS (1994).

O *Componente de Controle de Processos* é o que sofre maiores acréscimos de funcionalidades. Dentre elas está o *apoio à especificação do Ambiente Configurado* (responsável pela identificação das necessidades da organização em relação ao Ambiente Configurado,

apoiando também a elaboração de uma proposta de configuração) e o *apoio à criação de ambientes* (responsável pela geração e teste dos ambientes configurados ou instanciados), além do apoio à execução do processo de fato. Este apoio é iniciado com o *apoio à instanciação de ADSOrg* para projetos específicos através da descrição dos projetos, o planejamento dos seus respectivos processos e a criação dos ambientes propriamente ditos. Neste sentido, *apoiar a descrição de um projeto* implica em apoiar a descrição de suas características e das características do produto a ser desenvolvido ou mantido, de forma a possibilitar o *apoio ao planejamento do processo* através da sugestão de modelos de ciclo de vida, de mapeamentos entre atividades do processo e o modelo de ciclo de vida selecionado, e da sugestão das técnicas de avaliação da qualidade mais adequadas ao projeto. Os Ambientes Instanciados têm como funcionalidades controlar a execução do processo, apoiar a execução das atividades de engenharia e gerência, além de apoiar a gerência de artefatos. *Controlar a execução do processo* implica em fornecer acesso a roteiros, ferramentas e artefatos de acordo com a atividade sendo executada, além de apoiar o acompanhamento e a alteração do processo de software durante a sua execução conforme proposto por (ARAÚJO, 1998) e evoluído por (MAFRA, 2004). *Apoiar a gerência de artefatos* refere-se ao apoio ao acesso compartilhado dos artefatos, ao controle de versões e à gerência de configuração. Estes ambientes expandem o *apoio dado à execução das atividades de engenharia e gerência* fornecendo informações e conhecimentos sobre a organização, sobre o domínio de aplicação e de Engenharia de Software, sobre o processo de software, além do conhecimento organizacional em desenvolvimento e manutenção de software adquirido através da experiência em projetos específicos. Experiências do projeto são coletadas e as Teorias de Domínio e a descrição da organização podem ser evoluídas.

O *Componente de Conhecimento* da Estação Taba passa a ser composto também pelo *apoio à descrição da organização* através da descrição de sua estrutura, seus profissionais e processos organizacionais; do *apoio à gerência do conhecimento organizacional* com a evolução das Teorias de Domínio disponíveis no ambiente e no apoio à produção de conhecimento organizacional a partir dos conhecimentos adquiridos em projetos específicos.

O *Componente de Reutilização* está relacionado à reutilização de conhecimento produzido e armazenado pelos ambientes e, também, das descrições de processos padrão e especializados presentes nos Ambientes Configurados e utilizados pelos Ambientes Instanciados.

O *Componente de Repositório Comum*, além dos dados produzidos pelos projetos relativos à execução do processo, passa ser composto também pelos repositórios de

Experiências da Organização e de *Experiências do Projeto*, que contêm conhecimento que foi adquirido através da experiência, que é complementar ao conhecimento teórico.

5.5.2 A Implementação Atual da Estação Taba em Plataforma Web

A evolução tecnológica e as mudanças econômicas ocorridas nos últimos anos vêm alterando significativamente as características das organizações e, conseqüentemente, dos projetos de desenvolvimento de software. Observa-se, por exemplo, uma tendência de descentralização das equipes dos projetos (AUDY e PRIKLADNICKI, 2007), com seus integrantes trabalhando distribuídos em localidades distintas. Sendo assim, são comuns situações tais como: parte da equipe do projeto ficar localizada no cliente e a outra na sede da empresa que está desenvolvendo o software; a impossibilidade de o gerente do projeto ficar no mesmo local dos demais integrantes da equipe do projeto, ou, duas equipes de projetos diferentes de uma mesma organização, localizadas em unidades distantes fisicamente uma da outra, precisarem acessar uma base de dados única relativa a todos os projetos da organização (MONTEIRO e SANT'ANNA, 2006). Além disso, numa corporação, é comum que as organizações estejam em localidades diferentes e, também neste contexto, aumenta a necessidade de acesso aos dados e às ferramentas entre a corporação e as organizações e mesmo dentre as próprias organizações de uma mesma corporação. Esse cenário impõe alterações na forma de gerenciar projetos de software para garantir o seu sucesso. Um dos pontos que merece destaque é a necessidade de acompanhamento e execução das atividades a serem desempenhadas, que devido à sua complexidade requer a utilização de uma ferramenta ou ambiente de apoio específico.

A arquitetura da Estação Taba, desenvolvida entre 1994 e 2004 (TRAVASSOS, 1994; OLIVEIRA, 1999; VILLELA, 2004), porém, não foi concebida para apoiar a descentralização do uso das ferramentas, ambientes e bases de dados. Além disso, ao longo do tempo também percebeu-se através do relato dos usuários que uma mudança de plataforma de forma a apoiar a descentralização do uso das ferramentas, ambientes e bases de dados seria necessária para atender às crescentes expectativas em relação à Estação Taba. Sendo assim, uma reformulação da arquitetura dos ambientes fez-se necessária. O levantamento de requisitos da nova infra-estrutura foi feito com base na construção de um protótipo e na aplicação de um instrumento específico para a coleta de requisitos que a nova implementação deveria atender.

5.5.2.1 A Estação Taba na Web

As ações para a definição da versão Web da Estação Taba foram iniciadas de forma paralela. O protótipo teve início assumindo a premissa de que alguns requisitos fundamentais seriam ratificados pela coleta de requisitos. Ao final do levantamento de requisitos a lista inicial foi revista e o protótipo aprimorado para contemplar as alterações identificadas e os novos requisitos. Dentre os requisitos fundamentais definidos para a construção do protótipo constavam o uso de linguagens e *frameworks* gratuitos (mas não necessariamente *open source*), multi-plataforma (pelo menos Linux e Windows), com suporte a orientação a objetos, com acesso nativo a banco de dados e com apoio a internacionalização de interfaces. Outro requisito importante era manter a compatibilidade com o formato da base de dados utilizada pela Estação Taba de forma a não causar perda de dados pelas empresas que já a utilizavam e, assim, não impactar negativamente na adoção da versão na Web por estas empresas. Além disso, desejou-se a escolha de uma tecnologia que pudesse contar com comunidades ativas de desenvolvedores de forma a garantir a sobrevivência do projeto a longo prazo.

O instrumento para a identificação de requisitos foi aplicado com os usuários e os desenvolvedores da Estação Taba, além dos consultores que a utilizavam em implantação de processos nas organizações. Foi solicitado que os respondentes identificassem características que a nova Estação Taba Web deveria ter/possibilitar e as características que ela não deveria ter/possibilitar. Os formulários foram preenchidos livremente pelos respondentes. A pesquisa foi enviada para 81 pessoas, sendo 47 membros de 20 de empresas que já tinha utilizado a Estação Taba ou estavam utilizando no momento, 26 consultores de implantação de processos e 14 membros da equipe de desenvolvimento, sendo que 6 eram tanto consultores quanto membros da equipe de desenvolvimento. Destas, 22 pessoas responderam os questionários. Foram identificados 200 itens referentes a diversos aspectos tanto com impacto em decisões arquiteturais quanto relativos a melhorias ou correções identificadas nas ferramentas da Estação Taba. Os itens relacionados a correções e melhorias foram direcionados à equipe de suporte e os demais foram analisados pelo responsável pela definição da arquitetura da Estação Taba. Chegou-se, então, a uma lista de 20 requisitos agrupados nas categorias arquitetura, desempenho, concorrência, usabilidade, portabilidade, manutenibilidade e segurança.

Estes requisitos, que englobaram o conjunto inicial utilizado para a construção do protótipo e podem ser vistos na Tabela 5.5, representam os Requisitos de Arquitetura do Produto da Estação Taba e devem ser respeitados e apoiados em sua nova implementação.

Tabela 5.5 – Requisitos de Arquitetura do Produto da Estação Taba

Categoria	#	Descrição
Arquitetura	1	Possuir uma arquitetura bem definida, utilizando componentização para facilitar o desenvolvimento, e que minimize o esforço e custo de construção das funcionalidades e ferramentas.
	2	Permitir a utilização do sistema em diferentes locais sem a necessidade de instalação prévia.
	3	Ser construído usando tecnologias escaláveis, robustas e bem aceitas no mercado, de preferência gratuitas, evitando a utilização de tecnologias incipientes, instáveis e/ou não confiáveis.
	4	Possuir uma interface desacoplada, de forma a permitir sua evolução sem impacto nos demais elementos da aplicação.
	5	Possuir uma integração de dados entre as ferramentas evitando a duplicidade de entrada de informações.
	6	Possibilitar a utilização do sistema de qualquer lugar através de interface Web.
	7	Manter compatibilidade com a infra-estrutura atual da Estação Taba.
Concorrência	8	Possibilitar o acesso concorrente de múltiplos usuários ao sistema nas mesmas ferramentas e ambientes.
Desempenho	9	Possuir um tempo de resposta aceitável para as ações do usuário.
Usabilidade	10	Possuir facilidades de uso para a inclusão e edição das informações a serem digitadas.
	11	Possuir uma interface amigável, mantendo o padrão da interface atual da Estação Taba, mas adicionando outras facilidades.
	12	Possuir suporte à internacionalização dos elementos de interface e mensagens para os usuários.
	13	Ter uma interface padrão entre as ferramentas.
Portabilidade	14	Possuir independência de sistema operacional do cliente ou do servidor. Além disso, possibilitar a utilização de diferentes navegadores.
	15	Poder ser utilizado com diferentes tipos de banco de dados.
	16	Possuir dispositivo para atualização automática de versões.
	17	Possuir dispositivo para envio automático de exceções e ocorrências de falha.
	18	Facilitar a manutenção das funcionalidades do ambiente, permitindo a evolução dos elementos de forma independente sem impacto nos demais elementos do ambiente.
Segurança	19	Armazenar o log das ações tomadas pelos usuários tanto para fins de auditoria quanto para auxiliar na identificação de erros nas ferramentas.
	20	Possibilitar a definição de diferentes perfis de controle de acesso para o acesso às ferramentas e às funcionalidades de determinadas ferramentas mais críticas de acordo com o papel nos projetos.

Os trabalhos para uma nova implementação começaram de fato no final de 2005 como um trabalho de final de curso, sob minha orientação, desenvolvido com o objetivo de investigar as soluções tecnológicas que melhor se adequariam aos requisitos fundamentais a apoiar o desenvolvimento dos ambientes da Estação Taba. A implementação do protótipo foi feita utilizando a tecnologia JEE (SUN, 2008a) e, ao final de 2006, foi gerada uma primeira versão de um Ambiente de Desenvolvimento de Software instanciado pela Estação Taba na plataforma Web (MONTEIRO e SANT’ANNA, 2006). A partir desta implementação, algumas deficiências e melhorias foram identificadas, boa parte causada pela evolução da tecnologia em que a infra-estrutura se baseava com a conclusão da especificação do JEE 5.0 (SUN, 2008b) e pelo amadurecimento das idéias relacionadas à arquitetura que os ambientes deveriam possuir. Percebeu-se neste momento, também, que a integração entre as classes de modelo da infra-estrutura Web e a camada de

dados da Estação Taba não estava adequada devido a particularidades do mapeamento Objeto-Relacional utilizado. Como um dos requisitos a ser respeitado na implementação era a integração a base de dados da Estação Taba, evitando a perda de dados e possibilitando a manutenção por um tempo das duas versões em funcionamento na empresas, foi necessária a reformulação destas classes pela equipe de desenvolvimento da Estação Taba. Em paralelo a isso, novas tecnologias começaram a ser desenvolvidas ou então adaptadas à arquitetura do JEE 5.0. Com isso, soluções dadas para criação das classes de modelo, vista e controle foram revistas e reformuladas.

A estrutura atual da Estação Taba na Web já contempla a maior parte dos requisitos da arquitetura do produto identificados anteriormente. Uma discussão sobre a implementação destes requisitos pode ser vista na Tabela 5.6.

Tabela 5.6 – Solução Atual para Implementação dos Principais Requisitos de Arquitetura do Produto da Estação Taba

#	Requisito	Solução
1	Possuir uma arquitetura bem definida, utilizando componentização para facilitar o desenvolvimento, e que minimize o esforço e custo de construção das funcionalidades e ferramentas.	A arquitetura foi definida de modo a possuir baixo acoplamento entre camadas e baixa dependência de tecnologias específicas. O esforço para desenvolvimento de novas funcionalidades e ferramentas foi minimizado visto que implementações de atividades básicas já foram feitas e o padrão de desenvolvimento foi definido com o objetivo de ser simples e intuitivo. A aplicação de padrões de projeto facilitou cumprimento deste requisito.
2	Permitir a utilização do sistema em diferentes locais sem a necessidade de instalação prévia.	Como está sendo utilizada uma arquitetura Web, a Estação Taba poderá ser utilizada em diferentes locais através de um navegador e acesso à internet ou intranet da empresa.
3	Ser construído usando tecnologias escaláveis, robustas e bem aceitas no mercado, de preferência gratuitas, evitando a utilização de tecnologias incipientes, instáveis e/ou não confiáveis.	A infra-estrutura foi construída utilizando a tecnologia Java EE, padrão adotado pelo mercado mundial para aplicações de grande porte em Java. Utilizou, também, <i>frameworks open-source</i> (como Hibernate, log4j, JUnit, etc.) desenvolvidos e testados por organizações reconhecidas.
4	Possuir uma interface desacoplada, de forma a permitir sua evolução sem impacto nos demais elementos da aplicação.	Utilização do padrão MVC, permitindo uma definição clara da interface dos componentes, utilização de uma arquitetura em 3 camadas (dados, negócio e apresentação), que restringe o impacto de mudanças através de camadas e a utilização da infra-estrutura para componentes de negócio e apresentação definidos na especificação Java EE. A utilização de padrões definidos na especificação Java EE (como EJB, JPA e JSF) facilitou a adoção destes padrões, permitindo isolamento até mesmo do fornecedor do sistema gerenciador de banco de dados.
5	Possuir uma integração de dados entre as ferramentas evitando a duplicidade de entrada de informações.	Este requisito é atingido pelo compartilhamento da camada de dados, única para todos os ambientes e ferramentas internas da Estação Taba.
6	Possibilitar a utilização do sistema de qualquer lugar através de interface Web.	A arquitetura atual está toda baseada em tecnologias Web. Como todas as ferramentas atuais da Estação Taba ainda não foram reimplementadas na nova arquitetura, durante algum tempo as ferramentas já existentes continuarão ser utilizados junto com a versão Web dos ambientes Taba.

Tabela 5.6 – Solução Atual para Implementação dos Principais Requisitos de Arquitetura do Produto da Estação Taba

#	Requisito	Solução
7	Manter compatibilidade com a infraestrutura atual da Estação Taba.	O banco de dados utilizado pela Estação Taba foi definido a partir do modelo de classes utilizadas na implementação do sistema, definido na linguagem C++. As mesmas regras utilizadas para geração do banco de dados foram feitas para criação do modelo de classes Java da Estação Taba na Web. Dessa forma, a Estação Taba utiliza o mesmo banco de dados da Estação Taba, através de mapeamentos utilizando tecnologia JPA, resultando na criação de um novo modelo de classes compatível ao já existente em C++.
8	Possibilitar o acesso concorrente de múltiplos usuários ao sistema nas mesmas ferramentas e ambientes.	Uma solução robusta para controle de concorrência entre os usuários dos ambientes ainda não foi implementado.
9	Possuir um tempo de resposta aceitável para as ações do usuário.	<i>Frameworks</i> com tecnologia AJAX foram adotados para otimizar o tráfego na rede e melhorar a navegabilidade e a percepção do usuário quanto ao tempo de resposta. Também foram adotados <i>pool</i> de objetos de negócio (seguindo a especificação EJB) e <i>frameworks</i> de cache para acesso ao banco de dados. O tempo de resposta ainda pode ser aumentado utilizando <i>clusters</i> específicos para as camadas de negócio e apresentação (conforme a especificação JEE) e dados (dependendo do sistema gerenciador de banco de dados escolhido).
10	Possuir facilidades de uso para a inclusão e edição das informações a serem digitadas.	<i>Frameworks</i> com tecnologia AJAX foram adotados para otimizar o tráfego na rede e melhorar a navegabilidade e a percepção do usuário quanto à interação com o sistema.
11	Possuir uma interface amigável, mantendo o padrão da interface atual da Estação Taba, mas adicionando outras facilidades.	A interface padrão das ferramentas da Estação Taba, com a orientação a processo e integração com a ferramenta de gerência de conhecimento, foi mantida e adaptada para o padrão de aplicações Web.
12	Possuir suporte à internacionalização dos elementos de interface e mensagens para os usuários.	Implementado através da utilização de frameworks para a tradução da interface (como definidos na especificação JEE), utilizando arquivos de propriedades para exibição de mensagens na interface com o usuário, mensagens de negócio e textos associados a enumerações. A internacionalização será viabilizada com o comprometimento dos desenvolvedores em cada uma das novas funcionalidades e ferramentas.
13	Ter uma interface padrão entre as ferramentas.	Todo o conjunto de ferramentas e ambientes possui uma interface padrão e os componentes de vista utilizam uma biblioteca padrão que possibilita ao usuário adaptar certos elementos da interface (por exemplo, cores dos elementos de tela), porém sem descaracterizar os componentes utilizados.
14	Possuir independência de sistema operacional do cliente ou do servidor. Além disso, possibilitar a utilização de diferentes navegadores.	A utilização de Java EE como plataforma base permite que diferentes plataformas e sistemas operacionais, como Unix, Windows ou Macintosh, sejam utilizados. Todas as tecnologias utilizadas permitem, também, a utilização de diferentes navegadores em quaisquer destas plataformas.
15	Poder ser utilizado com diferentes tipos de banco de dados.	Este requisito é conseguido utilizando-se um mapeamento objeto-relacional que permite a utilização de qualquer servidor de banco de dados, seguindo a especificação JPA.
16	Possuir dispositivo para atualização automática de versões.	Requisito ainda não implementado.
17	Possuir dispositivo para envio automático de exceções e ocorrências de falha.	Requisito ainda não implementado.
18	Facilitar a manutenção das funcionalidades do ambiente, permitindo a evolução dos elementos de forma independente sem impacto nos demais elementos do ambiente.	Testes automatizados garantem eficiência aos testes de integração, permitindo que uma execução freqüente e que impactos em outros elementos do ambiente sejam detectados precocemente. Além disso, o desacoplamento entre os diversos componentes reduz a probabilidade de eventos desta natureza.

Tabela 5.6 – Solução Atual para Implementação dos Principais Requisitos de Arquitetura do Produto da Estação Taba

#	Requisito	Solução
19	Armazenar o log das ações tomadas pelos usuários tanto para fins de auditoria quanto para auxiliar na identificação de erros nas ferramentas.	A adoção do <i>framework</i> log4j facilita a captura de informações ao longo da execução da aplicação e permite seu armazenamento em diversas mídias, como arquivos, bancos de dados ou até mesmo o envio remoto das informações através de <i>sockets</i> .
20	Possibilitar a definição de diferentes perfis de controle de acesso para o acesso às ferramentas e às funcionalidades de determinadas ferramentas mais críticas de acordo com o papel nos projetos.	A utilização de componentes que verificam automaticamente a permissão de acesso do usuário às páginas valida seu acesso, impedindo a execução de ações indevidas ou a visualização de dados sensíveis.

5.5.2.2 Arquitetura Atual da Estação Taba

Com a nova implementação da Estação Taba, foram novamente revistos os seus componentes e a filosofia de integração foi adaptada à nova realidade. Além de trazer melhoria no gerenciamento e execução de um processo de software em uma organização que utiliza a Estação Taba, uma mudança arquitetural como a que foi feita implica também em ganhos no seu desenvolvimento e manutenção. Esta mudança beneficia diretamente não só as empresas que utilizam a Estação Taba como os seus desenvolvedores. A Estação Taba, por ser um projeto acadêmico e de pesquisa, tem suas funcionalidades implementadas à medida que dissertações e teses agregam ao sistema novas ferramentas de apoio às atividades de um processo de software. Devido a essa característica, há uma grande rotatividade e diversidade na equipe de desenvolvedores. Dessa forma, é de extrema importância que os desenvolvedores sigam um padrão de desenvolvimento com o objetivo de manter a uniformidade da implementação e, assim, facilitar a manutenção. Além disso, os desenvolvedores devem se preocupar prioritariamente com a lógica de funcionamento de suas ferramentas e não com aspectos gerais da Estação Taba, tais como controle de acesso e apresentação. Desse modo, é possível o desenvolvimento de novas ferramentas de forma ágil, possibilitando que os alunos dêem maior ênfase à pesquisa e elaboração de suas dissertações e teses (MONTEIRO e SANT'ANNA, 2006).

As diferentes ferramentas da Estação Taba às vezes respondem por um conjunto específico de funcionalidades e são responsáveis por prover algum tipo de serviço. Assim, podem ser agrupadas na forma de componentes. Cada um dos diferentes tipos de ambientes da Estação Taba possui componentes específicos responsáveis por atenderem aos requisitos identificados na Tabela 2.3. Além disso, há um conjunto de serviços básicos, exibidos na Figura 5.20.

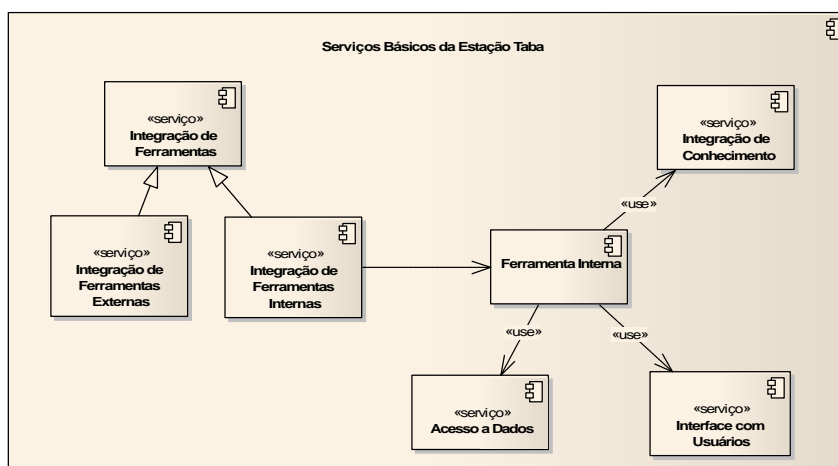


Figura 5.20 – Serviços Básicos da Estação Taba

Para esta figura e as demais que descrevem os serviços da Estação Taba foi utilizada a notação UML. Os serviços básicos da Estação Taba são representados com o estereótipo <<serviço>>.

Estes requisitos fazem parte da infra-estrutura mínima dos ambientes e são responsáveis por prover funcionalidades como integração de ferramentas internas e externas, padronização da interface com usuários e prover acesso a diferentes fontes de dados, como bases de dados e conhecimento internas e externas. Estes serviços são responsáveis pela implementação da estratégia de integração definida por TRAVASSOS (1994) e são compartilhados por todas as ferramentas, componentes e ambientes da Estação Taba.

A Figura 5.21 apresenta detalhes de como a implementação da Estação Taba na Web foi realizada em relação às soluções tecnológicas adotadas e detalhes de implementação em relação à estrutura em camadas e a filosofia de integração adotadas.

	Função	Filosofia de Integração da Estação Taba	Solução Tecnológica	Detalhes da Implementação
Camada de Apresentação	Sua principal função é traduzir as tarefas e resultados de forma que o usuário possa entender.	Integração de Apresentação e Interação - possibilita a homogeneização das formas de apresentação das informações e das técnicas de interação com o usuário.	A vista do padrão MVC model 2 está presente nesta camada. Aqui é realizado o controle de fluxo entre as telas através da arquitetura do JSF, a construção de telas utilizando tags JSF, a exibição de dados nestas telas e o callback para a camada de negócio através de Managed Beans, presentes da especificação do JSF.	Todas as ferramentas possuem os mesmos padrões para construção da interface e utilizam os mesmos componentes, todos da biblioteca RichFaces, que já trazem de forma automática e transparente, recursos de AJAX.
Camada Lógica	Esta camada coordena a aplicação, processo comandos, toma decisões lógicas e desempenha cálculos. Também transporta e processo dados entre as duas outras camadas.	Integração de Gerenciamento e Controle - provê, para ferramentas e para os ambientes, serviços e funcionalidades que permitem o funcionamento de forma organizada ao longo do processo de desenvolvimento. Integração de Conhecimento - torna possíveis os serviços básicos de armazenamento, gerenciamento e utilização do conhecimento descrito e adquirido ao longo do processo de desenvolvimento.	Na camada de negócio, classes de controle implementadas como Session Beans interpretam, organizam e executam as operações nas camadas de modelo implementadas como Entities.	Os componentes e serviços da Estação Taba compartilham uma estrutura comum e são integrados de forma a trabalharem em conjunto e compõem os diferentes ambientes. A integração de conhecimento é feita com o uso de ontologias para derivar as classes de modelo. Conhecimento de processo é capturado e disponibilizado através de ferramentas.
Camada de Dados	Esta camada é responsável pela armazenamento e recuperação de informação de uma base de dados ou arquivo. A informação é passada para a camada lógica para processamento e, se for o caso, daí para o usuário final.	Integração de Dados - estabelece a forma como as ferramentas da Estação realizarão o tratamento das informações.	O mapeamento Objeto-Relacional é feito através do recurso de Annotations nos próprios Entities, de acordo com a especificação da JPA. A JPA traz recursos como a possibilidade de acesso simultâneo a diferentes bases em diferentes SGBDs de forma transparente para a camada de modelo. Assim, a tradução das operações realizadas sobre o modelo para comandos SQL e a execução destes comandos é feita automaticamente, de forma transparente e independente da implementação do SGBD.	As bases de dados dos ambientes corporativo, organização e projetos são armazenados em SGBD Oracle Relacional, mas a aplicação é totalmente independente de implementações específicas do Oracle, facilitando futuras migrações para outros SGBDs. O SGBD cuida do armazenamento e recuperação de informações mantendo um alto desempenho. Os dados podem ser acessados e alterados através da linguagem SQL.

Figura 5.21 – Detalhes de Implementação da Estação Taba na Web e dos AESCorp

As próximas subseções apresentam as principais características atuais de cada uma das estratégias de integração adotadas. Os componentes atuais são discutidos em conjunto com a Integração de Gerenciamento e Controle.

5.5.2.3 Integração de Gerenciamento e Controle

A Integração de Gerenciamento e Controle é obtida na Estação Taba em parte através do Serviços de Integração de Ferramentas Externas e Internas e em parte pelo compartilhamento e herança das características comuns do componente Ferramenta Interna por todas as ferramentas e componentes construídos como parte da infra-estrutura da Estação Taba. O Serviço de Integração de Ferramentas Externas é responsável pela integração de ferramentas externas aos ambientes, como, por exemplo, editores de textos, compiladores e ferramentas de uso geral necessárias para apoiar a execução das tarefas previstas nos diferentes ambientes e nos processos de engenharia de software a serem executados. O Serviço de Integração de Ferramentas Internas é responsável pela integração das ferramentas que utilizam a infra-estrutura geral da Estação Taba. Cada Ferramenta

Interna da Estação Taba, por sua vez, utiliza os serviços relativos à Integração de Conhecimento, Interface com Usuários e Acesso a Dados (estes serviços serão detalhados nas subseções seguintes).

Como regra geral as ferramentas internas da Estação Taba possuem uma arquitetura em três camadas (ECKERSON, 1995) (dados, lógica e apresentação) e uma estrutura seguindo o modelo MVC Modelo 2 (SINGH *et al.*, 2002)⁹. Com a reformulação da arquitetura, houve uma distinção entre o uso dos termos ferramentas e componentes. Dentro do novo contexto, um componente é responsável por implementar uma determinada funcionalidade enquanto as ferramentas são meras “abstrações lógicas” de funcionalidades que podem estar relacionadas a um ou vários componentes. Cada componente pode ser dividido em subcomponentes conforme adequado visando a separação de responsabilidades de cada parte e aumento da capacidade de reuso das funcionalidades que encapsula. Por exemplo, as ferramentas existentes para planejamento, gerência e monitoração de tempo, custo e recursos humanos foram agrupadas em um único componente, pois as tarefas apoiadas por estas ferramentas são muito dependentes entre si. De forma semelhante, as tarefas relacionadas à descrição da estrutura organizacional de uma organização foram agrupadas em um componente e este passou a ter comportamento diferenciado quando descrevendo uma organização ou corporação.

Os componentes/ferramentas foram agrupados de acordo com os diferentes tipos de ambientes da Estação Taba. O Meta-Ambiente é responsável por prover serviços através de componentes relacionados aos Gerenciadores de Definição de Processos, de Melhoria de Processos, de Ambientes, de Conhecimento e de Versão e Configuração, conforme pode ser visto na Figura 5.22.

⁹ O MVC (BUSCHMANN *et al.*, 1996) é um estilo arquitetural que visa separar o modelo de negócio, a interface com o usuário e a lógica de controle de uma aplicação em três componentes distintos, reduzindo o acoplamento da aplicação, de forma que alterações em um componente causem o mínimo impacto possível nos outros. As aplicações que adotam o MVC Modelo 2 (SINGH *et al.*, 2002) são mais fáceis de manter e estender porque a vista não acessa outras vistas diretamente, há um único ponto de controle que geralmente encapsula a entrada de dados para o modelo MVC que roda ao fundo.

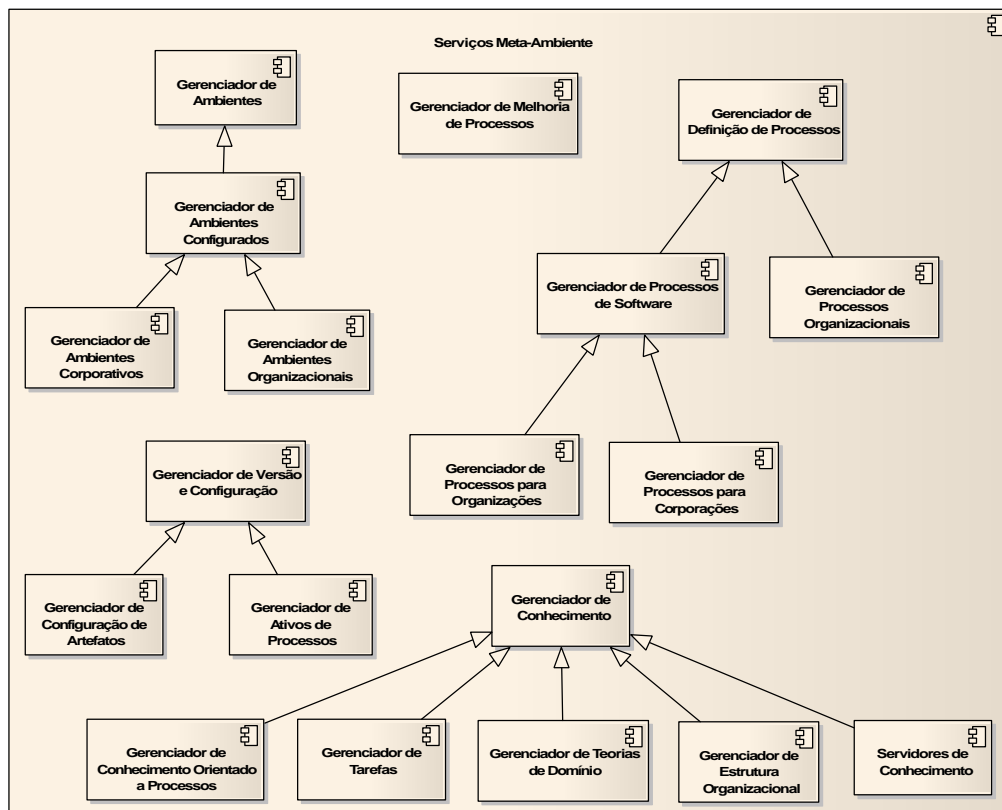


Figura 5.22 – Componentes do Meta-Ambiente

Os Gerenciadores de Ambientes presentes no Meta-Ambiente são responsáveis pela definição e manutenção dos Ambientes Organizacionais, o que engloba a ferramenta Config (VILLELA, 2004), o dos Ambientes Corporativos, através da nova funcionalidade construída decorrente desta tese.

O Gerenciador de Melhoria de Processos do Meta-Ambiente e dos Ambientes Organizacionais está sendo desenvolvido como parte de duas teses de doutorado (respectivamente, (CAMPOS, 2005) e (ALBUQUERQUE, 2008)). O Gerenciador de Processos Organizacionais é representado pela ferramenta ProcKnow (VILLELA, 2004), enquanto que a definição e manutenção dos Processos de Software para Organizações engloba a ferramenta Config (VILLELA, 2004) e a definição e manutenção dos Processos de Software para Corporações é provida por nova funcionalidade construída decorrente desta tese. A Gerência do Controle de Versão e Configuração de artefatos e dos ativos de processo é feita por dois outros componentes específicos, representados pelas ferramentas GConf (FIGUEIREDO *et al.*, 2004) e a Biblioteca de Ativos.

A Gerência de Conhecimento é de responsabilidade de cinco componentes distintos, o Gerenciador de Tarefas (que engloba a ferramenta EdiTar (ZLOT, 2002)), o Gerenciador de Teorias de Domínio (que engloba a ferramenta EdiTeD (OLIVEIRA,

1999)), os Servidores de Conhecimento (propostos por FALBO (1998)), o Gerenciador de Estrutura Organizacional (que engloba a ferramenta Sapiens (SANTOS, 2003)) e o Gerenciador de Conhecimento Orientado a Processos (que engloba a ferramenta ACKNOWLEDGE (MONTONI, 2003) e a Comunidade de Prática (MIRANDA, 2004)). Alguns dos serviços disponibilizados pelo Meta-Ambiente são utilizados ou especializados nos demais ambientes, de acordo com necessidades específicas.

Os Ambientes Corporativos e Organizacionais possuem muitas funcionalidades em comum que são adequadas, conforme o caso, para o âmbito corporativo e organizacional, respectivamente. Dessa forma, seus componentes foram agrupados em Serviços dos Ambientes Configurados e especializados conforme pertinente, como mostra a Figura 5.23.

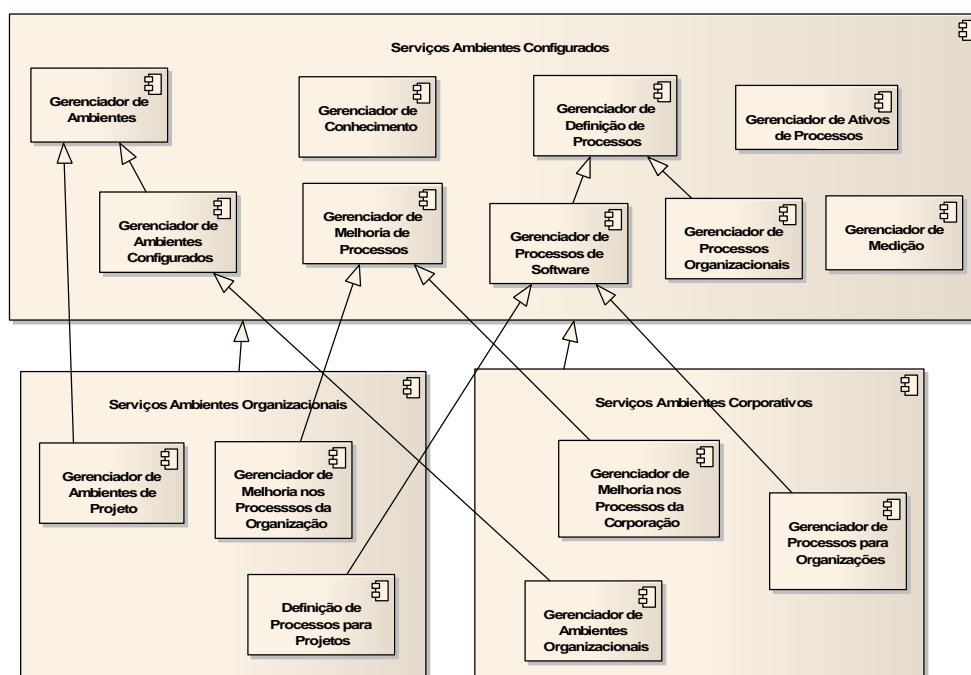


Figura 5.23 – Serviços dos Ambientes Corporativos e Organizacionais

Os Ambientes Configurados também provêem serviços específicos para a Gerência de Ativos de Processos, Definição de Processos de Software (para a Corporação, Organização ou de Projetos), Definição de Processos Organizacionais, Conhecimento, Melhoria de Processos (Corporativos e Organizacionais) e Ambientes (Corporativos, Organizacionais, de Projetos), além de serviços para Gerência de Medição (que engloba as ferramentas MedPlan e Metrics (SCHNAIDER *et al.*, 2004)).

Por fim, os Ambientes de Projeto possuem serviços diferenciados especializados para a execução e controle de processos, conforme pode ser visto na Figura 5.24.

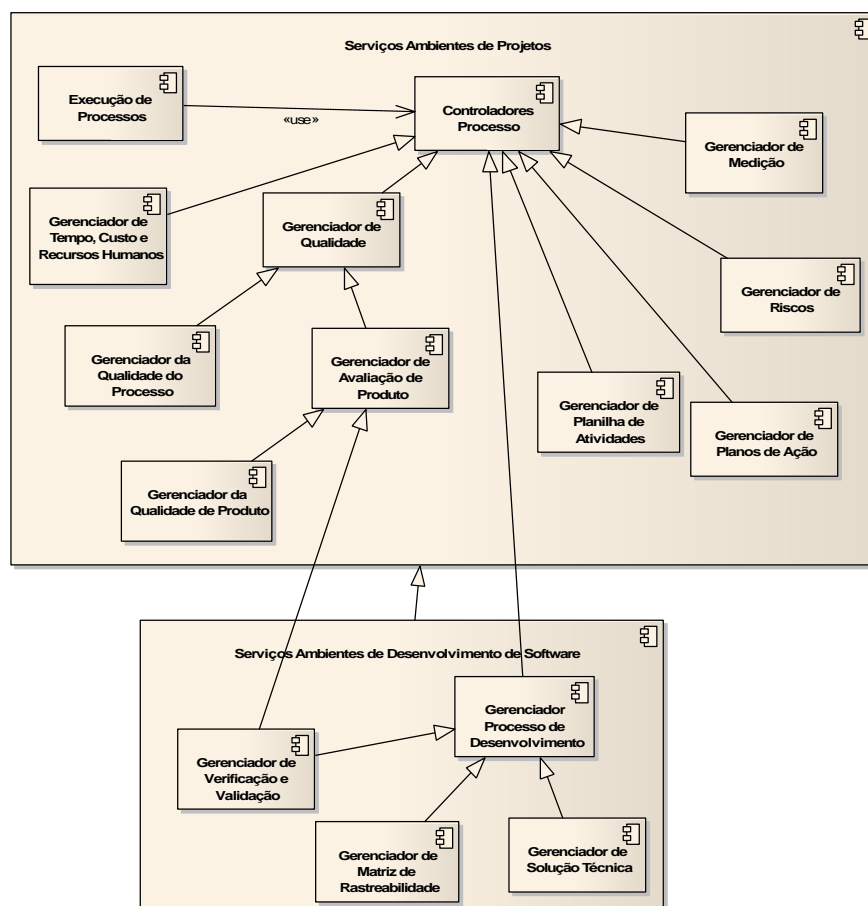


Figura 5.24 – Serviços dos Ambientes de Projetos

Os serviços gerais, que podem ser executados em projetos de todos os tipos, incluem além do componente de Execução de Processos, Controladores de Processos específicos para atividades de Gerência de Tempo, Custo e Recursos Humanos (representados pelas ferramentas TempManager, CustManager (BARCELLOS, 2003), RHManager (SCHNAIDER, 2003), ControlManager, OrgPlan), Gerência de Qualidade de Produto e do Processo, Gerência de Planilha de Atividades, Gerência de Medição, Gerência de Riscos (representado pela ferramenta RiscManager (FARIAS, 2002)) e Gerência de Planos de Ação. Além disso, ambientes para contextos específicos podem se beneficiar de componentes mais especializados, como, no caso de Ambientes de Desenvolvimento de Software, Gerência de Matriz de Rastreabilidade (representado pela ferramenta ReqManager), Gerência de Verificação e Validação (representado pelas ferramentas VerificationManager (BARRETO, 2006) e ValidationManager (NATALI, 2006)) e Gerência de Solução Técnica (representado pelas ferramentas MBR e TecSolution (FIGUEIREDO *et al.*, 2006)).

5.5.2.4 Integração de Apresentação e Interação

Com a reimplementação, um novo padrão de interface com os usuários foi definido para ser seguido por todas as ferramentas. Foram mantidos da versão atual da Estação Taba a orientação a processos e a integração com a ferramenta de gerência de conhecimento, como pode ser visto na Figura 5.25. As principais modificações foram feitas para adequação dos elementos da interface aos padrões comumente adotados na Web e em decorrência das observações da equipe de desenvolvimento sobre a utilização das ferramentas e ambientes na prática pela indústria.

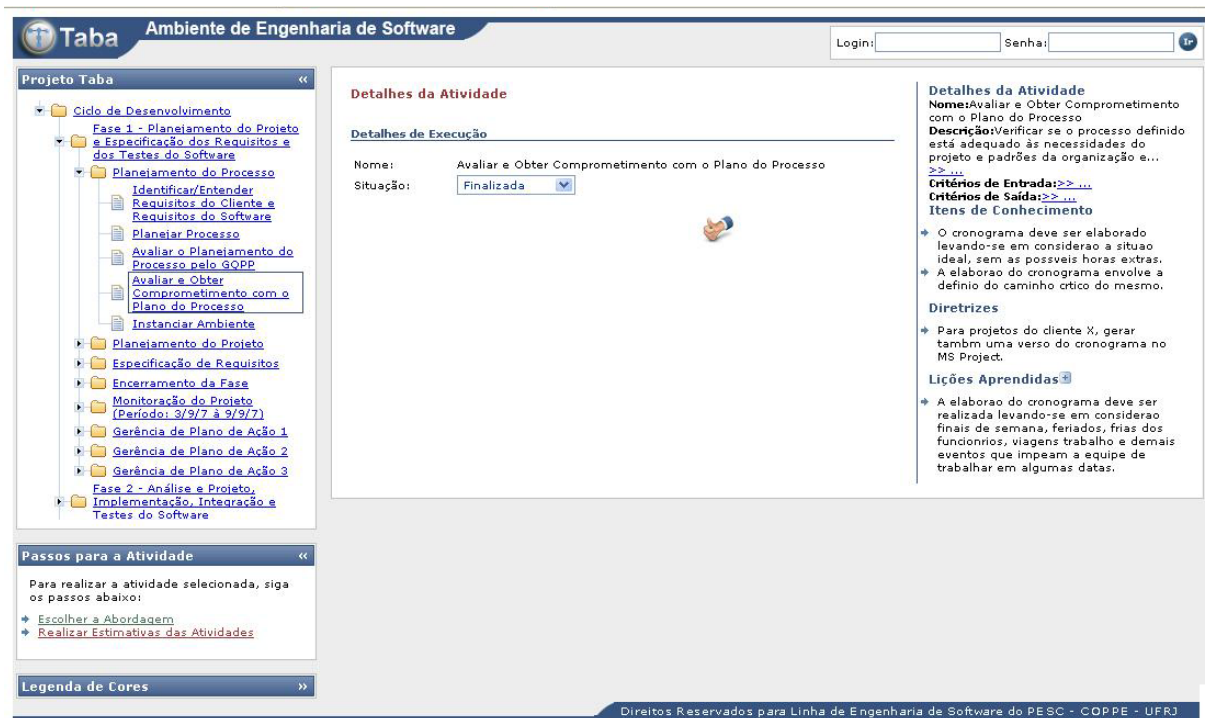


Figura 5.25 – Padronização de Interface

A orientação a processos pode ser vista na lateral direita da tela: é exibido o processo interno que guia a utilização da ferramenta ou, no caso de um ambiente de projeto, o processo de software utilizado durante a definição do ambiente. A integração com a ferramenta ACKNOWLEDGE pode ser vista na lateral direita da tela. O conhecimento associado à atividade sendo executada é exibido proativamente, não sendo mais, portanto, dependente da interação do usuário para ser exibido. A área central da figura contém, de fato, as funcionalidades de cada ferramenta. Para os elementos de tela necessários para a utilização das ferramentas foi utilizado uma biblioteca padrão de componentes do RichFaces (JBOSS, 2008) que garante a padronização destes elementos.

5.5.2.5 Integração de Dados – Mapeamento Objeto-Relacional na Estação Taba

O primeiro passo para possibilitar a criação da versão Web da Estação Taba foi feito ainda em 2004, com a adequação da estrutura da Estação Taba para trabalhar com repositórios contidos em bases de dados relacionais (anteriormente era utilizada uma base de dados em arquivo binário de formato proprietário próprio do MS Visual C++). Foi construída uma camada de reflexão computacional nas classes em C++, mecanismo não existente em modo nativo, para possibilitar a identificação das classes existentes e dos atributos e relacionamentos destas classes. Tanto as classes de modelo quanto as de controle da Estação Taba foram revistas e alteradas, o que envolveu a alteração de quase todo o código-fonte. Essa alteração permitiu, no entanto, que todas as ferramentas e ambientes construídos até então continuassem a serem utilizados com o mínimo de esforço para adequação da nova estrutura. Além disso, permitiu a criação de regras automáticas para verificação de regras para remoção de objetos, a criação de mecanismos para atualização da estrutura da base de dados devido à alteração dos atributos das classes e, também, possibilitou o uso de qualquer sistema gerenciador de banco de dados para o armazenamento dos dados manipulados pelas ferramentas e ambientes.

5.5.2.6 Integração de Dados – Repositórios dos Ambientes

As ferramentas da Estação Taba armazenam e utilizam os dados contidos em repositórios específicos para cada um dos tipos de ambientes disponíveis. Dessa forma, há três tipos de repositórios disponíveis para os AESCorp:

- (i) *Repositório da Corporação*: armazena informações sobre a corporação e conhecimento de interesse de toda a corporação e é parcialmente alimentado por informações coletadas pelas organizações que constituem a corporação e pelos projetos executados ou em execução. Estas informações incluem os dados e/ou conhecimentos que sejam considerados de interesse da corporação, por exemplo: dados/conhecimento referente às organizações que compõem a corporação e também das organizações clientes e/ou parceiras da corporação; informações disponíveis sobre a execução de projetos e de seus processos associados e que sejam relevantes para a corporação; informações sobre o desempenho de projetos e aderência a modelos de maturidade e normas nacionais e internacionais; conhecimento especializado sobre Engenharia de Software em geral além de conteúdo relativo a engenharia de software específico para corporação, como, por exemplo, informações sobre boas práticas e diretrizes a serem adotadas.

- (ii) *Repositório da Organização*: armazena informações sobre a organização (por exemplo, sua estrutura, dados de projetos realizados, informações sobre clientes e parceiros) e é parcialmente alimentado por informações coletadas ao longo da execução de seus projetos. Estas informações incluem: a descrição de tarefas genéricas, comuns a diferentes domínios e organizações, com são, por exemplo, as tarefas de reservar e diagnosticar; teorias de domínio utilizadas para organizar o conhecimento sobre os domínios de negócio, utilizando ontologias e identificando tarefas genéricas que são executadas no contexto dos domínios; dados sobre a organização que utiliza o ambiente e as organizações clientes e parceiras, estabelecendo, dentro do contexto da estrutura e dos processos organizacionais, quais as tarefas genéricas que são executadas e o conhecimento de Engenharia de Software que é requerido; o conhecimento e os dados relevantes para a organização que foram obtidos ao longo de seus vários projetos de software; os ativos de processo que sejam considerados relevantes para a organização.
- (iii) *Repositório do Projeto*: armazena todas as informações disponíveis sobre a execução de um projeto e coletadas através das diferentes ferramentas que compõem os ambientes instanciados pela Estação Taba. Estas informações incluem, por exemplo, dados sobre cronograma e custos de projetos, riscos mais comumente identificados, resultados de avaliações *post mortem*.

A Tabela 5.7 sintetiza os repositórios presentes nos AESCorp.

Tabela 5.7 – Repositórios Presentes nos AESCorp

Repositório	Armazena Informações Sobre	Fonte de Alimentação
Projeto	Execução de um projeto (dados sobre custos, riscos, alocação de pessoal, avaliações <i>post mortem</i> etc.).	Ferramentas dos ambientes instanciados.
Organização	A organização (estrutura organizacional, dados de projetos, dados sobre clientes e parceiros, ativos de processo etc.).	Ferramentas dos ambientes organizacionais. Informações coletadas ao longo da execução de projetos.
Corporação	A corporação e conhecimento de interesse de toda a corporação (ativos de processo, o desempenho de projetos, aderência a modelos de maturidade e normas internacionais etc.).	Ferramentas do ambiente corporativo. Informações coletadas nos ambientes organizacionais e ao longo da execução de projetos.

5.5.2.7 Integração de Conhecimento

Parte da Integração de Conhecimento na Estação Taba é feita hoje através do uso de Servidores de Conhecimento, propostos por FALBO (1998), como forma de integração do conhecimento definido através de ontologias em ambientes de desenvolvimento de software. A arquitetura de Servidor do Conhecimento corresponde, basicamente, à definição de módulos de conhecimento que representam ontologias e suas instanciações e

que se tornam disponíveis para o compartilhamento e reuso do conhecimento. A utilização das ontologias visa minimizar ambigüidades na comunicação entre os agentes computacionais e humanos envolvidos no contexto da Estação Taba e dos Ambientes de Engenharia de Software Orientados à Corporação. Atualmente há servidores de domínio relacionados a Conhecimento de Processo (derivados da ontologia de processo de software definida por FALBO (1998) e evoluída por VILLELA (2004)) e Conhecimento Organizacional (definida por VILLELA (2004)).

Outra forma de utilização do conhecimento ontológico na Estação Taba está na derivação inicial de modelos de classes (e conseqüente modelo de dados) para representar conceitos já mapeados em ontologias. Os conceitos e relações entre conceitos descritos na ontologia são utilizados para derivar uma versão inicial das classes e estas classes mantêm uma relação com o conceito de onde foram originados. Ferramentas como a Sapiens (responsável pela descrição da estrutura organizacional e das competências requeridas ao longo desta estrutura e também as possuídas pelas pessoas) utilizam esta relação entre o conceito ontológico e as classes de implementação para derivar novas relações e auxiliar na busca por informações (SANTOS, 2003; SANTOS *et al.*, 2004). O uso das ontologias para derivar o modelo de classes pode ser visto também em virtualmente quase todas as ferramentas da Estação Taba que utilizam ou manipulam processos, pois tais classes foram derivadas da estrutura definida por uma ontologia.

5.5.3 Avaliação do Modelo de Integração da Estação Taba

Atualmente, um conjunto de ferramentas está sendo revisto e reimplementado visando à definição de ambientes de apoio às atividades corporativas (como definição e gerência de ambientes de apoio às atividades organizacionais relacionadas a software) e também de apoio à execução de projetos baseados em processos de software (como projetos de desenvolvimento, manutenção, melhoria, medição etc.). Ao longo do tempo muitas ferramentas foram construídas utilizando a infra-estrutura anterior da Estação Taba. Construir novas versões dessas ferramentas para a infra-estrutura atual é um trabalho demorado, por isso, o conjunto de componentes e serviços apresentados é em parte representado por ferramentas já existentes na plataforma anterior. Está previsto a reimplementação de todas as ferramentas na infra-estrutura atual, mas, enquanto isso não acontece, as ferramentas novas e antigas serão executadas concomitantemente de forma complementar.

Apesar dos 14 anos desde a primeira implementação, o modelo de integração utilizado manteve-se constante e pertinente, apenas a tecnologia utilizada para tratar cada dimensão foi atualizada. Esta constatação, por si só, constitui uma importante prova de conceito de que a proposição inicial do modelo de integração estava correta e responde por parte da longevidade e atualidade dos Ambientes Taba. Além disso, o modelo de ciclo de vida para ADSs proposto por BROWN (1992), constituído pela Estruturação, População, Adaptação e Uso dos ADSs, é de difícil avaliação por ser naturalmente longo e dependente de uso em campo e, por isso, tem-se pouca ou nenhuma evidência da sua avaliação ao longo das pesquisas relacionadas a ambientes. Dessa forma, os resultados obtidos pela utilização dos Ambientes Taba em organizações no apoio a implantação de melhoria de processos de software ao longo do tempo e suas constantes evoluções demonstram, também, suas robustez, aplicabilidade e pertinência ao contexto em que são aplicados.

5.6 Modelo de Desenvolvimento da Estação Taba

Não só a definição de uma arquitetura para o desenvolvimento da Estação Taba na Web é suficiente para facilitar a criação de novas ferramentas para conter a sua infraestrutura. Da mesma forma que a Estação Taba é utilizada para apoiar a implantação de processos de software em organizações, o seu desenvolvimento também pode se beneficiar da existência de um processo de software para a sua construção.

5.6.1 Processo do Escritório de Projetos da Estação Taba

Dessa forma, foi elaborado um conjunto de processos-padrão visando suprir todo o ciclo de desenvolvimento de uma nova funcionalidade da Estação Taba. A esse conjunto de processos padrão deu-se o nome de Processo do Escritório de Projetos da Estação Taba, como pode ser visto Figura 5.26. Este escritório representa a estrutura da equipe de desenvolvimento da Estação Taba, que é composta por um responsável pelas alterações na arquitetura e decisões relacionadas à disponibilização de diferentes versões e atualizações aos usuários e, também, por definir as diretrizes gerais do desenvolvimento, outros responsáveis por processos específicos relacionados ao ciclo de vida do produto (por exemplo, responsável pelas atividades de Garantia da Qualidade do Processo e do Produto ou, então, pelas atividades relacionadas a Medição) e a própria equipe de desenvolvimento composta por gerentes de projetos, analistas de sistemas, desenvolvedores e alunos de graduação, mestrado e doutorado (que podem assumir quaisquer dos papéis da equipe de

desenvolvimento) durante a construção das ferramentas previstas em seus trabalhos acadêmicos.

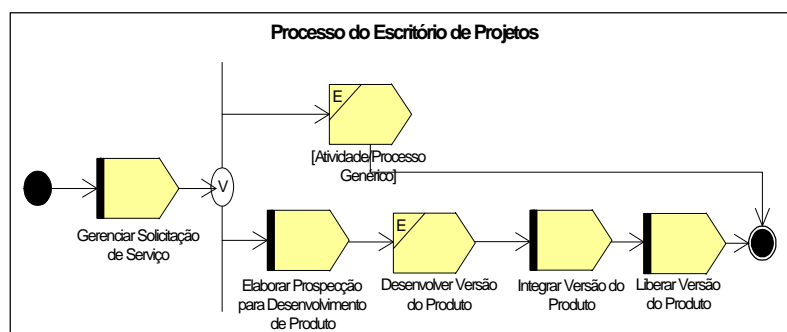


Figura 5.26 – Visão Geral do Processo do Escritório de Projetos

Um dos efeitos colaterais de uma equipe reduzida para o desenvolvimento da Estação Taba é a realização de várias atividades por um uma mesma pessoa. Isso também foi determinante para a definição de um processo para o Escritório de Projetos da Estação Taba que possibilitasse a execução de um grande número de tarefas específicas (como, por exemplo, criação de nova funcionalidade, manutenção de produto específico, adaptação de processos a uma empresa específica, disponibilização de versão a usuários, prospecção para futuro desenvolvimento de funcionalidade etc.) e não apenas o desenvolvimento ou manutenção de software.

Devido a isso, o Processo do Escritório de Projetos da Estação Taba prevê a execução de uma tarefa genérica denominada Solicitação de Serviço. Para a construção dos AESCorp foi utilizada uma solicitação de serviço que englobava a prospecção para o desenvolvimento, o desenvolvimento de uma nova versão, a integração e liberação de uma versão da Estação Taba. A prospecção para o desenvolvimento é responsável pelo levantamento de requisitos inicial que dá origem a um plano de desenvolvimento que prevê o desenvolvimento destes requisitos em uma ou várias versões que são integradas conforme apropriado e, então, liberadas para o usuário final. A existência desses passos garante à equipe de desenvolvimento um planejamento de, por exemplo, quais requisitos serão implementados em decorrência de um trabalho acadêmico e quais podem (ou deveriam) ser implementados apenas se a ferramenta precisar ser utilizada por empresas.

5.6.2 Processo de Desenvolvimento

A implementação dos requisitos identificados durante a prospecção é feita seguindo-se um processo específico de desenvolvimento. O desenvolvimento da Estação Taba possui algumas características peculiares que devem ser respeitadas e levadas em

consideração para a definição de seus processos de apoio. Dentre elas está o fato de a equipe de desenvolvimento ser reduzida em relação ao tamanho do produto desenvolvido (atualmente são cerca de 400.000 linhas de código em C++) e, muitas vezes, pouco experiente na linguagem (fato derivado de muitas ferramentas surgirem como produtos de trabalhos acadêmicos e terem que ser implementadas por pessoas com pouco conhecimento sobre o desenvolvimento na Estação Taba). Devido a isto foi feita a opção pela definição de um processo de desenvolvimento que utilizou a experiência do Grupo de Engenharia de Software na definição e implantação de processos em empresas para definir um processo que procurasse ser leve (não confundir com utilização de métodos ágeis) o suficiente para conter os formalismos comuns a processos de software, mas procurando não criar barreiras que dificultassem o desenvolvimento (como atividades muito burocráticas ou que impeçam a continuidade de outras fora do caminho crítico). Atualmente o processo de desenvolvimento é compatível com o Nível E do MR-MPS (SOFTEX, 2007) e é composto por três fases¹⁰ cujos escopos são, respectivamente: (i) Planejamento do Projeto e Especificação dos Requisitos e dos Testes do Software; (ii) Análise e Projeto, Implementação, Integração e Testes do Software; e (iii) Documentação, Homologação e Encerramento do Projeto.

5.6.3 Estratégia de Versionamento

Durante o desenvolvimento da Estação Taba é comum a necessidade de haver diferentes versões em desenvolvimento ao mesmo tempo. Não é adequado, por exemplo, que o desenvolvimento de novas ferramentas atrapalhe a geração de versões de manutenção ou que o desenvolvimento de uma nova ferramenta atrapalhe o desenvolvimento de outra que esteja ocorrendo ao mesmo tempo. Outro cenário comum é o plano de desenvolvimento derivado de uma prospecção prever a geração de várias versões intermediárias até a liberação para o usuário final. Neste meio tempo atualizações importantes devido a manutenção do código podem necessitar de ser disponibilizadas para as versões ainda em andamento.

Devido a isso, foi definida uma Estratégia de Versionamento que prevê a utilização de uma ferramenta de controle de versão (geralmente Subversion (SVN, 2008)) para armazenamento e gerência de código fonte que dê suporte ao desenvolvimento em paralelo

¹⁰ Outros processos, visando uma avaliação formal no MPS.BR, como o de Medição, Reuso, Garantia da Qualidade, Definição de Processos também foram definidos mas fogem do escopo da construção do software em si e, por isso, não serão descritos neste trabalho.

através de ramos (*branches*) independentes que são, quando pertinente, integrados e, se for o caso, liberados para os usuários.

Para o desenvolvimento da Estação Taba na Web, todas as ferramentas possuem testes automatizados que são executados (e devem ser considerados satisfeitos) antes que o código seja considerado efetivamente integrado e próprio para liberação interna para a equipe de desenvolvimento ou para usuários finais das ferramentas e ambientes da Estação Taba. O planejamento e construção destes testes são garantidos pelas atividades específicas do processo de desenvolvimento utilizado.

5.6.4 Ferramentas de Apoio

Como apoio às atividades previstas no Processo do Escritório de Projeto da Estação Taba, Processo de Desenvolvimento e Estratégia de Versionamento é utilizado um conjunto de ferramentas composto pela Estação Taba (responsável pelo apoio à gerência e execução dos projetos de desenvolvimento, definição e melhoria de processos, medição, gerência de conhecimento e gerência de ativos de processos), Enterprise Architect (responsável pelo apoio à modelagem e gerência de requisitos, inerentes às atividades de engenharia do processo de desenvolvimento e de prospecção), Bugzilla (responsável pelo apoio à gerência dos itens que irão compor versões de manutenção da Estação Taba e também para apoiar a identificação de melhorias nos processos e a gerência de ações corretivas) e Subversion (para controle de versão de código fonte).

5.7 Modelo para Construção dos Ambientes de Engenharia de Software Orientados a Corporação na Estação Taba

A partir dos requisitos dos Ambientes de Engenharia de Software (sumarizados na seção 4.2.6), dos Ambientes Orientados a Corporação (sumarizados na seção 4.3.7) e da Estação Taba (sumarizados na seção 5.3), procurou-se, com base no Modelo para Construção de Ambientes de Engenharia de Software Orientados a Corporação (proposto na seção 4.4), identificar os principais itens que deveriam compor o núcleo básico dos AESCorp. Este núcleo básico envolve componentes (conforme estrutura discutida na seção 5.5.2) relacionados à gerência de ambientes, processos, ativos de processos, estrutura organizacional e execução de processos, que serão discutidas nas subseções a seguir.

A implementação dos Ambientes de Engenharia de Software Orientados a Corporação na Estação Taba utilizou boa parte da infra-estrutura já existente. Esta infra-estrutura vem sendo construída desde a primeira implementação em C++, iniciada por

OLIVEIRA (1999)¹¹ com a definição dos Ambientes de Desenvolvimento de Software Orientados a Domínio (ADSOD). A maior parte das ferramentas, no entanto, foi elaborada a partir da definição dos Ambientes de Desenvolvimento de Software Orientados a Organização (ADSOrg), realizada por VILLELA (2004)¹². Assim, nestes nove anos muitos trabalhos de alto interesse prático foram realizados no contexto de ambientes de desenvolvimento de software. No entanto, de forma semelhante ao que ocorreu quando houve a transição da implementação na linguagem Eiffel para o C++, não seria possível, de imediato, re-implementar o resultado de todos esses projetos. Decidimos, então, re-implementar na nova infra-estrutura apenas as funcionalidades básicas que deveriam ser providas pelos ambientes: a definição e configuração de Ambientes Corporativos e Organizacionais e a execução de processos a partir dos Ambientes de Projeto. No entanto, diferentemente do que aconteceu no passado, as duas versões (em Java e em C++) passam a co-existir: ferramentas podem ser acessadas, de forma integrada, pelos novos ambientes.

Para ilustrar a implementação realizada será apresentado um conjunto de diagramas utilizado para a definição de requisitos e funcionalidades a serem apoiados por cada ferramenta. Não é objetivo, neste capítulo ou nesta tese, descrever toda modelagem ou implementação realizada a fundo. Dessa forma, serão exibidos apenas os principais elementos do levantamento de requisitos realizado. O Anexo V apresenta mais detalhes do levantamento de requisitos realizado seguindo os modelos de documentos do processo de desenvolvimento da Estação Taba (descrito na seção 5.6). Para o diagramas de requisitos apresentados neste capítulo será utilizada a notação apresentada na Figura 5.27, derivada da UML.

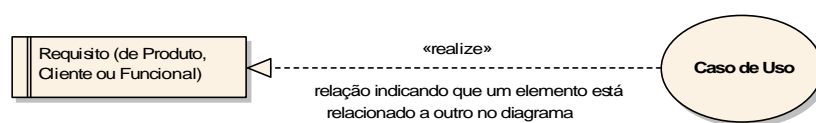


Figura 5.27 – Notação dos Diagramas de Requisitos

5.7.1 Gerência de Ambientes

A Figura 5.28 exibe o diagrama de requisitos da Gerência de Ambientes Configurados.

¹¹ Parte desta implementação foi realizada com a ajuda de SANTOS e ZLOT (1999).

¹² Desde antes da definição dos ADSOrg estive envolvido diretamente com a definição e implementação das ferramentas por ser responsável pela gerência das atividades de desenvolvimento da Estação Taba.

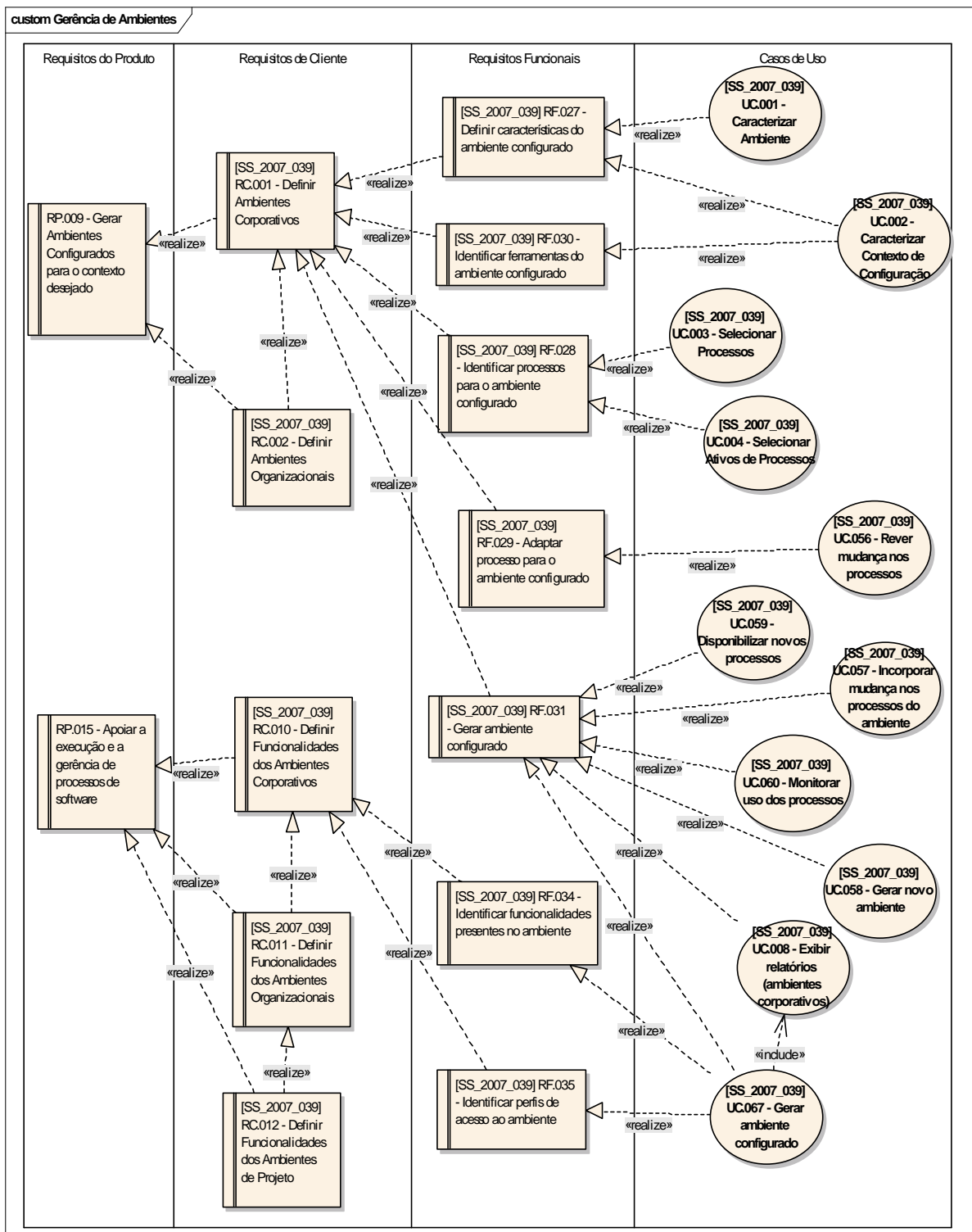


Figura 5.28 – Definição de Ambientes Configurados – Requisitos e Casos de Uso

Na primeira coluna são exibidos os Requisitos do Produto (identificados na Tabela 2.3) relacionados à Gerência de Ambientes. A partir dos Requisitos do Produto foram derivados Requisitos de Cliente (que podem ser vistos na segunda coluna) e a partir destes, Requisitos Funcionais (que podem ser vistos na terceira coluna). Por fim, são exibidos os

Casos de Uso que foram derivados dos Requisitos Funcionais (o que pode ser visto na quarta e última coluna).

De forma similar à Figura 5.28, a Figura 5.29 mostra o diagrama de requisitos referentes à Gerência de Ambientes de Projeto.

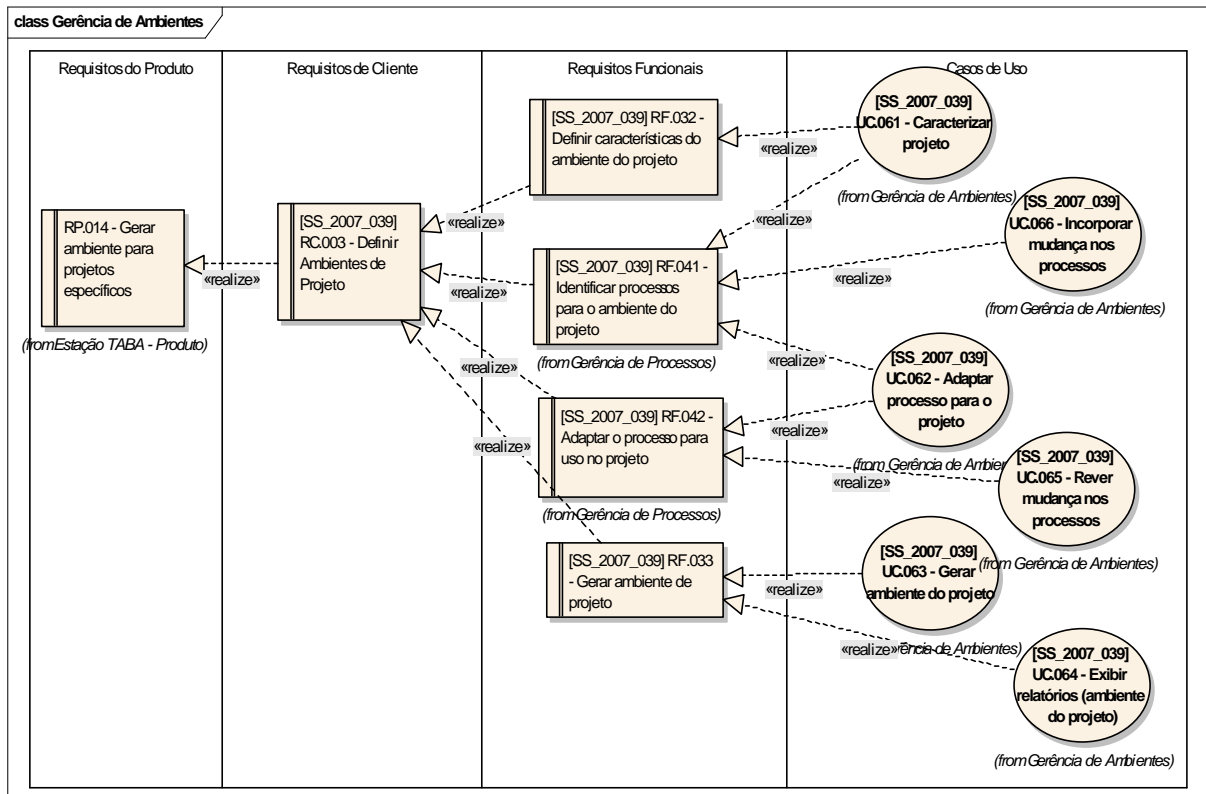


Figura 5.29 – Definição de Ambientes de Projeto – Requisitos e Casos de Uso

As ferramentas da Estação Taba são, assim como seus ambientes, orientadas a processo. Assim, é necessário que sejam definidos processos de apoio a serem utilizadas por cada uma das ferramentas. Dessa forma, a partir destes casos de uso foram gerados cinco diferentes processos: Definir Ambientes Configurados (que atende pelos Ambientes Corporativos e Organizacionais), Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais, Definir Ambientes de Projetos e Alterar Ambientes de Projetos. A Figura 5.30 exibe o processo Definir Ambientes Configurados (os detalhes da notação utilizada podem ser vistos no Anexo I). Cada atividade pode estar associada a um ou mais casos de uso e representam, de fato, uma tela da ferramenta a ser construída.

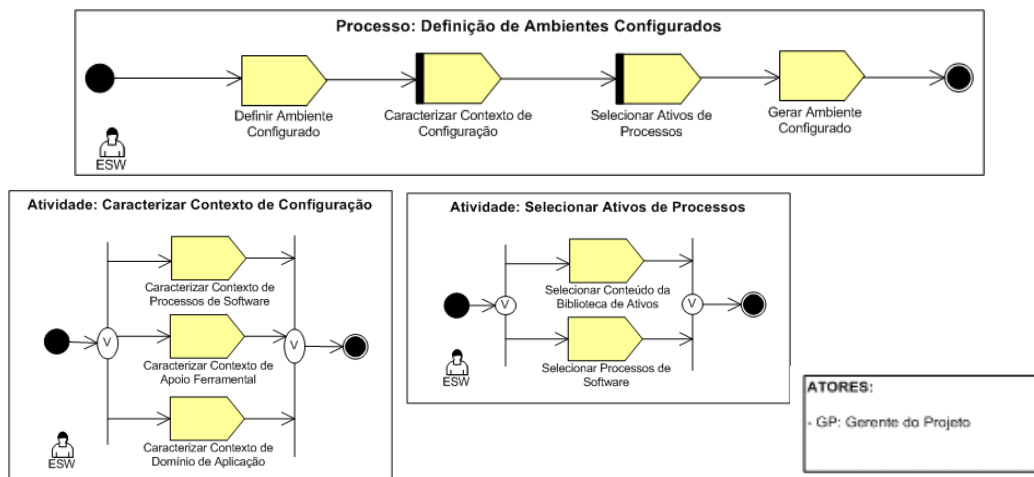


Figura 5.30 – Definição de Ambiente Configurado – Processo de Apoio

Com base nesta estrutura, foi definida a ferramenta para definição de Ambientes Configurados, como pode ser visto na Figura 5.31. Pode-se perceber ao lado esquerdo da tela o processo de apoio da ferramenta (descrito na Figura 5.30). A tela sendo exibida corresponde à atividade “Definir Ambiente Configurado”. Mais detalhes desta ferramenta poderão ser vistos na seção 5.8.



Figura 5.31 – Definição de Ambiente Configurado – Definir Ambiente Configurado

A Figura 5.32 exibe o processo Definir Ambientes de Projeto.

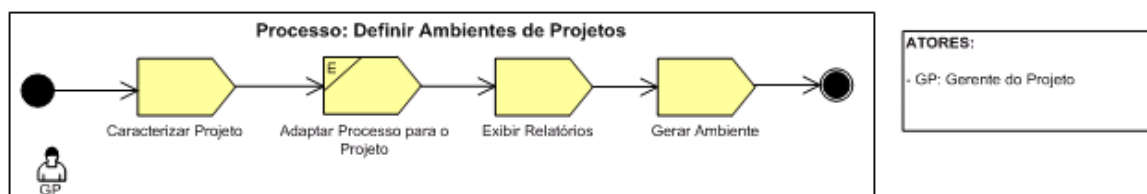


Figura 5.32 – Definição de Ambiente de Projeto – Processo de Apoio

Este processo é compatível com a ferramenta AdaptPro já existente para a definição de Ambientes de Projeto existente na Estação Taba (BERGER, 2003). Esta

ferramenta é responsável não só pela definição do Ambiente do Projeto como também a adaptação do processo padrão para o uso no projeto. Não foi feita uma reimplantação dessa ferramenta na nova infra-estrutura. Preferiu-se, neste momento, apenas adaptá-la para (i) gerar Ambientes de Projeto para quaisquer tipos de processos que a organização possuir (por exemplo, para apoiar atividades de Medição, Garantia da Qualidade ou Aquisição) e não apenas os processos de Desenvolvimento e Manutenção; e (ii) flexibilizar a utilização dos diferentes critérios de especialização do processo padrão durante a adaptação do processo ao projeto. Mais detalhes desta ferramenta poderão ser vistos na seção 5.8.

Os processos Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais não foram implementados e, por isso, não são descritos neste capítulo. Para maiores detalhes sobre as características de tais funcionalidades deve-se consultar o Anexo V. O processo Alterar Ambientes de Projetos é implementado através da disponibilização da ferramenta AdaptPro também nos Ambientes de Projeto. Nestes ambientes sua funcionalidade é limitada, não permitindo a geração de novos ambientes, apenas a edição do processo do projeto em andamento.

5.7.2 Gerência de Processos

A Figura 5.33 mostra o diagrama de requisitos referentes à Gerência de Processos Padrão. Na primeira coluna são exibidos os Requisitos do Produto (identificados na Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes) relacionados à Gerência de Ambientes. A partir dos Requisitos do Produto foram derivados Requisitos de Cliente (que podem ser vistos na segunda coluna) e a partir destes, Requisitos Funcionais (que podem ser vistos na terceira coluna). Por fim, são exibidos os Casos de Uso que foram derivados dos Requisitos Funcionais (o que pode ser visto na quarta e última coluna).

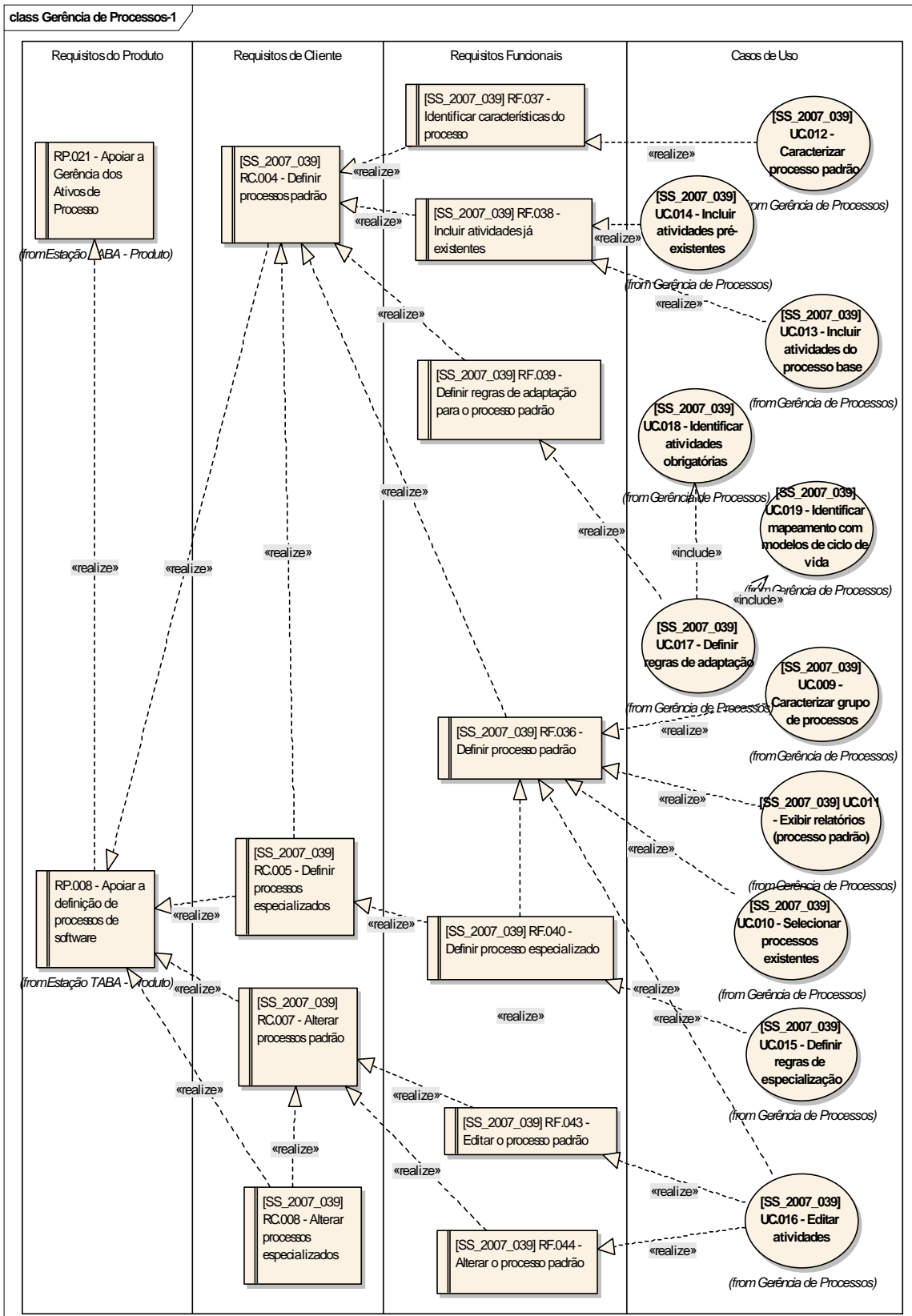


Figura 5.33 – Definição de Processos Padrão e Especializados – Requisitos e Casos de Uso

De forma similar à Figura 5.33, a Figura 5.34 mostra o diagrama de requisitos referentes à Gerência de Processos Definidos para os Projetos. Detalhes dos processos derivados destes requisitos podem ser vistos no Anexo V.

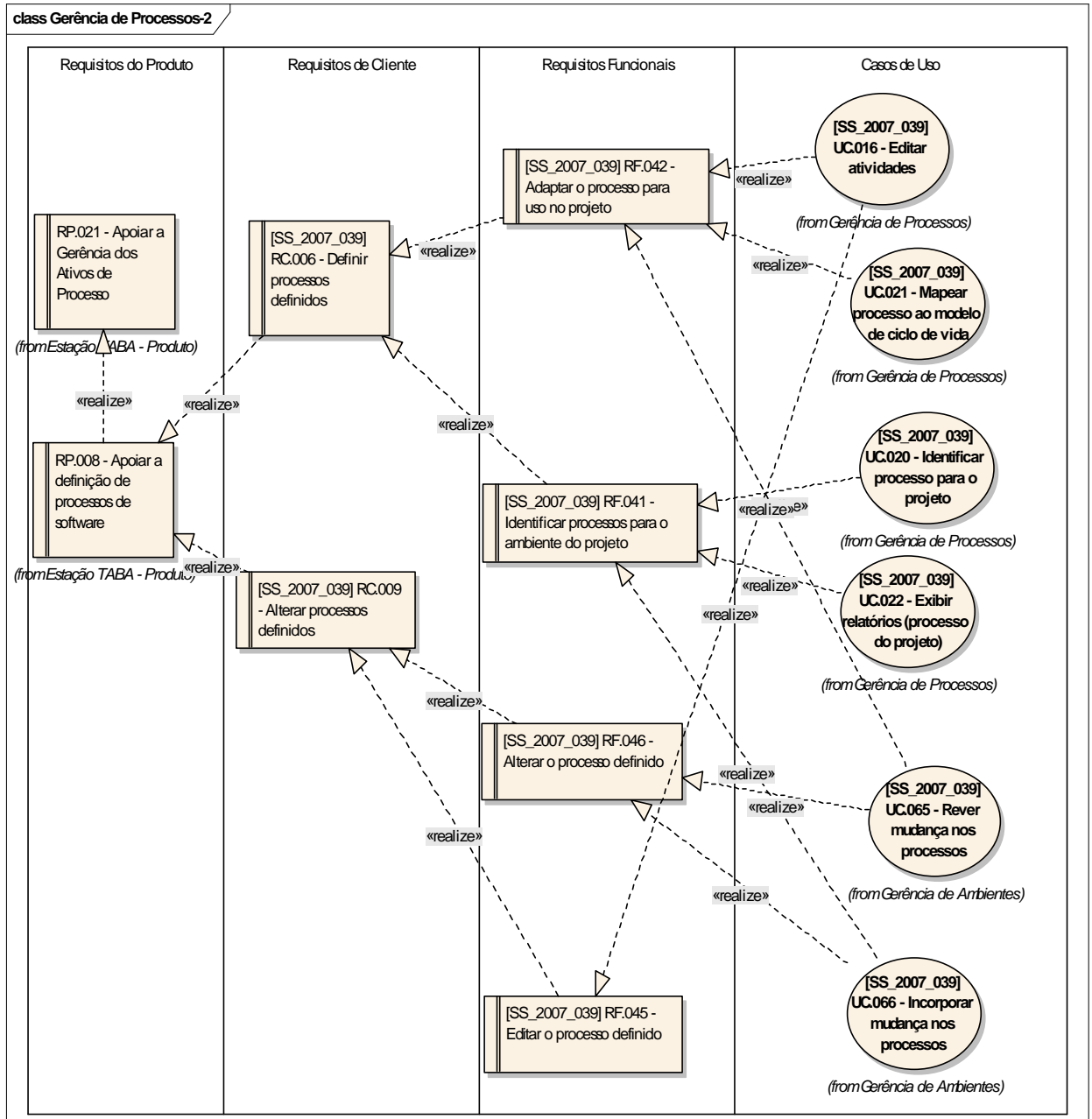


Figura 5.34 – Definição de Processos Definidos para o Projeto – Requisitos e Casos de Uso

A definição de Processos Padrão e Especializados na Estação Taba era feita em conjunto com a definição do Ambiente Configurado Organizacional por meio da ferramenta Config (VILLELA, 2004). Com a criação de um novo configurador de

ambientes, as funcionalidades para definição destes tipos de processos foi desmembrada e passou a compor uma nova ferramenta específica para este fim. Uma alteração nas funcionalidades já existentes foi feita para permitir a flexibilização das regras de especialização de processos. A partir de agora é possível definir diferentes critérios de especialização (por exemplo, tamanho do projeto e tipo de software). Antes a especialização era feita automaticamente apenas para o paradigma de desenvolvimento de software. Mais detalhes desta ferramenta poderão ser vistos na seção 5.8.

A definição do Processo Definido para o Projeto é feita em conjunto com a definição do Ambiente do Projeto (como descrito na seção 5.7.1) e, assim, não será descrita novamente.

5.7.3 Gerência de Ativos de Processos

A Figura 5.35 mostra o diagrama de requisitos referentes à Gerência de Ativos de Processo. Na primeira coluna são exibidos os Requisitos do Produto (identificados na Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes) relacionados à Gerência de Ambientes. A partir dos Requisitos do Produto foram derivados Requisitos de Cliente (que podem ser vistos na segunda coluna) e a partir destes, Requisitos Funcionais (que podem ser vistos na terceira coluna). Por fim, são exibidos os Casos de Uso que foram derivados dos Requisitos Funcionais (o que pode ser visto na quarta e última coluna).

De forma similar à Figura 5.35, a Figura 5.36 mostra o diagrama de requisitos referentes à Gerência de Versão de Ativos de Processos. Detalhes dos processos derivados destes requisitos podem ser vistos no Anexo V.

A definição e gerência da biblioteca de ativos de processos da Estação Taba são feitas através de uma ferramenta (SANTOS *et al.*, 2008a) disponibilizada nos Ambientes Configurados integrada à ferramenta de Gerência de Configuração (FIGUEIREDO *et al.*, 2004). Mais detalhes destas ferramentas poderão ser vistos na seção 5.8.

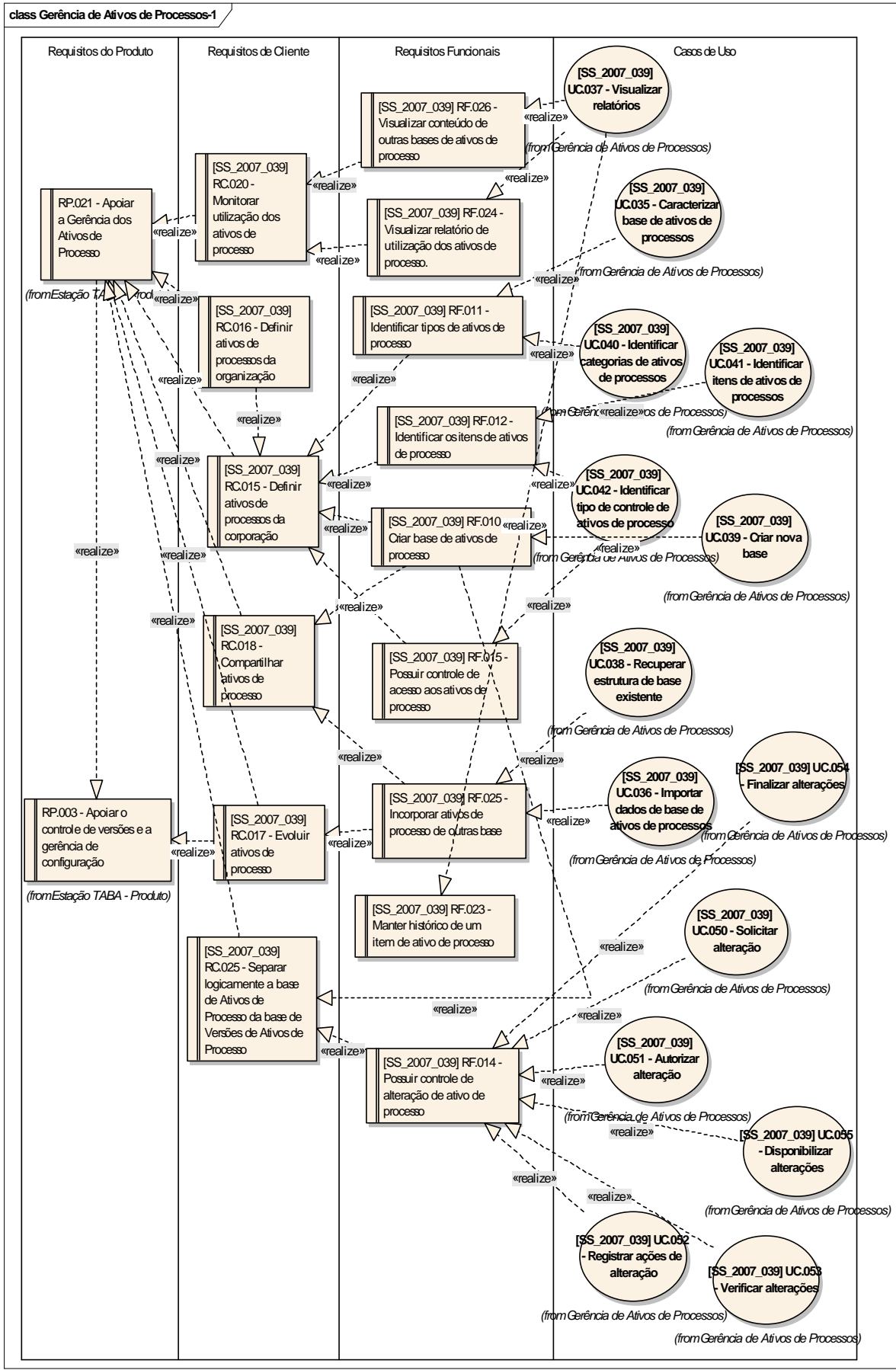


Figura 5.35 – Biblioteca de Ativos de Processos – Requisitos e Casos de Uso

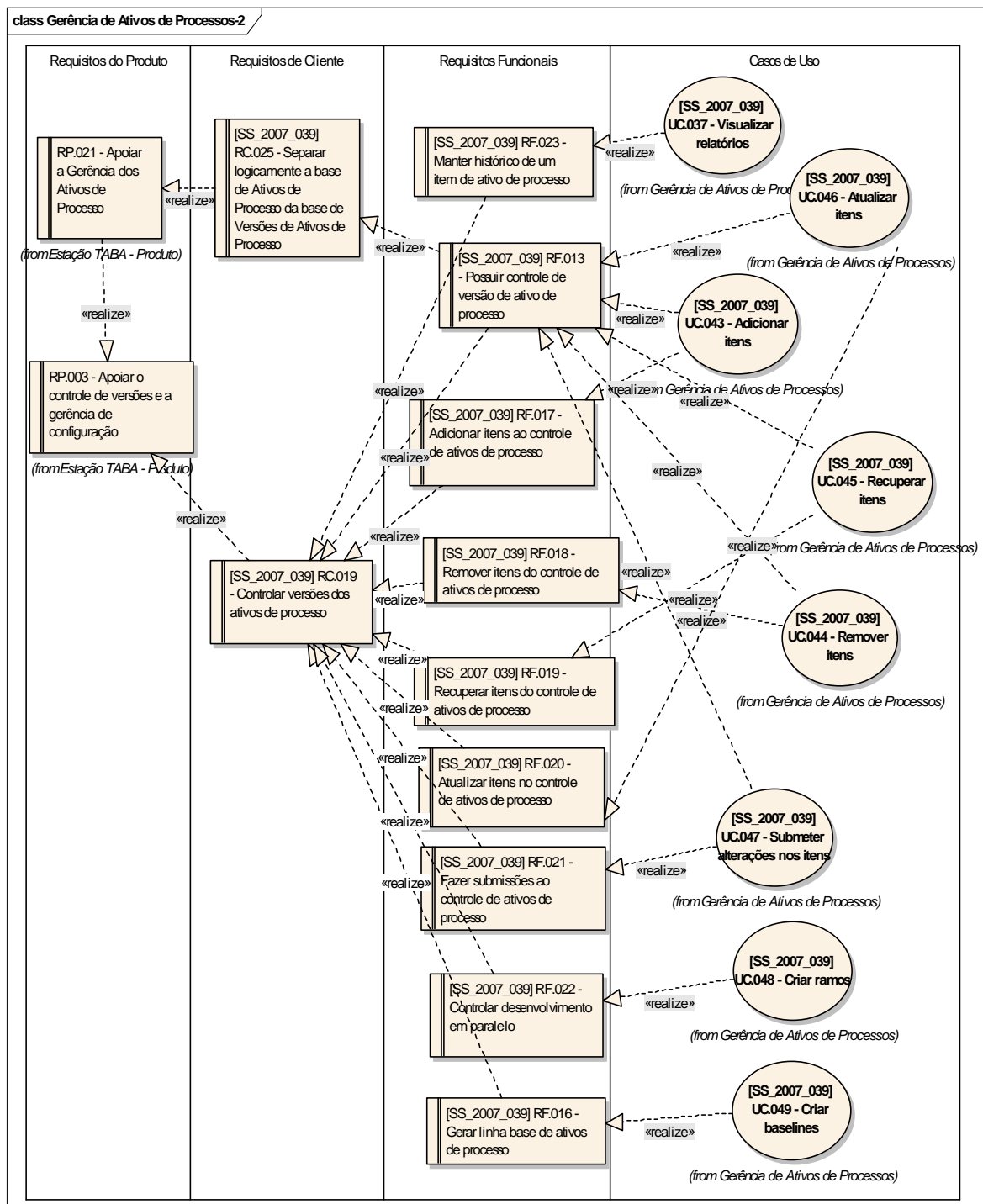


Figura 5.36 – Gerência de Versão de Ativos de Processos – Requisitos e Casos de Uso

5.7.4 Gerência de Execução de Processos

A Figura 5.37 mostra o diagrama de requisitos referentes à Gerência de Execução de Processos. Na primeira coluna são exibidos os Requisitos do Produto (identificados na Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes) relacionados à Gerência de Ambientes. A partir dos Requisitos do Produto foram derivados Requisitos de Cliente (que podem ser vistos na segunda coluna) e a partir destes, Requisitos Funcionais (que podem

ser vistos na terceira coluna). Por fim, são exibidos os Casos de Uso que foram derivados dos Requisitos Funcionais (o que pode ser visto na quarta e última coluna). Detalhes do apoio ferramental para a execução de processos poderão ser vistos na seção 5.8.

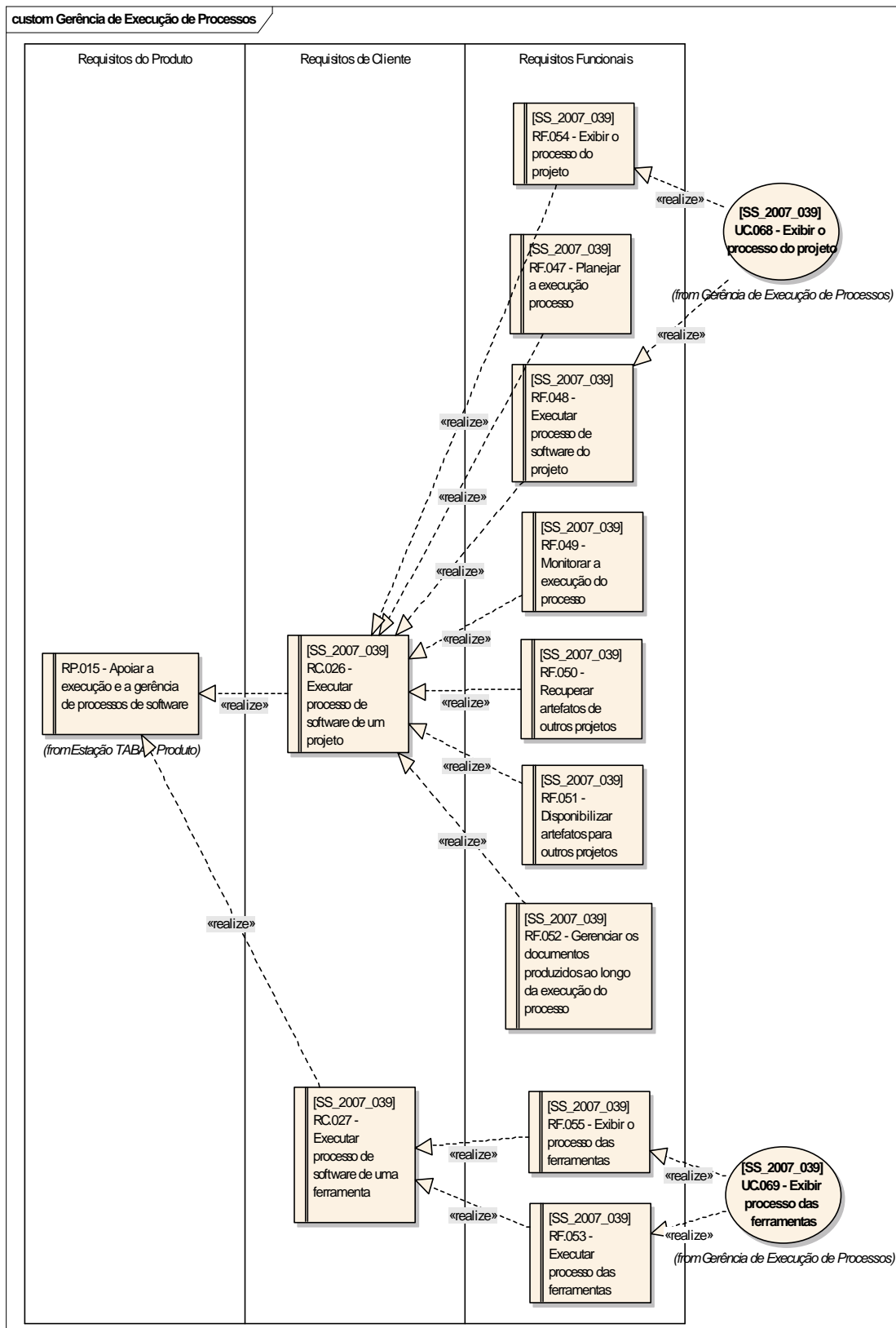


Figura 5.37 – Execução de Processos nos Ambientes de Projeto – Requisitos e Casos de Uso

5.7.5 Gerência de Conhecimento

A Gerência de Conhecimento atualmente na Estação Taba está presente através da Aquisição de Conhecimento Integrada às Ferramentas através da ferramenta Acknowledge (MONTONI, 2003), Comunidades de Prática (MIRANDA, 2004), Mapa de Conhecimento da Organização (SANTOS, 2003) e inclusão de conhecimento sobre o domínio da aplicação tratado pela Organização (OLIVEIRA, 1999) e de conhecimento de Tarefas (ZLOT, 2002). As soluções atuais foram consideradas parcialmente adequadas e sofrerão apenas pequenas modificações para serem totalmente adequadas ao novo modelo (por exemplo, permitir descrever a estrutura de uma corporação além de organizações).

A Figura 5.38 e a Figura 5.39 mostram o diagrama de requisitos referentes à Gerência de Estrutura Organizacional. Em ambas as figuras, na primeira coluna são exibidos os Requisitos do Produto (identificados na Tabela 5.4 – Requisitos da Estação Taba e de seus Ambientes) relacionados à Gerência de Ambientes. A partir dos Requisitos do Produto foram derivados Requisitos de Cliente (que podem ser vistos na segunda coluna) e a partir destes, Requisitos Funcionais (que podem ser vistos na terceira coluna). Por fim, são exibidos os Casos de Uso que foram derivados dos Requisitos Funcionais (o que pode ser visto na quarta e última coluna). Detalhes da execução das ferramentas Sapiens e Acknowledge poderão ser vistos na seção 5.8.

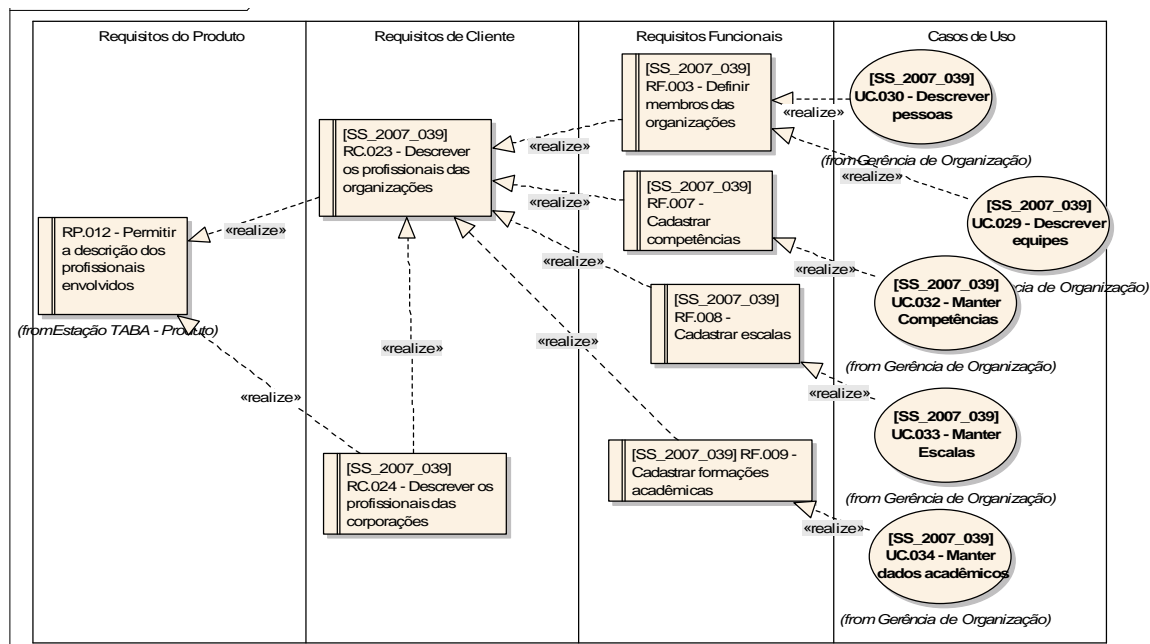


Figura 5.38 – Definição da Estrutura Organizacional I – Requisitos e Casos de Uso

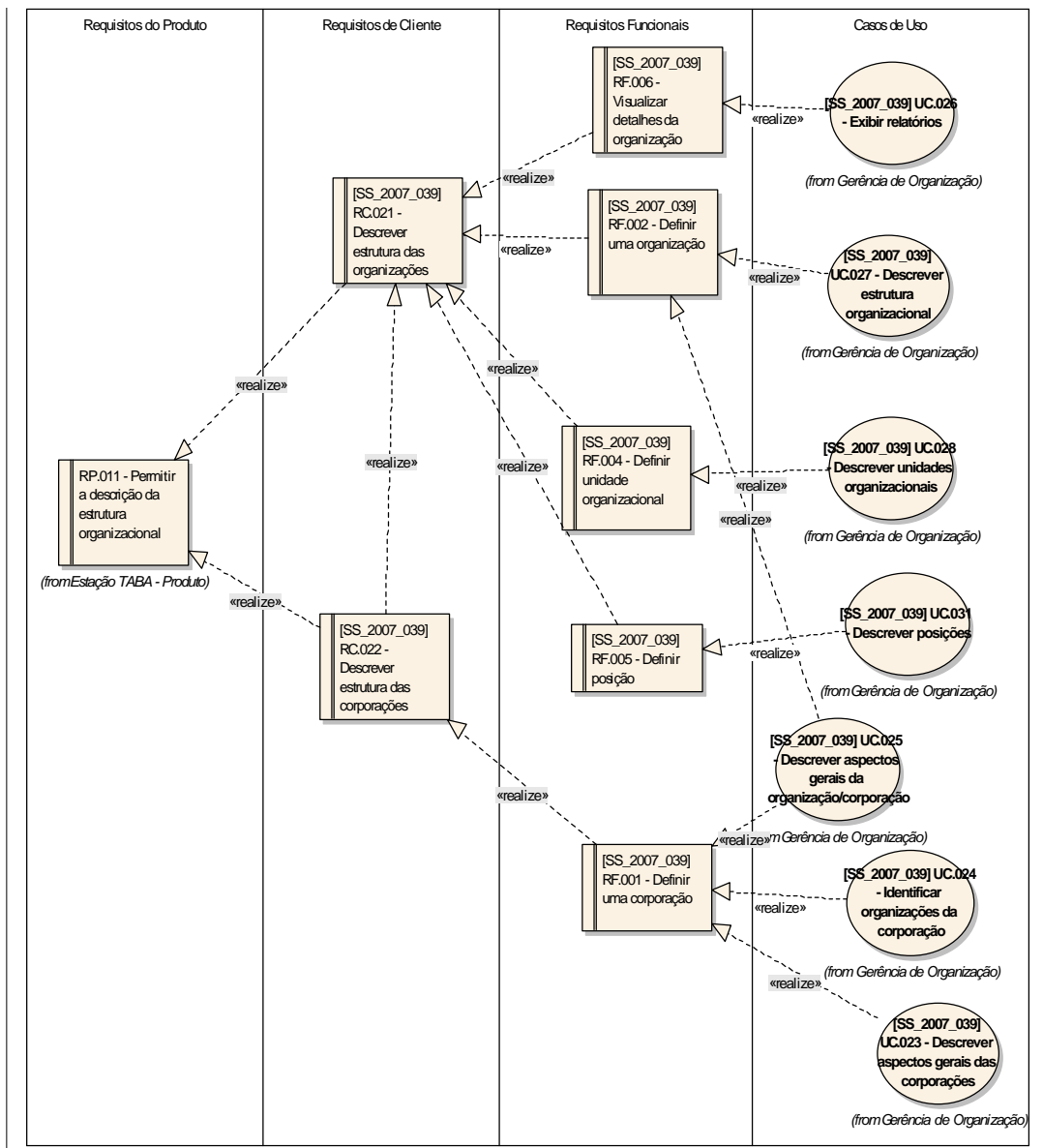


Figura 5.39 – Definição da Estrutura Organizacional II – Requisitos e Casos de Uso

5.8 Cenário de Uso dos Ambientes de Engenharia de Software Orientados a Corporação

Esta seção apresenta um cenário de uso dos AESCorp no contexto de uma corporação como uma prova de conceito da infra-estrutura desenvolvida no contexto deste trabalho. A Aeronáutica é uma corporação que possui dentro de sua estrutura algumas organizações que desenvolvem software. Em 2005, o Centro de Computação da Aeronáutica em São José dos Campos (CCA SJ) iniciou uma experiência de implantação de processos com base no Nível E do MPS.BR que culminou com o resultado positivo de

uma avaliação oficial no final de 2006 (SCHEID *et al.*, 2007)¹³. Devido ao sucesso da experiência, em 2007 a COPPE/UFRJ foi contatada pelo Centro de Computação da Aeronáutica em Brasília (CCA BR) visando ao apoio para a implantação de processos. Esta implantação começou a ser realizada em março de 2008 utilizando a infra-estrutura da Estação Taba.

Assim, o início da cooperação com mais um Centro de Computação da Aeronáutica serve para demonstrar um cenário de utilização dos AESCorp em uma corporação real. Para a descrição do cenário de uso será caracterizada a iniciativa de melhoria de processos de software com apoio ferramental procurando deixar claro alguns dos aspectos considerados relevantes para o entendimento das razões que levaram à iniciativa de melhoria de processos, o uso do apoio ferramental e os benefícios obtidos. Como, de fato, o ciclo de utilização dos AESCorp na corporação em questão não terminou, será utilizado detalhado o que se espera que sejam os benefícios e resultados a serem obtidos ao final da experiência, considerando, também a experiência anterior no CCA SJ.

5.8.1 Iniciativa de Melhoria de Processos na Aeronáutica

A iniciativa de melhoria de processos será acompanhada em duas organizações desenvolvedoras de software da Aeronáutica: o Centro de Computação da Aeronáutica em São José dos Campos (CCA SJ) e o Centro de Computação da Aeronáutica em Brasília (CCA BR). O CCA SJ já possui um processo padrão institucionalizado compatível com o Nível E do MPS.BR enquanto que o CCA BR não utiliza nenhum tipo de processo padrão.

Os objetivos desta iniciativa de melhoria de processos são: (i) a definição de um processo padrão de desenvolvimento de software para a Aeronáutica compatível com o Nível G do MPS.BR (menor nível de maturidade deste modelo); (ii) institucionalização de um processo padrão de desenvolvimento de software para o CCA BR compatível com o Nível G do MPS.BR; e (iii) manutenção da institucionalização do processo padrão de desenvolvimento de software compatível com Nível E do MPS.BR no CCA SJ de forma a garantir a continuidade do projeto de melhoria iniciado em 2005 e permitir, no futuro, que este processo seja evoluído para o Nível C do MPS.BR.

¹³ Neste projeto de melhoria de processos de software eu fui alocado como consultor na tarefa de auxiliar a empresa na implantação dos novos processos e na condução dos projetos de desenvolvimento de software.

Com esta experiência deseja-se, também, avaliar o uso dos Ambientes de Engenharia de Software Orientados a Corporação no contexto da implantação de MPS.BR em unidades desenvolvedoras e mantenedoras de software da Aeronáutica.

5.8.2 Apoio Ferramental

O apoio ferramental a ser utilizado será baseado principalmente no conjunto de ferramentas disponibilizadas pela Estação Taba através dos Ambientes Configurados Corporativos, Organizacionais e de Projeto. Estas ferramentas são descritas nas seções 3.3.1 e 5.7 desta tese. Além disso, serão utilizadas outras ferramentas de apoio possuídas pelas organizações envolvidas como o Caliber RM (para gerência de requisitos), MS Project Server (para gerência de cronograma e de alocação de recursos humanos), CVS (para controle de versão de código e alguns documentos) e ClearQuest (para controle de requisições de funcionalidades pelos usuários e gerência de ações corretivas).

A utilização das ferramentas e dos Ambientes Taba será descrita a seguir.

A tela exibida na Figura 5.37 corresponde à atividade “Definir Ambiente Configurado” da ferramenta de Definição de Ambientes Configurados (cujo processo foi descrito na seção 5.7.1). Nesta tela, pode-se identificar o nome do ambiente configurado a ser gerado no final e a organização ou corporação a que ele se destina (a escolha entre uma organização e uma corporação derivará seu tipo, respectivamente: Ambiente Organizacional ou Ambiente Corporativo). Neste caso, foi definido o Ambiente Configurado Corporativo para a Aeronáutica.



Figura 5.40 – Definição de Ambiente Configurado – Definir Ambiente Configurado

Durante a execução da atividade “Caracterizar Contexto de Configuração”, que pode ser vista na Figura 5.41, podem ser definidos os contextos de Processos de Software (o que inclui os tipos de software), de Apoio Ferramental e de Domínio de Aplicação que o Ambiente apoiará.

Taba Ambiente de Engenharia de Software

Logini: Senha:

Projeto Taba

- Configuração de Ambientes
 - Definir Ambiente Configurado
 - Caracterizar Contexto de Configuração
 - Selecionar Ativos de Processo
 - Gerar Ambiente Configurado

Passos para a Atividade

Para realizar a atividade selecionada, siga os passos abaixo:

- Escolher a Abordagem
- Realizar Estimativas das Atividades

Legenda de Cores

Caracterizar Contexto de Configuração

Contexto de Processos de Software

Tipos de Software:

<input type="checkbox"/>	Nome	Descrição
<input checked="" type="checkbox"/>	Web	Portais para Internet
<input checked="" type="checkbox"/>	Não Web	Sistemas Gerais
<input type="checkbox"/>	Sistema Especialista	

Tipos de Processo:

<input type="checkbox"/>	Nome	Tipo	Adaptação ao MCV	Especializável
<input checked="" type="checkbox"/>	Processo de Desenvolvimento	Fundamental	Sim	Sim
<input checked="" type="checkbox"/>	Gerência de Projeto (GPR)	Gerência	Não	Não
<input checked="" type="checkbox"/>	Gerência de Requisitos (GRE)	Apoio	Não	Sim
<input type="checkbox"/>	Aquisição (AQU)	Fundamental	Não	Não
<input checked="" type="checkbox"/>	Gerência de Configuração (GCO)	Apoio	Não	Não
<input checked="" type="checkbox"/>	Garantia da Qualidade (GQA)	Apoio	Não	Não
<input checked="" type="checkbox"/>	Medição (MED)	Apoio	Não	Não
<input checked="" type="checkbox"/>	Avaliação e Melhoria do Processo Organizacional (AMP)	Organizacional	Não	Não
<input checked="" type="checkbox"/>	Definição do Processo Organizacional (DFP)	Organizacional	Não	Não
<input checked="" type="checkbox"/>	Gerência de Recursos Humanos (GRH)	Organizacional	Não	Não

Paradigmas de Desenvolvimento:

<input type="checkbox"/>	Nome	Descrição
<input checked="" type="checkbox"/>	Orientado a Objetos	
<input type="checkbox"/>	Estruturado	

Modelos de Maturidade:

<input type="checkbox"/>	Nome	Descrição	Nível
<input checked="" type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível G
<input type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível F
<input checked="" type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível E
<input type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível D
<input type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível C
<input type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível B
<input type="checkbox"/>	MPS.BR	Modelo Brasileiro de Melhoria de Processos de Software	Nível A
<input type="checkbox"/>	CMMI	Capability Maturity Model Integration	Nível 2
<input type="checkbox"/>	CMMI	Capability Maturity Model Integration	Nível 3
<input type="checkbox"/>	CMMI	Capability Maturity Model Integration	Nível 4

Contexto de Apoio Ferramental

Ferramentas:

<input type="checkbox"/>	Nome	Descrição
<input checked="" type="checkbox"/>	AdaptPro	Adaptação de Processos para Projetos
<input checked="" type="checkbox"/>	Biblioteca de Ativos	Biblioteca de Ativos de Processo
<input checked="" type="checkbox"/>	Config	Configuração de Ambientes
<input checked="" type="checkbox"/>	CustManager	Gerência de Custos
<input checked="" type="checkbox"/>	TempManager	Gerência de Tempo
<input checked="" type="checkbox"/>	GConf	Gerência de Configuração
<input checked="" type="checkbox"/>	OrgPlan	Definição do Plano Organizacional para o Projeto
<input type="checkbox"/>	Pilot	Gerência de Projetos Piloto
<input checked="" type="checkbox"/>	Planilha de Atividades	Planilha de Atividades do Projeto
<input type="checkbox"/>	MBR	Análise de Fazer, Comprar ou Reutilizar

Contexto de Domínio de Aplicação

Teorias de Domínio:

<input type="checkbox"/>	Nome	Descrição
<input type="checkbox"/>	Cardiologia	
<input type="checkbox"/>	Processo de Software	
<input type="checkbox"/>	Acústica Submarina	

Itens de Conhecimento

- O cronograma deve ser elaborado levando-se em consideração a situação ideal, sem as possíveis horas extras.
- A elaboração do cronograma envolve a definição do caminho crítico do mesmo.

Diretrizes

- Para projetos do cliente X, gerar também uma versão do cronograma no MS Project.

Lições Aprendidas

- A elaboração do cronograma deve ser realizada levando-se em consideração finais de semana, feriados, férias dos funcionários, viagens trabalho e demais eventos que impeçam a equipe de trabalhar em algumas datas.

Confirmar Cancelar

Direitos Reservados para Linha de Engenharia de Software do PESC - COPPE - UFRJ

Figura 5.41 – Definição de Ambiente Configurado – Caracterizar Contexto de Configuração

O contexto de Processos de Software inclui os tipos de software (por exemplo, software para Web), tipos de processo (por exemplo, Desenvolvimento, Aquisição, Medição), paradigma de desenvolvimento (por exemplo, orientado a objetos ou estrutura), modelos e níveis de maturidade apoiados (por exemplo, CMMI Nível 2 ou MPS.BR Nível E). O contexto de Apoio Ferramental permite possibilita quais ferramentas devem ser disponibilizadas no ambiente a ser gerado. O contexto de Domínio de Aplicação possibilita escolher quais teorias de domínio devem estar disponíveis no ambiente a ser gerado.

Para a Aeronáutica, foi definida utilização do paradigma de desenvolvimento orientado a objetos e a possibilidade de construção de software Web e não Web. Além disso, como há na Corporação organizações com diferentes níveis de maturidade, escolheu-se os tipos de processo compatíveis com os níveis G e E do MPS.BR, além do Processo de Desenvolvimento. Além disso, selecionaram-se as ferramentas da Estação Taba compatíveis e adequadas a estes níveis.

Na atividade “Selecionar Ativos de Processos”, Figura 5.42, é possível incluir na base inicial do Ambiente os processos padrão e especializados (ver seção 5.7.2) a serem utilizados e também os ativos de processo existentes (ver seção 5.7.3). No caso de estar-se fazendo a configuração de Ambiente Corporativo são exibidos os processos e ativos de processos disponíveis na base do Meta-Ambiente. No caso de estar-se fazendo a configuração de Ambiente Organizacional são exibidos os processos e ativos de processos disponíveis na base do Ambiente Corporativo.

Para a Aeronáutica, foram escolhidos, novamente, ativos de processos compatíveis com os níveis que devem ser apoiados para comporem a base inicial da biblioteca de ativos corporativos. Estes ativos poderão ser evoluídos no futuro a critério da corporação e/ou de suas organizações.

Por fim, a atividade “Gerar Ambiente Configurado”, que pode ser vista na Figura 5.43, finaliza o processo e permite a disponibilização do ambiente a seus usuários finais. A partir deste momento, o Ambiente Corporativo para a Aeronáutica já pode ser utilizado. De posse de seu Ambiente Configurado, a Corporação em questão pode utilizá-lo para gerar ambientes para as organizações da mesma forma que o descrito anteriormente.

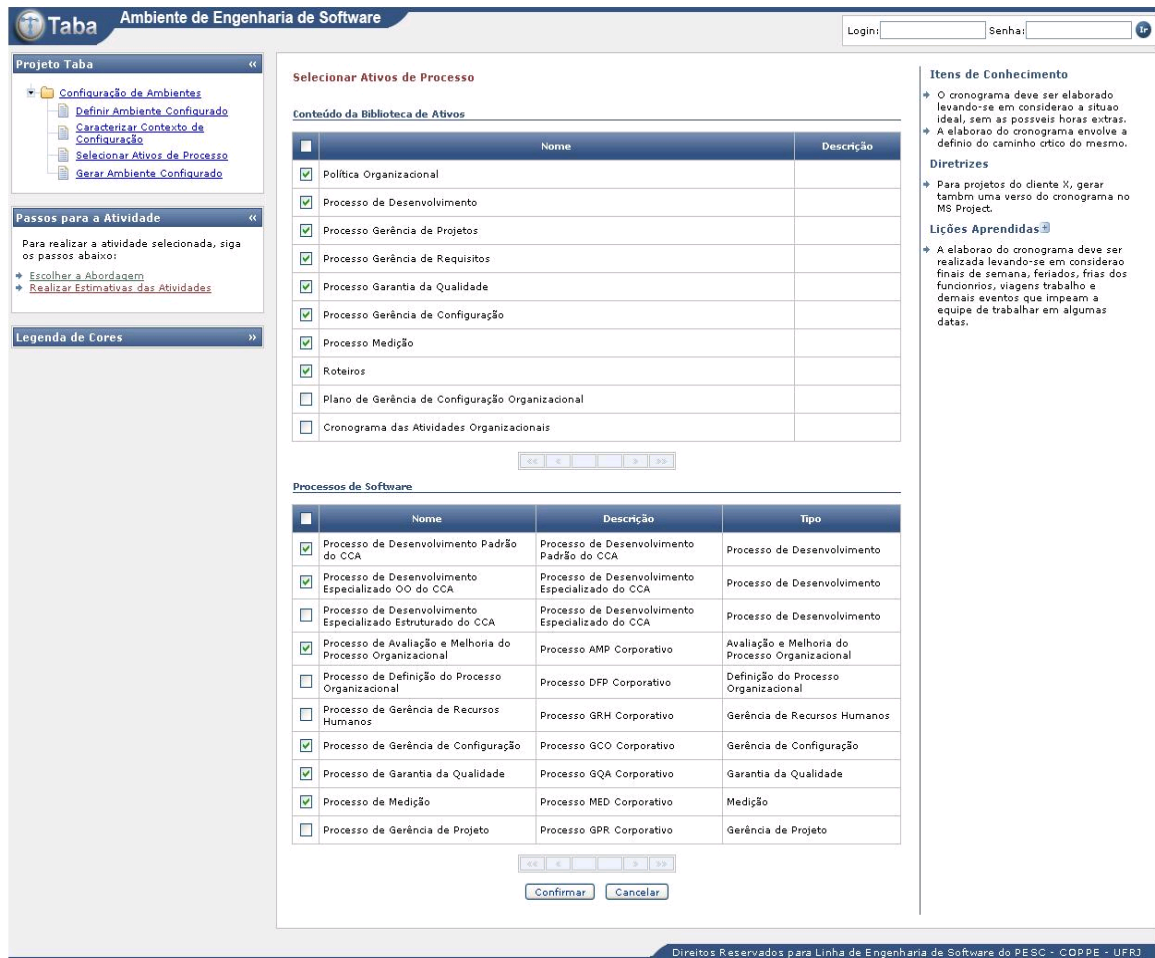


Figura 5.42 – Definição de Ambiente Configurado – Selecionar Ativos de Processos

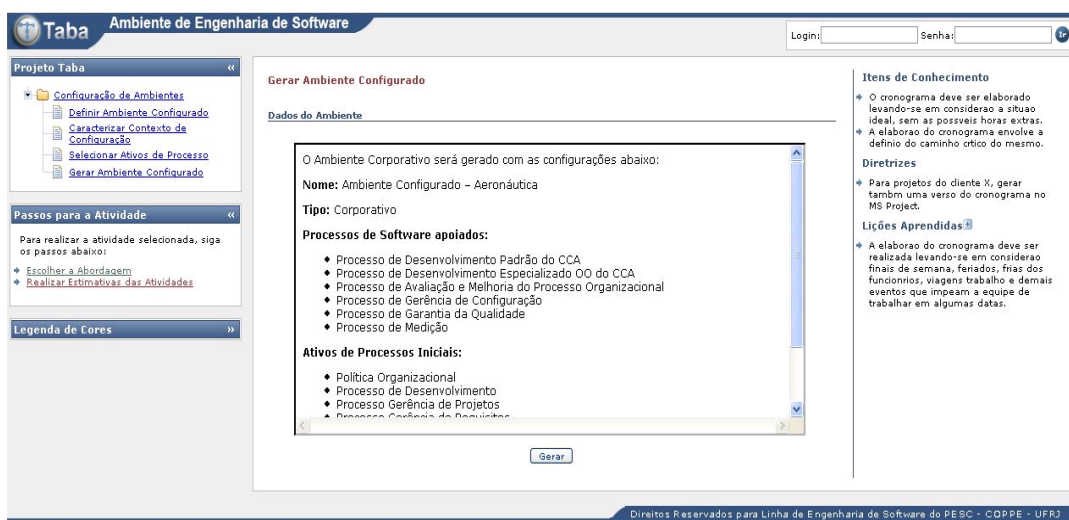


Figura 5.43 – Definição de Ambiente Configurado – Gerar Ambiente Configurado

A definição dos processos para a Corporação é feita num momento anterior à definição do Ambiente Corporativo através da ferramenta EditPro, que pode ser vista na Figura 5.44. A tela exibida permite a derivação de processos especializados a partir dos processos padrão definidos. No exemplo, está sendo definido o processo especializado para o paradigma Orientado a Objetos.

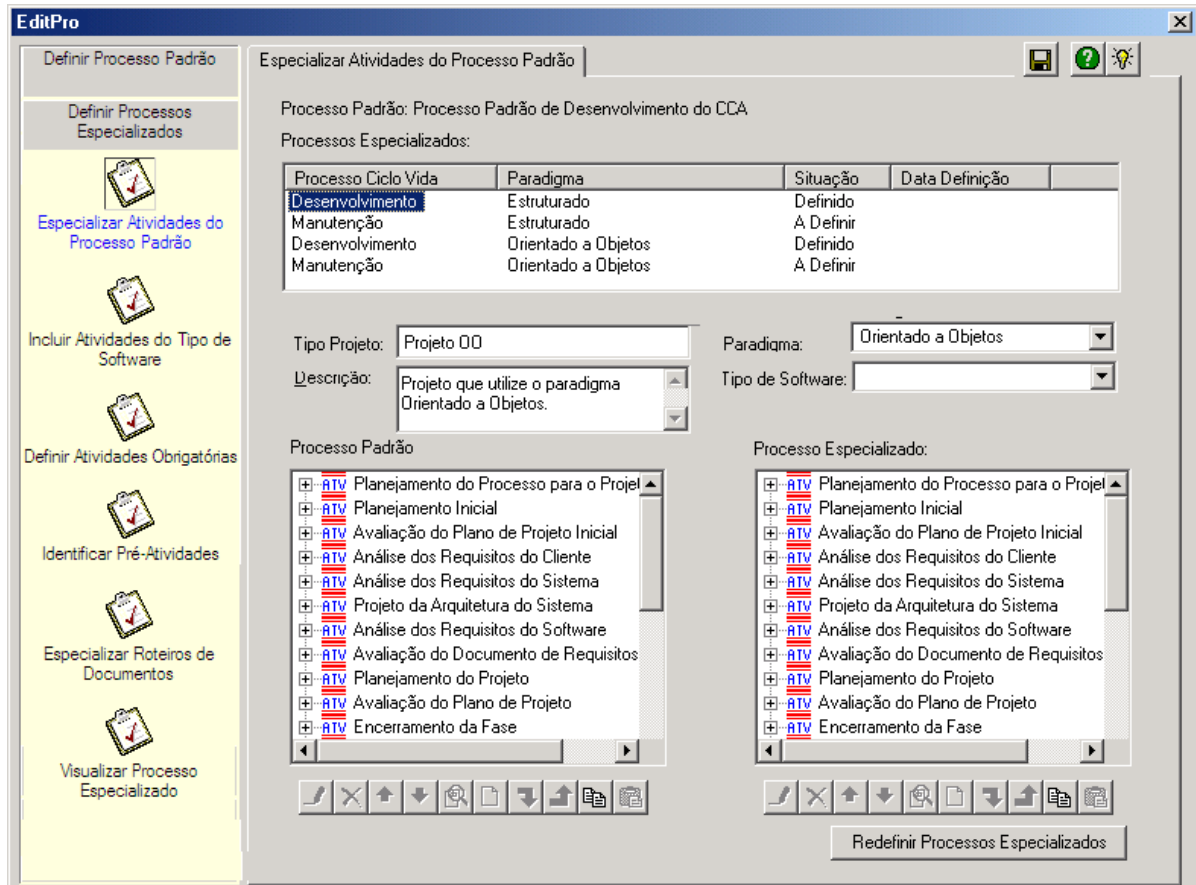


Figura 5.44 – Definição de Processos Especializados – Implementação

Antes da definição do Ambiente para uma corporação ou organização é necessário o seu cadastro. A definição da Estrutura Organizacional na Estação Taba é feita através da ferramenta Sapiens (SANTOS, 2003; SANTOS *et al.*, 2004). Esta ferramenta permite a identificação da estrutura organizacional (composta pelas unidades organizacionais) e a distribuição de competências através desta estrutura por meio da identificação das pessoas que compõem a organização e as competências possuídas por elas. Também é possível identificar as equipes disponíveis na organização e os requisitos de competência para que as pessoas ocupem determinadas posições ao longo da estrutura organizacional. A partir de

agora é possível definir também corporações através da associação da corporação com suas organizações subordinadas, como pode ser visto na Figura 5.45. O ícone “C” ao lado do nome organização na árvore exibida na esquerda da tela a qualifica como uma corporação. Para cada corporação é possível indicar quais são suas organizações subordinadas, como pode ser visto na parte direita da tela.

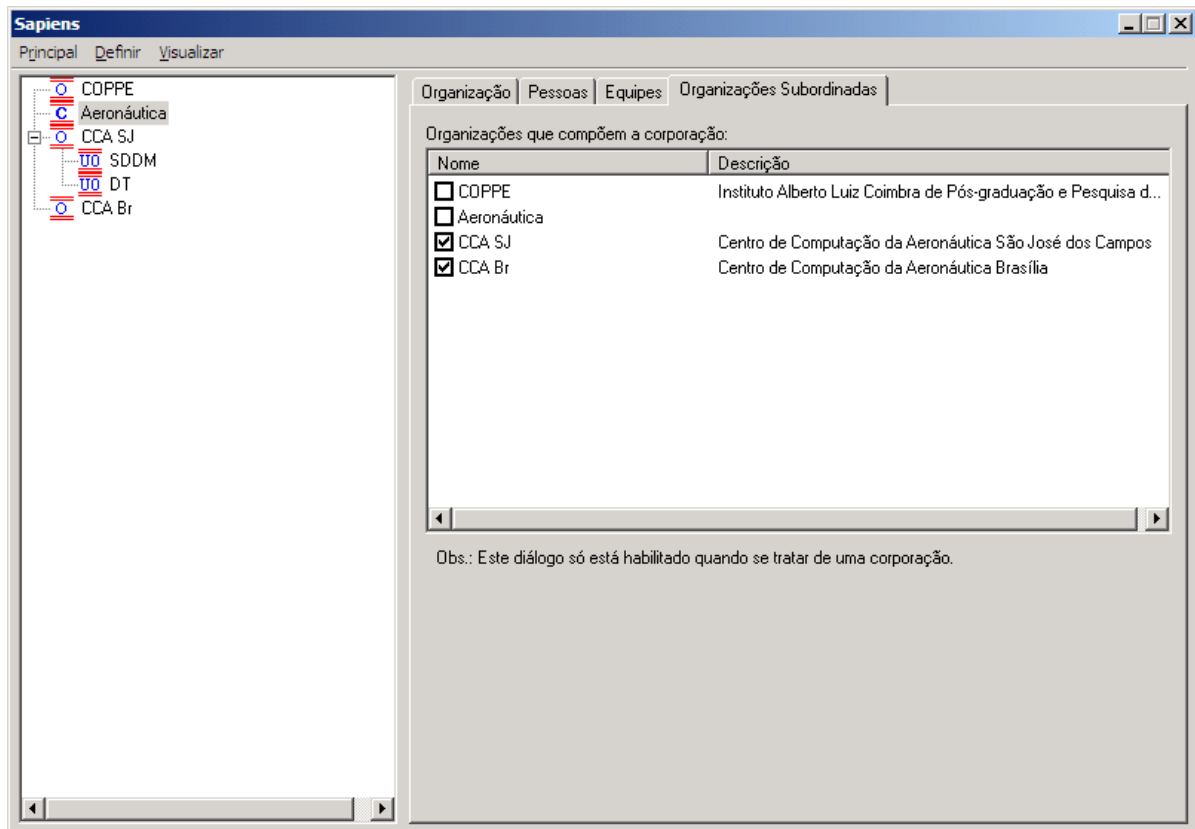


Figura 5.45 – Descrição da Estrutura Organizacional da Corporação – Implementação

Uma vez de posse de seu Ambiente Configurado, a Corporação pode prosseguir o povoamento e evolução de sua biblioteca de ativos de processos. A Figura 5.46 exibe a definição da estrutura da biblioteca de ativos de processos para a Aeronáutica com base no conteúdo inicial definido durante a configuração do ambiente. Neste momento, é possível incluir na lista de ativos gerenciáveis quaisquer documentos desejados pela organização/corporação. Podem-se associar, também, ferramentas aos itens de forma a facilitar sua utilização.

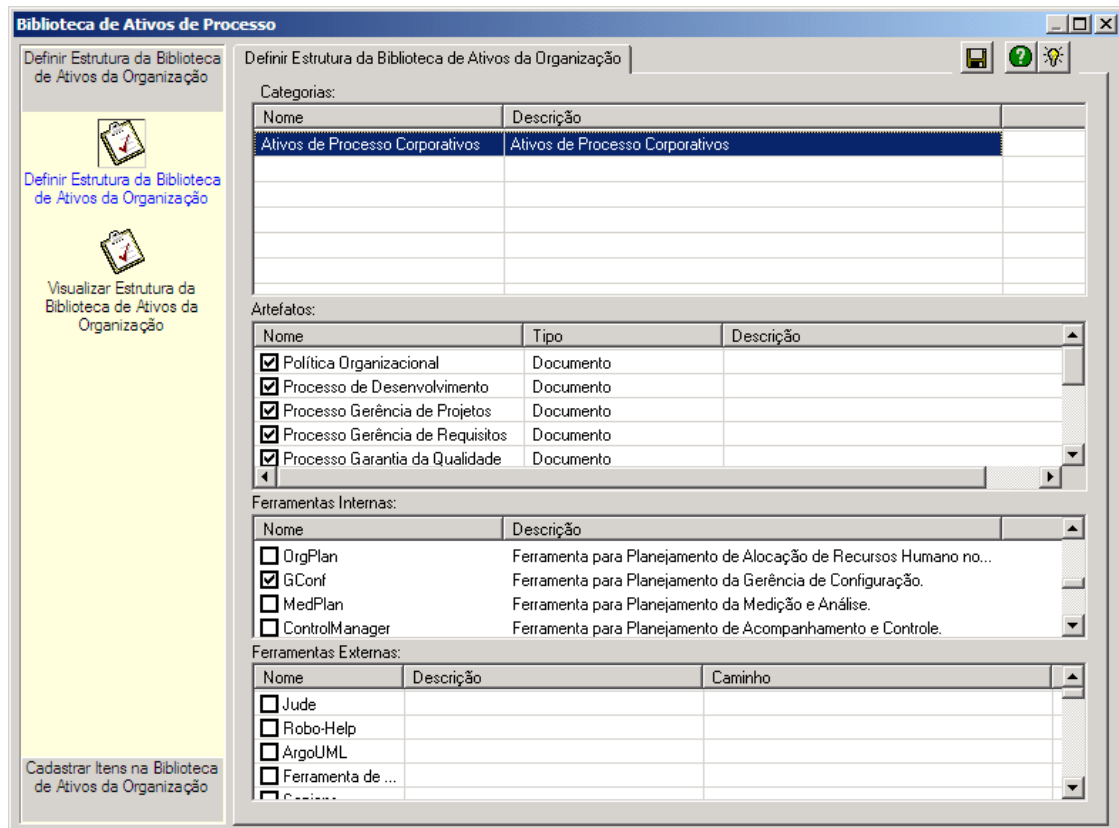


Figura 5.46 – Biblioteca de Ativos de Processos - Implementação

Os itens da biblioteca de ativos de processos depois poderão ser colocados nos níveis apropriados de gerência de configuração, como pode ser visto na Figura 5.47.

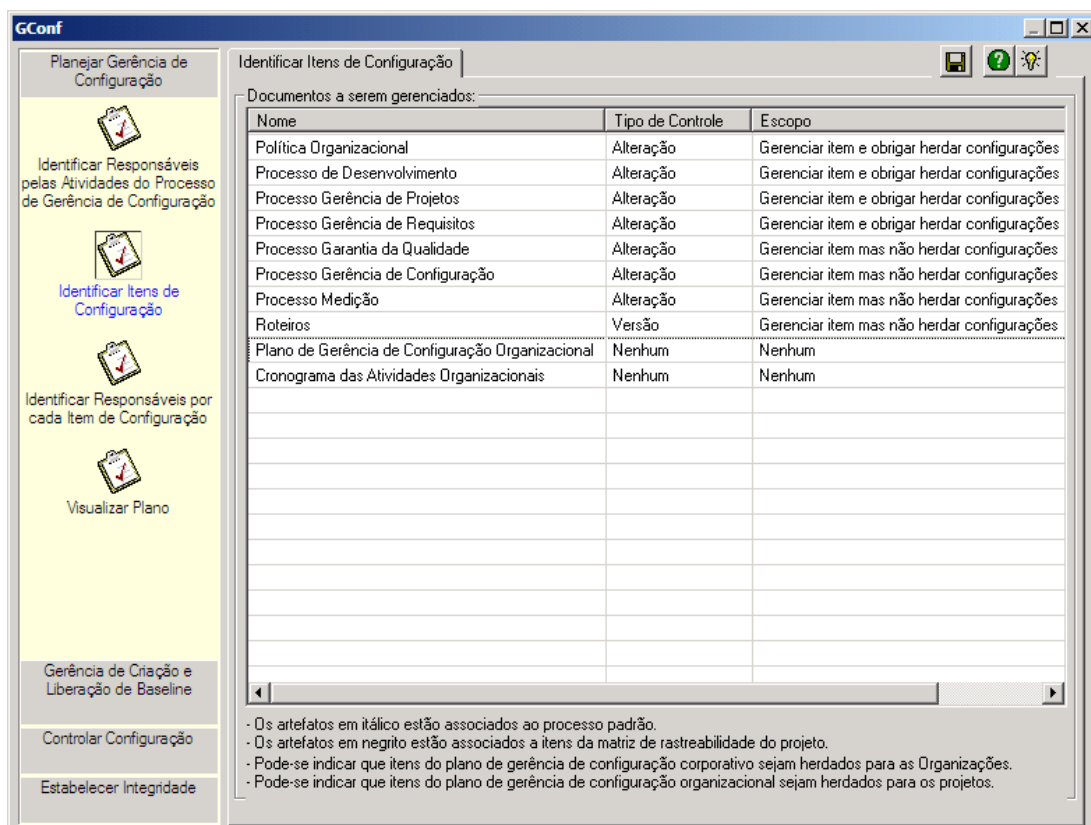


Figura 5.47 – Gerência de Configuração de Ativos de Processos - Implementação

A ferramenta de gerência de configuração foi modificada para: (i) permitir o controle de versão dos documentos além do controle de alteração; (ii) permitir que uma Corporação possa obrigar suas Organizações a seguir um plano mínimo de gerência de configuração dos seus ativos de processo; e (iii) permitir que uma Organização defina um plano mínimo de gerência de configuração para os produtos dos projetos a serem executados.

A Figura 5.48 mostra a definição de um Ambiente de Projeto para o projeto piloto de utilização do processo padrão no CCA BR através da ferramenta AdaptPro. Esta tela se presta à caracterização do projeto a ser apoiado pelo ambiente a ser gerado. Pode-se perceber a possibilidade de escolha do tipo de processo a ser utilizado (foi escolhido dentre as opções disponíveis, o de “Desenvolvimento”) e também a definição de qual processo especializado a ser utilizado (no caso, “Projeto OO”).

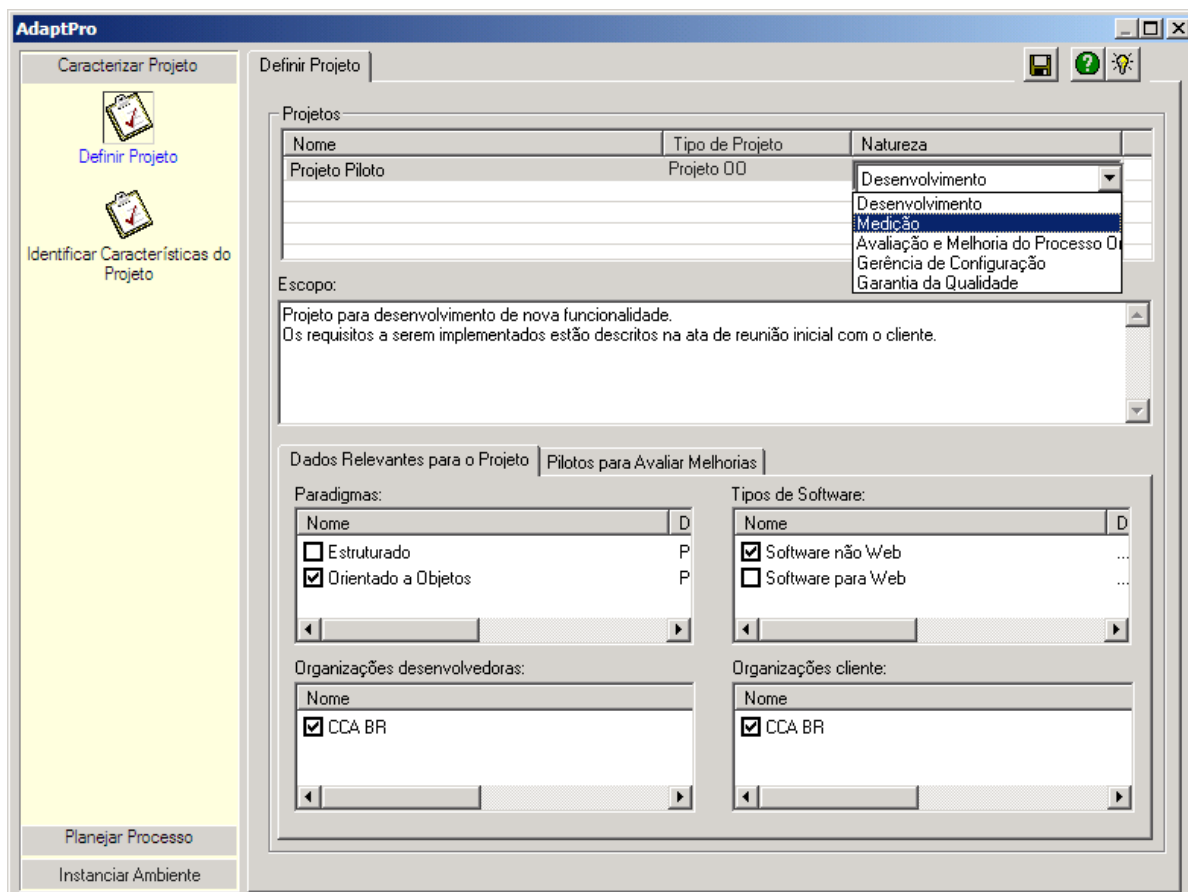


Figura 5.48 – Definição de Ambiente de Projeto – Implementação

Após a definição do Ambiente do Projeto, o processo contido nele deve ser executado. A Figura 5.49 exibe a tela principal de um Ambiente de Projeto, responsável pela execução do processo deste projeto. A orientação a processos pode ser vista na lateral direita da tela: é exibido o processo interno que guia a utilização da ferramenta ou, no caso

de um ambiente de projeto, o processo de software utilizado durante a definição do ambiente. A integração com a ferramenta Acknowledge pode ser vista na lateral direita da tela. O conhecimento associado à atividade sendo executada é exibido proativamente, não sendo mais, portanto, dependente da interação do usuário para ser exibido. A área central da figura contém, de fato, as funcionalidades de cada ferramenta.

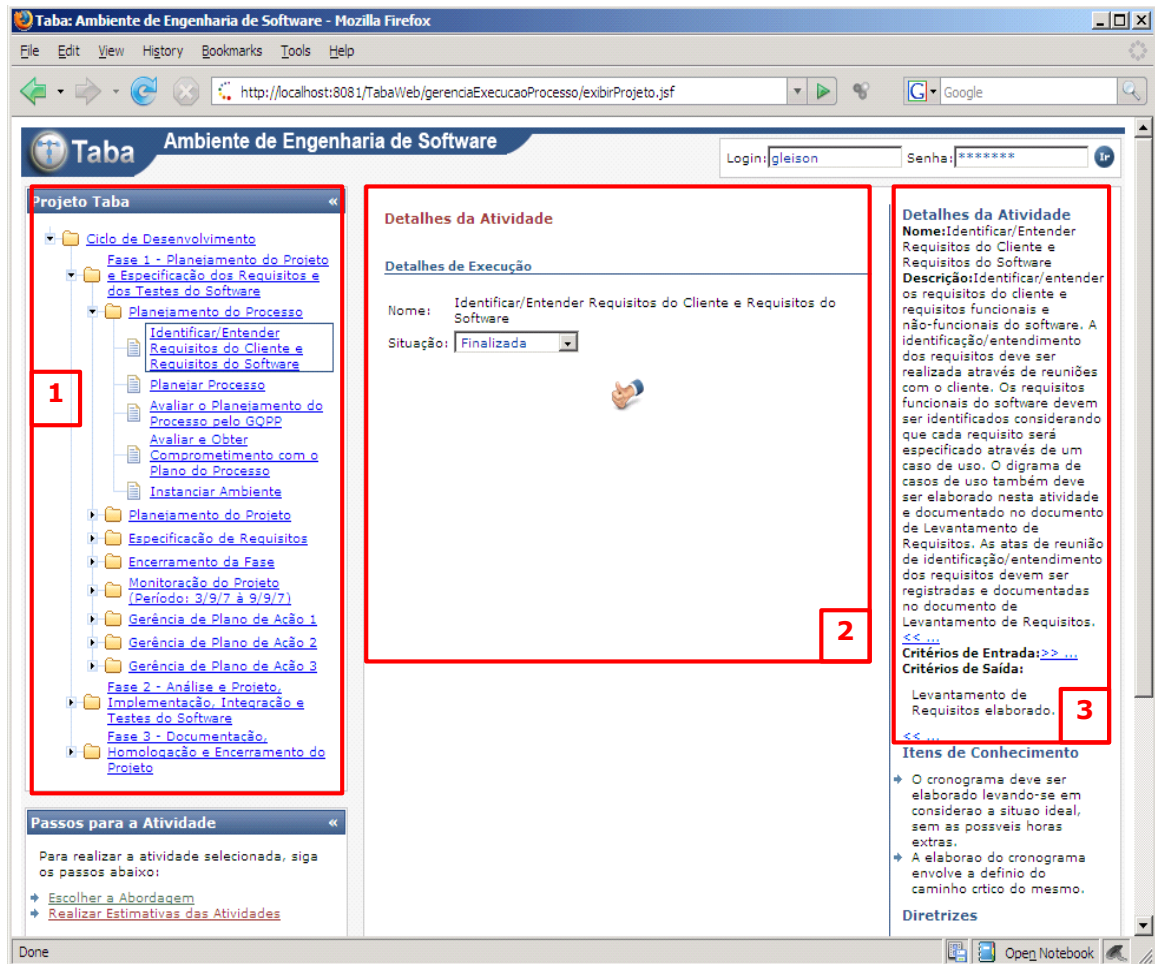


Figura 5.49 – Execução de Processos nos Ambientes de Projeto - Implementação

Dessa forma, a partir da tela principal do Ambiente do Projeto, o usuário pode percorrer todo o processo de software do projeto, como pode ser visto na Área 1 da Figura 5.49. Ao se selecionar uma das atividades (identificada pelo símbolo de uma folha de papel) (i) seus detalhes também são exibidos na Área 2 e (ii) os itens de conhecimento associados a ela são exibidos na Área 3.

A integração da ferramenta Acknowledge também pode ser vista na Figura 5.49. Quando o usuário seleciona uma atividade, sua descrição é exibida, como pode-se ver na Área 3. Os itens de conhecimento, a princípio, são exibidos na forma reduzida. Nesta forma de exibição, são mostrados os 100 primeiros caracteres acompanhados de um link para a descrição completa, representado pelos caracteres ">> ...". Ao se clicar sobre este

link, é exibido o conteúdo completo do conhecimento associado à atividade, desta vez, acompanhado de um *link* para a descrição resumida, representado pelos caracteres “<< ...”.

Durante a execução do programa de melhoria de processos, caso uma evolução nos processos padrões da corporação ou de uma das organizações for identificada e realizada, é possível definir uma evolução do ambiente configurado correspondente através de uma nova configuração. Neste caso, os projetos em andamento podem continuar a utilizar e executar seus processos normalmente enquanto novos projetos terão acesso à versão melhorada dos processos através dos novos ambientes de projeto que forem gerados.

5.8.3 Resultados da Iniciativa de Melhoria de Processos

Os resultados obtidos na experiência passada no CCA SJ demonstram que o apoio ferramental da Estação Taba é adequado a uma organização que deseja ter seus processos institucionalizados no Nível E do MPS.BR. Como as ferramentas a serem utilizadas no CCA BR, que almeja o Nível G do MPS.BR, também foram utilizadas no CCA SJ, espera-se que se obtenha o mesmo tipo de resultado. Outro fato que corrobora essa expectativa é o fato de as organizações e seus membros possuírem um perfil semelhante, o que também nos leva a crer que os outros aspectos da envolvidos com a condução da iniciativa de melhoria de processos também serão compatíveis. Além disso, a definição dos processos e configuração dos ambientes utilizando a infra-estrutura dos AESCorp também se mostrou satisfatória até o momento, apesar de não termos definido nenhum instrumento para avaliá-la com mais rigor, além da prova de conceito realizada até o momento.

O uso das ferramentas desenvolvidas no contexto dos AESCorp pela Aeronáutica é demorado pois depende da execução de vários projetos no contexto de suas organizações desenvolvedoras de software. Devido a isto, a experiência de uso descrita nesta seção ainda está em andamento no momento. Entretanto, é intenção de todos os envolvidos que ela continue e permita, assim, a avaliação da infra-estrutura construída para os AESCorp.

5.9 Considerações Finais

Este capítulo apresentou as modificações a serem realizadas na Estação Taba para a definição dos Ambientes de Engenharia de Software Orientados a Corporações, englobando alterações na sua arquitetura e funcionalidades para apoio a processos, definição de Ambientes, Gerência de Conhecimento e Gerência de Ativos de Processos. No próximo capítulo serão apresentadas considerações finais e as contribuições deste trabalho.

CAPÍTULO 6 - CONCLUSÃO E PERSPETIVAS FUTURAS

Este capítulo conclui este trabalho, resumindo sua motivação e proposta, apresentando as suas contribuições. As perspectivas futuras fornecem a direção para que seja dada continuidade ao trabalho relacionado aos Ambientes de Engenharia de Software Orientados a Corporação (AESCorp).

6.1 Conclusão

Esta tese propôs os *Ambientes de Engenharia de Software Orientados a Corporações* (AESCorp), seus principais requisitos e características de infra-estrutura, englobando apoio à definição, execução e melhoria de processos, definição de ambientes, gerência de ativos de processos e gerência de conhecimento.

Os AESCorp foram definidos de forma a fornecem apoio computacional suficiente para que uma corporação possa gerenciar a diversidade e os estágios de maturidade dos processos de software de cada uma das organizações que a compõem de forma adequada às suas necessidades. Também foram pensados de forma a permitir que as corporações e organizações sejam capazes de gerenciar e controlar os diversos processos de software de que dispõem e/ou necessitem e o conhecimento organizacional envolvido.

Estes ambientes possuem duas vertentes específicas: (i) o apoio à execução e gerência de diferentes tipos de processos de software, além dos de desenvolvimento e manutenção, através de um ambiente para projetos específicos; e (ii) apoio à configuração de ambientes para que as corporações e organizações possam gerenciar a definição e evolução de seus processos de software, além de poderem definir ambientes de projetos para possibilitar a execução destes processos.

Também foram discutidos detalhes da implementação dos AESCorp utilizando a infra-estrutura fornecida pela Estação Taba. Esta infra-estrutura foi evoluída para possibilitar a construção e execução dos ambientes na plataforma Web enquanto a infra-estrutura atual é descontinuada. Ferramentas existentes foram alteradas para refletir o novo esquema de definição de ambientes e outra foi construída para permitir esta definição.

Também foi descrito como a infra-estrutura criada para os AESCorp pode ser utilizada na prática para apoiar uma corporação na definição de processos e ambientes para suas corporações. Para isso foi apresentado um exemplo de definição de ambientes e

processos para a Aeronáutica, uma corporação, e duas de suas organizações: o Centro de Computação da Aeronáutica em São José dos Campos e o Centro de Computação da Aeronáutica de Brasília.

6.1.1 Contribuições

As principais contribuições desta tese são:

- A definição dos Ambientes de Engenharia de Software Orientados a Corporação (AESCorp), o que implicou na:
 - definição dos requisitos para os AESCorp;
 - definição de um modelo de componentes para a construção de AESCorp;
 - definição da ontologia de corporação a partir da evolução da ontologia de organização, de forma a fornecer um vocabulário comum para a descrição das organizações.
- A extensão da Estação Taba para possibilitar a construção de AESCorp, o que implicou na:
 - definição e implementação de uma estratégia para a construção de AESCorp na Estação Taba, o que envolve a configuração do Meta-ambiente Taba para as corporações e organizações e instanciação de ambientes para os projetos específicos;
 - revisão das funções, ambientes, requisitos e arquitetura da Estação Taba, de forma a possibilitar a construção dos Ambientes Configurados Corporativos e dos Ambientes de Projeto para quais processos;
- A definição de uma nova infra-estrutura para a Estação Taba para a construção e disponibilização de seus ambientes na plataforma Web.
- A definição, em conjunto com a equipe de desenvolvimento da Estação Taba, de um processo padrão de desenvolvimento para apoiar a construção e evolução de novas ferramentas no contexto da Estação Taba e dos AESCorp.
- Realização de um estudo baseado em revisão sistemática para caracterizar o apoio ferramental utilizado por organizações na execução, avaliação e melhoria de iniciativas de melhoria de processos de software.
- A configuração e preparação de um Ambiente Corporativo para a Aeronáutica, de forma a verificar, na prática, a viabilidade do que foi aqui proposto.

6.2 Perspectivas Futuras

Apesar de todo o trabalho realizado no contexto desta tese no sentido de prover uma infra-estrutura para apoiar a definição, execução e melhoria de processos de software no contexto de uma corporação, ainda existe trabalho a ser realizado.

Trabalhos futuros podem ser realizados em no sentido de evoluir e melhorar ferramentas ou soluções já existentes na Estação Taba, por exemplo:

- Definição e implementação de uma abordagem para Gerência de Conhecimento na Corporação que possibilite a análise dos itens de conhecimento coletados e utilizados nas organizações de forma a permitir a construção de uma base de conhecimento corporativa visando uma maior disseminação do conhecimento adquirido pelas Organizações;
- Definição e implementação de uma abordagem para Melhoria de Processos na Corporação que possibilite, a partir da identificação de melhorias realizadas nos processos organizacionais, a definição de evoluções nos processos corporativos;
- Definição e implementação de mecanismos para a implantação de melhoria nos processos dos projetos em execução a partir de evoluções dos processos organizacionais;
- Definição de ferramentas e funcionalidades específicas para outros tipos de Ambientes de Projeto que podem ser apoiados pela Estação Taba, e não somente as ferramentas específicas para o processo de Desenvolvimento de software existentes atualmente;
- Definição e implementação de mecanismos de outros de apoio à execução dos processos, como a construção de uma máquina de processo ou sistema de gerência de *workflows*, ou o uso de agentes ou simulação;
- Definição e implementação de ferramentas que possibilitem o apoio à execução de processos de software baseados em metodologias ágeis;
- Definição e implementação de mecanismos que permitam à corporação a monitoração das ações realizadas a definição, execução e melhoria de processos de software desempenhadas pelas organizações subordinadas;
- Definição e implementação de mecanismos que permitam a alteração de processos em vó decorrentes de melhorias identificadas nos processos;

- Adaptação do modelo de construção dos AESCorp para o apoio a iniciativas de melhoria de processos desempenhadas por diferentes unidades organizacionais de uma mesma organização.

Reimplementações de ferramentas já existentes na Estação Taba para adequação à nova infra-estrutura incluem, por exemplo:

- Implementação de uma ferramenta para a definição de processos padrão e especializados também na infra-estrutura da Estação Taba na Web visando a uma maior integração com a ferramenta de definição de Ambientes Configurados Corporativos e Organizacionais;
- Reimplementação da ferramenta AdaptPro, para definição de processo para o projeto e Ambiente de Projeto para o projeto em questão, na nova infra-estrutura da Estação Taba na Web;
- Reimplementação da ferramenta de gerência de ativos de processos integrada à ferramenta de gerência de configuração também na nova plataforma na Web (neste sentido, um trabalho de final de curso de graduação já está em andamento com previsão de término em setembro/2008).

Além disso, pode-se citar como importante a atualização periódica do estudo baseado em revisão sistemática realizado de forma a continuar a caracterização do uso de apoio ferramental em iniciativas de melhoria de processos de software pelas organizações.

Ainda este ano está previsto o início do uso da Estação Taba na Web nas empresas que já utilizam os ambientes atuais, o que possibilitará a sua avaliação através da observação de campo e de instrumentos específicos e, conseqüentemente, a realização de melhorias. Espera-se dessa forma que este ciclo de avaliação da Estação Taba e de suas ferramentas seja executado de forma mais rápida possibilitando uma interferência positiva na sua evolução e continuidade da Estação Taba como projeto de pesquisa e apoio a iniciativas de melhoria de processos de software nas organizações brasileiras. Também está prevista a conclusão da experiência de uso dos AESCorp na Aeronáutica visando a avaliação da infra-estrutura desenvolvida.

Dessa forma, a partir das experiências de uso dos Ambientes Configurados e de Projetos na indústria será possível avaliar a eficácia da abordagem proposta neste trabalho.

REFERÊNCIAS

- AAEN, I., 2003, "Software process improvement: Blueprints versus recipes", v. 20, n. 5, pp. 86--93.
- ABNT, 2000a, *NBR ISO 9000 - Sistema de Gestão da Qualidade – Fundamentos e Vocabulário*, Associação Brasileira de Normas Técnicas, Rio de Janeiro, Brasil.
- ABNT, 2000b, *NBR ISO 9001 - Sistema de Gestão da Qualidade – Requisitos*, Associação Brasileira de Normas Técnicas, Rio de Janeiro, Brasil.
- ABREU, M., SCHLEBBE, H., REIS, C., *et al.*, 2003, "APSEE: Uma Abordagem Integrada para Automação da Gerência do Processo de Software". In: *Anais da X Sessão de Ferramentas - XVII Simpósio Brasileiro de Engenharia de Software*, pp. 91-96, Manaus, Brasil, Out. 2003.
- AGUIAR, H.V., ROUILLER, A.C., MOREIRA, R.T., *et al.*, 2005, "Implantando o Modelo CMMI em uma Empresa de Software de Pequeno Porte Jovem e Imatura". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 313 - 320, Porto Alegre - RS, Junho 2005.
- AGUIAR, T.C., 1992, *Um Sistema Especialista de Suporte à Decisão para Planejamento de Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- AISC, "Artemis 7". In: <http://www.aisc.com/Product/1>, acessado em 17/01/2008.
- ALBUQUERQUE, A.B., 2008, *Avaliação e Melhoria de Ativos de Processos Organizacionais em Ambientes de Desenvolvimento de Software*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- AMESCUA, A., GARCIA, J., SANCHEZ-SEGURA, M.I., *et al.*, 2006, "Approaching software process improvement to organizations", v. 5, n. 3, pp. 507--514.
- ANACLETO, A., WANGENHEIM, C.G.V., 2005, "Método e Modelo de Avaliação para Melhoria de Processos de Software em Micro e Pequenas Empresas". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 155 - 169, Porto Alegre - RS, Junho 2005.
- ANDRADE, J.M.S., 2005, *Avaliação de Processos de Software em ADSOrg*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- AOYAMA, M., 1998, "Web-Based Agile Software Development", *IEEE Software*, v. 15, n. 6 (Nov/Dec. 1998), pp. 56-65.

- ARAÚJO, M.A.P., 1998, *Automatização do Processo de Desenvolvimento de Software nos Ambientes Instanciados pela Estação TABA*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- ARBAOUI, S., DERNIAME, J., OQUENDO, F., *et al.*, 2000, "A Comparative Review of Process-Centered Software Engineering Environments". In: *Annals of Software Engineering*, v. 14, pp. 311-340, Dec. 2002.
- ASSIS, S.G., 1992, *Aquisição de Conhecimento para Sistemas Baseados no Conhecimento: Uma Experiência em Engenharia de Software*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- AUDY, J., PRIKLADNICKI, R., 2007, *Desenvolvimento Distribuído de Software*, 1a ed., Campus / Elsevier.
- BADDOO, N., HALL, T., 2002, "Motivators of Software Process Improvement: An analysis of practitioners' views", *Journal of Systems and Software*, v. 62, n. 2, pp. 85-96.
- BADDOO, N., HALL, T., 2003, "De-motivators for software process improvement: An analysis of practitioners' views", *Journal of Systems and Software*, v. 66, n. 1, pp. 23-33.
- BALDASSARRE, M.T., CAIVANO, D., VISAGGIO, C.A., *et al.*, 2002, "ProMisE: A Framework for Process Models Customization to the Operative Context". In: *Proceedings of the 2002 International Symposium on Empirical Software Engineering*.
- BANDINELLI, S., DI NITTO, E., FUGGETTA, A., 1996, "Supporting Cooperation in the SPADE-1 Environment", *IEEE Trans. on Software Engineering*, v. 22, n. 12, pp. 841-865.
- BARCELLOS, M.P., 2003, *Planejamento de Custos em Ambientes de Desenvolvimento de Software Orientados à Organização*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- BARCELLOS, M.P., FIGUEIREDO, S.M.D., ROCHA, A.R., *et al.*, 2003, "Utilização de Métodos Paramétricos, Analogias, Julgamento de Especialistas e Conhecimento Organizacional no Planejamento de Tempo e Custos de Projetos de Software". In: *II Simpósio Brasileiro de Qualidade de Software - SBQS 2003*, Fortaleza - CE.
- BARRETO, A., 2006, *Apoio à Verificação de Software em Ambientes de Desenvolvimento de Software Orientados à Organização*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- BARROS, M.O., 1996, *Recuperação de Componentes em Biblioteca de Software: Uma Abordagem Conexionista*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.

- BARRY, E.J., KEMERER, C.F., SLAUGHTER, S.A., 2007, "How software process automation affects software evolution: A longitudinal empirical analysis", *Journal of Software Maintenance and Evolution*, v. 19, n. 1, pp. 1-31.
- BASILI, V.R., CALDIERA, G., ROMBACH, H.D., 1994, "The Experience Factory". In: MARCINIAK, J.J. (eds), *Encyclopedia of Software Engineering* New York, John Wiley & Sons.
- BEN-SHAUL, I.Z., SKOPP, P.D., HEINEMAN, G.T., *et al.*, 1994, "Integrating groupware and process technologies in the Oz environment". In: *Proc. of the 9th Int. Software Process Workshop*, pp. 114–116, 5-7 Oct.
- BERGER, P., 2003, *Instanciação de Processos de Software em Ambientes Configurados na Estação TABA*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- BERTOLLO, G., 2006, *Definição de Processos em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.
- BERTOLLO, G., RUY, F., MIAN, P., *et al.*, 2002, "ODE – Um Ambiente de Desenvolvimento de Software Baseado em Ontologias". In: *Anais da IX Sessão de Ferramentas - XVI Simpósio Brasileiro de Engenharia de Software*, pp. 438-443, Gramado, Brasil, Out. 2002.
- BOEHM, B., 2006, "A view of 20th and 21st century software engineering". In: *Proceedings of the 28th international conference on Software engineering*, Shanghai, China.
- BORLAND, "Products for the Software Development Life Cycle". In: <http://www.borland.com/us/products/index.html>, acessado em 10/05/2008.
- BORSSATTO, Í.B., MORO, A.R., 2007, "Medições de uma implementação de MPS.BR Nível F". In: *VI Simpósio Brasileiro de Qualidade de Software - SBQS 2007*, pp. 397 - 404, Porto de Galinhas - PE, Junho 2007.
- BRAGA, R., WERNER, C., MATTOSO, M., 1999, "Odyssey: A Reuse Environment based on Domain Models". In: *Proceedings. of 2nd IEEE Symposium on Application-Specific Systems and Software Engineering Technology*, pp. 49-57, Richardson, USA, Mar. 1999.
- BREWSTERA, C., O'HARA, K., 2007, "Knowledge representation with ontologies: Present challenges—Future possibilities", *Int. J. Human-Computer Studies*, v. 65 (April 2007), pp. 563–568.
- BRIETZKE, J., LÓPEZ, P.A.D.P., ALBERTUNI, I., *et al.*, 2007, "A Conquista do MPS.BR Nível F na Qualidade Informática: Um Caso de Sucesso". In: *VI Simpósio*

- Brasileiro de Qualidade de Software - SBQS 2007*, pp. 381 - 388, Porto de Galinhas - PE, Junho 2007.
- BROWN, A., EARL, A., MCDERMID, J., 1992, *Software Engineering Environments: Automated Support for Software Engineering*, England, McGraw-Hill Book Company.
- BUSCHMANN, F., MEUNIER, R., ROHNERT, H., *et al.*, 1996, *Pattern-Oriented Software Architecture, Volume 1: A System of Patterns.*, 1st edition ed., John Wiley & Sons.
- CAMPOS, F.B., 2005, *Melhoria de Processos e Evolução do Meta-Ambiente TABA*, Exame de Qualificação, COPPE, UFRJ, Rio de Janeiro, Brasil.
- CAMPOS, F.B., ALBUQUERQUE, A.B., ANDRADE, J.M., *et al.*, 2006, "Abordagem em Níveis para Avaliação e Melhoria de Processos de Software". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 100-114, Vila Velha - ES.
- CANFORA, G., GARCIA, F., PIATTINI, M., *et al.*, 2006, "Applying a framework for the improvement of software process maturity", *Software - Practice and Experience*, v. 36, n. 3, pp. 283-304.
- CASS, A., VOLCKER, C., SUTTER, P., *et al.*, 2002, "SPiCE in action - experiences in tailoring and extension". In: *Euromicro Conference, 2002. Proceedings. 28th*, pp. 352-360.
- 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.
- CHANDRASEKARAN, B., JOSEPHSON, J.R., R., B.V., 1999, "What Are Ontologies, and Why Do We Need Them?" *IEEE Intelligent Systems & their applications*, v. 14, n. 1 (Jan/Feb, 1999), pp. 20-26.
- CHANWOO, Y., JUNHO, Y., BYUNGJEONG, L., *et al.*, 2004, "An integrated model of ISO 9001:2000 and CMMI for ISO registered organizations". In: *Software Engineering Conference, 2004. 11th Asia-Pacific*, pp. 150-157.
- CHOI, Y., LEE, E., HA, S., 2005, "The management of software processes with software process improvement tool based on ISO 15504", v. 2, pp. 933--936, Phoenix Park.
- CHRISIS, M.B., KONRAD, M., SHRUM, S., 2006, *CMMI (Second Edition): Guidelines for Process Integration and Product Improvement*, Addison Wesley Professional.
- CIMA, A.M., 1996, *Desenvolvimento de Software Com Reutilização Baseada em Frameworks Orientados a Objetos*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- COALLIER, F., 2003, "International Standardization in Software and Systems Engineering", *CrossTalk*, v. 16, n. 2 (Feb. 2003), pp. 18 - 22.

- COELHO, A.C.B., 1997, *Guias de Qualidade para Construção de Frameworks Orientados a Objetos*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- COLETTA, A., 2007, "An industrial experience in assessing the capability of non-software processes using ISO/IEC 15504", *Software Process Improvement and Practice*, v. 12, n. 4, pp. 315-319.
- COSTA, A., SALES, E., REIS, C.A.L., *et al.*, 2007, "Apoio a Reutilização de Processos de Software através de Templates e Versões". In: *Simpósio Brasileiro de Qualidade de Software - SBQS 2007*, pp. 47-61, Porto de Galinhas - PE, Junho 2007.
- COSTA, V., 2003, *TecKnowledge, Um Ambiente de Gerência de Conhecimento para uma Organização Fornecedora de Software*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- CRISPIM, M.E.H., 1991, *Avaliação de Métodos de Desenvolvimento de Software*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- CUGOLA, G., GHEZZI, C., 1999, "Design and Implementation of PROSYT: a Distributed Process Support System". In: *Proceedings of the 8th Workshop on Enabling Technologies on Infrastructure for Collaborative Enterprises*, pp. 32-39, Palo Alto, Jun. 1999.
- DECK, M., 2001, "Managing Process Diversity While Improving Your Practices", *IEEE Computer Society Press*, v. 18, n. 3, pp. 21-27.
- DEMING, W.E., 1986, *Out of the Crisis*, Cambridge: MIT Center for Advanced Engineering Study.
- DEMIRORS, O., DEMIRORS, E., TARHAN, A., *et al.*, 2000, "Tailoring ISO/IEC 12207 for instructional software development". In: *Euromicro Conference, 2000. Proceedings of the 26th*, v. 2, pp. 300-307 vol.2.
- DOSSICK, S., KAISER, G., 1999, "CHIME: a metadata-based distributed software development environment". In: *Proceedings of the 7th European Software Engineering Conference – 7th ACM SIGSOFT Symposium on Foundations of Software Engineering*, pp. 464-475, Toulouse, France, Sep. 1999.
- DUARTE, E., SILVA, R., ROCHA, A.R., *et al.*, 2005, "Uma Abordagem para Implantação do Modelo de Referência para Melhoria de Processos de Software". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 41 - 48, Porto Alegre - RS, Junho 2005.
- EBERT, C., 1999a, "Technical controlling and software process improvement", *Elsevier Science Inc*, v. 46, n. 1, pp. 25--39.
- EBERT, C., 1999b, "Technical controlling in software development", v. 17, n. 1, pp. 17--28.

- EBERT, C., DE MAN, J., 2002, "e-R&D - Effectively managing process diversity", v. 14, n. 1-4, pp. 73--91.
- ECKERSON, W.W., 1995, "Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications", *Open Information Systems*, v. 10, n. 1 (January 1995).
- EKDAHL, F., LARSSON, S., 2006, "Experience Report: Using Internal CMMI Appraisals to Institutionalize Software Development Performance Improvement". In: *Software Engineering and Advanced Applications, 2006. SEAA '06. 32nd EUROMICRO Conference on*, pp. 216-223.
- EMMERICH, W., FINKELSTEIN, A., FUGGETTA, A., *et al.*, 1999, "Software Process - Standards, Assessments and Improvement", pp. 15-26, 1999.
- FALBO, R., 1998, *Integração de Conhecimento em um Ambiente de Desenvolvimento de Software*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- FALBO, R., NATALI, A.C., MIAN, P., *et al.*, 2003, "ODE – Ontology-based software Development Environment". In: *IX Congreso Argentino de Ciencias de la Computación*, pp. 1124-1135, La Plata, Argentina, Out. 2003.
- FALBO, R.A., BORGES, L.S.M., VALENTE, F.F.R., 2004, "Using knowledge management to improve software process performance in a CMM level 3 organization". In: *Proceedings - Fourth International Conference on Quality Software, QSIC 2004*, pp. 162-169, Braunschweig.
- FARIAS, L., 2002, *Planejamento de Riscos em Ambientes de Desenvolvimento de Software Orientados à Organização*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- FATEEV, M., 1998, *O Meta-Gerador de Ferramentas da Estação TABA*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- FERNANDEZ, A., GARZALDEEN, B., GRUTZNER, I., *et al.*, 2004, "Guided support for collaborative modeling, enactment and simulation of software development processes", *Software Process Improvement and Practice*, v. 9, n. 2, pp. 95-106.
- FERREIRA, A.I.F., CERQUEIRA, R., SANTOS, G., *et al.*, 2007a, "Retorno de Investimento da Melhoria de Processo de Software na BL Informática". In: *VI Simpósio Brasileiro de Qualidade de Software - SBQS 2007*, pp. 413 - 420, Porto de Galinhas - PE, Junho 2007.
- FERREIRA, A.I.F., CERQUEIRA, R., SANTOS, G., *et al.*, 2006a, "ISO 9001:2000, MPS.BR Nível F e CMMI Nível 3: Uma Estratégia de Melhoria de Processos na BL

- Informática". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 375 - 382, Vila Velha - ES, Junho 2006.
- FERREIRA, A.I.F., SANTOS, G., CERQUEIRA, R., *et al.*, 2007b, "Applying ISO 9001:2000, MPS.BR and CMMI to Achieve Software Process Maturity: BL Informatica's Pathway". In: *29th Int. Conference on Software Engineering (ICSE)*, pp. 642-651, Minneapolis, USA, May.
- FERREIRA, A.I.F., SANTOS, G., CERQUEIRA, R., *et al.*, 2007c, "ROI of Software Process Improvement at BL Informática - SPI is Really Worth". In: *EuroSPI 2007 - Industrial Proceedings*, pp. 12.27 - 12.34, Postdam, Germany, September 2007.
- FERREIRA, A.I.F., SANTOS, G., CERQUEIRA, R., *et al.*, 2006b, "Taba workstation: Supporting software process improvement initiatives based on software standards and maturity models". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 4257 LNCS, pp. 207-218, Joensuu.
- FIGUEIREDO, S., ROCHA, A.R., SANTOS, G., *et al.*, 2006, "Uma Abordagem de Apoio à Solução Técnica em Ambientes de Desenvolvimento de Software Orientados à Organização". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 293-307, Vila Velha - ES.
- FIGUEIREDO, S., SANTOS, G., ROCHA, A.R., 2004, "Gerência de Configuração em Ambientes de Desenvolvimento de Software Orientados a Organização". In: *III Simpósio Brasileiro de Qualidade de Software - SBQS 2004*, Brasília - DF.
- FOURO, A.M.M., 2002, *Apoio à Construção de Base de Dados de Pesquisa em Ambientes de Desenvolvimento de Software Orientados a Domínio*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- FUGGETTA, A., 2000, "Software Process: a roadmap". In: *Proceedings of the Conference on the Future of Software engineering - International Conference on Software Engineering*, pp. 25-34, Limerick, Ireland.
- GALOTTA, C., 2000, *Netuno: Um Ambiente de Desenvolvimento de Software Orientado ao Domínio de Acústica Submarinha*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- GALOTTA, C., 2003, *Uma Proposta para Apoio Integrado aos Processos de Negócio e de Software*, Exame de Qualificação, COPPE, UFRJ, Rio de Janeiro, Brasil.
- GALOTTA, C., ZANETTI, D., ROCHA, A.R., *et al.*, 2004, "Organizational Learning Based on a Customizable Environment for Knowledge Management Using

- Intranet". In: *E-LEARN 2004 – World Conference on e-Learning in Corporate, Government, Healthcare & Higher Education*, v. 2, pp. 2626-2633, Washington, EUA.
- GARCIA, F., PIATTINI, M., RUIZ, F., *et al.*, 2006, "FMESP: Framework for the modeling and evaluation of software processes", *Journal of Systems Architecture*, v. 52, n. 11, pp. 627-639.
- GARCIA, F., SERRANO, M., CRUZ-LEMUS, J., *et al.*, 2007, "Managing software process measurement: A metamodel-based approach", *Information Sciences*, v. 177, n. 12, pp. 2570-2586.
- GARCIA, J., RIMAWI, Y., SANCHEZ, M.I., *et al.*, 2005, "RAMALA: A knowledge base for software process improvement". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 3792 LNCS, pp. 106-117, Budapest.
- GARG, P., JAZAYERI, M., 1996, "Process-centered Software Engineering Environments", *IEEE Computer Society Press*.
- GOMAA, H., FARRUKH, G.A., 1999, "Methods and tools for the automated configuration of distributed applications from reusable software architectures and components", *IEE Proceedings - Software*, v. 146, n. 6 (Dec. 1999), pp. 277-285.
- GOMES, A.G., 2001, *Avaliação de Processos de Software Baseada em Medições*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- GONÇALVES, F.M.G.S., ARAÚJO, S.A., MARINHO, F., *et al.*, 2006, "Experiência na definição e implantação do processo de análise e tomada de decisão alinhado ao CMMI-SW". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 383 - 390, Vila Velha - ES, Junho 2006.
- GONG, B., HE, X., LIU, W., 2006a, "A process improvement framework and a supporting software oriented to chinese small organizations", v. 3840 LNCS, pp. 277--286, Beijing.
- GONG, B., HE, X., LIU, W., 2006b, "A process improvement framework and a supporting software oriented to chinese small organizations". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 3840 LNCS, pp. 277-286, Beijing.
- GRESSE VON WANGENHEIM, C., VARKOI, T., SALVIANO, C.F., 2006, "Standard based software process assessments in small companies", v. 11, n. 3, pp. 329--335.
- GUERRA, E., TRAVASSOS, G.H., SANTOS, G., *et al.*, 2006, "Melhoria de Processos no Desenvolvimento de Software e Hardware - O Caso Maxtrack". In: *V Simpósio*

- Brasileiro de Qualidade de Software - SBQS 2006*, pp. 326 - 333, Vila Velha - ES, Junho 2006.
- GUIZZARDI, G., 2005, *Ontological Foundations for Structural Conceptual Models*, PhD Thesis, Departamento de Informática, Centre for Telematics and Information Technology, Enschede, The Netherlands.
- GUZMAN, J.G., GARCIA, R.L.C., DE SECO, A.A., *et al.*, 2006, "CASE STUDY: A practical approach for SPI in large Spanish companies", v. 11, n. 3, pp. 261--268.
- HARRISON, W., OSSHER, H., TARR, P., 2000, "Software Engineering Tools and Environments: A Roadmap". In: *Proceedings of the Conference on the Future of Software engineering - International Conference on Software Engineering*, pp. 261-277, Limerick, Ireland.
- HIKICHI, K., FUSHIDA, K., IIDA, H., *et al.*, 2006, "A software process tailoring system focusing to quantitative management plans", v. 4034 LNCS, pp. 441--446, Amsterdam.
- HOLZ, H., KÖNNECKER, A., MAURER, F., 2001, "Task-Specific Knowledge Management in a Process-Centred SEE". In: *Advances in Learning Software Organizations, v. 2176, Lecture Notes in Computer Science*, pp. 163-177.
- HOYER, C., CHROUST, G., 2006, "Evolving Standard Process Reference Models for Product Line Development". In: *Software Engineering and Advanced Applications, 2006. SEAA '06. 32nd EUROMICRO Conference on*, pp. 320-327.
- HUNDHAUSEN, R., 2006, *Working with Microsoft Visual Studio 2005 Team System*, 5th ed., Redmond, USA, Microsoft Press.
- IBM, "IBM Rational Software". In: <http://www.ibm.com/software/rational/>, acessado em 10/05/2008.
- IMPPROS, "Ambiente de Implementação Progressiva de Processo de Software". In: <http://www.cin.ufpe.br/~imppros/>, acessado em 17/01/2008.
- ISO/IEC, 1995, "Systems and software engineering – Software life cycle processes", *The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 12207:1995.
- ISO/IEC, 2002, "Information Technology - Amendment 1 to ISO/IEC 12207", *The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC-12207:Amd1.

- ISO/IEC, 2003, "Information Technology – Software Process Assessment", *Parts 1-9, The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 15504.
- ISO/IEC, 2004, "Information Technology - Amendment 2 to ISO/IEC 12207", *The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC-12207:Amd2.
- ISO/IEC, 2008a, "Systems and Software Engineering - System life cycle processes", *The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 15228:2008.
- ISO/IEC, 2008b, "Systems and software engineering – Software life cycle processes", *The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 12207:2008.
- JABLONSKI, S., FAERBER, M., 2007, "Integrated Management of Company Processes and Standard Processes: A Platform to Prepare and Perform Quality Management Appraisals". In: *Proceedings of the 5th International Workshop on Software Quality*.
- JARVINEN, J., 2000, "Measurement based continuous assessment of software engineering processes", *VTT Publ.*, n. 426, pp. 4--97.
- JBOSS, "JBoss RichFaces". In: <http://www.jboss.org/jbossrichfaces/>, acessado em 30/05/2008.
- JOKIKYYNY, T., 1999, "Using the Internet to communicate software metrics in a large organization", *IEEE*, v. 3, pp. 1947--1953, Rio de Janeiro, Braz.
- JUN, D., RUI, L., YI-MIN, H., 2007, "Software Processes Improvement and Specifications for Embedded Systems". In: *Software Engineering Research, Management & Applications, 2007. SERA 2007. 5th ACIS International Conference on*, pp. 13-18.
- KESS, P., HAAPASALO, H., 2002, "Knowledge creation through a project review process in software production", v. 80, n. 1, pp. 49--55.
- KETTUNEN, P., LAANTI, M., 2005, "How to steer an embedded software project: tactics for selecting the software process model", *Information & Software Technology*, v. 47, n. 9, pp. 587-608.
- KILPI, T., 2001, "Implementing a Software Metrics Program at Nokia", *IEEE Computer Society Press*, v. 18, n. 6, pp. 72-77.
- KOBIALKA, H., 1998, *Technical Report Number 15*, GMD Research Series, Sankt Augustin, Germany.

- KOENIG, S., 2003, "Integrated process and knowledge management for product definition, development and delivery". In: *Software: Science, Technology and Engineering, 2003. SmSTE '03. Proceedings. IEEE International Conference on*, pp. 133-141.
- KOLANO, P., DANG, Z., KEMMERER, R., 1999, "The design and analysis of real-time systems using the ASTRAL software development environment". In: *Annals of Software Engineering*, v. 7, pp. 177-210, Oulu.
- KURNIAWATI, F., JEFFERY, R., 2004, "The long-term effects of an EPG/ER in a small software organisation". In: *Proceedings of the Australian Software Engineering Conference, ASWEC*, v. 2004, pp. 128-136, Melbourne, Vic.
- KURNIAWATI, F., JEFFERY, R., 2006, "The use and effects of an electronic process guide and experience repository: a longitudinal study", *Information and Software Technology*, v. 48, n. 7, pp. 566-577.
- LEUNG, H.K.N., LI, L., YUZHONG, Q., 2007, "Automated support of software quality improvement", *International Journal of Quality and Reliability Management*, v. 24, n. 3, pp. 230-243.
- LIMA FILHO, J.M.S., FLORENCIO, A.L.A., COSTA, M.C.C., *et al.*, 2002, "Operação do SEPG na Motorola Brasil". In: *I Simpósio Brasileiro de Qualidade de Software - SBQS 2002*, pp. 206-212, Gramado - RS.
- LINDVALL, M., RUS, I., 2000, "Guest Editors' Introduction: Process Diversity in Software Development", v. 17, n. 4, pp. 14-18.
- MACEDO, C.C., LIMA, S.H.C.D., ROCHA, A.R., *et al.*, 2006, "Implantação de Melhoria de Processos de Software no Tribunal Superior Eleitoral". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 351 - 358, Vila Velha - ES, Junho 2006.
- MACHADO, L.F., 2000, *Modelo para Definição, Especialização e Instanciação de Processos de Software na Estação TABA*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- MAFRA, S., 2004, *Infra-estrutura para Automatização de Processos de Software*, Monografia de Final de Curso de B.Sc., DCC, UFRJ, Rio de Janeiro, Brasil.
- MALHEIROS, V., PAIM, F.R., GUZZO, H., *et al.*, 2006, "Uma Abordagem para Melhoria Contínua do Processo de Desenvolvimento de Software". In: *V Simpósio Brasileiro de Qualidade de Software, SBQS'2006*, pp. 218-230, Vila Velha - ES.
- MARÇAL, A.S.C., BUCHMANN, C.D.A.F., MORAIS, E.M., *et al.*, 2007, "Implantando uma Gestão Colaborativa de Processos aderentes ao CMMI em um Instituto de

- Inovação". In: *VI Simpósio Brasileiro de Qualidade de Software - SBQS 2007*, pp. 313 - 320, Porto de Galinhas - PE, 381 - 388.
- MARCZAK, S., AUDY, J.L.N., SÁ, L., 2005, "Proposta e aplicação de um instrumento de acompanhamento da implantação do SW CMM Nível 2". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 243 - 253, Porto Alegre - RS, Junho 2005.
- MARINHO, F.G., MONTEIRO, T.C., ARAÚJO, S.A., *et al.*, 2006, "Evoluindo do SW-CMM Nível 2 para o CMMI-SW nível 3: a experiência do Instituto Atlântico". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 310 - 317, Vila Velha - ES, Junho 2006.
- MARTINEZ, P., AMESCUA, A., GARCÍA, J., *et al.*, 2005, "Requirements for a knowledge management framework to be used in software intensive organizations", v. 2005, pp. 554--559, Las Vegas, NV.
- MASSOLAR, J.L., 1993, *SIM: Um Gerador Semi Automático de Documentos*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- MIAN, P.G., 2003, *ODEd: Uma Ferramenta de Apoio ao Desenvolvimento de Ontologias em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.
- MICROSOFT, "Visual Studio Team System 2005 Developer Center". In: <http://msdn2.microsoft.com/en-us/teamsystem/default.aspx>, acessado em 17/01/2008.
- MINH, N.N., WANG, A.I., CONRADI, R., 1997, "Total Software Process Model Evolution in EPOS Experience Report". In: *Proc. of the 19th Int. Conf. on Software Engineering*, pp. 390–399, May 17-23.
- MIRANDA, R., 2004, *Um Ambiente de Apoio a Comunidades de Prática no Contexto da Estação TABA*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- MOE, N.B., DINGSØYR, T., NILSEN, K.R., *et al.*, 2005, "Project web and electronic process guide as software process improvement". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 3792 LNCS, pp. 175-186, Budapest.
- MONTANGERO, C., DERNIAME, J.-C., KABA, B.A., *et al.*, 1999, "The Software Process: Modelling and Technology", pp. 1-14, 1999.
- MONTEIRO, L., SANT'ANNA, R., 2006, *Uma Infra-estrutura Web para a Construção de Ambientes de Desenvolvimento de Software na Estação TABA*, Projeto Final de Curso, Departamento de Ciência da Computação, UFRJ, Rio de Janeiro, Brasil.

- MONTEIRO, T., MARINHO, F., GONÇALVES, F.M.G.S., *et al.*, 2006, "Processo de Gestão de Processos: uma implementação da gerência de processos organizacionais do CMMI-SW Nível 3". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 359 - 366, Vila Velha - ES, Junho 2006.
- MONTONI, M., 2003, *Aquisição de Conhecimento: Uma Aplicação no Processo de Desenvolvimento de Software*, Dissertação de M.Sc., COPPE, Dissertação, Rio de Janeiro, Rj, Brasil.
- MONTONI, M., 2007, *Uma Abordagem para Condução de Iniciativas de Melhoria de Processos de Software*, Exame de Qualificação, COPPE, UFRJ, Rio de Janeiro, Brasil.
- MONTONI, M., CERDEIRAL, C., ZANETTI, D., *et al.*, 2007a, "Uma Abordagem para Condução de Iniciativas de Melhoria de Processos de Software", *PROQUALITY - Qualidade na Produção de Software*, v. 3, n. 3 (Novembro/2007), pp. 19-24.
- MONTONI, M., ROCHA, A.R., 2007, "Uma Abordagem para Gerência de Programas de Melhoria de Processo de Software". In: *V Workshop de Teses e Dissertações em Qualidade de Software, VI Simpósio Brasileiro de Qualidade de Software*, Porto de Galinhas, PE, Junho.
- MONTONI, M., SANTOS, G., FIGUEIREDO, S., *et al.*, 2006a, "Uma Abordagem de Garantia de Qualidade de Processos e Produtos de Software com Apoio de Gerência de Conhecimento na Estação TABA". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 87-99, Vila Velha - ES.
- MONTONI, M., SANTOS, G., ROCHA, A.R., *et al.*, 2006b, "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*, v. 4034 LNCS, pp. 249--262, Amsterdam.
- MONTONI, M., SANTOS, G., ROCHA, A.R., *et al.*, 2006c, "Taba Workstation: Supporting Software Process Deployment based on CMMI and MR-MPS.BR", *7th International Conference on Product Focused Software Process Improvement, Lecture Notes of Computer Science (LNCS)*.
- MONTONI, M., SANTOS, G., ROCHA, A.R., *et al.*, 2007b, "MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings". In: *Fifth Workshop on Software Quality held in conjunction with the 29th Int. Conference on Software Engineering (ICSE)*, Minneapolis, USA, May.
- MONTONI, M., SANTOS, G., VILLELA, K., *et al.*, 2005, "Enterprise-oriented software development environments to support software products and processes quality improvement". In: *Lecture Notes in Computer Science*, v. 3547, pp. 370-384, Oulu.

- MOREIRA, L.S.R., SAMRSLA, V.C., MÓRA, M., *et al.*, 2005, "Uma Avaliação das Equipes de Projeto em um Momento Pré-Avaliação Oficial em um Programa de Melhoria de Processo de Software visando o CMMI nível 2". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 271 - 278, Porto Alegre - RS, Junho 2005.
- MOURA, L., 1992, *Taxonomia de Ambientes de Desenvolvimento de Software*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- MUNCH, J., VIERIMAA, M., 2006, "Product-Focused Software Process Improvement, 7th International Conference, PROFES 2006, Proceedings". In: *Product-Focused Software Process Improvement*, v. 4034, Amsterdam, The Netherlands, June 12-14, 2006.
- MURADAS, F.M., 2006, *Processo de Avaliação MPS.BR: Definição e Ambiente de Apoio*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- NATALI, A.C., 2006, *Verificação e Validação de Software Apoiadas por Gerência de Conhecimento*, Exame de Qualificação, COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- NIAZI, M., WILSON, D., ZOWGHI, D., 2005, "A framework for assisting the design of effective software process improvement implementation strategies", *Journal of Systems and Software*, v. 78, n. 2, pp. 204-222.
- NIELBOCK, D., 1997, *Uma Ferramenta para Planejamento e Produção de Desenvolvimento de Software*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- NIKULA, U., SAJANIEMI, J., 2005, "Tackling the complexity of requirements engineering process improvement by partitioning the improvement task". In: *Proceedings of the Australian Software Engineering Conference, ASWEC*, v. 2005, pp. 48-57, Brisbane.
- NUNES, E.D., PINTO, R., ROCHA, A.R., *et al.*, 2006, "MPS.BR Nível E - Uma Avaliação em Verde e Amarelo". In: *V Simpósio Brasileiro de Qualidade de Software - SBQS 2006*, pp. 318 - 325, Vila Velha - ES, Junho 2006.
- OGASAWARA, H., ISHIKAWA, T., MORIYA, T., 2006, "Practical approach to development of SPI activities in a large organization~Toshiba's SPI history since 2000~". In: *Proceedings - International Conference on Software Engineering*, v. 2006, pp. 595-599, Shanghai.
- OGASAWARA, H., KUSANAGI, T., FUJIMAKI, N., 2005, "Practical Approach to Evolve SPI Activities in a Large-Scale Organization". In: *3rd World Congress for Software Quality*, Munich, Germany, 26 - 30 September.
- OGASAWARA, H., YAMADA, A., KUSANAGI, T., *et al.*, 2002, "How to Effectively Promote the Software Process Improvement Activities in a Large-Scale

- Organization". In: *Software Quality - ECSQ 2002, 7th International Conference*, v. 2349, pp. 124-134, Helsinki, Finland, June 9-13, 2002.
- OLIVEIRA, K.M., 1999, *Modelo para Construção de Ambientes de Desenvolvimento de Software Orientados a Domínio*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- OLIVEIRA, S.R.B., VASCONCELOS, A.M.L., 2006, "A Continuous Improvement Model in ImPProS". In: *Computer Software and Applications Conference, 2006. COMPSAC '06. 30th Annual International*, v. 2, pp. 370-371.
- OSTERWEIL, L.J., 1996, "Improving the quality of software quality determination processes", pp. 90-105, 1996.
- PARK, E.-J., KIM, H.-K., LEE, R.Y., 2007, "Frameworks of Integration Repository for Software Process Improvement using SOA". In: *Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on*, pp. 200-206.
- PAULK, M.C., CURTIS, B., C., B., *et al.*, 1993a, *Capability Maturity Model for Software, Version 1.1*, CMU/SEI-93-TR-24, DTIC Number ADA263403 ed., Software Engineering Institute.
- PAULK, M.C., CURTIS, B., CHRISSIS, M.B., *et al.*, 1993b, "Capability maturity model, version 1.1", *Software, IEEE*, v. 10, n. 4, pp. 18-27.
- PAXIÚBA, C., PEREIRA, M., REIS, C.A.L., *et al.*, 2007, "Acompanhamento e Avaliação de Projetos através da Monitoração de Eventos em um Ambiente de Gestão de Processos de Software." In: *Simpósio Brasileiro de Qualidade de Software - SBQS 2007*, pp. 15-29, Porto de Galinhas - PE, Junho 2007.
- PMI, 2004, *A Guide to the Project Management Body of Knowledge*, 3th ed., USA, PMBOK Guides.
- PRACCHIA, L., 2004, "TheAV-8B team learns synergy of EVM and TSP accelerates software process improvement", n. 1, pp. 20--22.
- PRIKLADNICKI, R., GOMES, G., MAJDENBAUM, A., *et al.*, 2005, "Um Caso Prático de Implantação de Gerência de Risco em Ambientes de Desenvolvimento Distribuído de Software, baseado no Modelo CMMI". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 95 - 102, Porto Alegre - RS, Junho 2005.
- RAMACHANDRAN, M., 2005, "A process improvement framework for XP based SMEs", v. 3556, pp. 202--205, Sheffield.
- ROCHA, A.R., MONTONI, M., SANTOS, G., *et al.*, 2005a, "Reference model for software process improvement: A Brazilian experience". In: *Lecture Notes in*

- Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 3792 LNCS, pp. 130-141, Budapest.
- ROCHA, A.R., MONTONI, M., SANTOS, G., *et al.*, 2005b, "Estação TABA: Uma Infra-estrutura para Implantação do Modelo de Referência para Melhoria de Processos de Software". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 49 - 61, Porto Alegre - RS, Junho 2005.
- ROCHA, A.R., MONTONI, M., SANTOS, G., *et al.*, 2006, "Success Factors and Difficulties in Software Process Deployment Experiences based on CMMI and MR-MPS.BR". In: *6th International Workshop on Learning Software Organizations (LSO'2006)*, pp. 77-87, Rio de Janeiro, RJ, Brasil, Setembro.
- ROCHA, A.R.C.D., SOUZA, J.M., AGUIAR, T.C., 1990, "TABA: A Heuristic Workstation for Software Development". In: *COMPEURO 90*, pp. 126-129, Tel Aviv, Israel, Maio.
- ROSSI, S., MARTTIIN, P., 2000, "Adoption of Integrated Process and Product Support for Software Engineering in SP Jyväskylä". In: *International Conference on software Methods and Tools*, pp. 97-105, Wollongong, Australia, Nov. 2000.
- ROUT, T.P., TUFFLEY, A., 2007, "Harmonizing ISO/IEC 15504 and CMMI", v. 12, n. 4, pp. 361-371.
- RUIZ, M., RAMOS, I., TORO, M., 2002, "A Dynamic Integrated Framework for Software Process Improvement", *Softw. Qual. J.*, v. 10, n. 2, pp. 181--194.
- RUY, F.B., 2006, *Semântica em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.
- SAKAMOTO, K., NAKAKOJI, K., TAKAGI, Y., *et al.*, 1998, "Toward computational support for software process improvement activities", *IEEE Comp Soc*, pp. 22--31, Kyoto, Jpn.
- SANTOS, G., 2003, *Representação da Distribuição do Conhecimento, Habilidades e Experiências através da Estrutura Organizacional*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- SANTOS, G., MONTONI, M., FIGUEIREDO, S., *et al.*, 2008a, "Uma Estratégia de Gerência de Configuração de Ativos de Processos de Software Apoiada por um Ambiente de Engenharia de Software Centrado em Processos". In: *VII Simpósio Brasileiro de Qualidade de Software - SBQS 2008*, Florianópolis - SC, Junho 2008.

- SANTOS, G., MONTONI, M., FIGUEIREDO, S., *et al.*, 2007a, "SPI-KM - Lessons Learned from Applying a Software Process Improvement Strategy Supported by Knowledge Management", *Product-Focused Software Process Improvement*.
- SANTOS, G., MONTONI, M., KATSURAYAMA, A.E., *et al.*, 2008b, "Aplicação da Estratégia SPI-KM para Apoiar a Implementação do MPS.BR Níveis G e F em Pequenas e Médias Empresas do Rio de Janeiro". In: *VII Simpósio Brasileiro de Qualidade de Software - SBQS 2008*, Florianópolis - SC, Junho 2008.
- SANTOS, G., MONTONI, M., VASCONCELLOS, J., *et al.*, 2007b, "Implementing Software Process Improvement Initiatives in Small and Medium-Size Enterprises in Brazil". In: *Quality of Information and Communications Technology, 2007. QUATIC 2007. 6th International Conference on the*, pp. 187-198, Lisboa, Portugal.
- SANTOS, G., VILLELA, K., MONTONI, M., *et al.*, 2005, "Knowledge management in a software development environment to support software processes deployment", v. 3782 NAI, pp. 111-120, Kaiserslautern, Germany.
- SANTOS, G., VILLELA, K., SCHNAIDER, L., *et al.*, 2004, "Building Ontology Based Tools for a Software Development Environment". In: *Advances in Learning Software Organizations*, pp. 19-30, Banff, Canada.
- SANTOS, G., ZLOT, F., 1999, *Definição e Instanciação e Ambientes na Estação TABA*, Projeto Final de Curso, Departamento de Ciência da Computação, UFRJ, Rio de Janeiro, Brasil.
- SCHEID, M., PESSOA, M.V., GOMES, R.F., *et al.*, 2007, "Implantação do MR-MPS Nível E no Centro de Computação da Aeronáutica de São José dos Campos", *ProQuality (UFLA)*.
- SCHNAIDER, L., 2003, *Planejamento da Alocação de Recursos Humanos em Ambientes de Desenvolvimento, de Software Orientados à Organização*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- SCHNAIDER, L., SANTOS, G., MONTONI, M., *et al.*, 2004, "Uma Abordagem para Medição e Análise em Projetos de Desenvolvimento de Software". In: *III Simpósio Brasileiro de Qualidade de Software - SBQS 2004*, Brasília - DF.
- SCOTT, L., CARVALHO, L., JEFFERY, R., *et al.*, 2002, "Understanding the use of an electronic process guide", *Inf Software Technol*, v. 44, n. 10, pp. 601--616.
- SCOTT, L., JEFFERY, R., CARVALHO, L., *et al.*, 2001, "Practical software process improvement - the IMPACT project". In: *Software Engineering Conference, 2001. Proceedings. 2001 Australian*, pp. 182-189.

- SEI, "FAQ ISO/IEC 15504". In: <http://www.sei.cmu.edu/cmml/faq/15504-faq.html>, acessado em 10/05/2008.
- SIKAKAS, K.V., GEORGIADOU, E., 2002, "Empirical Measurement of the Effects of Cultural Diversity on Software Quality Management", *Kluwer Academic Publishers*, v. 10, n. 2, pp. 169-180.
- SILVA FILHO, R.C., 2006, *Uma Abordagem para Avaliação de Propostas de Melhoria em Processos de Software*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.
- SINGH, I., STEARNS, B., JOHNSON, M., *et al.*, 2002, *Designing Enterprise Applications with the J2EE(TM) Platform*, 2nd Edition ed., Prentice Hall PTR.
- SIQUEIRA, F.L., SILVA, P.S.M., 2005, "Mapeamento da NBR ISO/IEC 12207 em Papéis da Gerência de Projetos de Software". In: *IV Simpósio Brasileiro de Qualidade de Software - SBQS 2005*, pp. 103 - 117, Porto Alegre - RS, Junho 2005.
- SOFTEX, 2007, "MPS.BR – Guia Geral".
- SOFTEX, "Artigos MPS.BR". In: <http://www.softex.br/portal/mpsbr/artigos/default.asp>, acessado em 10/05/2008.
- SOFTEX, "FAQ MPS.BR". In: <http://www.softex.br/portal/mpsbr/faq/faqDiversos.asp>, acessado em 10/05/2008.
- STARK, J.A., CROCKER, R., 2003, "Trends in Software Process: The PSP and Agile Methods", *IEEE Computer Society Press*, v. 20, n. 3, pp. 89-91.
- SUN, "Java 2 Platform, Enterprise Edition (J2EE) Overview". In: <http://java.sun.com/j2ee/overview.html>, acessado em 30/05/2008.
- SUN, "Java EE at a Glance". In: <http://java.sun.com/javace/>, acessado em 30/05/2008.
- SVN, "SVN - Version Control System". In: <http://subversion.tigris.org/>, acessado em 30/05/2008.
- TAVARES, D.P.D., FABBRI, S.C.P.F., SANCHES, R., 2002a, "Uma Abordagem para Melhoria Contínua do Processo de Desenvolvimento de Software". In: *V Simpósio Brasileiro de Qualidade de Software, SBQS'2006*, pp. 189-197, Vila Velha - ES.
- TAVARES, H.C., PAIM, F.R.S., CARVALHO, A.E., 2002b, "Implantando CMM Nível 2: A Estratégia SERPRO". In: *I Simpósio Brasileiro de Qualidade de Software - SBQS 2002*, pp. 181-188, Gramado - RS.

- TIANFIELD, H., 2003, "An autonomic framework for quantitative software process improvement". In: *Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on*, pp. 446-450.
- TRAVASSOS, G.H., 1994, *Modelo de Integração de Ferramentas da Estação TABA*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- VARKOI, T., 2002, "Management of continuous software process improvement", v. 1, pp. 334--337, Cambridge.
- VASCONCELOS, F., 1997, *Reutilização de Processos de Desenvolvimento de Software baseada em Padrões*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- VASQUES, R.C., GOMES, I.M., "CMMI, MSF e Tecnologias Microsoft – uma visão de integração". In: http://www.isdbrasil.com.br/artigo_cmmi_msf.html, acessado em 17/01/2008.
- VILLELA, K., 2004, *Definição e Construção de Ambientes de Desenvolvimento de Software Orientados à Organização*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- VILLELA, K., SANTOS, G., MONTONI, M., *et al.*, 2004, "Definição de Processos em Ambientes de Desenvolvimento de Software Orientados a Organização". In: *III Simpósio Brasileiro de Qualidade de Software - SBQS 2004*, pp. 4-18, Brasília - DF.
- VILLELA, K., SANTOS, G., SCHNAIDER, L., *et al.*, 2005, "The use of an enterprise ontology to support knowledge management in software development environments", *Journal of the Brazilian Computer Society*, v. 11, n. 2, pp. 45 - 59.
- WANG, Q., LI, M., 2005a, "Measuring and improving software process in China", pp. 183-192, Queensland.
- WANG, Q., LI, M., 2005b, "Measuring and improving software process in China". In: *2005 International Symposium on Empirical Software Engineering, ISESE 2005*, pp. 183-192, Queensland.
- WANG, Q., LI, M., 2006, "Software process management: Practices in China", v. 3840 NCS, pp. 317-331, Beijing, China.
- WANG, Q., XIAO, J., LI, M., *et al.*, 2006, "A Process-Agent construction method for software process modeling in SoftPM". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 3966 LNCS, pp. 204-213, Shanghai.
- WANG, R., WANG, L., GENG, C., 2002, "The Design of VPP Software Development Environment". In: *IEEE Instrumentation and Measurement Technology Conference*, pp. 403-408, Anchorage, USA, May. 2002.

- WANG, Y., BRYANT, A., 2002, "Process-based software engineering: Building the infrastructures", v. 14, n. 1-4, pp. 9--37.
- WEIER, M.H., MURPHY, C., "Can General Motors Change How Companies Think About IT Outsourcing?" In: <http://www.informationweek.com/showArticle.jhtml?articleID=201803575>, acessado em 19/01/2008.
- WERNECK, V.M., 1995, *Ambiente para Desenvolvimento de Software Baseado em Conhecimento*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- WERNECK, V.M.B., 1990, *Taxonomia de Domínios de Aplicação*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- WERNER, C.M., TRAVASSOS, G.H., ROCHA, A.R., *et al.*, 1997, "Memphis: A Reuse Based O. O. Software Development Environment". In: *Proceedings of TOOLS*, Beijing, China, Sep. 1997.
- XU, R., XUE, Y., NIE, P., *et al.*, 2006, "Research on CMMI-based software process metrics". In: *First International Multi- Symposiums on Computer and Computational Sciences, IMSCCS'06*, v. 2, pp. 391-397, Hangzhou, Zhejiang.
- ZAHARAN, S., 1998, *Software Process Improvement – Practical Guidelines for Business Success*, 5th ed., USA, Addison-Wesley.
- ZLOT, F., 2002, *Conhecimento de Tarefa em Ambientes de Desenvolvimento de Software Orientados a Domínio*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.

ANEXO I - LINGUAGEM PARA MODELAGEM DE PROCESSOS ORGANIZACIONAIS

Uma série de linguagens e formalismos para modelagem de processo de software têm sido propostas, mas a complexidade das linguagens existentes e a forte orientação à execução e à automação dos processos dificultam a adoção dos mesmos na prática, onde o mais importante é a comunicação e entendimento dos processos (VILLELA, 2004). REIS et al. (2002) propuseram uma linguagem de modelagem gráfica para descrever processos de software com o objetivo de apoiar a instanciação, considerando que modelagem, instanciação e execução são frequentemente intercaladas no ciclo de vida do processo. O Object Management Group (OMG) definiu um meta-modelo para engenharia de processos de software, que pode ser considerado uma UML (Unified Modeling Language) Profile, utilizando estereótipos e diagramas UML básicos para fornecer diferentes perspectivas de um modelo de processo de software (OMG, 2002). Este anexo descreve a linguagem para modelagem de processos organizacionais em uso para modelagem de processos na Estação Taba. Esta linguagem permite que modelos de processos de software sejam definidos de forma sistemática e incremental e utilizados para guiar os desenvolvedores nas atividades a serem executadas.

1 Introdução

Dois tipos de processo são importantes no contexto da Estação Taba (VILLELA, 2004): os processos de software, que têm a finalidade de orientar o desenvolvimento, reparo e melhoria de produtos de software; e os processos da organização e, quando pertinente, das suas organizações clientes que podem ser apoiados, controlados ou automatizados por produtos de software. Na Estação Taba, os modelos de processos de software são prescritivos, ou seja, fornecem instruções de como os processos devem ser executados para que os objetivos pretendidos sejam atingidos; enquanto os modelos dos demais processos são descritivos, pois a finalidade é promover o entendimento desses processos e facilitar a comunicação, possibilitando uma visão compartilhada dos mesmos. Esta distinção resulta em tratamentos diferenciados para os modelos de processo neste contexto.

Neste anexo não se discutirá como os modelos de processos de software podem ser definidos de forma sistemática e incremental e utilizados para guiar os desenvolvedores nas

atividades a serem executadas. O foco é a linguagem de modelagem a ser utilizada neste contexto, que deve ser a mais simples e intuitiva possível.

No que se refere à modelagem de processos organizacionais quaisquer, a *Rational Software* propôs uma *UML Profile* para modelagem de negócio (RATIONAL, 2001) e muitos Sistemas de Gerência de *Workflow* e Ferramentas de Definição de Processos têm sido desenvolvidos e comercializados. Diante da variedade de produtos de *workflow* no mercado, muitas vezes voltados para necessidades específicas, a *Workflow Management Coalition (WFMC)* definiu um modelo de referência para promover a integração de produtos de *workflow* (HOLLINGSWORTH, 1995). Este modelo inclui um meta-modelo para definição e intercâmbio de processos (WFMC, 1999), que estabelece um conjunto básico de tipos de objeto para a troca de definições de processo simples e possibilita que outros tipos de objetos sejam adicionados por extensões específicas de vendedor. O *Ultimus Workflow Suite* e o *ARIS Easy Design* e *ARIS Toolset* são exemplos de produtos de *workflow* que utilizam linguagem de modelagem gráfica para definição de processos, sendo que o *ARIS Easy Design* permite a documentação e acesso de conhecimentos associados aos processos utilizando os objetos gráficos definidos em (ALLWEYER, 1999). No entanto, enquanto o *Ultimus Workflow Suite* não contempla a representação dos conhecimentos requeridos e produzidos ao longo dos processos, os produtos ARIS prevêem um conjunto de 34 objetos com desenhos, muitas vezes, bastante semelhantes e o *ARIS Easy Design* utiliza objetos com desenhos desnecessariamente complexos (por exemplo: pessoa/organização, função, conhecimento documentado).

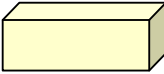





A próxima seção apresenta uma linguagem gráfica para modelagem de processos organizacionais quaisquer, inclusive os de software, com fins descritivos. A linguagem difere das demais por que: (i) busca definir um conjunto mínimo de objetos, suficientes para representar os conceitos da ontologia de organização considerados pertinentes no contexto, as primitivas mencionadas em (WFMC, 1999) (CHANG *et al.*, 2000) e alguns conceitos que se mostraram necessários na prática; (ii) permite a representação dos conhecimentos requeridos e produzidos ao longo dos processos e (iii) utiliza objetos gráficos simples de desenhar. O resultado é uma linguagem fácil de usar, intuitiva, que produz modelos fáceis de serem lidos, visualmente agradáveis, além de compatíveis com o desejo de modelar o fluxo de conhecimento ao longo dos processos organizacionais.






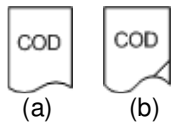



2 Linguagem para Modelagem de Processos Organizacionais

A linguagem para modelagem de processos organizacionais descrita neste anexo é composta de elementos gráficos que podem ser do tipo área, objeto ou ligação, onde uma ligação estabelece uma relação entre dois objetos e uma área agrupa objetos, definindo um contexto para os mesmos. Objetos ainda permitem adornos, utilizados para representar explicitamente características dos objetos. Esta linguagem foi definida originalmente por VILLELA (2004) e foi revista para melhorar a apresentação e diagramação de alguns de seus elementos, notadamente os referentes a artefatos e atores.

A seguir, cada elemento da linguagem é apresentado nas tabelas a seguir.

Tabela 1 - Definição e Notação dos Objetos

Objeto	Notação	Definição
Processo		Objeto referente a um conjunto de atividades estruturadas e destinadas a resultar em um artefato ou serviço de valor para a organização ou para um determinado cliente ou mercado. Implica em uma ordenação específica das atividades com começo, fim, insumos e produtos claramente identificados (VILLELA, 2004). <i>Atributos Especiais:</i> Origem (Interno, Externo)
Evento		Objeto que representa um acontecimento no ambiente que provoca o início ou fim de um processo. A notação é proveniente do produto comercial de <i>workflow</i> ARIS <i>ToolSet</i> .
Ator	 COD	Objeto que representa uma pessoa, agente ou unidade organizacional. <i>Pessoas</i> são fundamentais para o funcionamento de uma organização, atuando na execução de atividades necessárias ao cumprimento da missão da organização; um <i>Agente</i> representa uma especificação de um perfil necessário para que a organização cumpra a sua missão através da execução de atividades, pode representar um cargo ou uma posição; <i>Unidades Organizacionais</i> são agrupamentos de componentes da organização (por exemplo: atividades e pessoas) de acordo com a homogeneidade de conteúdo para que a organização possa ser econômica e eficiente (VILLELA, 2004). A notação foi baseada na proposta por KRUCHTEN (2000) para representação dos <i>workflows</i> básicos do <i>Rational Unified Process</i> e posteriormente adaptada de (SOFTEX, 2007). <i>Atributos Especiais:</i> Código do Ator descrito numa Área de Legenda
Atividade		Objeto que representa uma ação de transformação que pode requerer competências para a sua execução e, ao ser executada, fazer uso de bens de produção (recursos), consumir matérias-primas e artefatos de entrada (insumos), além de produzir artefatos de saída (produtos) (VILLELA, 2004). A notação foi utilizada por KRUCHTEN (2000) para representação dos <i>workflows</i> básicos do <i>Rational Unified Process</i> . <i>Atributos Especiais:</i> Origem (Interna, Externa) Granularidade (Elementar ou Composta)
Estado Inicial		Objeto puramente notacional, proveniente dos diagramas de estado da UML e que indica onde é iniciado o fluxo de atividades que definem um processo ou uma atividade composta
Estado Final		Objeto puramente notacional, proveniente dos diagramas de estado e que indica onde é encerrado o fluxo de atividades que definem um processo ou uma atividade composta

Objeto	Notação	Definição
Conhecimento Explícito		Objeto que representa um conhecimento que pode ser expresso em palavras e números e ser facilmente transmitido e compartilhado. A notação foi proposta por (ALLWEYER, 1999).
Conhecimento Implícito		Objeto que representa um conhecimento que é altamente pessoal e difícil de formalizar, o que o torna também difícil de ser compartilhado. A notação foi proposta por (ALLWEYER, 1999).
Comunicação		Objeto que representa a comunicação de dados ou informações a partir da, ou para a, execução de uma atividade. A comunicação pode ser verbal ou escrita e exemplos são e-mail e fax.
Repositório (Meio Magnético)		Objeto que representa um meio magnético para o armazenamento de dados e informações. A notação é proveniente do produto comercial de <i>workflow ARIS ToolSet</i> .
Arquivo (Local Físico)		Objeto que representa um local físico para armazenamento de documentos e comunicações escritas.
Artefato	-	Objeto que representa qualquer elemento produzido pelo homem e não por causas naturais, podendo exercer diferentes papéis em uma organização, tais como o de insumo ou produto de uma atividade (VILLELA, 2004). Não possui uma representação própria, devendo ser utilizado algum de seus tipos: Documento, Componente de Hardware, Componente de Software, Peça, Matéria, Prima ou Bem.
Documento		Objeto que representa um artefato escrito ou que pode ser impresso, cuja função é fornecer informação, conhecimento ou prova (VILLELA, 2004). A notação foi baseada na do produto comercial de <i>workflow ARIS ToolSet</i> e posteriormente adaptada de (SOFTEX, 2007). Um documento pode ser de dois tipos: (a) Documento de entrada, ou seja, insumo para uma Atividade; (b) Documento de saída, ou seja, produto de uma Atividade. <i>Atributos Especiais:</i> Código do Documento descrito numa Área de Legenda
Componente	-	Objeto que representa um artefato que é utilizado na composição de outros artefatos, mas que não é documento (VILLELA, 2004). Não possui uma representação própria, devendo ser utilizado algum de seus tipos: Componente de Hardware, Componente de Software ou Peça.
Componente de Hardware		Objeto que representa um componente especificamente utilizado na composição de artefatos de hardware (VILLELA, 2004). A notação é baseada na notação de componente da UML.
Componente de Software		Objeto que representa um componente constituído por conjuntos de instruções e dados, podendo ser produzidos em uma atividade de desenvolvimento de software ou ser obtidos de uma biblioteca de componentes (VILLELA, 2004). A notação é baseada na notação de componente da UML.
Peça		Objeto que representa um componente de propósito geral, que podem entrar na composição de equipamentos ou de artefatos de hardware (VILLELA, 2004).




Objeto	Notação	Definição
Bem		Objeto que representa um artefato concluído, no sentido de não participar da composição de outros artefatos, além de não ser documento (VILLELA, 2004). Em outras palavras, bens são artefatos que já passaram por todas as transformações necessárias para que incorporassem suas funcionalidades e que, conseqüentemente, fazem parte do seu processo de produção. Bens podem ser classificados em bens de usufruto e bens de produção. A notação fornecida pode ser substituída por uma mais significativa para o objeto específico do modelo como, por exemplo, o logotipo do software. <i>Atributos Especiais:</i> Tipo (Usufruto, Software, Hardware e Equipamento de Produção)
Matéria-Prima		Objeto que representa uma matéria bruta ou pouco elaborada utilizada como insumo em uma atividade, no sentido de ser um objeto de transformação desta atividade (VILLELA, 2004).
Nota Explicativa		Objeto que permite que notas explicativas sejam adicionadas ao modelo. <i>Atributos Especiais:</i> Texto

Tabela 2 - Definição e Notação dos Adornos

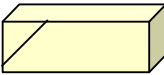
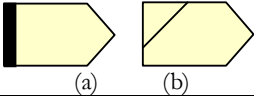
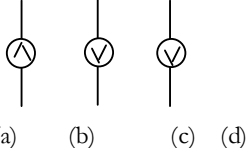

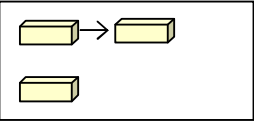
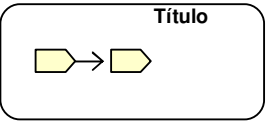
Objeto	Notação com Adornos	Definição dos Adornos
Processo		Adorno que indica que o processo é externo, ou seja, que é executado por outra organização ou em outro contexto.
Atividade		(a) Adorno que indica que a atividade é composta, o que significa que ela pode ser decomposta em subatividades; (b) Adorno que indica que a atividade é externa, ou seja, que é executada por outra organização.
Operação Lógica		(a) Adorno que indica a operação lógica E; (b) Adorno que indica a operação lógica OU; (c) Adorno que indica a operação lógica OU Exclusivo; (d) Notação sem adorno, que é utilizada como elemento de junção após uso de uma operação lógica que atua como elemento de divisão.
Conhecimento Explícito		Adorno que indica que foi especificado um caminho para acesso ao conhecimento disponível em meio magnético. Este adorno só deve ser utilizado se a visualização do modelo for apoiada por uma ferramenta de software que permita o acesso ao conhecimento. <i>Atributos Especiais:</i> Localização do Arquivo

Tabela 3 – Definição e Notação das Áreas

Objeto	Notação	Definição
Grupo de Processos		Área que agrupa processos relacionados. Opcionalmente pode conter um título.
Grupo de Atividades		Área que agrupa atividades, provavelmente parte de uma macro-atividade ou de um processo. Opcionalmente pode conter o nome da atividade ou processo no título presente na parte superior. A definição de cada Ator deve ser feita em cada atividade. <i>Atributos Especiais:</i> Título

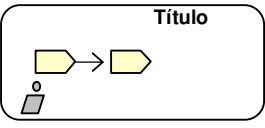
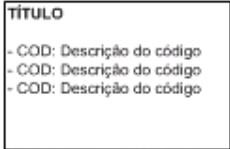
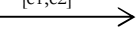



Objeto	Notação	Definição
Área de Ator		Área que agrupa atividades, provavelmente parte de uma macro-atividade ou de um processo, executadas por um ator ou grupo de atores. O ator ou o grupo de atores também precisa estar contido na área. O Ator ou grupo de atores é assumido como associado com todas as atividades apresentadas. Opcionalmente pode conter o nome da atividade ou processo no título presente na parte superior. <i>Atributos Especiais:</i> Título
Área de Legenda		Área que agrupa a descrição de códigos utilizados durante a elaboração dos diagramas. Cada Área de Legenda só pode se referir a Documentos ou Atores, nunca os dois ao mesmo tempo. <i>Atributos Especiais:</i> Título, Código do Ator ou do Documento, Descrição do Ator ou do Documento

Tabela 4 – Definição e Notação das Ligações

Objeto	Notação	Definição
Fluxo de Controle		Ligação que indica a passagem de controle do objeto origem para o objeto destino. O c1 e o c2 indicados na notação são os rótulos das condições estabelecidas para que a passagem de controle ocorra. <i>Atributo Especial:</i> Condição, formada por rótulo e descrição
Fluxo de Entrada/Saída		Ligação que estabelece um insumo (se o fluxo é de entrada) ou um produto de uma atividade (se o fluxo é de saída). Quando o objeto de origem ou destino é um armazenador (repositório ou arquivo), a notação pode incluir os rótulos das informações trafegadas, existindo, então, um atributo especial. Note que há uma notação especial para Documentos que sejam insumo ou produto de uma atividade. <i>Atributo Especial:</i> Informação, formada por rótulo e descrição
Ligação Não Direcionada		Ligação que não indica passagem de controle nem estabelece insumos e produtos para uma atividade, sendo utilizada para conectar bens de produção (software, hardware e equipamentos) utilizados como recursos para execução das atividades e para conectar eventos que atuam sobre processos, provocando o seu início ou fim. No segundo caso, um atributo especial é definido. <i>Atributo Especial:</i> Papel do Evento (Iniciador, Terminador)
Ligação para Nota Explicativa		Ligação que estabelece que uma nota explicativa é referente a um elemento do modelo.

As regras para composição dos elementos gráficos da linguagem são:

- Um Grupo Processo agrupa um¹ ou mais Processos e um Processo pode estar ou não em um Grupo Processo;

¹ Zero é admitido enquanto o modelo está sendo elaborado.

- Uma Área Ator agrupa um ou mais Atores e um Ator sempre está contido em uma Área Ator;
- Uma Área Ator agrupa uma¹ ou mais Atividades e uma Atividade pode estar ou não contida em uma Área Ator;
- O uso de uma Área de Ator não é obrigatório, neste caso deve-se representar diretamente a associação de um Ator uma Atividade;
- Os Atores devem ser exibidos no canto inferior esquerdo da Atividade;
- Apenas um elemento referente a Ator pode estar associado uma Atividade, no caso de este elemento representar, na prática, mais de um Ator, deve-se listar todos os códigos referentes a eles separados por vírgula;
- Todos os códigos referentes a Atores presentes em um diagrama devem estar descritos em uma Área de Legenda;
- Um Fluxo de Controle pode conectar um Processo a outro Processo ou a uma Operação Lógica;
- Um Fluxo de Controle pode conectar uma Atividade a outra Atividade, a uma Operação Lógica, a um Estado Inicial ou a um Estado Final, sendo que, quando a ligação envolve um dos dois últimos objetos, o Estado Inicial é obrigatoriamente a origem e o Estado Final é obrigatoriamente o destino;
- Tanto Processos quanto Atividades admitem apenas um Fluxo de Controle com destino na atividade/processo;
- Um Fluxo de Controle pode conectar uma Operação Lógica a um Processo, uma atividade, outra Operação Lógica, um Estado Inicial ou um Estado Final. Quando a ligação envolve um dos dois últimos objetos, o Estado Inicial é obrigatoriamente a origem e o Estado Final é obrigatoriamente o destino;
- Uma Operação Lógica implementa uma primitiva de Junção (*Join*) ou de Divisão (*Split*), o que significa, respectivamente, que apenas um Fluxo de Controle tem origem na Operação Lógica ou apenas um Fluxo de Controle tem como destino a Operação Lógica;
- Uma Ligação Não Direcionada pode conectar um Processo e um Evento ou uma Atividade e um Bem;
- Através de Ligações Não Direcionadas, um Processo pode estar conectado a zero ou mais Eventos, e um Evento pode estar conectado a um⁶ ou mais Processos;

- Através de Ligações Não Direcionadas, uma Atividade pode estar conectada a zero ou mais Bens, e um Bem pode estar conectado a zero ou mais Atividades;
- Os Documentos são ligados diretamente a uma Atividade sem a utilização de Fluxo de Entrada/Saída;
- Os Documentos de entrada (ou seja, insumo para uma Atividade) devem ser exibidos no canto superior esquerdo da Atividade;
- Os Documentos de saída (ou seja, produto de uma Atividade) devem ser exibidos no canto inferior direito da Atividade;
- Apenas um elemento referente a Documento de entrada (ou seja, insumo para uma Atividade) pode estar associado uma Atividade, no caso de este elemento representar, na prática, mais de um Documento, deve-se listar todos os códigos referentes a eles separados por vírgula;
- Apenas um elemento referente a Documento de saída (ou seja, produto de uma Atividade) pode estar associado uma Atividade, no caso de este elemento representar, na prática, mais de um Documento, deve-se listar todos os códigos referentes a eles separados por vírgula;
- Todos os códigos referentes a Documentos presentes em um diagrama devem estar descritos em uma Área de Legenda;
- Um Fluxo de Entrada/Saída pode conectar uma Atividade a um Conhecimento Implícito, a um Conhecimento Explícito, a uma Comunicação, a um Repositório, a um Arquivo, a um Componente de Hardware ou de Software, a uma Peça, a um Bem ou a uma Matéria-Prima, sendo que, quando a ligação envolve Matéria-Prima, este objeto é obrigatoriamente a origem;
- Através de Fluxos de Entrada/Saída, uma Atividade pode estar conectada a zero ou mais dos elementos pertinentes e cada um desses elementos pode estar conectado a uma² ou mais Atividades;
- Uma Ligação para Nota Explicativa pode conectar qualquer elemento da linguagem a uma Nota Explicativa;
- Através de Ligações para Nota Explicativa, uma Nota Explicativa pode estar conectada a zero ou mais elementos da linguagem e um elemento da linguagem pode estar conectado a zero ou mais Notas Explicativas;

² Zero é admitido enquanto o modelo está sendo elaborado. Além disso, Bens podem existir no modelo sem estar associados a Atividades através de Fluxos de Entrada/Saída.

- Uma mesma Área de Legenda não pode ser utilizada para descrever os códigos de Documentos e de Atores.

Algumas considerações sobre o objeto Operação Lógica ainda são oportunas. Com o objetivo de modelar *workflows*, CHANG *et al.* (2000) discutem como as primitivas *AND-Join*, *AND-Split*, *OR-Join*, *OR-Split*, Iteração e Casualidade podem ser expressas utilizando o diagrama de atividade da UML (Figura 1).

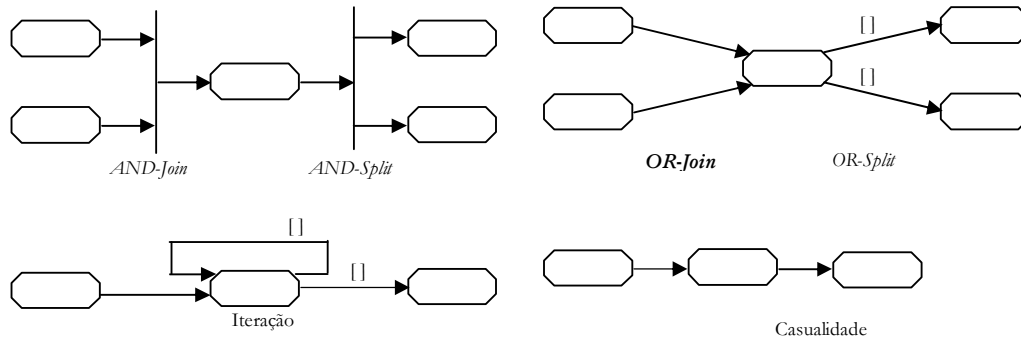


Figura 1 - Primitivas utilizando a Notação do Diagrama de Atividade UML

No meta-modelo definido em (WFMC, 1999), a descrição de uma atividade pode ser dividida em elemento de junção, corpo e elemento de divisão, onde o elemento de junção trata das restrições sobre as transições de entrada, o corpo pode conter uma repetição e o elemento de divisão trata das restrições sobre as transições de saída. No entanto, as restrições sobre as transições são especificadas através do uso das palavras-chave AND e XOR, sendo que um AND com condições no elemento de divisão pode ser considerado um OR. Já a linguagem proposta possibilita a representação das primitivas *AND-Join*, *AND-Split*, *OR-Join*, *OR-Split*, *XOR-Join* e *XOR-Split* (Figura 2), pois também foi considerado importante representar o OR sem condições associadas. Outro aspecto a ser destacado é a representação explícita do OR e do XOR, principalmente nas primitivas de junção³, de forma a possibilitar combinações de primitivas (por exemplo: [Atividade1 ou Atividade 2] e Atividade 3).

³ A primitiva *OR-Split* também pode ser representada explicitamente. A representação implícita visa apenas economizar espaço de desenho nos diagramas, o que pode ser crítico em modelos mais complexos.

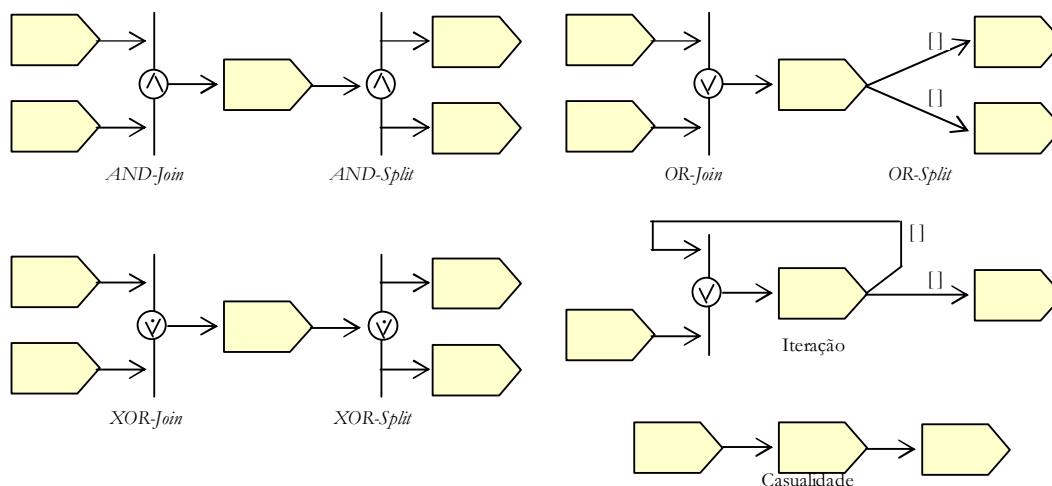


Figura 2 - Primitivas utilizando a Notação Proposta

A linguagem para modelagem de processos proposta nesta subseção tem sido avaliada por estudantes e pesquisadores da linha de Engenharia de Software da COPPE/UFRJ, que a têm utilizado para modelar os processos apoiados pelas ferramentas desenvolvidas. Além de estar sendo avaliada, a evolução da linguagem é inerente à estratégia adotada para a sua definição, segundo a qual o conjunto mínimo, inicialmente proposto, de elementos e possibilidades de conexão desses elementos deve evoluir a partir da necessidade prática dos modeladores. Portanto, um requisito importante para uma ferramenta de apoio à modelagem de processo baseada na linguagem é o de que ela seja flexível, oferecendo mecanismos para a evolução da linguagem.

Referências Bibliográficas

- ALLWEYER, T., 1999, *A Framework for Re-designing and Managing Knowledge Processes*, ARIS White Paper, IDS Scheer AG, Saarbrücken, Germany. In: <http://www.changeware.net/doc/km.pdf>, acessado em 02/11/2007.
- CHANG, Y., CHEN, S., CHEN, C., *et al.*, 2000, "Workflow Process Definition and Their Applications in e-Commerce". In: *2000 International Symposium on Multimedia Software Engineering*, Taipei, Taiwan, Nov. 2000.
- HOLLINGSWORTH, D., 1995, *The Workflow Reference Model*, TC00-1003, version 1.1, Workflow Management Coalition.
- KRUCHTEN, P., 2000, *The Rational Unified Process: An Introduction*, Addison-Wesley.

- OMG, 2002, *Software Process Engineering – Metamodel Specification*, formal/02-11-14, version 1.0, Object Management Group.
- RATIONAL, 2001, *Business Modeling with UML and Rational Suite AnalystStudio*, A Rational White Paper.
- REIS, C., REIS, R., SCHLEBBE, H., *et al.*, 2002, "A Policy-based Resource Instantiation Mechanism to Automate Software Process Management". In: *Proceedings of 14th International Conference on Software Engineering & Knowledge Engineering*, Ischia, Italy, Jul. 2002.
- SOFTEX, 2007, *MPS.BR - Guia de Avaliação*, v1.1, SOFTEX - Associação para Promoção da Excelência do Software Brasileiro. In: <http://www.softex.br/mps.br>, acessado em 02/11/2007.
- VILLELA, K., 2004, *Definição e Construção de Ambientes de Desenvolvimento de Software Orientados à Organização*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- WFMC, W.G., 1999, *Interface 1: Process Definition Interchange - Process Model*, TC00-1016-P, version 1.1, Workflow Management Coalition.

ANEXO II - PERFIL UML PARA MODELAGEM DE ONTOLOGIAS

Este anexo descreve o perfil UML utilizado para modelagem de ontologias no contexto desta tese. Este perfil utiliza os princípios da linguagem de modelagem de ontologias denominada LINGO.

1 Perfil UML para Modelagem de Ontologias

Dá-se o nome de perfil UML a um subconjunto pré-definido de elementos padrão da UML, associados a estereótipos, valores etiquetados e restrições, que conjuntamente especializam e configuram a UML para um determinado domínio de aplicação ou propósito particular. Com o crescimento da UML como linguagem de modelagem para ontologias, um perfil UML foi proposto, guardando os princípios definidos na linguagem de modelagem de ontologias LINGO (MIAN, 2003).

O perfil UML mostrado na Figura 1, define elementos para representar conceitos, relações e propriedades. Este perfil também permite derivar alguns axiomas a partir de determinadas relações, tornando desnecessário que sejam tornados explícitos e, assim, diminuindo substancialmente o esforço de escrita de axiomas formais.

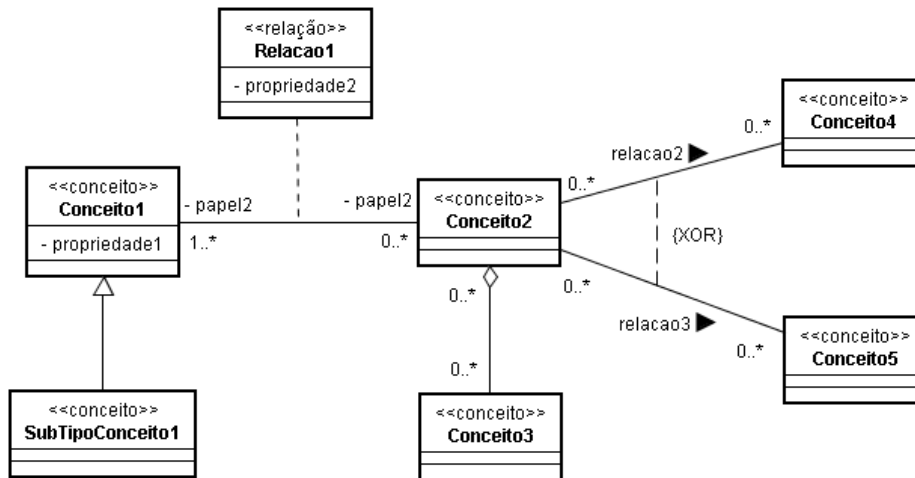


Figura 1 – Perfil UML (MIAN, 2003)

Para dar uma semântica diferente aos elementos de modelo da UML, foi utilizado o mecanismo de extensão estereótipo. Sendo assim, classes com estereótipo <<conceito>>

representam conceitos da ontologia, da mesma forma que relações que contêm propriedades ou que possuem aridade maior que dois, são representadas como classes associativas com estereótipo `<<relação>>`.

As relações binárias sem propriedades são definidas como associações nomeadas. As propriedades de conceitos e relações são representadas como atributos de classes estereotipadas. Relações de subtipo e todo-parte são representadas como relações de generalização/especialização e de agregação/composição, respectivamente. Condicionantes entre relações são representados por outro mecanismo de extensão da UML, as restrições entre associações (MIAN, 2003).

Os axiomas derivados da Figura 1 são:

Todo-parte:

- (AE1) $(\forall x) \neg \text{parteDe}(x, x)$
- (AE2) $(\forall x, y) \text{parteDe}(y, x) \cdot \text{todoDe}(x, y)$
- (AE3) $(\forall x, y) \text{parteDe}(y, x) \rightarrow \neg \text{parteDe}(x, y)$
- (AE4) $(\forall x, y) \text{parteDe}(y, x) \rightarrow (\exists z) (\text{parteDe}(z, x))$
- (AE5) $(\forall x, y, z) \text{parteDe}(z, y) \wedge \text{parteDe}(y, x) \rightarrow \text{parteDe}(z, x)$

Sub-tipo-de:

- (AE6) $(\forall x, y, z) \text{subTipoDe}(x, y) \wedge \text{subTipoDe}(y, z) \rightarrow \text{subTipoDe}(x, z)$
- (AE7) $(\forall x, y) \text{subTipoDe}(x, y) \rightarrow \text{superTipoDe}(y, x)$

Ou-exclusivo (XOR):

- (AE8) $(\forall a \in C2) ((\exists b) (b \in C3) \wedge R2(a, b)) \rightarrow \neg ((\exists c \in C4) \wedge R3(a, c))$
- (AE9) $(\forall a \in C2) ((\exists c) (c \in C4) \wedge R3(a, c)) \rightarrow \neg ((\exists b \in C3) \wedge R2(a, b))$

Axiomas relacionados à multiplicidade das relações, axiomas relacionados à classificação de indivíduos como instâncias de um conceito e axiomas relacionados a propriedades são igualmente tratados implicitamente. Assim, sejam x e y instâncias, respectivamente, de *Conceito1* e *Conceito2*. Assume-se, implicitamente, que as seguintes sentenças estão definidas:

- `conceito1(x)`
- `conceito2(y)`
- `propriedade1(x, p1)`, onde $p1$ é o valor da propriedade1 para x .
- $(\forall y \in \text{Conceito2}) ((\exists x) (x \in \text{Conceito1}) \wedge \text{relaçãol}(x, y))$

Referências Bibliográficas

MIAN, P.G., 2003, *ODEd: Uma Ferramenta de Apoio ao Desenvolvimento de Ontologias em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.

ANEXO III - ESTUDO BASEADO EM REVISÃO SISTEMÁTICA – INFRA-ESTRUTURA DE APOIO A INICIATIVAS DE MELHORIA DE PROCESSOS DE SOFTWARE

Este anexo apresenta os dados referentes ao planejamento e à execução de estudo baseado em revisão sistemática visando a identificar os elementos que compõem as diferentes abordagens e instrumentos de apoio utilizados em iniciativas de melhoria de processos de software em organizações e corporações e identificar as abordagens existentes de apoio à execução e melhoria de processos de software no escopo de projetos, organizacionais ou corporativos, através de ferramentas CASE isoladas e/ou ambientes específicos. Parte do resultado obtido com este estudo pode ser visto no Capítulo II e, em especial, no Capítulo III desta tese, que tratam, respectivamente, de Melhoria de Processos de Software e Infra-estrutura para Iniciativas de Melhoria de Processos de Software.

1 Introdução

Vários fatores influenciam a definição, gerência, implantação, execução e melhoria de seus processos de software e no controle dos diferentes estágios de maturidade e características possuídos por cada organização e por corporações. Esta tese teve como objetivo propor, através dos Ambientes de Engenharia de Software Orientados a Corporações (AESCorp), um apoio computacional para este fim. Devido a isto foi de interesse a identificação de elementos que compõem as diferentes abordagens e instrumentos de apoio empregados em iniciativas de melhoria de processos de software reais em organizações e corporações. Foi realizado, então, um estudo baseado em revisão sistemática procurando reduzir o viés de uma revisão informal e, também, permitir que tal pesquisa bibliográfica possa ser atualizada com novas publicações disponibilizadas ao longo do tempo.

Segundo MAFRA e TRAVASSOS (2006), a menos que a revisão da literatura seja conduzida de forma confiável e abrangente, seus resultados possuirão pouco valor científico. Muitas revisões da literatura são conduzidas informalmente, sem um planejamento e critérios de seleção estabelecidos *a priori* caracterizam-se por serem pouco

abrangentes, não passíveis de repetição, pouco confiáveis e dependentes dos revisores. Dessa forma, o desenvolvimento de uma abordagem sistemática de revisão visa a estabelecer um processo formal para conduzir este tipo de investigação, evitando a introdução de eventuais vieses da revisão de literatura informal. A revisão sistemática é um tipo de estudo secundário (KITCHENHAM, 2004), cujo processo de pesquisa segue um conjunto de passos metodologicamente bem definidos de acordo com um protocolo prévio (BIOLCHINI *et al.*, 2005) e cuja adoção procura reduzir o viés inerente a uma revisão informal (SILVA FILHO, 2006).

Parte do trabalho envolvido em uma revisão sistemática é a calibragem das palavras-chave para a busca. DIESTE e PADUA (2007) comentam sobre a estratégia de busca por artigos para fazer uma revisão sistemática. Analisam o uso de termos com o objetivo de encontrar estudos experimentais na literatura e analisa o custo-benefício do uso de diferentes combinações de palavras-chave. Segundo os autores é aceitável uma taxa de 72-80% de sensibilidade (total de artigos identificados dentro do universo de busca) e 15-25% de precisão (total de artigos realmente relevantes dentro dos artigos encontrados pela busca). Algumas possíveis limitações dos mecanismos de busca que devem ser levadas em consideração: recursos bibliográficos limitados, problemas com o algoritmo de busca, falha em reconhecimento de plural e textos completos ou *abstracts* incompletos. Os autores aconselham a busca no título e no *abstract* em vez de no texto completo para melhores resultados. Além disso, comentam que dependendo do objetivo dos revisores o objetivo da estratégia de busca selecionada pode ser: maximizar a quantidade de material encontrado, maximizar a quantidade de material relevante encontrado e otimizar a quantidade de material relevante encontrado.

DIESTE e PADUA (2007) também fazem algumas considerações¹ sobre os termos utilizados na busca utilizada para ilustrar o artigo: o uso combinado da palavra-chave principal com sinônimos mais proximamente relacionados melhora as propriedades da busca (sensibilidade e precisão); o uso apenas da palavra-chave principal para procurar nos títulos e *abstracts* não é de todo uma má estratégia; para reduzir o número de artigos relevantes não encontrados através da palavra-chave principal devem-se utilizar sinônimos próximos e geralmente aceitos utilizados com frequência; se for adicionado termos mais gerais à busca do que aos sinônimos mais frequentes e aceitos pode-se detectar mais artigos relevantes, mas também pode aumentar o número de artigos irrelevantes; não se deve

¹ Apesar destes comentários se aplicarem aos termos utilizados no artigo não é difícil imaginar que as recomendações também podem ser estendidas para outros termos em situações similares.

utilizar somente os sinônimos da palavra-chave principal pois eles podem omitir um conjunto expressivo de resultados importantes. Para finalizar, duas recomendações: se a revisão sistemática deve ser exaustiva e encontrar todos os experimentos em um determinado domínio, não se deve limitar a busca apenas em publicações de renome; dependendo do tópico ou tecnologia de interesse, publicações de outras áreas devem ser exploradas.

2 Protocolo do Estudo Baseado em Revisão Sistemática

O protocolo utilizado para o estudo foi derivado dos trabalhos produzidos por MAFRA (2006) e SILVA FILHO (2006).

Para cada uma das subseções a seguir serão apresentados o que se espera a partir do protocolo (texto em *{italico}*) e o conteúdo de fato utilizado no estudo em questão.

2.1 Contexto

{Descrever um breve relato sobre o problema que motivou a realização do estudo; delimitar o problema; identificar o que é importante e o que está fora do escopo; justificar a necessidade de conduzir o estudo para tratar o problema apresentado (Por que este tipo de estudo é indicado?).}

Vários fatores influenciam a definição, gerência, implantação, execução e melhoria de seus processos de software e no controle dos diferentes estágios de maturidade e características possuídos por cada organização e por corporações. Para a definição dos Ambientes de Engenharia de Software Orientados a Corporações (AESCorp) é de interesse a identificação de elementos que compõem as diferentes abordagens e instrumentos de apoio empregados em iniciativas de melhoria de processos de software reais em organizações e corporações. A partir dessa identificação, é possível realizar uma análise das abordagens investigadas com o objetivo de caracterizá-las e, se possível, influenciar a definição dos requisitos e infra-estrutura dos AESCorp.

2.2 Objetivo e Questões de Pesquisa

{Descrever o objetivo e as questões de pesquisa do estudo.}

2.2.1 Objetivo

{Descrever o objetivo do estudo através do seguinte padrão: Analisar [Fontes de evidências/indícios: relatos de experiência/publicações científicas/estudos primários/software etc.]; Com o propósito de [caracterizar/identificar] o [objeto de estudo: abordagens/processo/tecnologia/técnica/método etc.]; Com relação à [características de interesse do objeto de estudo]; Do ponto de vista [organizações de

software/grupo de processo/alta gerência/Instituições Implementadoras de Processos/pesquisador/desenvolvedor etc.]; No contexto [industrial/acadêmico/ambos] [com foco em estudos/projetos/programas/iniciativas] [com tais características].}

Analisar relatos de experiência e publicações científicas através de um estudo baseado em revisão sistemática.

Com o propósito de identificar os elementos que compõem as abordagens de apoio empregadas em iniciativas de melhoria de processos de software em organizações e corporações.

Com relação aos instrumentos de apoio empregados para a definição, implantação, execução e melhoria dos processos de software em organizações e corporações.

Do ponto de vista dos pesquisadores.

No contexto acadêmico e industrial com foco em programas de melhoria de processos de software em organizações e corporações.

2.2.2 Questões de pesquisa

{Identificar que questões serão respondidas a partir da identificação e caracterização do objeto de estudo. Ou seja, uma vez identificados/caracterizados os objetos de estudo, que questões relevantes ao problema descrito poderão ser respondidas/discutidas?}

Questão principal

Quais são as principais características e apoio computacional das abordagens para execução de iniciativas de melhoria de processos de software em organizações e corporações?

Questões secundárias

Uma vez que as abordagens e o apoio computacional sejam identificados, propõe-se uma caracterização através das seguintes questões:

- Qual a caracterização da agregação do apoio ferramental apresentado? (se ferramentas isoladas, ferramentas integradas ou ambiente)
- Qual a caracterização do tamanho da organização? (se pequena, média, grande ou se corporação)

- A iniciativa de melhoria de processos de software foi descrita em uma organização individual ou num grupo de organizações?
- Qual o uso do apoio ferramental apresentado? (se foi usado ou se é apenas um protótipo);
- Qual a abrangência do apoio ferramental apresentado? (que processos de software ou elementos da iniciativa de melhoria são apoiados)
- Quais os modelos de maturidade, normas, técnicas ou métodos mencionados?

Intervenção

Instrumentos utilizados no apoio a projetos de melhoria de processos de software.

Comparação

Não se aplica.

População

Trabalhos publicados em conferências e periódicos relatando projetos melhoria de processos de software em corporações ou organizações.

Resultados

A partir da identificação dos elementos que compõem as diferentes abordagens e instrumentos de apoio empregados em iniciativas de melhoria de processos de software pretende-se realizar uma análise mais detalhada das abordagens investigadas com o objetivo de caracterizá-las e, se possível, influenciar a definição dos requisitos e infra-estrutura dos AESCorp.

2.3 Escopo da pesquisa

{Delimitar os tipos de mecanismos que serão utilizados para realizar as buscas, por exemplo, bibliotecas digitais através dos seus respectivos engenhos de busca, bibliotecas, consultas a especialistas, catálogo especializado de produtos etc. }

Para delinear o escopo da pesquisa foram estabelecidos critérios para garantir, de forma equilibrada, a viabilidade da execução (custo, esforço e tempo), acessibilidade aos dados e abrangência do estudo. A pesquisa dar-se-á a partir de bibliotecas digitais através

dos seus respectivos engenhos de busca e, quando os dados não estiverem disponíveis eletronicamente, através de consultas manuais.

2.3.1 Critérios adotados para seleção das fontes

{Identificar os critérios que devem ser adotados para a seleção das fontes de pesquisa.}

Para as bibliotecas digitais é desejado:

- Possuir engenho de busca que permita o uso de expressões lógicas ou mecanismo equivalente;
- Incluir em sua base publicações da área de exatas ou correlatas que possuam relação direta com o tema a ser pesquisado;
- Os engenhos de busca deverão permitir a busca no texto completo das publicações.

Além disso, deve-se garantir que as publicações pertençam a uma das editoras listadas no Portal de Periódicos da CAPES.

Os mecanismos de busca utilizados devem garantir resultados únicos através da busca de um mesmo conjunto de palavras-chaves. Quando isto não for possível, deve-se estudar e documentar uma forma de minimizar os potenciais efeitos colaterais desta limitação.

Serão considerados, também, simpósios patrocinados pela Sociedade Brasileira de Computação (SBC) na área de Engenharia de Software.

2.3.2 Restrições

{Identificar todas as restrições associadas ao estudo. Identificar o intervalo de tempo válido para as buscas. O acesso aos dados, em geral, não deve incorrer em ônus para a pesquisa.}

A pesquisa está restrita à análise de publicações obtidas, exclusivamente, a partir das fontes selecionadas a partir dos critérios supracitados.

O estudo englobará os dados disponíveis nas fontes considerando o período de 01 de janeiro de 1998 até 31 de dezembro de 2007.

2.4 Idiomas

{Deve-se identificar os idiomas das publicações que serão aceitas para a pesquisa. Se possível, deve-se justificar essa escolha.}

Para a realização desta pesquisa foram selecionados os idiomas inglês e português. A escolha do idioma inglês deve-se à sua adoção pela grande maioria das conferências e periódicos internacionais relacionados como tema de pesquisa e por ser o idioma utilizado

pela maioria das editoras relacionadas com o tema listadas no Portal de Periódicos da CAPES. A escolha do idioma português deve-se à sua adoção pelas principais conferências e periódicos nacionais da área de Engenharia de Software.

2.5 Métodos de Busca de Publicações

{Deve-se descrever a forma de busca (manual e/ou eletrônica) além da expressão de busca: expressão lógica contendo uma combinação de palavras chaves extraída do objetivo do estudo relacionada ao objeto de estudo, características de interesse e respectivos sinônimos.}

As fontes digitais serão acessadas via Web, através de expressões de busca pré-estabelecidas. Caso não seja possível obter o artigo completo através dos sites de busca, os autores dos artigos deverão ser contatados via e-mail.

As publicações das fontes não-digitais serão analisadas manualmente, quando disponíveis, considerando a expressão de busca definida.

2.5.1 Expressão de Busca

Para artigos em inglês deve-se utilizar a expressão de busca abaixo²:

(enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations) AND (infrastructure OR environment OR tool OR tools OR support OR platform OR framework OR “project management system”) AND (“software processes” OR “software process” OR “process evolution”) AND (improvement OR enactment OR execution OR control)

Para artigos em Português deve-se utilizar a expressão de busca abaixo:

(infraestrutura OR infra-estrutura OR ambiente OR ferramenta OR apoio OR suporte OR plataforma OR framework OR arcabouço OR “sistema de gerência de projetos”) AND (empresa OR organização OR corporação OR companhia OR associação OR corporativo OR organizações OR companhias OR empresas OR corporações OR negócio OR indústria OR indústrias) AND (melhoria OR execução OR controle) AND (“processo de software” OR “processos de software” OR “evolução de processo”)

² Mais detalhes sobre a definição da expressão de busca podem ser obtidos na seção 3.2.

2.5.2 Busca Manual

Quando a consulta for manual, devem-se procurar as palavras-chave presentes na expressão de busca nos títulos e resumos (*abstracts*) dos artigos. Para artigos em Português, mas com *abstracts* em inglês, deve-se primeiro pesquisar no *abstract*, em caso de dúvida sobre a seleção do artigo, deve-se pesquisar as palavras-chave no resumo.

2.6 Procedimentos de Seleção e Critérios

{Deve-se descrever os procedimentos para seleção das publicações, incluindo procedimentos avaliação da inclusão de publicações no escopo da pesquisa e critérios de inclusão e exclusão.}

A estratégia de busca será aplicada por um pesquisador para identificar as publicações em potencial. As publicações identificadas serão selecionadas pelos demais pesquisadores (incluindo o que fará a busca) através da verificação dos critérios de inclusão e exclusão e de qualidade estabelecidos. Os pesquisadores deverão entrar em consenso sobre a seleção das publicações cujas avaliações se mostrem conflitantes.

Em caso de impasse entre os pesquisadores, a publicação deverá ser incluída na lista de selecionadas. Para diminuir o risco que uma publicação seja excluída prematuramente em uma das etapas do estudo, sempre que existir dúvida a publicação não deverá ser excluída.

Serão aceitas publicações que descrevam pelo menos provas de conceito e/ou relatos de experiência na academia ou na indústria.

2.6.1 Procedimentos de Seleção

{Identificar as etapas necessárias para seleção das publicações para o estudo.}

A seleção dos estudos dar-se-á em 4 etapas:

i) Seleção e catalogação preliminar dos dados coletados. A seleção preliminar das publicações será feita a partir da aplicação da expressão de busca às fontes selecionadas. Cada publicação será catalogada em um banco de dados criado especificamente para este fim e armazenada em um repositório para análise posterior;

ii) Seleção dos dados relevantes - [1º filtro]. A seleção preliminar com o uso da expressão de busca não garante que todo o material coletado seja útil no contexto da pesquisa, pois a aplicação das expressões de busca é restrita ao aspecto sintático.

Dessa forma, após a identificação das publicações através dos mecanismos de buscas, deve-se ler os resumos/*abstracts* e analisá-los seguindo os critérios de inclusão e

exclusão identificados a seguir. Neste momento, poder-se-ia classificar as publicações apenas quanto aos critérios de exclusão, entretanto, para facilitar a análise e reduzir o número de publicações das quais possa-se ter dúvidas sobre sua aceitação, deve-se também classificá-las quanto aos critérios de inclusão.

Na impossibilidade de classificar a publicação quanto a um dos critérios abaixo, os pesquisadores deverão entrar em consenso sobre a classificação da publicação quanto aos critérios definidos ou, então, definir um novo critério de inclusão ou exclusão.

Devem ser excluídas as publicações contidas no conjunto preliminar que:

- **CE1-01** - Não serão selecionadas publicações em que as palavras chave não estão presentes na publicação e não há variações destas palavras chave (exceto plural).
- **CE1-02** - Não serão selecionadas publicações em que as palavras chave da busca não apareçam no título, resumo e/ou texto da publicação (excluem-se daí o campo 'palavras chave', as seções agradecimentos, biografia dos autores, referências bibliográficas e anexos).
- **CE1-03** - Não serão selecionadas publicações em que descrevam e/ou apresentam '*keynote speeches*', tutoriais, cursos, workshops e similares.
- **CE1-04** - Não serão selecionadas publicações em que as palavras chave '*software process improvement*' qualifique normas ou modelos de maturidade (como por exemplo o SPICE, CMM e CMMI) sem descrever de fato uma abordagem para elas.
- **CE1-05** - Não serão selecionadas publicações em que o contexto em que as palavras chave são utilizadas no artigo levem a crer que a publicação não descreve um programa de melhoria de processos de software de fato.
- **CE1-06** - Não serão selecionadas publicações em que o contexto em que as palavras chave são utilizadas no artigo levem a crer que a publicação não descreva a utilização de ambientes e/ou ferramentas CASE em iniciativas de melhoria de processos.
- **CE1-07** - Não serão selecionadas publicações em que as ferramentas citadas não estejam relacionadas a um programa de melhoria de processos de software e/ou não apoiem aspectos da abordagem descrita.
- **CE1-08** - Não devem ser selecionadas publicações que falem sobre ambientes no sentido de meio físico/cultural/organizacional.

- **CE1-09** - Não devem ser selecionadas publicações que apenas citam a existência de ferramentas CASE e/ou ambientes.
- **CE1-10** - Não devem ser selecionadas publicações que comentem sobre a avaliação/capacitação de pessoal em vez de mencionar características de programas de melhoria de processos de software.
- **CE1-11** - Não serão selecionadas publicações em que a sigla SPI não signifique "*software process improvement*".
- **CE1-12** - Não serão selecionadas publicações que apresentam apoio ferramental mas não apresentam subsídios que as relacione a uma iniciativa real ou hipotética de melhoria de processos de software.
- **CE1-13** - Não serão selecionadas publicações que apresentam uma iniciativa de melhoria de processos de software mas não falam de apoio ferramental.
- **CE1-14** - Não serão selecionadas publicações que apresentam uma compilação de dados sobre iniciativas de melhoria de processos de software sem contextualizar a utilização de apoio ferramental.
- **CE1-15** - Não serão selecionadas publicações que apresentam uma compilação de dados sobre apoio ferramental sem contextualizar a utilização em uma iniciativa de melhoria de processos de software.
- **CE1-16** - Não serão selecionadas publicações que apresentam uma abordagem específica para um único tipo de processo de engenharia de software (por exemplo, gerência de requisitos ou aquisição).
- **CE1-17** - Não serão selecionadas publicações que não apresentam uma abordagem de melhoria de processos de software e só discutem a aderência ou aplicabilidade de um modelo ou norma de maturidade.
- **CE1-18** - Não serão selecionadas publicações que não falem de melhoria de processos de software nem de apoio ferramental.
- **CE1-19** - Não serão selecionadas publicações que descrevem abordagem ou iniciativa de melhoria de processos de software mas não apresentem subsídios que permitam identificar se foi aplicada.
- **CE1-20** - Não serão selecionadas publicações que descrevam problemas relacionados a melhoria de processos de software sem foco numa abordagem e/ou ferramenta.

Podem ser incluídas apenas as publicações contidas no conjunto preliminar que:

- **CI1-01** - Podem ser selecionadas publicações que mencionam ambientes no sentido computacional do termo.
- **CI1-02** - Podem ser selecionadas publicações que descrevam características desejáveis para um ambiente ou o apoio ferramental para processos de software desde que estejam relacionados a melhoria de processos de software.
- **CI1-03** - Podem ser selecionadas publicações que descrevam um protótipo de ambiente desde que seja acompanhado de uma justificativa das características deste ambiente em relação a um programa de melhoria de processos de software.
- **CI1-04** - Podem ser selecionadas publicações que apresentam abordagem para melhoria de processos de software ainda não implementada desde que sua aplicação seja descrita para um contexto específico.
- **CI1-05** - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco na evolução dos processos.
- **CI1-06** - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco em pequenas e médias empresas.
- **CI1-07** - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software em uma organização ou aplicável a uma organização.
- **CI1-08** - Podem ser selecionadas publicações que apresentam apoio ferramental para execução de processos em uma organização.
- **CI1-09** - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco em uma grande organização ou corporação.
- **CI1-10** - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software baseado em uma norma ou modelo de maturidade.
- **CI1-11** - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

- **CI1-12** - Podem ser selecionadas publicações que discutam aspectos relacionados a abordagens de melhoria de processos de software, incluindo apoio ferramental.

iii) Seleção dos dados relevantes - [2º filtro]. Apesar de limitar o universo de busca, o 1º filtro empregado não garante que todo o material coletado seja útil no contexto da pesquisa. Por isso, após a leitura dos artigos selecionados no 1º filtro, deve-se verificar que as publicações respeitem os critérios abaixo:

- **CS2 -SPI -Fer -Uso** - Não devem ser selecionadas publicações que não descrevam uma iniciativa de melhoria de processos de software ou o apoio ferramental e não dêem indícios que a iniciativa de melhoria de processos é real ou que o apoio ferramental foi utilizado na prática.
- **CS2 -SPI -Fer +Uso** - Não devem ser selecionadas publicações que descrevam um caso prático mas não apresentem apoio ferramental e contextualize uma iniciativa de melhoria de processos de software.
- **CS2 -SPI +Fer -Uso** - Não devem ser selecionadas publicações que apresentem apoio ferramental mas não o contextualize numa iniciativa de melhoria de processos de software na prática.
- **CS2 -SPI +Fer +Uso** - Não devem ser selecionadas publicações que apresentem apoio ferramental e comente sobre seu uso na prática mas não o contextualize numa iniciativa de melhoria de processos de software.
- **CS2 +SPI -Fer -Uso** - Não devem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software mas não descrevam apoio ferramental nem dêem indícios que a iniciativa de melhoria de processos é real ou que o apoio ferramental foi utilizado.
- **CS2 +SPI -Fer +Uso** - Não devem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software mesmo que real mas não dêem indícios que o apoio ferramental foi utilizado.
- **CS2 +SPI +Fer -Uso** - Não devem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software e o apoio ferramental mas não dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Dessa forma, todas as publicações devem respeitar o critério abaixo:

- **CI2 +SPI +Fer +Uso** - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

2.6.2 Critérios de Inclusão

{Identificar critérios adicionais para inclusão de publicações no escopo do estudo.}

Devem ser consideradas ainda as publicações que:

- Sejam citadas nas referências bibliográficas e forem considerados relevantes apesar de não terem sido identificados pelas palavras chave do estudo.

2.7 Procedimentos para Extração dos Dados

{Identificar os procedimentos para extração de dados a partir das publicações.}

2.7.1 Na seleção e catalogação preliminar dos dados coletados

Armazenamento das referências completas selecionadas a partir da fonte consultada no repositório de dados do estudo.

2.7.2 Na seleção dos dados relevantes

Cada referência catalogada deve ser examinada com o objetivo de ser submetida aos critérios de seleção dos filtros identificados. Os dados que atenderem aos critérios de seleção deverão ser marcados como “verificado no [número do filtro]º filtro, passou”, do contrário, o registro deverá ser marcado como “verificado no [número do filtro]º filtro, não passou no critério [número do critério]”.

2.7.3 Extração de Dados

Os dados extraídos das publicações selecionadas deverão ser armazenados em um banco de dados e devem conter:

- Dados da publicação:
 - Título,
 - Autor(es),
 - Data de publicação,
 - Veículo de publicação;

- Resumo da publicação;
- Dados derivados das características de interesse declaradas no objetivo do estudo:
 - **Geral** - Representa uma categoria de itens gerais sobre o artigo. O preenchimento dos itens dessa seção é obrigatório quando a publicação for considerada válida para o estudo baseado em revisão sistemática.
 - *Objetivo* - Descrição do objetivo do artigo. O preenchimento deste item é sempre obrigatório independente da publicação ser ou não válida para o estudo baseado em revisão sistemática.
 - *Escopo revisão da literatura* - Identificação de elementos que sejam interessantes considerar em uma revisão da literatura.
 - **Organizações apoiadas** - Descrição das características das organizações onde a iniciativa de melhoria de processos foi definida ou executada e/ou as ferramentas foram utilizadas. O preenchimento dos itens dessa seção é obrigatório quando o artigo for considerado válido para o estudo baseado em revisão sistemática.
 - *Descrição* - Descrição das organizações citadas na publicação.
 - *Localização* - Descrição da localização (tipicamente o país) das organizações citadas no artigo.
 - **Iniciativa de melhoria de processos** - Descrição de características da iniciativa de melhoria de processos descrita no artigo, pode ser uma estratégia definida e estabelecida ou apenas a abordagem utilizada no contexto do estudo de caso apresentado. O preenchimento dos itens dessa seção é obrigatório quando o artigo for considerado válido para o estudo baseado em revisão sistemática.
 - *Objetivos* - Descrição dos objetivos da iniciativa de melhoria de processos ou da abordagem utilizada.
 - *Resultados obtidos* - Descrição da execução ou os resultados obtidos da adoção ou execução da iniciativa de melhoria de processos nas organizações descritas no artigo.
 - **Apoio ferramental** - Descrição das características do apoio ferramental apresentado. O preenchimento dos itens dessa seção é obrigatório quando o artigo for considerado válido para o estudo baseado em revisão sistemática.

- *Descrição* - Descrição do apoio ferramental apresentado em detalhes suficientes para compreender seu uso na iniciativa de melhoria de processos descrita.
 - *Uso em contexto real* - Descrição de como (ou onde) o apoio ferramental apresentado foi ou pode ser utilizado em um contexto real.
- Dados derivados das características de interesse declaradas nas questões secundárias de pesquisa:
 - Caracterização da agregação do apoio ferramental apresentado (se ferramenta(s) isolada(s), ferramentas integradas ou ambiente);
 - Caracterização do tamanho da organização (se pequena, média, grande ou se corporação);
 - Caracterização se a iniciativa de melhoria de processos de software foi descrita em uma organização individual ou num grupo de organizações;
 - Descrição do uso do apoio ferramental apresentado (se foi usado ou se é apenas um protótipo);
 - Identificação da abrangência do apoio ferramental apresentado;
 - Identificação das modelos de maturidade, normas, técnicas ou métodos mencionados;
- Comentários adicionais do pesquisador.

2.7.4 Sumarização dos resultados

Os resultados serão tabulados. Nenhuma meta-análise será realizada.

2.8 Procedimentos para Análise

{Identificar os procedimentos para análise dos dados coletados. Incluir totalização das mais diversas e relevantes para o objetivo do estudo e questões de pesquisa.}

2.8.1 Análise Quantitativa

A análise quantitativa dar-se-á pela extração direta dos dados a partir do banco de dados com os registros dos achados.

A análise quantitativa consiste em fornecer:

- Número de publicações selecionadas para fazerem parte do estudo;
- Número de abordagens de iniciativa de melhoria de processos descritas nas publicações selecionadas para fazerem parte do estudo.

2.8.2 Análise Qualitativa

A análise qualitativa deverá utilizar como base, os dados quantitativos e realizar considerações com o intuito de discutir os achados com relação às questões de pesquisa declaradas.

3 Planejamento e Execução

O protocolo descrito na seção anterior é a base para a execução do estudo baseado em revisão sistemática, entretanto, o seu planejamento começou antes de sua elaboração. Para a construção do protocolo e condução do estudo foi utilizado o processo definido por MONTONI (2007), que consiste das seguintes atividades: Realizar Prospecção sobre o Tema de Interesse, Definir Protocolo, Testar Protocolo e Avaliar o Protocolo; Executar a Pesquisa e Analisar Resultados da Pesquisa; Empacotar Resultados e Publicar Resultados.

3.1 Definição do Escopo e Estudos Preliminares

A primeira etapa do planejamento foi a prospecção sobre o tema de interesse para o estudo. Num primeiro momento pensou-se em realizar dois estudos diferentes: (i) identificar instrumentos de apoio empregados para a definição, implantação, execução e melhoria dos processos de software das corporações e suas organizações; e (ii) caracterizar diferentes abordagens de apoio à execução de processos de software que utilizassem ferramentas e/ou ambientes específicos.

Neste momento, foram testadas, também, algumas opções de palavras-chave para cada um dos estudos. Para o primeiro as palavras-chave giravam em torno de “*software process improvement*” e sinônimos para *organization* e *corporation* (como, por exemplo, *large-organization*, *enterprise*). Para o segundo, em torno de variantes “*software process improvement*” (por exemplo, “*software proces*”, “*process execution*”) e sinônimos para *environment* e “*case tool*” (por exemplo, *support* e *tool*). Os resultados obtidos com as buscas preliminares³ levaram a crer que, apesar da abrangência das palavras-chave, poucos eram as publicações que trariam alguma boa contribuição para o tema de cada estudo. Isso poderia indicar (i) problemas com as palavras-chave e/ou (ii) que os temas de cada estudo deveriam

³ As buscas preliminares utilizaram a Biblioteca Digital da ACM (<http://portal.acm.org/dl.cfm>) que é conhecida por ter problemas no seu mecanismo de buscas principalmente na acurácia dos resultados retornados e no fato de diferentes execuções de uma mesma consulta nem sempre trazer os mesmos resultados. Entretanto, isso não foi determinante para os resultados obtidos, pois neste momento o que mais interessava era a prospecção de diferentes artigos para compor o grupo de controle dos estudos.

ser revistos. Além disso, percebeu-se com estes testes que o esforço para a execução de um estudo baseado em revisão sistemática é muito grande e que, devido a isso, seria melhor concentrar esforços em um único estudo que fosse melhor aproveitado.

Assim, decidiu-se identificar um tema que conseguisse unir os dois estudos iniciais, trouxesse melhores resultados e, principalmente, trouxesse melhores resultados úteis para o contexto desta tese. Neste momento, chegou-se o tema do estudo descrito neste anexo: a identificação de apoio ferramental utilizado por organizações no apoio às atividades relacionadas a melhoria de processos de software.

3.2 Identificação de Publicações de Controle e Palavras-Chave

Após ter-se definido o tema do estudo, foi feita uma nova rodada de testes da expressão de busca. Desta vez foi utilizada a máquina de busca da Compendex (<http://www.engineeringvillage.com/>).

Para a escolha das palavras-chave, é bom se ater, num primeiro, a publicações de controle. Uma lista inicial destas publicações foi identificada durante os testes dos dois estudos iniciais. Outras foram sendo adicionadas durante os testes com as expressões de busca a medida que novos artigos eram retornados.

Esta lista de publicações tem de aparecer no estudo através da calibragem da expressão de busca. Se muitas publicações estiverem sendo retornadas é necessário (i) verificar se o que está sendo procurado é essencial ou se está misturando outros requisitos do pesquisar que não aqueles inerentes ao estudo de fato (por exemplo, necessidades da revisão da literatura da tese ou dissertação); (ii) verificar se não há palavras-chave demais, desnecessárias; (iii) verificar se a máquina de busca é eficiente. Não se indica também a inclusão de muitas cláusulas restritivas (por exemplo, *AND NOT (critério-qualquer)*) pois podem limitar o escopo da busca desnecessariamente (a lista de publicações pode ser revista através da aplicação dos filtros presentes no protocolo).

Dessa forma, o processo para calibragem da expressão de busca envolveu:

1. Definição da máquina de busca para testes do protocolo;
2. Identificação de publicações que deveriam compor o grupo de controle;
3. Identificação de expressão de busca inicial;
4. Testes da expressão de busca;
5. Análise dos resultados retornados pela expressão de busca;

Esse processo foi feito de forma iterativa, com os passos 2 a 5 sendo executados continuamente até que o resultado fosse considerado satisfatório.

3.2.1 Primeira Rodada

Num primeiro momento, a base de publicações foi povoada com cerca de 500 artigos que apareceram nas buscas realizadas nos testes dos estudos preliminares (descritos na seção anterior), destes 6 foram classificados como dentro do grupo de controle (o Grupo 1) e 26 tinham indícios que deveriam compor o escopo da busca (o Grupo 2).

Neste momento os testes começaram com a expressão de busca: (enterprise OR organization OR organisation OR corporation OR company OR association OR megacorp OR firm OR multinational OR corporate) AND (improvement OR infrastructure OR environment OR "CASE tool" OR support) AND ("improvement approach" OR initiative) AND ("software process").

A cada consulta foi verificado o número de publicações que faziam parte do grupo de controle e que faziam parte do grupo com indícios de aceitação. Para esta primeira consulta apenas 2 das 6 publicações do Grupo 1 e 3 das 26 do Grupo 2 foram retornadas.

Ao final dessa rodada, a expressão de busca escolhida foi: (enterprise OR organization OR organisation OR corporation OR company OR association OR firm) AND (infrastructure OR environment OR "CASE tool" OR support) AND ("software process") AND (improvement). As seguintes palavras-chave também foram testadas, mas no final, foram desconsideradas: 12207, 15504, affiliate, approach, CMM, CMMI, corporate, initiative, megacorp, MPS.BR, multinational, PSEE, SDE, SPI.

Esta última busca retornou 170 publicações, sendo 4 no Grupo 1 e 9 no Grupo 2. Percebeu-se neste momento que a abrangência da busca não estava sendo boa, pois a relação de publicações de interesse para o estudo em relação ao universo total retornado era baixo. Dessa forma, decidiu-se por uma mudança de estratégia.

3.2.2 Segunda Rodada

Na segunda rodada de testes, reviu-se as publicações presentes no grupo de controle (Grupo 1) e eliminou-se a existência do Grupo 2. Neste momento, foram identificados 29 publicações no grupo de controle. O conteúdo da base de publicações também foi revisto para impedir que o número excessivo (e sem o foco adequado) de publicações identificadas durante os testes preliminares continuassem a influenciar na definição da expressão de busca. Assim, procurou-se, nesta rodada e ao longo de todo o teste de palavras-chave, não expandir demasiadamente o universo de publicações retornadas para a inclusão de uma publicação específica no grupo de controle. Nestes casos, procurou-se olhar o custo-benefício dessa inclusão e a real adequação desta publicação no grupo de controle. Na maioria das vezes, quando a inclusão de uma

publicação causava o retorno de um número muito grande publicações no universo do estudo, era porque a publicação não estava totalmente aderente ao objetivo do estudo. Esse procedimento foi feito para reduzir o esforço associado com a execução do estudo, principalmente na fase de leitura e análise das publicações. Desta forma, ao final deste processo, novos critérios de inclusão e exclusão foram definidos para delinear mais claramente o escopo do estudo a ser realizado.

Na primeira busca realizada nesta rodada, foram identificadas 99 publicações, sendo que 11 pertenciam ao grupo de controle. Foi, então, escolhida aleatoriamente uma publicação do grupo de controle que não tinha sido retornada e procurou-se identificar que palavra-chave poderia ser acrescentada à expressão de busca para retorná-la.

Este processo foi repetido até se chegar à expressão de busca: (enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations OR business OR industry OR industries) AND (infrastructure OR environment OR tool OR tools OR support OR platform OR framework OR "project management system") AND ("software process") AND (improvement OR enactment OR execution OR control).

Ao longo do percurso, novas palavras-chave foram sendo testadas e novas publicações foram consideradas dentro do grupo de controle. A única palavra-chave testada nesta rodada que ficou de fora da expressão de busca foi “CASE tool”.

Os testes, como dito anteriormente, foram realizados na base de dados da Compendex, que deveria, em tese, indexar também as publicações da IEEE. Entretanto, verificou-se que a partir de uma busca na IEEE por “*software process improvement*”, pelo menos dois artigos interessantes para o estudo (“*OWPL: A Gradual Approach for Software Process Improvement In SMEs*” e “*Requirements for a software process improvement support and learning environment*”) não eram indexados pela Compendex. O primeiro não foi indexado assim como nenhum outro publicado na mesma conferência “*32nd Euromicro Conference On Software Engineering And Advanced Applications, SEAA*”. O segundo era indexado pelas duas bibliotecas, mas com conteúdo de *abstracts* diferentes. Estas observações levaram à decisão de que, após a definição da expressão de busca, para a execução do estudo, outras máquinas de busca além da Compendex deveriam ser consideradas.

Por fim, havia 30 publicações no grupo de controle sendo que 24 estavam sendo retornadas pela expressão de busca (em um universo de 300 publicações).

3.2.3 Terceira Rodada

Novamente as publicações no grupo de controle foram revistas. Com esta revisão, novos critérios de inclusão e exclusão foram definidos para delinear mais claramente o escopo do estudo a ser realizado. Percebeu-se que 4 publicações não eram retornadas pela expressão de busca. Destas, duas não eram retornadas pela falta das palavras-chave “*process evolution*” e “*software processes*”. As outras duas não eram retornadas por que não eram, de fato, indexadas pelas máquinas de busca.

Assim, ao final desta última rodada de testes, a expressão de busca foi definida como: (enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations) AND (infrastructure OR environment OR tool OR tools OR support OR platform OR framework OR "project management system") AND ("software processes" OR "software process" OR "process evolution") AND (improvement OR enactment OR execution OR control).

3.3 Definição das Máquinas de Busca

Durante os testes do protocolo e da expressão de busca, verificou-se que a base de dados da Scopus (<http://www.scopus.com/>) retornava a maior parte dos artigos em relação às máquinas de busca da Compendex e da IEEE (<http://ieeexplore.ieee.org/>). Entretanto, para garantir uma maior cobertura dos resultados, decidiu-se utilizar as três máquinas de busca em conjunto, pois havia artigos indexados apenas por uma delas.

A base de dados da IEEE foi incluída porque nem todas as publicações são indexadas corretamente e alguns poucos artigos interessantes para o foco do estudo ficaram de fora do escopo ao utilizar apenas as bases da Scopus e da Compendex. Foi considerado que a cobertura destas três máquinas de busca seria suficiente, pois elas indexam quase tudo o que é relevante na literatura no contexto deste estudo.

A máquina de busca da ACM não foi utilizada por não retornar registros confiáveis (com muitos falsos positivos e com comportamento diferente para buscas com uma mesma expressão de busca).

Para a busca manual, seguindo os critérios do protocolo, foram considerados os anais do Simpósio Brasileiro de Engenharia de Software (SBES) desde 1998 (sua 12ª edição e data de início do escopo deste estudo) e do Simpósio Brasileiro de Qualidade de Software (SBQS) desde 2002 (data de sua 1ª edição).

3.3.1 Expressão de Busca na Biblioteca Digital da Compendex

((enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations) AND (infrastructure OR environment OR tool OR tools OR support OR platform OR framework OR "project management system") AND ("software processes" OR "software process" OR "process evolution") AND (improvement OR enactment OR execution OR control)) wn KY

3.3.2 Expressão de Busca na Biblioteca Digital da IEEE

((('enterprise' 'organization' 'organisation' 'corporation' 'company' 'association' 'corporate' 'organizations' 'organisations' 'companies' 'enterprises' 'corporations') ('infrastructure' 'environment' 'tool' 'tools' 'support' 'platform' 'framework' 'project management system') ('software processes' 'software process' 'process evolution') ('improvement' 'enactment' 'execution' 'control')) (ab, ti) (pyr >= 1998 pyr <= 2007)

3.3.3 Expressão de Busca na Biblioteca Digital da Scopus

TITLE-ABS-KEY((enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations) AND (infrastructure OR environment OR tool OR tools OR support OR platform OR framework OR "project management system") AND ("software processes" OR "software process" OR "process evolution") AND (improvement OR enactment OR execution OR control)) AND PUBYEAR AFT 1998

3.3.4 Instrumento para Consulta Manual

Para a consulta manual foram elaborados dois documentos auxiliares. O primeiro, que pode ser visto na Figura 1, é um quadro com a divisão da expressão de busca (em inglês e em português) em quatro partes (onde cada parte é concatenada com a seguinte através da expressão booleana *AND*).

Após a identificação dos anais de cada publicação selecionada para o estudo, preencheu-se um formulário (cujo modelo pode ser visto na Figura 2) visando a identificação da conferência, ano, número do artigo nos anais, a língua principal do artigo, a primeira página da publicação, a presença de cada grupo de palavras-chave (conforme descrito na Figura 1) e o resultado final (aceito ou não para o escopo da pesquisa). Foi decidido não anotar dados mais completos dos artigos (como nome e autores) para deixar a

pesquisa mais rápida visto que, neste momento, o número do artigo na conferência e a página inicial servem como um bom identificador.

Critério	Inglês	Português
1	enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations	empresa OR organização OR corporação OR companhia OR associação OR corporativo OR organizações OR companhias OR empresas OR corporações OR negócio OR indústria OR indústrias
2	infrastructure OR environment OR tool OR tools OR support OR platform OR framework OR "project management system"	infraestrutura OR infra-estrutura OR ambiente OR ferramenta OR apoio OR suporte OR plataforma OR framework OR arcabouço OR "sistema de gerência de projetos"
3	"software processes" OR "software process" OR "process evolution"	"processo de software" OR "processos de software" OR "evolução de processo"
4	improvement OR enactment OR execution OR control	melhoria OR execução OR controle

Figura 1 – Quadro com Palavras-Chave

<u>Conferência</u>	<u>Ano</u>	<u># Artigo</u>	<u>Lingua</u>	<u>1ª pág.</u>	<u>C1</u>	<u>C2</u>	<u>C3</u>	<u>C4</u>	<u>Resultado</u>

Figura 2 – Quadro para Controle de Pesquisa Manual

Caso um grupamento de palavra-chave não pudesse ser identificado, os demais não eram procurados. Essa decisão também foi feita no intuito de agilizar o processo⁴.

3.4 Identificação do Período de Busca

A identificação do período de busca foi em função da identificação do artigo de controle mais antigo, datado de 1998. Dessa forma, julgou-se que se deveria buscar publicações a partir desta data.

O SBQS surgiu com este nome em 2002, anteriormente, outras edições do evento ocorreram, mas na forma de um evento paralelo ao SBES, o Workshop de Qualidade de Software (WQS). Foi considerado que o WQS não deveria ser incluído no escopo pois além de violar a restrição de escolha de fontes (*"simpósios patrocinados pela Sociedade Brasileira de Computação (SBC) na área de Engenharia de Software"*), na prática, poucos artigos tão antigos foram considerados de fato para o escopo do estudo.

⁴ A título de informação, 382 publicações foram pesquisadas manualmente.

3.5 Execução do Protocolo

A execução final da expressão de busca retornou 233 artigos, conforme pode ser visto na Figura 3. Destas, a maioria se concentrava nas bases da Compendex e Scopus. Das 382 publicações pesquisadas manualmente, apenas 7 passaram pela expressão de busca.

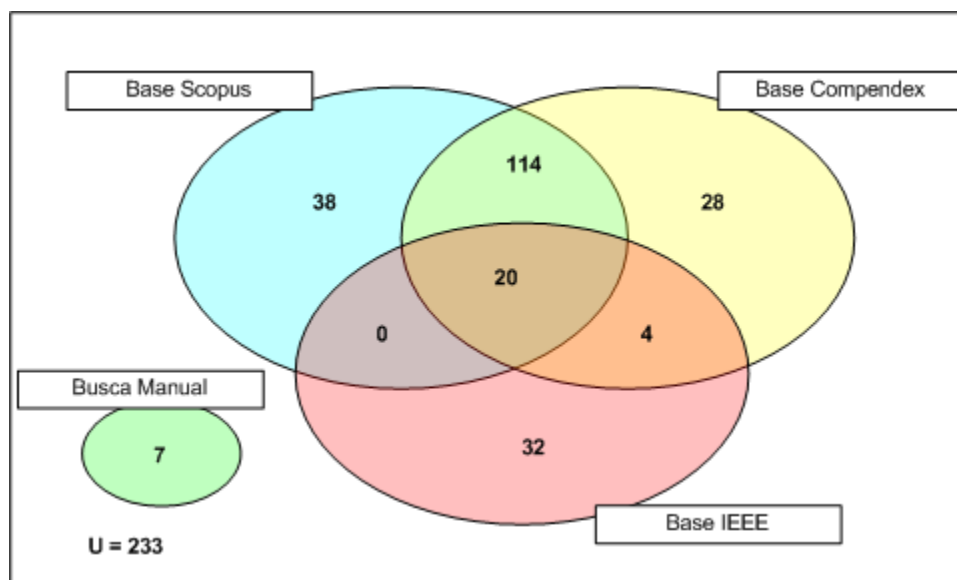


Figura 3 – Artigos Retornados pela Expressão de Busca

Destas publicações, todos os *abstracts* foram lidos e classificados segundo os critérios do primeiro filtro (ver seção 2.6). Foram feitas duas rodadas de leitura. Num primeiro momento, houve dúvidas sobre a aceitação ou não de algumas publicações. Nestes casos, uma nova leitura dos *abstracts* foi feita e um critério de inclusão ou exclusão foi selecionado. Dessa forma, não foi preciso a discussão entre os especialistas para decidir ou não a inclusão de uma publicação. Das 80 publicações, 14 não foram obtidas para análise⁵.

A Figura 4 mostra a distribuição das 80 publicações que passaram pelo primeiro filtro em relação à disponibilidade das máquinas de busca. Pode-se notar, novamente, uma concentração das publicações na Compendex e na Scopus.

⁵ Para os artigos que não se encontravam disponíveis para acesso nas bibliotecas digitais, foram procurados nos anais em meio físico, quando disponíveis, e solicitada uma cópia aos autores. Dessa forma, as 14 publicações citadas não foram obtidas após uma tentativa de contato com seus autores.

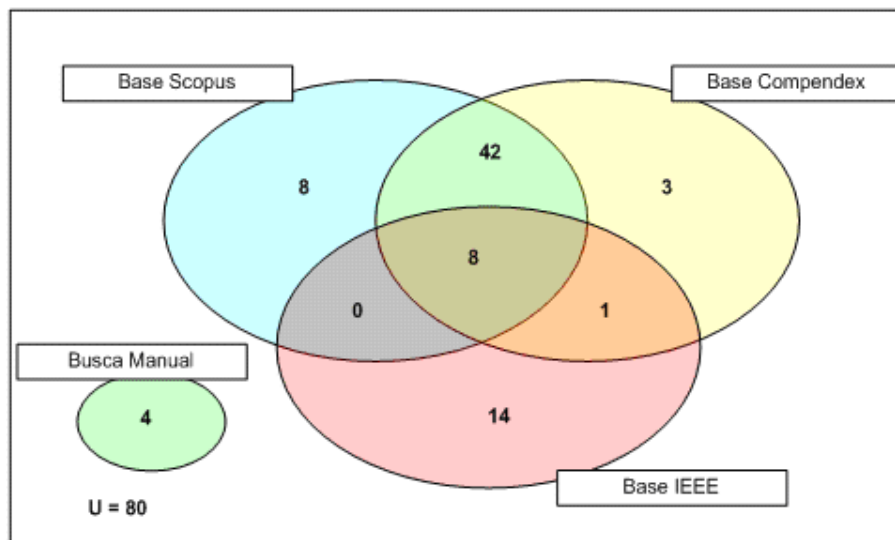


Figura 4 – Artigos Retornados que Passaram pelo Primeiro Filtro

Todas as 66 publicações obtidas foram lidas e novamente classificadas quanto ao primeiro filtro (algumas vezes a percepção que se tem dos artigos lendo apenas o *abstract* e lendo o arquivo completo, muda) e, também, em relação ao segundo filtro. O segundo filtro é mais rígido, para os propósitos do estudo, que o primeiro. Era necessário atender a três diferentes critérios: (i) descrever uma iniciativa de melhoria de processos; (ii) apresentar apoio ferramental; (iii) evidenciar, de alguma forma, o uso das ferramentas ou a aplicação em um caso real da iniciativa de melhoria de processos.

Para todos os artigos lidos, o formulário de coleta de dados foi preenchido. Para as publicações que não permaneceram no estudo (ou seja, falharam em atender o segundo filtro) apenas os itens do formulário de coleta que puderam ser respondidos foram preenchidos. O preenchimento do formulário de coleta foi completo para todas as publicações que passaram no segundo filtro. No total, 27 artigos foram selecionados para o escopo do estudo baseado em revisão sistemática, conforme pode ser visto na Figura 5. Novamente, a predominância de publicações indexadas pela Compendex e pela Scopus foi percebida. A Compendex e a Scopus concentram 87,55% (204 em 233) das publicações selecionadas pelas palavras-chave e 74,07% (20 em 27) das publicações incluídas no estudo.

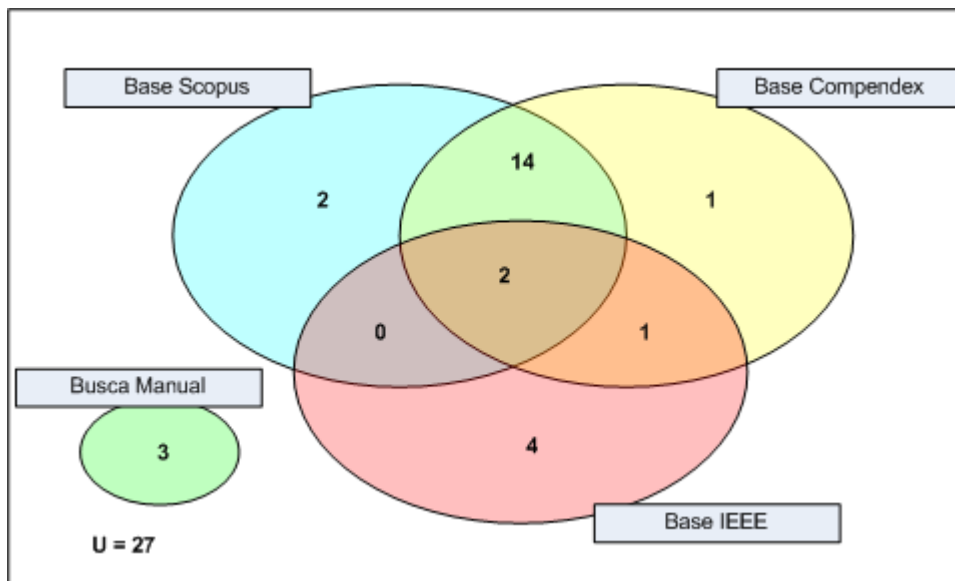


Figura 5 – Artigos Retornados que Passaram pelo Segundo Filtro

3.6 Considerações sobre o Resultado do Estudo

Em geral, o que pôde ser percebido através da execução do estudo é a falta de relatos na literatura especializada sobre apoio ferramental abrangente e específico para as iniciativas de melhoria de processos em curso nas organizações. Também é interessante notar a falta da simples menção de que algum apoio ferramental *foi* utilizado como facilitador destas iniciativas.

Mesmo nos relatos em que o apoio ferramental é descrito, é difícil perceber, também, qual a real influência do apoio ferramental para o sucesso das iniciativas. Não ficando claro, também, observar nos relatos, muitas vezes, se os resultados apresentados se referem ao uso das ferramentas (por exemplo, foram boas ou não) ou aos resultados obtidos na melhoria de processos (por exemplo, resultados bons ou ruins foram observados). Talvez isso advenha do fato de as ferramentas serem encaradas apenas como um mecanismo inerente à execução da iniciativa de melhoria e não como algo fundamental para tal.

Algumas vezes, no entanto, é difícil perceber, também, os reais objetivos de melhoria que a organização quer alcançar e os resultados efetivamente atingidos. Isto, no entanto, pode ser considerado um problema geral na descrição e caracterização de estudos de casos, não ficando restrito às publicações no escopo deste estudo. Não se espera que os artigos demonstrem um modelo para medir o retorno de investimento (ou, ROI, do inglês

return on investment), mas, que, pelo menos, dêem algum indicativo mesmo que subjetivo dos resultados obtidos.

De qualquer forma, após a realização deste estudo, sugere-se um padrão para descrição de iniciativas de melhoria de processos com o uso de apoio ferramental, conforme pode ser visto na Tabela 1.

Tabela 1 – Padrão para Descrição do Uso de Apoio Ferramental em Iniciativas de Melhoria de Processos

Campo do Padrão	Descrição
Caracterização do Relato	Descrição do objetivo do artigo. O preenchimento deste item é sempre obrigatório independente da publicação ser ou não válida para o estudo baseado em revisão sistemática.
Organizações Apoiadas	Descrição das características das organizações onde a iniciativa de melhoria de processos foi definida ou executada e/ou as ferramentas foram utilizadas.
Objetivos da Iniciativa de Melhoria de Processos	Descrição de características da iniciativa de melhoria de processos apresentada no artigo, pode ser uma estratégia definida e estabelecida ou apenas a abordagem utilizada no contexto do estudo de caso apresentado. Descrição dos objetivos da iniciativa de melhoria de processos ou da abordagem utilizada.
Descrição do Apoio Ferramental	Descrição do apoio ferramental apresentado em detalhes suficientes para compreender seu uso na iniciativa de melhoria de processos descrita.
Uso Apoio Ferramental em Contexto Real	Descrição de como (ou onde) o apoio ferramental apresentado foi ou pode ser utilizado em um contexto real.
Resultados Obtidos com a Iniciativa de Melhoria de Processos e o uso do Apoio Ferramental	Descrição da execução ou os resultados obtidos da adoção ou execução da iniciativa de melhoria de processos nas organizações descritas no artigo.

3.6.1 Caracterização das Organizações

A Figura 6 apresenta um gráfico de caracterização das organizações que foram descritas nos artigos do estudo. A grande maioria das empresas foi descrita como sendo pequena ou média⁶. Isto demonstra o pouco nível de informação sobre iniciativas de melhoria de processos de software em corporações ou até mesmo grandes organizações.

⁶ Em algumas publicações não foi possível determinar o tamanho das organizações relatadas. Em outros casos, publicações que tratavam de “pequenas e médias empresas” tiveram as organizações classificadas como “pequena” e “média”.

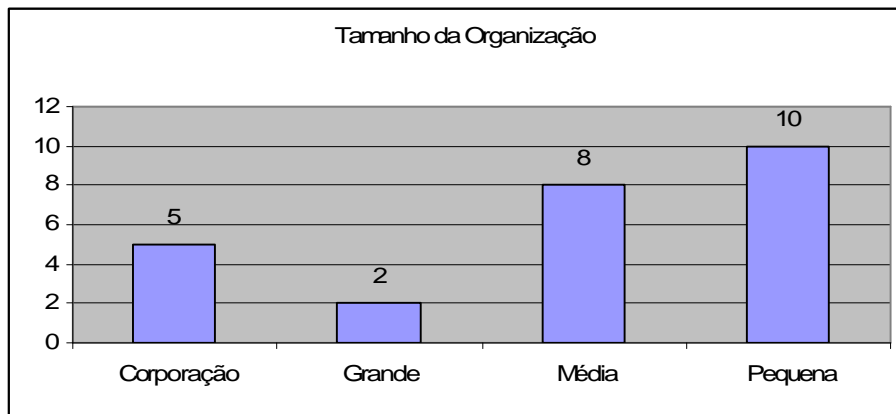


Figura 6 – Tamanho das Organizações

A Figura 7 apresenta um gráfico de caracterização da agregação das organizações durante a iniciativa de melhoria de processos. Em apenas 4 casos foi possível identificar que as organizações descritas estavam compartilhando esforços na iniciativa. Em 13 outros casos foi possível evidenciar que a melhoria de processos de software era uma iniciativa individual das organizações.

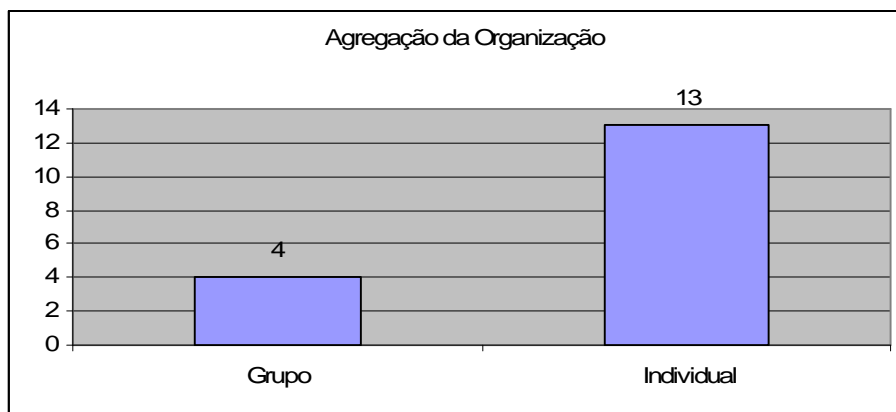


Figura 7 – Agregação das Organizações na Iniciativa de Melhoria de Processos

A Figura 8 apresenta um gráfico de caracterização das normas, modelos, técnicas e métodos mais citados pelas publicações do estudo. A presença neste gráfico não significa que os métodos, normas, modelos e técnicas foram de fato utilizados, apenas que foram citados. Dessa forma, pode-se medir, de algum modo, a influência deles nas organizações na hora de considerar os objetivos e condução das iniciativas de melhoria de processos de software, mas não a utilização deles de fato. Dentre os 27 artigos houve 23 menções ao SW-CMM ou ao CMMI, sendo de longe, o mais influente. Os próximos melhores colocados são normas ISO/IEC 15504 (e sua versão anterior, como relatório técnico, o projeto SPICE), ISO/IEC 12207 e as normas da série ISO 9000.

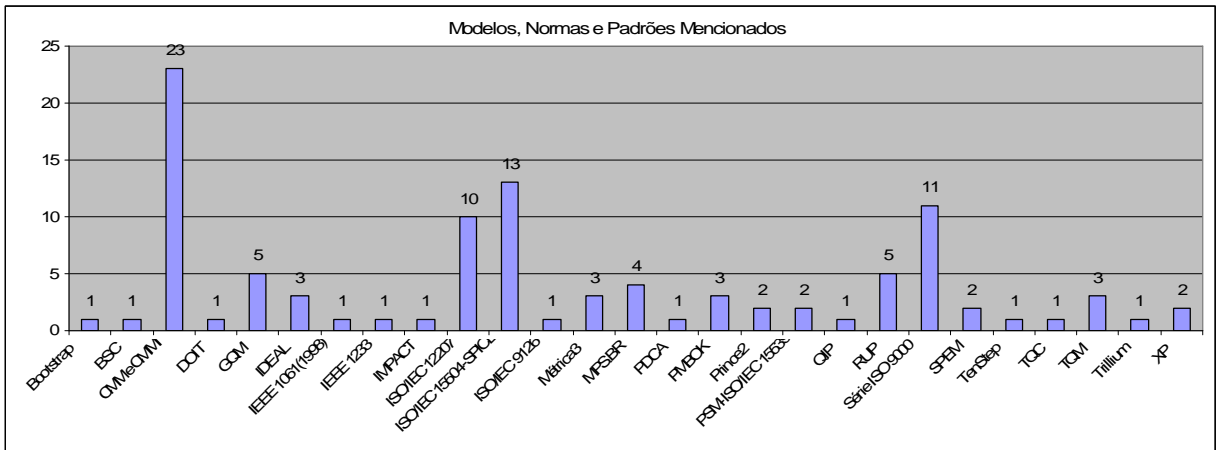


Figura 8 – Influência de Modelos, Normas, Técnicas ou Métodos

3.6.2 Caracterização do Apoio Ferramental

A Figura 9 apresenta um gráfico de caracterização da abrangência do apoio ferramental mencionado nas publicações. O maior número de citações se referia ao apoio à execução de um ou mais processos de software. Em segundo lugar ficaram, empatadas, ferramentas para avaliação da iniciativa de melhoria de processos (incluindo aí a avaliação/*assessment* dos processos utilizados ou a avaliação de processos para subsidiar a iniciativa de melhoria de processos), definição de processos e a gerência de conhecimento ou criação de um repositório de experiências. Outra característica bem mencionada do apoio ferramental foi o a disponibilização de guias eletrônicos de processos.

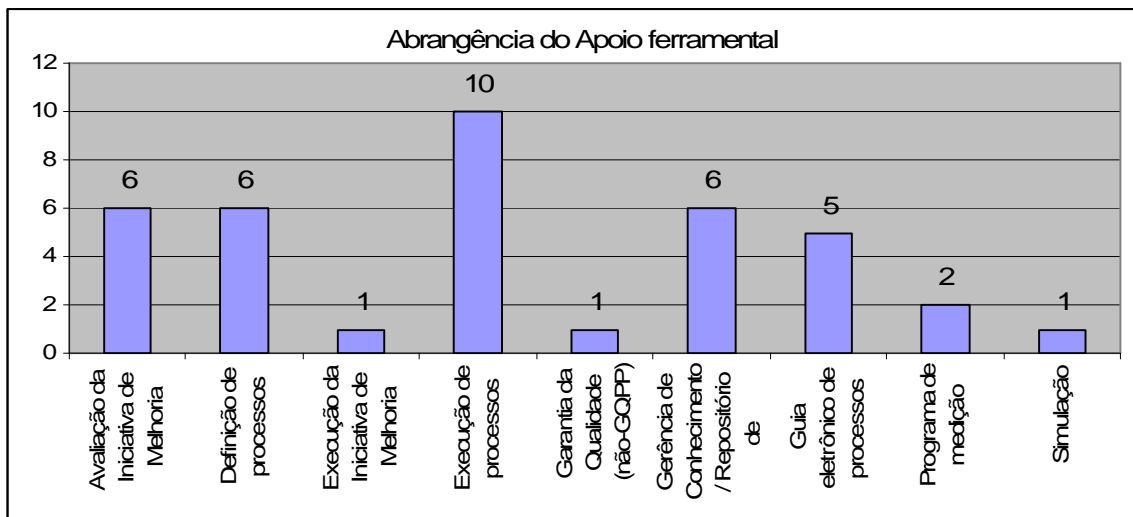


Figura 9 – Abrangência do Apoio Ferramental

A Figura 10 apresenta um gráfico de caracterização da agregação das ferramentas utilizadas no apoio às iniciativas de melhoria de processos de software. O uso de ferramentas isoladas foi mencionado 12 vezes enquanto o uso de ambientes (explicitamente ou de, pelo menos, ferramentas que, pela descrição do texto, poderiam ser consideradas como um ambiente de fato) foi mencionado 11 vezes, praticamente em um empate técnico. Em 4 outras situações foram mencionadas a utilização de ferramentas integradas (mas sem caracterizá-las como um ambiente, por exemplo, a integração de uma ferramenta de coleta de métricas com um sistema de requisição de problemas).

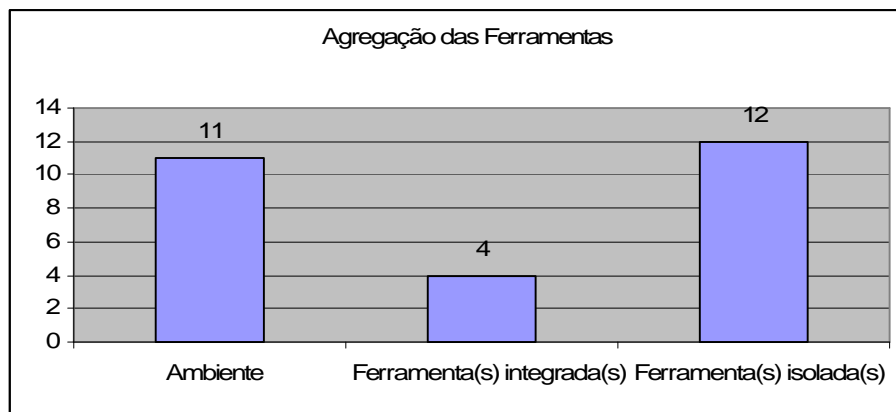


Figura 10 – Agregação das Ferramentas Utilizadas

A Figura 11 apresenta um gráfico de mostra o tipo de apoio ferramental descrito nas publicações. Na maior parte das vezes (24 em 27), foi possível evidenciar o uso do apoio ferramental funcional nas organizações.

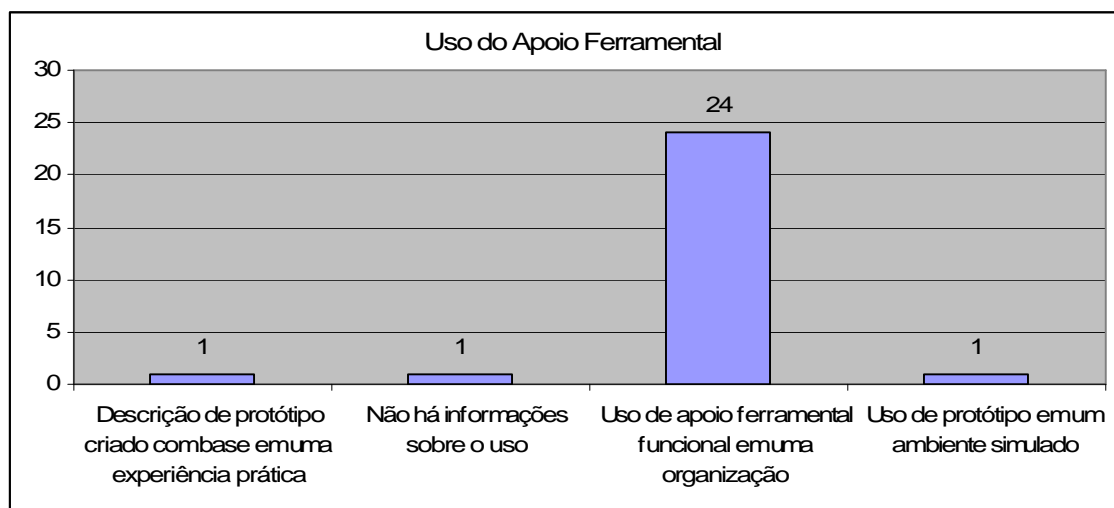


Figura 11 – Uso de Apoio Ferramental

4 Apoio Ferramental para Condução do Estudo

Foi criado um sistema para apoiar a condução do estudo baseado em revisão sistemática. Este sistema foi desenvolvido em MS-Access e apóia a execução do estudo e permite a geração de relatórios específicos. Além disso, como os dados são armazenados em um banco de dados, é fácil criar consultas para análise dos dados e informações.

4.1 Análise da Expressão de Busca

Para a análise da expressão de busca o sistema permite que seja cadastrada uma expressão de busca e o objetivo desta expressão, como pode ser visto na Figura 12. Após isso, pode-se importar os dados obtidos através das máquinas de busca da Compendex, IEEE e Scopus. Após a importação, os itens importados são analisados quanto à presença na lista de publicações de controle previamente preenchida. A partir da análise dos resultados, pode-se registrar comentários para documentar as decisões tomadas. Várias consultas podem ser testadas seguindo o mesmo princípio. A ferramenta permite, também, que se eliminem registros duplicados que sejam retornados pelas máquinas de busca.

The screenshot shows the 'Consultas' window in MS Access. The query 'stringbusca' is selected. The 'objeto' field contains the search expression: TITLE-ABS-KEY((enterprise OR organization OR organisation OR corporation OR company OR association OR corporate OR organizations OR organisations OR companies OR enterprises OR corporations) AND (infrastructure OR environment OR tool OR tools OR support OR platform OR)). The 'objetivo' field contains 'Teste da string de busca na Scopus.' The 'comentário' field contains 'Dos artigos retornados, 132 já tinham sido identificados na busca na Compendex, ou seja, apenas 27 artigos foram encontrados na Compendex.' The 'Resumo dos dados retornados' section shows a table with the following data:

id	num_registros_retornados	num_registros_talvez	num_registros_nao_servem	num_registros_serve
44	177	0	3	20

The 'Erros da Importação' section is empty. The window also has buttons for 'Importar dados da consulta', 'Eliminar Artigos Duplicados', 'Excluir Erros Importação', and 'Excluir Busca Associada'.

Figura 12 – Tela de Teste de Expressão de Busca

Para realizar a importação dos dados, é preciso, fazer a consulta na máquina de busca e salvar os dados retornados no formato BibTeX. Após isso, é preciso converter os dados para o formato XML, conforme pode ser visto na Figura 13, através da ferramenta BibTeXConverter⁷. Caso o arquivo XML gerado não possa ser entendido pelo MS-Access ou algum registro contiver um erro de formatação, a importação não é realizada e os erros encontrados são cadastrados no banco de dados para posterior análise.

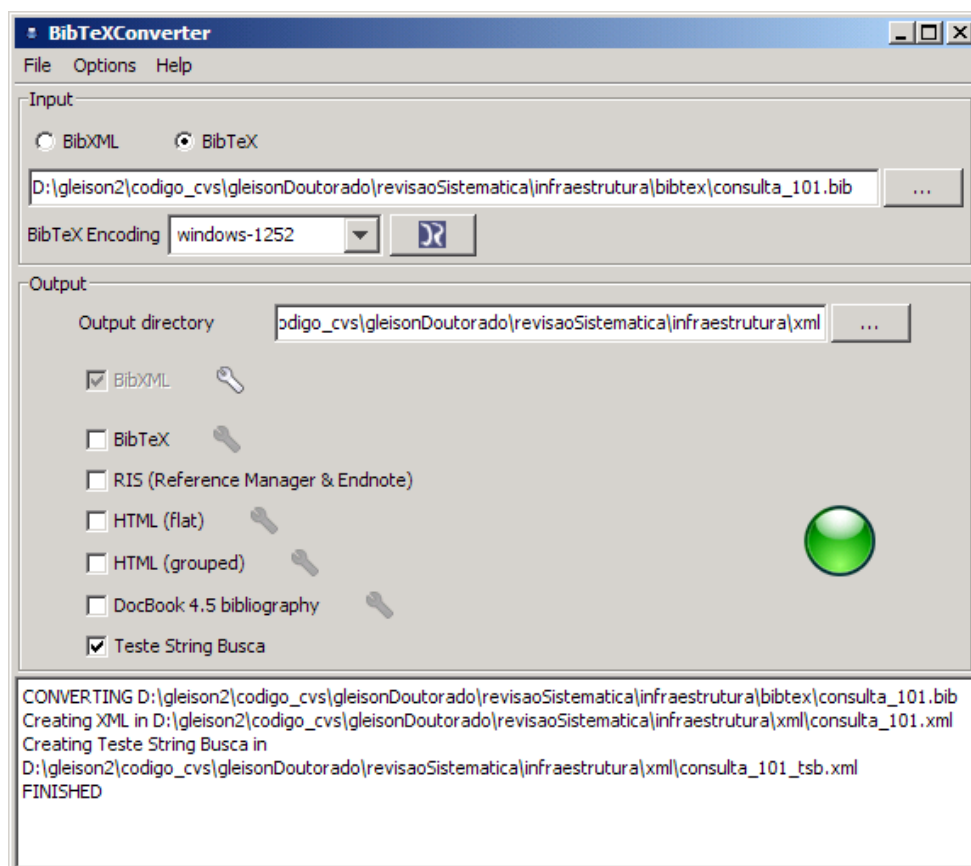


Figura 13 – Transformação de Arquivo no Formato BibTeX para BibXML

4.2 Análise das Publicações

Para análise das publicações é possível utilizar um formulário que, além de exibir os dados importados através do XML derivado do BibTeX, é possível associar a publicação aos critérios dos filtros dentre outras informações, como pode ser visto na Figura 14. Após a análise do primeiro filtro é possível indicar se a publicação está ou não disponível e, se estiver em meio eletrônico, é possível indicar o caminho onde o arquivo está localizado.

⁷ Este programa pode ser obtido no endereço <http://sourceforge.net/projects/bibtexml/>. Foi necessária a adaptação dos conversores disponibilizados na ferramenta para deixar o XML gerado num formato que o MS-Access pudesse interpretar.

Durante a análise da publicação após sua leitura, é possível associá-la novamente a um novo critério do primeiro filtro e ao critério escolhido do segundo filtro (que determinará, de fato, sua inclusão ou não no escopo do estudo). Além disso, é possível descrever os elementos pertinentes à publicação em relação ao formulário de coleta de dados, conforme descrito na seção 2.7. Os itens dos formulários são dinâmicos, podendo ser cadastrados pelos usuários, não estando, assim, restritivos ao escopo do estudo descrito neste anexo.

The screenshot shows a Microsoft Access form titled "Microsoft Access - [publicacoes]". The form is used for data entry and includes several sections:

- Metadata:** Fields for "bibtextkey" (06119759218) and "url" (http://dx.doi.org/10.1002/spe.697).
- Authors:** A text field containing "Canfora, Gerardo Garcia, Felix Piattini, Mario Ruiz, Francisco Visaggio, C.A."
- Journal Information:** "citetitle_journal" (Software - Practice and Experience) and a table with columns: "artpagenums", "volumenum", "pagenums", "issuenum", "confdates_ye", and "confrnum".
- Article Title and Abstract:** "citetitle_article" (Applying a framework for the improvement of software process maturity) and a large text area for "abstract" containing the article's summary.
- Classification and Inclusion:** Dropdown menus for "Classificação?" (Sim), "Inclusão?" (C11-07-Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software em uma organização ou aplicável a uma organização), and "Exclusão?".
- Analysis Section:** A section titled "Análise" with sub-sections for "Inclusão?" (C11-11-Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software), "Exclusão?", "Comentários Gerais", and "Itens da Análise".
- Criteria List:** A list of analysis criteria with dropdown menus for classification, such as "Tamanho da Organização - Média", "Agregação da Organização - Individual", "Modelos e Normas Mencionados - CMM e CMMI", "Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE", and "Modelos e Normas Mencionados - Série ISO 9000".

Figura 14 – Formulário de Coleta de Informações

4.3 Relatórios

Um relatório é disponibilizado de forma a sintetizar todas as informações coletadas sobre a publicação que passou pelos filtros e foi considerada dentro do escopo do estudo baseado em revisão sistemática, como pode ser visto na Figura 15. Através da integração entre ferramentas do MS-Office, é possível exportar o relatório do MS-Access para o MS-Word. Um exemplo deste tipo de relatório poderá ser visto na seção 5.1.

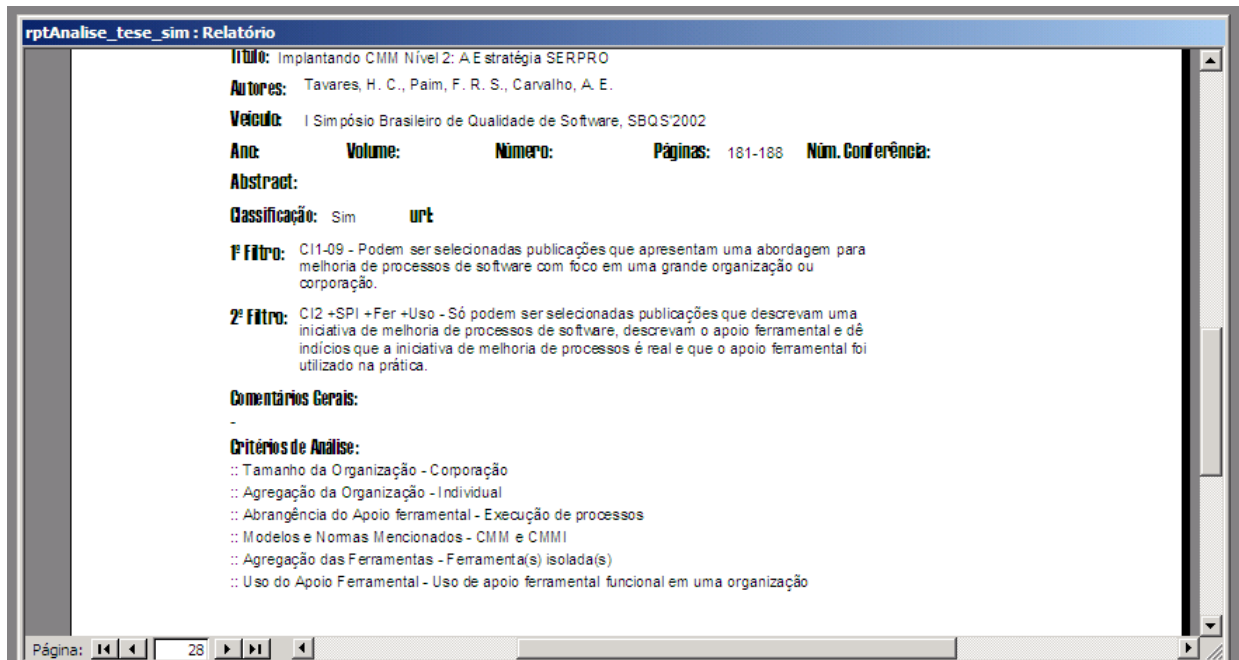


Figura 15 – Relatório Final do Estudo Baseado em Revisão Sistemática

Outro relatório é disponibilizado para facilitar a análise da publicação em relação ao primeiro filtro, que leva em consideração apenas o título e o *abstract*, como pode ser visto na Figura 16.

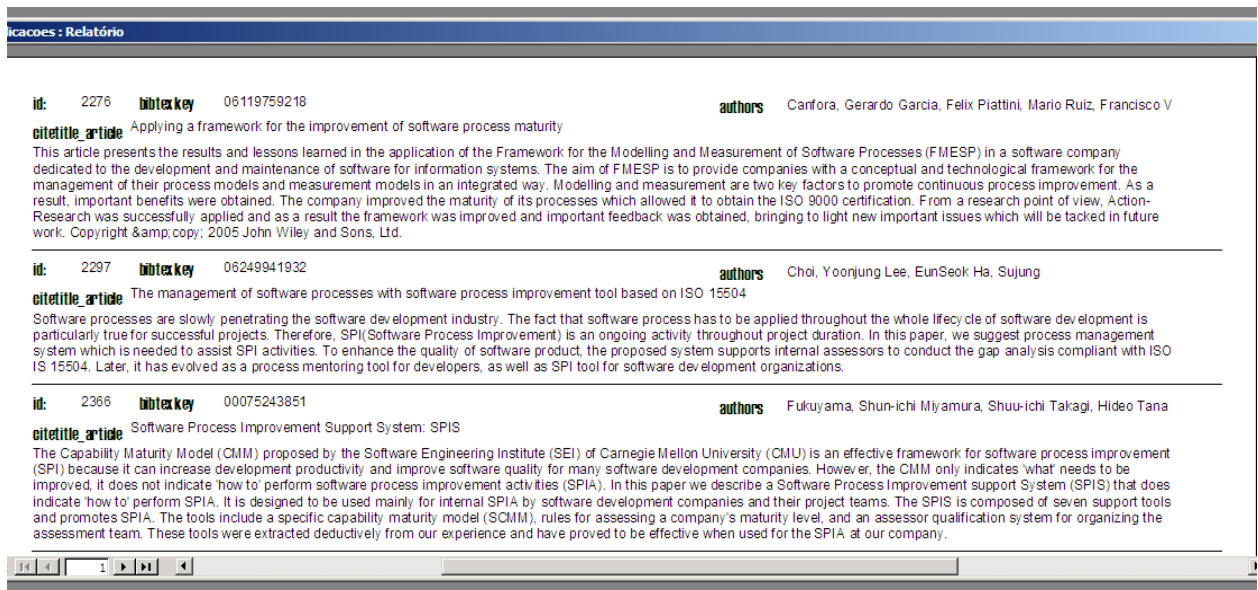


Figura 16 – Relatório Auxiliar para Análise de *Abstracts*

5 Dados Coletados

Esta seção apresenta os dados coletados a partir das publicações selecionadas para o estudo baseado em revisão sistemática segundo os formulários de coleta estabelecidos.

5.1 Dados das Publicações Dentro do Escopo do Estudo

Para cada publicação que passou pelo 2º filtro, é detalhado título, autores, veículo, *abstract*⁸ e, quando possível, ano de publicação, volume e número da revista, páginas, o número da conferência, URL do DOI⁹ da publicação e comentários gerais. Além disso, são exibidos os critérios que fizeram a publicação ser selecionada no primeiro e nos segundos filtros (ver seção 2.6), critérios de análise derivados das questões secundárias (ver seções 2.2 e 2.7) e os itens de análise derivados da questão principal de pesquisa (ver também as seções 2.2 e 2.7)

Título: A Dynamic Integrated Framework for Software Process Improvement

Autores: Ruiz, M. Ramos, I. Toro, M.

Veículo: Softw. Qual. J.

Ano: **Volume:** 10 **Número:** 2 **Páginas:** 181-194 **Núm. Conferência:**

Abstract:

Current software process models (CMM, SPICE, etc.) strongly recommend the application of statistical control and measure guides to define, implement, and evaluate the effects of different process improvements. However, whilst quantitative modeling has been widely used in other fields, it has not been considered enough in the field of software process improvement. During the last decade software process simulation has been used to address a wide diversity of management problems. Some of these problems are related to strategic management, technology adoption, understanding, training and learning, and risk management, among others. In this work a dynamic integrated framework for software process improvement is presented. This framework combines traditional estimation models with an intensive utilization of dynamic simulation models of the software process. The aim of this framework is to support a qualitative and quantitative assessment for software process improvement and decision making to achieve a higher software development process capability according to the Capability Maturity Model. The concepts underlying this framework have been implemented in a software process improvement tool that has been used in a local software organization. The results obtained and the lessons learned are also presented in this paper.

Classificação: Sim **URL:** <http://dx.doi.org/10.1023/A:1020580008694>

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crterios de Análise:

- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Simulação
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever o uso de simulação e dinâmica de sistemas como forma de promover melhoria de processos de software.
- :: Geral - Escopo revisão da literatura: Pode ser útil para falar de simulação e dinâmica de sistemas.
- :: Apoio ferramental - Uso em contexto real: A ferramenta foi utilizada com dados de um projeto passado para analisar se de fato conseguiria prever o comportamento dele ao longo do tempo. Os resultados foram bons, dentro da margem de erro.
- :: Apoio ferramental - Descrição: Há uma ferramenta de apoio à melhoria de processos utilizando simulação e dinâmica de sistemas.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Foi levantado durante essa análise, que durante a construção do modelo de simulação, o gerente do projeto ganhou muito "insight" nos aspectos do projeto

⁸ Apenas para as publicações identificadas através de busca eletrônica.

⁹ DOI é a sigla para *Digital Object Identifier System* que serve para identificar o conteúdo de objetos no ambiente digital. Para mais detalhes consultar: <http://www.doi.org/>.

relacionados ao processo de desenvolvimento que mais influenciam o sucesso do projeto (tempo, custo e qualidade). Também foi percebido que o uso das simulações providas pelo DIFSPI permitiu um melhor entendimento sobre a dinâmica relacionada com o processo de software.

- :: Iniciativa de melhoria de processos - Objetivos: Utilização de simulação em organizações para determinar a melhor abordagem para condução de um projeto. A abordagem é denominada DIFSPI. O objetivo é construir um framework de apoio à avaliação quantitativa e qualitativa para a melhoria de processos e tomada de decisão. Há diferenças de abordagem de acordo com o nível de maturidade CMM da empresa.
- :: Organizações apoiadas - Descrição: Uma empresa nível 1 do CMM.
- :: Organizações apoiadas - Localização: Espanha.

Título: A process improvement framework and a supporting software oriented to chinese small organizations

Autores: Gong, Bo He, Xingui Liu, Weihong

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2006 **Volume:** 3840 NCS **Número:** **Páginas:** 277-286 **Núm. Conferência:** 3840 NCS

Abstract:

Applying the widely accepted process improvement models, such as CMM and ISO 9001, to small software organizations is a challenge for Chinese software industry. Resistance comes from organization structure, software process improvement model, and the market. Small organizations have many characteristics suitable for process improvement, such as rapid communication. How to maximize advantages and minimize shortcomings is a long-term practical task that Chinese software organizations and academe must face. This paper analyzes difficulties that block small organizations, and provides suggestions to resolve these difficulties. And then this paper puts forwards a framework oriented to Chinese small organizations, 'consisting of three phases. To assist in implementing upper framework, supporting software Project Man was developed, which provides a convenient integrated environment for project management and process improvement. Experiments and practices have proved that the framework and supporting software can largely reduce the difficulties of process improvement in small organizations. © Springer-Verlag Berlin Heidelberg 2005.

Classificação: Sim **url:**

1º Filtro: C11-06 - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco em pequenas e médias empresas.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

CrITÉrios de Análise:

- :: Tamanho da Organização - Pequena
- :: Agregação da Organização - Grupo
- :: Abrangência do Apoio ferramental - Execução de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - IDEAL
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Modelos e Normas Mencionados - GQM
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descreve uma abordagem para melhoria de processos de software baseada no IDEAL e o apoio ferramental utilizado nesta abordagem. Mostra os resultados obtidos em uma das empresas em que a abordagem e as ferramentas foram utilizadas.
- :: Geral - Escopo revisão da literatura: Pode ser útil para falar de outras ferramentas com foco na execução de processos baseados no CMM/CMMI.
- :: Apoio ferramental - Uso em contexto real: Segundo os autores foi utilizada em mais de 20 empresas chinesas, inclusive uma subsidiária da Siemens.
- :: Apoio ferramental - Descrição: A ferramenta, Project Man, é composto de 4 unidades: ferramenta de garantia da qualidade (não GQPP) baseada em GQM, integração com o framework de SPI descrito, ferramentas para apoiar as práticas do nível 2 e 3 do CMM, apoio ao trabalho colaborativo.

Possui sistema de automatização de workflow e utiliza plataforma web, e-mail e base multimedia.

Utiliza a abordagem GQM para apoiar a medição dos resultados do processo relativo a requisitos.

- :: Iniciativa de melhoria de processos - Resultados obtidos: Foi utilizada em cerca de 20 pequenas empresas chinesas, incluindo uma subsidiária da Siemens. Os resultados foram verificados pelo menos em 1 dessas empresas e segundo o 90% das 88 pessoas pesquisadas disseram que o projeto de melhoria de processos aumentou enormemente a eficiência e visibilidade dos projetos, riscos foram previstos, comunicação foi facilitada e a quantidade de trabalho pode ser medida quantitativamente.
- :: Iniciativa de melhoria de processos - Objetivos: Apoiar a utilização dos níveis 2 e 3 do SW-CMM com base no IDEAL de forma adequada às pequenas empresas de software da China.

O framework tem 3 etapas: Definição de Processos (definição do programa de SPI incluindo infraestrutura, responsabilidades e comprometerimentos); Controle do processo (refinamento da estratégia, definição dos processos de desenvolvimento, integração das melhorias nos processos em projetos já em andamento, apoio às organizações no uso dos novos processos); Estabilidade do processo (estabelecimento de revisões formais, garantia da qualidade, engenharia de produto, testes formais, revisão por pares, monitoração dos processos via medição, análise de lições aprendidas e desenvolvimento de plano para guiar o programa de SPI).

:: Organizações apoiadas - Descrição: Segundo os autores foi utilizada em mais de 20 empresas chinesas, inclusive uma subsidiária da Siemens.

:: Organizações apoiadas - Localização: China

Título: Applying a framework for the improvement of software process maturity

Autores: Canfora, Gerardo Garcia, Felix Piattini, Mario Ruiz, Francisco Visaggio, C.A.

Veículo: Software - Practice and Experience

Ano: **Volume:** 36 **Número:** 3 **Páginas:** 283-304 **Núm. Conferência:**

Abstract:

This article presents the results and lessons learned in the application of the Framework for the Modelling and Measurement of Software Processes (FMESP) in a software company dedicated to the development and maintenance of software for information systems. The aim of FMESP is to provide companies with a conceptual and technological framework for the management of their process models and measurement models in an integrated way. Modelling and measurement are two key factors to promote continuous process improvement. As a result, important benefits were obtained. The company improved the maturity of its processes which allowed it to obtain the ISO 9000 certification. From a research point of view, Action-Research was successfully applied and as a result the framework was improved and important feedback was obtained, bringing to light new important issues which will be tackled in future work. Copyright © 2005 John Wiley and Sons, Ltd.

Classificação: Sim **url:** <http://dx.doi.org/10.1002/spe.697>

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crítérios de Análise:

- :: Tamanho da Organização - Média
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Definição de processos
- :: Abrangência do Apoio ferramental - Programa de medição
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Modelos e Normas Mencionados - PSM / ISO/IEC 15539
- :: Modelos e Normas Mencionados - SPEM
- :: Modelos e Normas Mencionados - Métrica3
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever as lições aprendidas na aplicação do FMESP (Framework for the Modelling and Measurement of Software Processes) numa organização dedicada ao desenvolvimento e manutenção de software. A organização obteve, como resultado, a certificação ISO 9000.
- :: Apoio ferramental - Uso em contexto real: Foi utilizado na empresa e ajudou a empresa a ser certificada ISO 9000.
- :: Apoio ferramental - Descrição: Ambiente FMESP (Framework for the Modelling and Measurement of Software Processes) que provê o apoio necessário para a representação e gerência do conhecimento relacionado a processos de software a partir das perspectivas de modelagem e medição, ao integrá-las. O ambiente possui componentes para descrição de ontologias (e é baseado no uso de ontologias de processo e de medição), modelagem de processos e apoio à medição de software.
- :: Iniciativa de melhoria de processos - Resultados obtidos: A organização conseguiu ser certificada na ISO 9000. Além disso, a empresa passou a ter seus processos definidos e institucionalizados e também um conjunto de ferramentas de apoio ao processo de medição.
- :: Iniciativa de melhoria de processos - Objetivos: Implantação de ferramentas para apoiar a área de medição numa organização (especialmente interessada numa certificação ISO 9000).
- :: Organizações apoiadas - Descrição: Cronos Ibéria, SA.
- Empresa de tamanho médio (100 funcionários) com foco no desenvolvimento e manutenção de sistemas de informação.
- :: Organizações apoiadas - Localização: Espanha

Título: Approaching software process improvement to organizations

Autores: Amescua, Antonio Garcia, Javier Sanchez-Segura, Maria-Isabel Medina-Dominguez, Fuensanta

Veículo: WSEAS Transactions on Computers

Ano: **Volume:** 5 **Número:** 3 **Páginas:** 507-514 **Núm. Conferência:**

Abstract:

Software engineering process models, methodologies, standards, and in general any kind of formalism in this field are important improvements in the development of software systems but the gap between their definition and the practice of them is still very big. This paper describes a model and a web based tool that represents an approach to software engineering process improvement programs to organizations in order to cover the gap between software engineering in theory and in practice, joining software process improvement and knowledge management disciplines.

Classificação: Sim **url:**

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Abrangência do Apoio ferramental - Avaliação da Iniciativa de Melhoria
- :: Abrangência do Apoio ferramental - Definição de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - PMBOK
- :: Modelos e Normas Mencionados - SPEM
- :: Modelos e Normas Mencionados - Prince2
- :: Modelos e Normas Mencionados - Métrica3
- :: Modelos e Normas Mencionados - DOIT
- :: Modelos e Normas Mencionados - TenStep
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever um modelo e uma ferramenta web para iniciativas de melhoria de processos de software que une gerência de conhecimento a SPI.
- :: Geral - Escopo revisão da literatura: Há parágrafo na introdução sobre apoio ferramental para SPI que pode ser útil.
- :: Apoio ferramental - Uso em contexto real: É descrito apenas que os participantes de um survey realizado para avaliar a aplicabilidade da ferramenta consideraram a ferramenta e a abordagem úteis. Não há muitas informações sobre o uso prático, de fato, da ferramenta ou da abordagem.
- :: Apoio ferramental - Descrição: A ferramenta, denominada PIBOK-Tool, é baseada no PMBOK e apóia a abordagem de SPI definida, denominada PIBOK-Model. Ela permite definir definir e avaliar processos de software. O foco dos processos são os processo gerenciais (daí a influência do PMBOK).
- :: Iniciativa de melhoria de processos - Resultados obtidos: É descrito apenas que os participantes de um survey realizado para avaliar a aplicabilidade da ferramenta consideraram a ferramenta e a abordagem úteis. Não há muitas informações sobre o uso prático, de fato, da ferramenta ou da abordagem.
- :: Iniciativa de melhoria de processos - Objetivos: A iniciativa é baseada na definição e aplicação de uma base de conhecimento sobre melhoria de processos de software, com foco nos processos gerenciais, e conta com o apoio de uma ferramenta para sua execução.
- :: Organizações apoiadas - Descrição: Não fica claro. Apenas são descritos os participantes do survey para avaliação da ferramenta.

Título: Diagnóstico, Definição e Melhoria do Processo de Software: um Estudo de Caso

Autores: Tavares, D. P. D., Fabbri, S. C. P. F., Sanches, R.

Veículo: I Simpósio Brasileiro de Qualidade de Software, SBQS'2002

Ano: **Volume:** **Número:** **Páginas:** 189-197 **Núm. Conferência:**

Abstract:

Classificação: Sim **url:**

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Pequena
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio Ferramental - Avaliação da Iniciativa de Melhoria
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever uma abordagem para diagnóstico e melhoria de processos de software em uma organização apoiada por uma ferramenta de avaliação (assessment) de processos.
- :: Geral - Escopo revisão da literatura: Não relevante.
- :: Apoio Ferramental - Uso em contexto real: Foi utilizado em duas rodadas de melhoria de processos de software com bons resultados (melhorias puderam ser identificadas, priorizadas e, algumas, implantadas).
- :: Apoio Ferramental - Descrição: Questionário de diagnóstico de processo de software com base no SW-CMM Nível 2 (e em 2 áreas de processo do Nível 3) apoiado por uma ferramenta específica (SproQ) de avaliação (assessment).
- :: Iniciativa de melhoria de processos - Resultados obtidos: Uma nova versão do processo de software (sem foco específico nas áreas de processo do SW-CMM) foi definida com base na utilização das ferramentas e uma segunda versão estava em andamento, desta vez mais focada nas áreas de processo do SW-CMM.

- :: Iniciativa de melhoria de processos - Objetivos: Melhoria de processos de forma geral e implantação das áreas de processo do SW-CMM a médio prazo.
- :: Organizações apoiadas - Descrição: Empresa de prestação de serviços que atua no segmento de internet.
- :: Organizações apoiadas - Localização: Brasil, Estado de São Paulo

Título: Enterprise-oriented software development environments to support software products and processes quality improvement

Autores: Montoni, Mariano Santos, Gleison Villela, Karina Rocha, Ana Regina Travassos, Guilherme H. Figueiredo, Savio Mafra, Somulo Albuquerque, Adriano Mian, Paula

Veículo: Lecture Notes in Computer Science

Ano: 2005 **Volume:** 3547 **Número:** **Páginas:** 370-384 **Núm. Conferência:** 3547

Abstract:

Software organizations have to adapt efficiently to cope with clients needs changes and new and evolving technologies in order to guarantee business success. Moreover, organizations must continuously enhance their capability to develop software in order to increase products and processes quality. These characteristics constitute dynamic environments that require specific competences from software engineers such as knowledge related to software technologies, ability to adapt software processes concerning project characteristics, and experience on product and process quality management. This paper presents enterprise-oriented software development environments that support software engineers to execute software processes more effectively and to produce products with better quality. A main feature of these environments is the support offered to organizational knowledge management. Thus the paper also presents the main characteristics of the knowledge management infrastructure integrated to those environments. The practical experience using the environments has shown several benefits, such as an increase of product and process quality, and the preservation of organizational knowledge related to software processes and the development of software products. © Springer-Verlag Berlin Heidelberg 2005.

Classificação: Sim **url:**

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentem apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

Critérios de Análise:

- :: Tamanho da Organização - Pequena
- :: Tamanho da Organização - Média
- :: Agregação da Organização - Grupo
- :: Abrangência do Apoio Ferramental - Gerência de Conhecimento / Repositório de Experiências
- :: Abrangência do Apoio Ferramental - Definição de processos
- :: Abrangência do Apoio Ferramental - Execução de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Apresentar as principais características da infraestrutura de gerência de conhecimento integrada aos ambientes da Estação Taba. Também comenta sobre experiências práticas sobre o uso dos ambientes e discute alguns benefícios percebidos.
- :: Geral - Escopo revisão da literatura: Não é relevante.

::Apoio ferramental - Uso em contexto real: As ferramentas foram utilizadas nas empresas citadas e foi percebido que o a implantação dos processos de software foi agilizada com o uso destas ferramentas. Além disso, a disseminação de boas práticas também foi facilitada. Há a descrição de um survey realizado que corrobora para estas opiniões. Houve redução na execução de tarefas repetitivas durante a definição de processo para projetos, nas atividades de planejamento e na execução e monitoração de projetos.

::Apoio ferramental - Descrição: Estação Taba - Ambiente de desenvolvimento de software com apoio a definição, implantação e melhoria, apóia organizações na execução de seus processos de software com apoio de gerência de conhecimento. Possui várias ferramentas relacionadas à gerência e monitoração de projetos, gerência de conhecimento e medição e análise dentre outras. Utiliza ontologias para auxiliar a gerência de conhecimento.

:: Iniciativa de melhoria de processos - Resultados obtidos: Duas empresas foram certificadas ISO 9000.

:: Iniciativa de melhoria de processos - Objetivos: Implantação de processos em pequenas e médias empresas brasileiras, como parte de um projeto chamado Qualisoft.

:: Organizações apoiadas - Descrição: 19 pequenas e médias empresas do Rio de Janeiro

:: Organizações apoiadas - Localização: Brasil

Título: e-R&D - Effectively managing process diversity

Autores: Ebert, C. De Man, J.

Veículo: Ann. Softw. Eng.

Ano: **Volume:** 14 **Número:** 1-4 **Páginas:** 73-91 **Núm. Conferência:**

Abstract:

Managing process diversity becomes increasingly relevant in software development. Software organizations typically do not work on the greenfield and thus need to integrate external workflows with R&D internal workflow management and heterogeneous development and maintenance processes. To stay competitive with its software development, Alcatel has put in place an orchestrated improvement program of its processes and the underlying engineering tools environment. Why do we call this 'e-R&D'? For two reasons. These improvement activities necessarily fit into the wider context of Alcatel's business process improvement and corporate e-business initiatives. The 'e-R&D' also means enabling of interactive R&D processes and increasing collaborative work across the globe. At Alcatel we realized, during a substantial reengineering of our development and industrialization processes, that the approach to acquire an off-the-shelf process and tailor it to our needs was not applicable. Different processes need to be seamlessly integrated to avoid inconsistencies and inefficiency caused by replicated work. Specific focus is given within this article on how we manage process diversity in a product line where various components are embedded in individual architectures, asking for different but defined development and maintenance processes depending on pre-selected criteria.

Classificação: Sim **url:** <http://dx.doi.org/10.1023/A:1020545406509>

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Corporação
- :: Agregação da Organização - Individual
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Agregação das Ferramentas - Ferramenta(s) integrada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever como a diversidade de processos numa linha de produtos foi tratada.
- :: Geral - Escopo revisão da literatura: Fala sobre diversidade de processos e como gerenciar esta diversidade (ver página 78 para objetivos da gerência desta diversidade).
- Fala sobre necessidade de apoio ferramental para processos.
- Problemas comuns em corporações (ver seção 2).
- :: Apoio ferramental - Uso em contexto real: Foi desenvolvido e utilizado pela Alcatel.
- :: Apoio ferramental - Descrição: Sistema de gerência de workflow e adaptação de processos, denominado e-R&D. Possui uma biblioteca de ativos com apoio de gerência de configuração.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Os benefícios conseguidos, segundo os autores foi: melhora da qualidade, redução do tempo de desenvolvimento, aumento da flexibilidade de engenharia, redução de esforço para tarefas, melhora na comunicação, melhora no alinhamento de processos e ferramentas, maior facilidade de geração de planos de treinamento.
- :: Iniciativa de melhoria de processos - Objetivos: Uso de uma ferramenta de workflow capaz de gerenciar a diversidade de processos dentro da corporação.
- :: Organizações apoiadas - Descrição: Alcatel.
- :: Organizações apoiadas - Localização: França.

Título: Implantando CMM Nível 2: A Estratégia SERPRO

Autores: Tavares, H. C., Paim, F. R. S., Carvalho, A. E.

Veículo: I Simpósio Brasileiro de Qualidade de Software, SBQS'2002

Ano: **Volume:** **Número:** **Páginas:** 181-188 **Núm. Conferência:**

Abstract:

Classificação: Sim **url:**

1º Filtro: CI1-09 - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco em uma grande organização ou corporação.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Corporação
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Execução de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever a estratégia do SERPRO na implantação do Nível 2 do CMM.
- :: Geral - Escopo revisão da literatura: Pode ser útil para descrever a estrutura de um SEPG corporativo. A estratégia de implantação dos processos utiliza uma estrutura de grupos de trabalho responsáveis pela definição e gerência dos processos. Esta estrutura é composta por: GEP (Grupo Executor do Projeto) formado por pessoas de cada unidade e responsável pelas diretrizes e aprovação das decisões e definições relacionadas ao Programa de Modernização do Desenvolvimento (PMoD); GTS (Grupo de Trabalho Serpro) que são grupos corporativos com função de definir a política, processos e procedimentos de cada uma das áreas de processo do nível 2 do CMM e para o processo de Engenharia; GTI (Grupo de Trabalho Interno) que são projeções dos GTS nas unidades de gestão formados por profissionais responsáveis pela elaboração e revisão do conteúdo do processo de desenvolvimento.
- :: Apoio ferramental - Uso em contexto real: As ferramentas foram utilizadas como forma de apoio à execução dos processos de Gerência de projetos, gerência de requisitos, gerência de configuração e divulgação do processo padrão da organização.

:: Apoio ferramental - Descrição: As ferramentas foram utilizadas como forma de apoio à execução dos processos de Gerência de projetos (apoio tarefas como cadastro de solicitações de usuários, medição de esforço e tamanho em pontos por função e controle e acompanhamento dos custos), gerência de requisitos, gerência de configuração e divulgação do processo padrão da organização. Não há elementos que permita identificar se estas ferramentas são integradas.

:: Iniciativa de melhoria de processos - Resultados obtidos: Não fica claro se alguma das unidades já tinha sido avaliada no nível 2 na época mas lições aprendidas são descritas que comprovam a institucionalização dos processos de software em, pelo menos, parte das unidades.

:: Iniciativa de melhoria de processos - Objetivos: Apoiar a implantação do Nível 2 do CMM em diversas unidades da organização.

:: Organizações apoiadas - Descrição: Serviço Federal de Processamento de Dados, vinculada ao Ministério da Fazenda, responsável pela execução de serviços de tratamento de informações e processamento de dados para o governo federal. A empresa está estruturada em unidades de gestão responsáveis por um segmento da administração pública. Cada segmento atende a pelo menos um órgão federal (com características e necessidades próprias), por intermédio de projeções da unidade de gestão em cada uma das unidades organizacionais.

:: Organizações apoiadas - Localização: Unidades espalhadas por 10 capitais brasileiras (Rio de Janeiro, Belém, Belo Horizonte, Curitiba, Fortaleza, Porto Alegre, Recife, Salvador, São Paulo e Brasília)..

Título: Integrated process and knowledge management for product definition, development and delivery

Autores: Koenig, S.

Veículo: Software: Science, Technology and Engineering, 2003. SwSTE '03. Proceedings. IEEE International Conference on

Ano: 2003 **Volume:** **Número:** **Páginas:** 133-141 **Núm. Conferência:**

Abstract:

We describe a software process improvement effort involving the development and deployment of DiME, a proprietary, integrated, collaborative environment for managing product definition, development and delivery processes and information. DiME has been developed as a means to improve and reengineer the management of these processes and facilitate the management of product development information. We describe the situation that led to the DiME initiative, the DiME system and its basic concepts, principles and capabilities, and its enterprise-wide deployment within Comverse. Although the project is still evolving, interim lessons learned from its introduction into the enterprise are presented and related work is discussed.

Classificação: Sim **url:**

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de

processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Grande
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Execução de processos
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descreve um sistema (DiME) cujo objetivo é gerenciar a definição, desenvolvimento e entrega de informação e processos de um produto.
- :: Geral - Escopo revisão da literatura: Não relevante.
- :: Apoio ferramental - Uso em contexto real: É utilizado em projetos na empresa (Comverse).

:: Apoio ferramental - Descrição: Apóia a gerência da definição, desenvolvimento e entrega de informação e processos de um produto.

Apóia os processos relativos a diferentes classes de pessoas nos projetos (por exemplo, gerente, arquiteto, testadores etc.). Além disso, provê um repositório central para armazenamento de dados, automatiza os workflows e possui uma interface padrão para as ferramentas.

:: Iniciativa de melhoria de processos - Resultados obtidos: Segundo os autores, a empresa está passando por uma evolução silenciosa na forma como definida, desenvolve e entrega produtos. O conhecimento resultante é mais rico, está mais disponível, acessível, uniforme e confiável, enquanto os processos estão sendo automatizados, padronizados e medidos em toda a organização.

:: Iniciativa de melhoria de processos - Objetivos: Melhorar a gerência da definição, desenvolvimento e entrega de informação e processos de um produto.

:: Organizações apoiadas - Descrição: Converse, provê soluções para o mercado de comunicações.

:: Organizações apoiadas - Localização: Israel

Título: Measurement based continuous assessment of software engineering processes

Autores: Jarvinen, J.

Veículo: VTT Publications

Ano: **Volume:** **Número:** 426 **Páginas:** 93- **Núm. Conferência:**

Abstract:

Software process assessments are routinely used by the software industry to evaluate software processes before instigating improvement actions. They are also used to assess the capability of an organisation to produce software. Since assessments are perceived as expensive, time-consuming and disruptive for the workplace there is a need to find alternative practices for software process assessment. Especially interesting for this research was to understand and improve the way an organisation can monitor the software process status between regular assessments and how this monitoring can be achieved feasibly in an industrial setting. This thesis proposes a complementary paradigm for software process assessment - measurement based continuous assessment. This approach combines goal-oriented measurement and an emerging standard for software process assessment as the background framework for continuous assessment of the software engineering process. Software tools have been created to support the approach that has been tested in an industrial setting. The results show that the proposed approach is feasible and useful, and provides new possibilities and insights for software process assessment.

Classificação: Sim **url:** http://www.vtt.fi/vtt_show_record.jsp?target=julk&form=sdefe&search=39655

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Grande
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Avaliação da Iniciativa de Melhoria
- :: Abrangência do Apoio ferramental - Programa de medição
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - GQM
- :: Modelos e Normas Mencionados - TQM
- :: Modelos e Normas Mencionados - Bootstrap
- :: Modelos e Normas Mencionados - BSC
- :: Modelos e Normas Mencionados - Trillium

- :: Modelos e Normas Mencionados - ISO/IEC 9126
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)

:: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

::Geral - Objetivo: O objetivo da tese é prover uma abordagem para avaliação (assessment) dos processos de software baseada em medições e com o uso de uma ferramenta.

::Geral - Escopo revisão da literatura: Se for necessário falar sobre assessment ou mais especificamente em assessment baseado em medições.

A estrutura da introdução da tese também é interessante.

A introdução fala sobre números relacionados a SPI e também indicadores de retorno de investimento.

Página 60 fala sobre monitoração da capacidade dos processos de uma organização envolvida com iniciativas de SPI de longo prazo.

::Apoio ferramental - Uso em contexto real: Foi utilizada na empresa como forma de monitoração do processo em uso pelo sistema OMEGA.

::Apoio ferramental - Descrição: Uma extensão da ferramenta MetriFlame denominada SPICE Mapper. O objetivo é associar as medições de um plano QGM com os processos e práticas básicas da ISO/IEC 15504.

:: Iniciativa de melhoria de processos - Resultados obtidos: Segundo os autores, a abordagem de avaliação (assessment) contínua proveu valor à empresa, por exemplo, mapeamento das atividades do projeto em relação a um modelo de referência de processo de software deu maior confiança para monitoração do sistema OMEGA. Um indicador do aumento da confiança foi a associação de um medida com novas questões relacionadas a capacidade. Informações sobre avaliações contínuas também proveu novos insights nas seções de feedback do QGM, por exemplo, fatores impactando a execução do processo se tornaram claras para os participantes do processo e, assim, resultaram numa melhor execução do processo.

:: Iniciativa de melhoria de processos - Objetivos: Abordagem para avaliação (assessment) dos processos de software baseada em medições e com o uso de uma ferramenta.

:: Organizações apoiadas - Descrição: A empresa se chama Tokheim e provê sistemas para estações de combustível self-service.

:: Organizações apoiadas - Localização: Não fica claro.

Título: Measuring and improving software process in China

Autores: Wang, Qing Li, Mingshu

Veículo: 2005 International Symposium on Empirical Software Engineering, ISESE 2005

Ano: 2005 **Volume:** **Número:** **Páginas:** 183-192 **Núm. Conferência:**

Abstract:

Measurement is an important facility to support effective and reasonable management. Software development has its inherent property of high-dependency on the personal capability of software engineers that causes the variance and instability of software processes. Therefore measurement for software process has long been a challenge. This paper discusses the major problems in software process measurement, presents an active measurement model (AMM) to support software process improvement (SPI). Based on the AMM, software organizations can establish adaptive measurement process and execute the measure just close-related the process goals, which focuses on their particular business environment. Subsequently, some methods of establishing the adaptive measurement process and the appropriate software process performance baseline to support higher level quantitative management are suggested. The application and implementation of AMM and the related methods is introduced in the end. These works were integrated in a toolkit called SoftPM, which was used widely in China. © 2005 IEEE.

Classificação: Sim **URL:** <http://dx.doi.org/10.1109/ISESE.2005.1541827>

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

CrITÉrios de Análise:

:: Modelos e Normas Mencionados - CMM e CMMI

:: Modelos e Normas Mencionados - Série ISO 9000

:: Modelos e Normas Mencionados - QGM

:: Modelos e Normas Mencionados - TQM

:: Modelos e Normas Mencionados - PSM / ISO/IEC 15539

:: Agregação das Ferramentas - Ambiente

:: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

::Geral - Objetivo: Apresentar um modelo para medição (active measurement model - AMM) para apoiar a melhoria de processos de software. Apresenta também um conjunto de ferramentas denominado SoftPM, utilizado por várias empresas na China.

::Geral - Escopo revisão da literatura: Fala de medição e também é um exemplo de ferramenta que auxilia a gerência de projetos.

Citar no capítulo sobre ambientes que apoiam a melhoria de processos.

Fala de custo de SPI e de problemas em SME.

Fala de necessidade de apoio automatizado nas organizações.

:: Apoio ferramental - Uso em contexto real: Auxilia a definição dos processo padrão e do projeto, estabelecer e manter a biblioteca de ativos de processos, desempenhar atividades de gerência de projeto e garantia de qualidade, medição e avaliação da situação e desempenho do projeto, dentre outros.

:: Apoio ferramental - Descrição: O SoftPM é um conjunto de ferramentas integradas que provêm um ambiente de trabalho cooperativo para gerentes seniores, gerentes de projetos, desenvolvedores, responsáveis pela qualidade, clientes e fornecedores, dentre outros.

A ferramenta mais importante do SoftPM é o PQM que é baseado no TQM e compatível com o ISO9000/CMM/CMMI. É composto por quatro ferramentas que podem ser combinadas de forma a aumentar a aderência ao CMM/CMMI. É baseado num ciclo do PDCA e apoia organizações de diferentes maneiras de acordo com o nível de maturidade pretendido.

:: Iniciativa de melhoria de processos - Resultados obtidos: Segundo os autores, há uma redução de 35% do esforço na iniciativa de melhoria de processos de software de acordo com informações dos usuários das ferramentas.

:: Iniciativa de melhoria de processos - Objetivos: Um modelo integrado de medição denominado Active Measurement Model (AMM) foi desenvolvido. Com ele uma organização pode gerar/adaptar um processo de medição específico para medir e analisar as características relacionadas aos objetivos do processo de acordo com as características da organização e seu cenário de negócio.

:: Organizações apoiadas - Descrição: Ferramentas utilizadas em mais de 100 organizações na China.

:: Organizações apoiadas - Localização: China.

Título: MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings

Autores: Montoni, Mariano Santos, Gleison Rocha, Ana Regina Weber, Kival C. Araujo, Eratostenes E.R. de

Veículo: Software Quality, 2007. WoSQ'07: ICSE Workshops 2007. Fifth International Workshop on

Ano: 2007 **Volume:** **Número:** **Páginas:** 4-4 **Núm. Conferência:**

Abstract:

Small and Medium-size Enterprises (SME) are usually hindered from improving their processes due to the complexity and costs involved in Software Process Improvement (SPI). In this context, Brazilian software industries and research universities are working cooperatively to implement a strategy aiming to improve software processes of Brazilian SME. The main goal of this initiative is to develop and disseminate a Brazilian software process model (named MPS Model) aligned to Brazilian software industry realities. In order to cope with factors that have influence on SPI success, we developed an approach to implement MPS Model-based initiatives through the support of a Process-centered Software Engineering Environment named TABA Workstation. This paper presents the MPS Model components and the main functionalities of the TABA Workstation to support MPS Model-based implementations in SME. We also discuss the achieved results of Brazilian SME that adopted the presented approach.

Classificação: Sim **url:**

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crerios de Análise:

:: Tamanho da Organização - Pequena

:: Tamanho da Organização - Média

:: Agregação da Organização - Grupo

:: Abrangência do Apoio ferramental - Execução de processos

:: Modelos e Normas Mencionados - CMM e CMMI

:: Modelos e Normas Mencionados - MPS.BR

:: Modelos e Normas Mencionados - IDEAL

:: Modelos e Normas Mencionados - ISO/IEC 12207

:: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE

:: Agregação das Ferramentas - Ambiente

:: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

:: Geral - Escopo revisão da literatura: Não há muita coisa relevante, a não ser para o capítulo do Taba.

:: Apoio ferramental - Uso em contexto real: As ferramentas foram utilizadas em empresas brasileiras como auxílio na implantação de CMMI e MPS.BR e foram consideradas como fatores de sucesso em tais avaliações.

:: Apoio ferramental - Descrição: Ambiente de engenharia de software com apoio de gerência de conhecimento e que possui funcionalidades e ferramentas para atender os principais requisitos dos níveis G, F, E, D e C do MPS.BR.

:: Iniciativa de melhoria de processos - Resultados obtidos: Nos últimos 4 anos, 5 empresas foram avaliadas no CMMI níveis 2 e 3, 4 empresas foram avaliadas no MPS.BR. Além disso, há relatos de empresas cuja redução de retrabalho, em relação ao tempo total do projeto, foi de 36%.

:: Iniciativa de melhoria de processos - Objetivos: Definir uma estratégia para implantação de processos baseado

no MPS.BR em pequenas e médias empresas com o uso de um ambiente de engenharia de software denominado Estação TABA.

:: Organizações apoiadas - Descrição: Pequenas e médias empresas do Rio de Janeiro.

:: Organizações apoiadas - Localização: Brasil

Título: Practical approach to development of SPI activities in a large organization Toshiba's SPI history since 2000

Autores: Ogasawara, Hideto Ishikawa, Takashi Moriya, Tetsuro

Veículo: Proceedings - International Conference on Software Engineering

Ano: 2006 **Volume:** 2006 **Número:** **Páginas:** 595-599 **Núm. Conferência:** 2006

Abstract:

For the effective promotion of software process improvement (SPI) activities in a large-scale organization, it is necessary to establish an organizational structure and a deployment method for promotion and to develop training courses, support tools, and other materials. Even if an organizational promotion system is established, the SPI activities of each development department cannot be promoted effectively without SPI community. To promote SPI activities throughout the TOSHIBA group, we organized a Corporate Software Engineering Process Group in April 2000. We also have been focused to establish SPI community, while promoting SPI activities in each development department. The fundamental our operating policy of SPI is 'bottom-up'. This paper discusses the problems encountered in the promotion of SPI activities and presents solutions to the problems. The actual results obtained show that the framework and solutions developed by us can be used to effectively promote SPI activities. Copyright 2006 ACM.

Classificação: Sim **url:**

1º Filtro: C11-09 - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco em uma grande organização ou corporação.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Corporação
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Garantia da Qualidade (não-GQPP)
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - IDEAL
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Modelos e Normas Mencionados - IEEE 1061-1998
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Não há informações sobre o uso

Itens de Análise:

- :: Geral - Objetivo: Descrever o sistema de promoção organizacional para atividades de melhoria de processos de software implementadas no grupo Toshiba.
- :: Geral - Escopo revisão da literatura: - A seção 2 é interessante para a revisão da literatura pois fala de infraestrutura e outras questões importantes para iniciativas de SPI.
- Pode ser útil para justificar necessidade de adaptação de processos (ver Introdução).
- Também pode ser útil para justificar uma estrutura hierárquica de grupos de processos (ver Figura 2).
- :: Apoio ferramental - Uso em contexto real: Não fala da utilização prática das ferramentas.
- :: Apoio ferramental - Descrição: Garantia da qualidade (não-GQPP)
- Apoio às atividades do SEPG

Cita apenas as ferramentas relacionadas a garantia da qualidade e ferramentas de apoio ao trabalho dos SEPG que são utilizadas.

ASQ tools: análise estática, gerência de configuração, gerência de bugs, estimativas de confiabilidades, geração de casos de testes etc.

Ferramentas do SEPG: website, newsletter, mailing list, relatórios de atividades de SPI.

:: Iniciativa de melhoria de processos - Resultados obtidos: - A estrutura do SEPG é hierárquica, existindo o SEPG corporativo e pelo menos 1 por organização subordinada. Também há SEPG relacionados à unidades de negócio que podem estar subordinados aos SEPG organizacionais ou aos corporativos.

- Várias unidades de negócio já foram avaliadas no SW-CMM níveis 2 e 3.

- Em 2006 havia cerca de 60 SEPG em unidades de negócio ou organizações contra 10 em 2001.

- Houve melhora significativa das unidades de negócio em relação a conhecimento e domínio de processos de software.

- O SEPG corporativo tem como responsabilidades: garantia de qualidade (testes, não GQPP) através de ferramentas de análise estática de programas; promoção de ferramentas automáticas de qualidade de software (ASQ tools: análise estática, gerência de configuração, gerência de bugs, estimativas de confiabilidades, geração de casos de testes etc); estabelecimento e execução de cursos para treinamento em engenharia de software.

:: Iniciativa de melhoria de processos - Objetivos: Atingir os objetivos de qualidade, custo e entregas no desenvolvimento de software e estabelecer uma cultura organizacional para tornar possível a melhoria contínua.

:: Organizações apoiadas - Descrição: Toshiba Corporation e subsidiárias

:: Organizações apoiadas - Localização: Japão

Título: Practical software process improvement - the IMPACT project

Autores: Scott, L. Jeffery, R. Carvalho, L. D'Ambra, J. Rutherford, P.

Veículo: Software Engineering Conference, 2001. Proceedings. 2001 Australian

Ano: 2001 **Volume:** **Número:** **Páginas:** 182-189 **Núm. Conferência:**

Abstract:

For many years now software process improvement (SPI) has been recognised as an effective way for companies to improve the quality of the software they produce and the productivity with which they work. Much work has gone into developing and selling improvement paradigms, assessment methods, modelling languages, tools and technologies. The challenge for -small-to-medium software development companies (SMEs) now is to find a way to

apply these SPI technologies to realise their company's improvement goals. For SMEs the most pressing requirements for improvement paradigms are that they are not only effective but that they realise tangible results quickly, can be implemented incrementally and utilise the many existing process improvement technologies. The paper presents a framework for SPI that realises these needs. The framework is designed to utilise a range of improvement technologies and supports continuous and highly focused improvement over many projects, thus producing timely, cost-effective and tangible improvements for SMEs. The effectiveness of the framework is illustrated with its application in a small, Sydney-based, Web development company

Classificação: Sim **url:** <http://dx.doi.org/10.1109/ASWEC.2001.948512>

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crerios de Análise:

- :: Tamanho da Organização - Pequena
- :: Abrangência do Apoio ferramental - Guia eletrônico de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - GQM
- :: Modelos e Normas Mencionados - QIP
- :: Modelos e Normas Mencionados - XP
- :: Modelos e Normas Mencionados - RUP
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: O artigo apresenta um framework de SPI baseado numa adaptação do QIP e do PDCA e mostra a aplicação desse framework em uma empresa australiana.
- :: Geral - Escopo revisão da literatura: Não é relevante.
- :: Apoio ferramental - Uso em contexto real: A ferramenta foi utilizada em um projeto como forma de apoiar a descrição do processo e dos templates a serem utilizados.
- :: Apoio ferramental - Descrição: O apoio ferramental para a abordagem inclui uma ferramenta que serve como um guia eletrônico de processos, o Spearmint/EPG tool. A ferramenta não apoia a abordagem em si, mas a aplicação da abordagem num contexto/objetivo específico.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Foi decidido aplicar a abordagem para melhorar a documentação na organização. Foi utilizado para isso um guia eletrônico de processos. Como resultados foi percebida uma melhora na coleta e documentação de requisitos, planejamento e estimativa mais estruturados do processo e um aumento na percepção da importância do processo na organização.
- :: Iniciativa de melhoria de processos - Objetivos: O framework age no nível do processo e dos projetos. No nível do projeto os projetos são executados de acordo com boas práticas de gerência de projetos (por exemplo, o ciclo PDCA). No nível do processo, experiências e o entendimento a partir de vários projetos são utilizados para entender e melhorar o modelo de processo genérico que é, então, utilizado em futuros projetos. O ciclo do projeto é executado seguindo a abordagem do PDCA e o framework do QIP. O ciclo do processo é composto pelos seguintes etapas e elementos: entendimento, modelo de processo, melhoria, guia de processo, medidas para monitoração.
- :: Organizações apoiadas - Descrição: Allette, pequena empresa da Austrália.
- :: Organizações apoiadas - Localização: Austrália.

Título: Project web and electronic process guide as software process improvement

Autores: Moe, Nils Brede Dingsoyr, Torgeir Nilsen, Ken Rune Villmones, Nils Jakob

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2005 **Volume:** 3792 NCS **Número:** **Páginas:** 175-186 **Núm. Conferência:** 3792 NCS

Abstract:

Software companies have to identify and manage numerous linked processes to function effectively. We describe how a medium-sized software company improved their software development methodology through implementing an electronic process guide. We discuss how involvement in creating an electronic process guide through process workshops influences the use of the guide over time. We have found that the workshop participators were more positive, and had a higher degree of use. Processes developed by the stakeholders themselves seem to be a perfect starting point when introducing a process guide. An evolutionary introduction of the guide created a high and continuous focus on software process improvement in the whole organization. We also found that integrating the existing administrative systems and tools supporting project work with the process guide increased its usefulness. © Springer-Verlag Berlin Heidelberg 2005.

Classificação: Sim **URL:** http://dx.doi.org/10.1007/11586012_17

1º Filtro: C11-12 - Podem ser selecionadas publicações que discutam aspectos relacionados a abordagens de melhoria de processos de software, incluindo apoio ferramental.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

Critérios de Análise:

- :: Tamanho da Organização - Média
- :: Abrangência do Apoio ferramental - Guia eletrônico de processos
- :: Abrangência do Apoio ferramental - Execução de processos
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Agregação das Ferramentas - Ferramenta(s) integrada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever a experiência de uma média empresa que melhorou sua metodologia de desenvolvimento de software através da implementação de um guia eletrônico de processos. Foram integradas, também, ferramentas e sistemas administrativos para apoiar o trabalho nos projetos.
- :: Geral - Escopo revisão da literatura: Requisitos e vantagens e desvantagens para um EPG.
- :: Apoio ferramental - Uso em contexto real: Uso do EPG por 13 meses. Ao longo do tempo o EPG evoluiu para englobar as ferramentas (denominada Project Web, ou PW) necessárias para o planejamento e acompanhamento dos projetos da empresa, a partir do 4o mês, o PW passou a ser mais utilizado do que o EPG.

Foi percebido que o fato de o processo do projeto e as ferramentas mais utilizadas serem disponibilizadas numa única página web fez com que a utilização do sistema aumentasse. Que o armazenamento de todas as informações do projeto no mesmo lugar facilitou o compartilhamento e troca de informações entre os projetos. O sistema se tornou um banco de dados de experiências.

Como ponto negativo, as ferramentas disponibilizadas tinham a concorrência de outras ferramentas já pré-estabelecidas (como MS Word e MS Project). Também percebeu-se que novas ferramentas devem disponibilizadas junto com um maior treinamento para os usuários.

:: Apoio ferramental - Descrição: Guia eletrônico de processos para apoiar a melhoria de processos de software e fazer a ligação dos processos com as ferramentas de apoio, ligadas a planejamento e gerência, disponíveis na empresa.

Estas ferramentas de apoio incluem: planejamento de orçamento e estimativas, com relatórios de progresso; rastreamento de ações; planejamento de riscos; planejamento de pagamentos; documentações iniciais e finais do projeto; inventário; planejamento do projeto; lista de itens a serem entregues com o projeto; arquivamento de documentos e estatísticas relacionadas à execução do projeto.

:: Iniciativa de melhoria de processos - Resultados obtidos: A abordagem evolutiva adotada resultou num foco contínuo na melhoria do processo de software na organização. O alto grau de envolvimento é provavelmente a razão pelo qual o Project Web foi considerado um sucesso. A estratégia de dar foco nas ferramentas e a integração tornou o sistema como um todo mais útil. O uso do EPG e o PW também tornou possível para empresa a manutenção da certificação ISO 9001:2000.

:: Iniciativa de melhoria de processos - Objetivos: Uso de um EPG para melhorar e documentar a metodologia de desenvolvimento da empresa.

:: Organizações apoiadas - Descrição: Kongsberg Spacotec AS ("Spacotec"), produz equipamentos para satélites de observação meteorológicos.

:: Organizações apoiadas - Localização: Noruega

Título: RAMALA: A knowledge base for software process improvement

Autores: Garcia, Javier Rimawi, Yaser Sanchez, Maria Isabel Amescua, Antonio

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2005 **Volume:** 3792 NCS **Número:** **Páginas:** 106-117 **Núm. Conferência:** 3792 NCS

Abstract:

The actual situation of small software organizations in software process definition and improvement is chaotic. Actually, deploying a software process improvement program within such organizations is very difficult, due to its

high cost and small ROI percentage that could be obtained. RAMALA is a knowledge base, supported by a software tool called also RAMALA, that contains a software process framework, which is mainly based on the PMBOK process framework [9], detailed by software engineering experts using the best practices of the main software reference models like CMMI [11] and ISO 15504 [3], and enriched with process assets of the most outstanding software development methodologies. RAMALA is a platform where best practices of any software engineering process are recollected in a process definition form. Small software organizations can define, assess and improve their software processes economically using RAMALA. © Springer-Verlag Berlin Heidelberg 2005.

Classificação: Sim **url:**

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Abrangência do Apoio ferramental - Avaliação da Iniciativa de Melhoria
- :: Abrangência do Apoio ferramental - Gerência de Conhecimento / Repositório de Experiências
- :: Abrangência do Apoio ferramental - Definição de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - PMBOK
- :: Modelos e Normas Mencionados - Prince2
- :: Modelos e Normas Mencionados - Métrica3
- :: Modelos e Normas Mencionados - RUP
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever uma ferramenta e base de conhecimento chamada RAMALA.
- :: Geral - Escopo revisão da literatura: Pode ser útil para falar sobre o planejamento do programa de melhoria de processos.
- :: Apoio ferramental - Uso em contexto real: Foi feito um survey com pessoas com poder de decisão de 8 empresas que utilizaram a ferramenta. O uso da base de conhecimento foi apontada positivamente por 85% dos respondentes.
- :: Apoio ferramental - Descrição: Uma base de conhecimento e uma ferramenta (ambas denominadas RAMALA) que captura o conhecimento de engenharia de software necessário para implantar um programa de melhoria de software em uma organização. Contém um framework de processo de software baseado no PMBOK e com boas práticas vindas do CMMI e da ISO/IEC 15504. A ferramenta apóia a definição de processos (com apoio de uma biblioteca de ativos) e avaliação (assessment) de processo.
- :: Iniciativa de melhoria de processos - Resultados obtidos: O software foi considerado muito capaz (cerca de 90%) de atingir os objetivos organizacionais necessários para representar o processo de software a ser utilizado. O resultado da análise da capacidade de avaliar a qualidade e a eficiência dos processos de gerência de projetos de software de uma organização também foi alto (37,5% disseram que os resultados refletiam a realidade e 62,5% consideraram-nos úteis). Também foi percebida uma redução no custo de avaliação e definição do conjunto de processos padrão da organização.
- :: Iniciativa de melhoria de processos - Objetivos: Prover uma base de conhecimento para apoiar a execução de um programa de melhoria de processos de software numa organização.
- :: Organizações apoiadas - Descrição: O software foi utilizado em 8 organizações de software.
- :: Organizações apoiadas - Localização: Espanha.

Título: Reference model for software process improvement: A Brazilian experience

Autores: Rocha, Ana Regina Montoni, Mariano Santos, Gleison Mafra, Somulo Figueiredo, Savio Albuquerque, Adriano Mian, Paula

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2005 **Volume:** 3792 NCS **Número:** **Páginas:** 130-141 **Núm. Conferência:** 3792 NCS

Abstract:

Recent research efforts about quality in the software area demonstrate that a concentrated effort is necessary to improve software process. Mainly in Brazil, there is an urge to enhance software processes performance aiming to improve the quality of software products and to increase Brazilian organizations competitive advantages both in the national and international markets. This work describes an approach developed to establish the base for Brazilian organizations to improve software processes. The focus of this work is to increase the software development capability of small and medium size companies in a fast pace. The presented approach consists of the development of a Reference Model for software process improvement and an appraisal method for the Brazilian software industry. This model has been deployed in several Brazilian companies thorough the support of Software Development Environments. The pilot experience and empirical validation results of application of the presented approach are also described in this paper. © Springer-Verlag Berlin Heidelberg 2005.

Classificação: Sim **url:**

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crerios de Análise:

- :: Tamanho da Organização - Pequena
- :: Tamanho da Organização - Média
- :: Agregação da Organização - Grupo
- :: Abrangência do Apoio ferramental - Gerência de Conhecimento / Repositório de Experiências
- :: Abrangência do Apoio ferramental - Definição de processos
- :: Abrangência do Apoio ferramental - Execução de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - MPS.BR
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever a abordagem desenvolvida com o objetivo de estabelecer a base para a melhoria de processos de software em empresas brasileiras. A abordagem consiste na adoção do MPS.BR em conjunto com as ferramentas dos ambientes da Estação Taba.
- :: Geral - Escopo revisão da literatura: Não é relevante.
- :: Apoio ferramental - Uso em contexto real: As ferramentas foram utilizadas nas empresas citadas e foi percebido que a implantação dos processos de software foi agilizada com o uso destas ferramentas. Além disso, a disseminação de boas práticas também foi facilitada. Há a descrição de um survey realizado que corrobora para estas opiniões.
- :: Apoio ferramental - Descrição: Estação Taba - Ambiente de desenvolvimento de software com apoio a definição, implantação e melhoria, apoia organizações na execução de seus processos de software com apoio de gerência de conhecimento. Possui várias ferramentas relacionadas à gerência e monitoração de projetos, gerência de conhecimento e medição e análise dentre outras.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Como resultado, 3 empresas obtiveram a certificação ISO 9000 e 1 conseguiu ser avaliada no CMMI nível 2. O prazo médio de implantação de processos foi de 8 meses. Segundo os autores 5 organizações devem ser avaliadas no MPS.BR no futuro.
- :: Iniciativa de melhoria de processos - Objetivos: Implantação de processos de software nas organizações visando a adoção de MPS.BR níveis F, G e E.
- :: Organizações apoiadas - Descrição: 18 pequenas e médias empresas do Rio de Janeiro
- :: Organizações apoiadas - Localização: Brasil

Título: Software process management: Practices in China

Autores: Wang, Qing Li, Mingshu

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2006 **Volume:** 3840 NCS **Número:** **Páginas:** 317-331 **Núm. Conferência:** 3840 NCS

Abstract:

Software process management has been proven a useful means to help software organizations improve their development processes and produce high quality products. It focuses on providing process-related products and services to software developer. Chinese software industry is developing rapidly, Effective software process methods, technology and tools that help them produce quality products while reducing the costs are in desperate need. This paper discusses the current state of project management in Chinese software companies and presents a solution and practices meeting this need. © Springer-Verlag Berlin Heidelberg 2005.

Classificação: Sim **url:**

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crerios de Análise:

- :: Abrangência do Apoio ferramental - Execução de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE

- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Modelos e Normas Mencionados - TQM
- :: Modelos e Normas Mencionados - PDCA
- :: Modelos e Normas Mencionados - TQC
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Discute a gerência de projetos em empresas de software chinesas e apresenta uma solução para o problema. Está solução é o SoftPM, um conjunto de ferramentas utilizado por várias empresas na China.
- :: Geral - Escopo revisão da literatura: Fala sobre o uso de CMMI como forma de melhoria de processos de software e dificuldades de aplicação de CMMI.
- Citar no capítulo sobre ambientes que apoiam a melhoria de processos.
- :: Apoio ferramental - Uso em contexto real: Foi utilizado com sucesso em mais de 100 empresas de software chinesas.
- :: Apoio ferramental - Descrição: Sistema integrado de apoio às atividades de gerentes de projetos, alta gerência, engenheiros, testadores, membros da área de garantia da qualidade e outros membros de áreas de apoio. Ajuda a compartilhar os dados coletados, entender o cronograma, esforço e qualidade do projeto e auxilia a comunicação entre os participantes do projeto.
- A ferramenta mais importante do SoftPM é a PQM (Plataforma para Gerência da Qualidade) que é baseado no TQM e compatível com o ISO9000/CMM/CMMI. É composto por quatro ferramentas que podem ser combinadas de forma a aumentar a aderência ao CMM/CMMI. É baseado num ciclo do PDCA e apóia organizações de diferentes maneiras de acordo com o nível de maturidade pretendido.
- Há quatro ferramentas principais no PQM:
 - gerência de projetos: foca no planejamento, monitoração e controle do projeto (de forma compatível com as áreas de processo do CMMI).
 - Biblioteca de ativos de processo: mantém os processos padrão organizacionais e os ativos de processo. Possibilita a definição de processos e permite que tais processos sejam exportados à ferramenta de gerência de projetos. Fornece também templates e guias para a execução das atividades dos processos.
 - Garantia da qualidade: planejamento de qualidade, auditoria de processo e produto, análise de qualidade de dados e controle de ações corretivas.
 - Medição e análise: provê infra-estrutura para apoiar as atividades de medição compatível com os níveis 2, 3, 4 e 5 do CMMI.
- Além destas ferramentas, há duas outras plataformas no SoftPM. Uma (Plataform for Product Engineering - PPE) relacionada a questões de engenharia (por exemplo, levantamento de requisitos, testes, gerência de conhecimento relacionada ao desenvolvimento de produtos) e outra (Plataform for Service Supporting - PSS) cujo foco é em prover serviços a clientes e fornecedores.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Segundo os autores, empresas que utilizam o SoftPM obtém a avaliação com maior efetividade. Ainda segundo os autores, a redução do esforço é de cerca de 35% em relação às atividades do SEPG, SQA e gerentes de projetos.
- :: Iniciativa de melhoria de processos - Objetivos: Uso do conjunto de ferramentas visando a uma avaliação CMMI.
- :: Organizações apoiadas - Descrição: Ferramentas utilizadas em mais de 100 organizações na China.
- :: Organizações apoiadas - Localização: China.

Título: Software Processes Improvement and Specifications for Embedded Systems

Autores: Jun, Dong Rui, Lin Yi-min, He

Veículo: Software Engineering Research, Management & Applications, 2007. SERA 2007. 5th ACIS International Conference on

Ano: 2007 **Volume:** **Número:** **Páginas:** 13-18 **Núm. Conferência:**

Abstract:

CMMI is virtual standard scaling software processes capability, but focuses on 'What' and are too general to be suitable for different enterprises, especially for development process. Hence, a suit of 'software processes improvement solution' referring CMMI level 2 3 for middle size enterprises is established. It is composed of: (1) The software processes specification, which is titled as 'Simplified Parallel Processes'; (2) The software processes management tool 'Future' based on SPP. SPP model has three layers and manage to answer 'How': the top level is project management processes set, the middle layer is project development processes set, the down layer is project supply processes set respectively. Meanwhile, concrete technical development specification for embedded systems is presented. It's frank and helpful for every process to obey in an orderly way. This specification and tool will be refined through practices.

Classificação: Sim **url:** <http://dx.doi.org/10.1109/SERA.2007.121>

1º Filtro: C11-08 - Podem ser selecionadas publicações que apresentam apoio ferramental para execução de processos em uma organização.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crterios de Análise:

- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Execução de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever uma abordagem para a implantação de CMMI níveis 2 e 3 para organizações de software chinesas que tem como foco o desenvolvimento de software embutido. A abordagem é composta pela definição dos processos e de apoio ferramental para a execução dos processos.
- :: Geral - Escopo revisão da literatura: Não há muitas informações novas relevantes.
- :: Apoio ferramental - Uso em contexto real: Apenas lista as organizações que já utilizaram/utilizam a ferramenta descrita sem prover muitos comentários.
- :: Apoio ferramental - Descrição: A ferramenta é baseada na estrutura do processo definido e tem o objetivo de apoiar a execução deste processo.
- Infraestrutura baseada em Java para Web, base de dados MySQL.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Não há informações sobre uso prático, apenas que foi aplicada em 6 empresas.
- :: Iniciativa de melhoria de processos - Objetivos: Definição de processos baseado no CMMI nível 2 adequado ao desenvolvimento de software embutido e execução destes processos apoiadas por uma ferramenta.
- :: Organizações apoiadas - Descrição: Não é específico, mas pelo conteúdo do artigo desenvolvem produtos com software embutido ou diretamente software embutido.
- :: Organizações apoiadas - Localização: China

Título: Taba workstation: Supporting software process deployment based on CMMI and MR-MPS.BR

Autores: Montoni, Mariano Santos, Gleison Rocha, Ana Regina Figueiredo, Savio Cabrai, Reinaldo Barcellos, Rafael Barreto, Ahilton Scares, Andrea Cerdeiral, Cristina Lupo, Peter

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2006 **Volume:** 4034 NCS **Número:** **Páginas:** 249-262 **Núm. Conferência:** 4034 NCS

Abstract:

Deployment of software processes based on reference models is a knowledge-intensive task, i.e., a great amount of technical knowledge must be applied in order to guarantee conformance and adherence of processes deployed to the reference models adopted. Moreover, software process deployers have to deal with organizational and individual cultural problems on a regular basis, for instance, resistances to organizational changes. Therefore, the success of software process deployment within an organization or organizational unit depends on both technical and social aspects of the software process deployment strategy definition and execution. This paper presents the Taba Workstation, an enterprise-oriented Process-centered Software Engineering Environment (PSEE) constituted of an integrated set of tools to support software process deployment based on the Capability Maturity Model Integration (CMMI) and the Reference Model for Brazilian Software Process Improvement (MR-MPS.BR). Software process appraisals demonstrated that the Taba Workstation constitutes one of the most important organizational assets to facilitate the success of software process deployment initiatives and to overcome the inherent difficulties.

© Springer-Verlag Berlin Heidelberg 2006.

Classificação: Sim **url:**

1º Filtro: C11-06 - Podem ser selecionadas publicações que apresentam uma abordagem para melhoria de processos de software com foco em pequenas e médias empresas.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

Critérios de Análise:

- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - MPS.BR
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Agregação das Ferramentas - Ambiente
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Descrever como a implantação de processos de software baseadas no CMMI e no MPS.BR pode ser realizada (e se beneficiar) com a utilização da Estação Taba. São descritas características das ferramentas relacionadas à definição e execução de processos e mostrados benefícios da utilização das ferramentas com base numa pesquisa para identificar fatores de sucesso e de dificuldades encontrados em iniciativas de melhoria de processos de software.
- :: Geral - Escopo revisão da literatura: Há a descrição de alguns ambientes de desenvolvimento de software e suas características relacionadas a definição, implantação e execução de processos de software.
- :: Apoio ferramental - Uso em contexto real: Não foi descrito o uso prático do ambiente ou das ferramentas, entretanto, foi mostrado o resultado de um survey realizado com consultores de software sobre fatores de sucesso de iniciativas de melhoria de processo de software e como a Estação Taba se relaciona com as principais

categorias identificadas.

:: Apoio ferramental - Descrição: Apóia a execução de processos baseados nos níveis 2 e 3 do CMMI. As ferramentas são integradas a uma ferramenta de gerência de conhecimento.

São descritas no artigo as funcionalidades para definição de processos padrão e para projetos específicos.

Descreve também as funções para execução de processos.

:: Iniciativa de melhoria de processos - Resultados obtidos: A iniciativa é baseada na utilização das ferramentas descritas no artigo. Não são dados detalhes da execução da abordagem em ferramentas, entretanto, são discutidos fatores de sucesso e dificuldades relacionadas à implantação de processos e estes são, então, relacionados à utilização da Estação Taba.

:: Iniciativa de melhoria de processos - Objetivos: Implantação de processos com base no CMMI e MPS.BR.

O ponto principal do artigo é justificar que a adoção do conjunto de ferramentas descritas apóiam a adoção de abordagens baseadas na implantação de modelos como o CMMI e o MPS.BR.

:: Organizações apoiadas - Descrição: Não foram descritas com detalhes as organizações que utilizaram as ferramentas.

:: Organizações apoiadas - Localização: Brasil.

Título: Taba workstation: Supporting software process improvement initiatives based on software standards and maturity models

Autores: Ferreira, Analia Irigoyen Ferreira Santos, Gleison Cerqueira, Roberta Montoni, Mariano Barreto, Ahilton Rocha, Ana Regina Figueiredo, Savio Barreto, Andrea Filho, Reinaldo C. Silva Lupo, Peter Cerdeiral, Cristina

Veículo: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)

Ano: 2006 **Volume:** 4257 NCS **Número:** **Páginas:** 207-218 **Núm. Conferência:** 4257 NCS

Abstract:

International software standards and maturity models play an important role in Software Process Improvement initiatives defining best practices and providing knowledge to the definition of software processes. Nevertheless, the definition and deployment of software processes based on that standards and models is an expensive and knowledge intensive task. This paper describes an approach to the definition and deployment of software processes in small and medium size Brazilian companies supported by a Process-centered Software Engineering Environment named Taba Workstation. It also presents results related to a software process improvement initiative undertaken in a Brazilian organization that demonstrates the feasibility of the presented approach. © Springer-Verlag Berlin Heidelberg 2006.

Classificação: Sim **url:**

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

:: Tamanho da Organização - Pequena

:: Tamanho da Organização - Média

:: Abrangência do Apoio ferramental - Definição de processos

:: Abrangência do Apoio ferramental - Execução de processos

:: Modelos e Normas Mencionados - CMM e CMMI

:: Modelos e Normas Mencionados - MPS.BR

:: Modelos e Normas Mencionados - ISO/IEC 12207

:: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE

:: Modelos e Normas Mencionados - Série ISO 9000

:: Agregação das Ferramentas - Ambiente

:: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

:: Geral - Objetivo: Descreve uma abordagem para definição e implantação de processos de software em pequenas e médias empresas brasileiras apoiada para um ambiente de engenharia de software centrado em processo (a Estação Taba). O artigo descreve, também, os resultados obtidos em uma empresa em particular, a BL Informática.

:: Geral - Escopo revisão da literatura: Não é relevante.

:: Apoio ferramental - Uso em contexto real: O ambiente descrito foi utilizado nas empresas citadas e também na empresa que está no foco do artigo. Nesta empresa foi utilizado também em conjunto com ferramentas próprias da organização.

:: Apoio ferramental - Descrição: Estação Taba - Ambiente de desenvolvimento de software com apoio a definição, implantação e melhoria, apóia organizações na execução de seus processos de software com apoio de gerência de conhecimento. As ferramentas estão aderentes a muitas práticas do nível 3 do CMMI.

:: Iniciativa de melhoria de processos - Resultados obtidos: A empresa foco do artigo foi certificada na ISO 9001, foi avaliada no MPS.BR nível F e será avaliada no CMMI nível 3 em breve. Para a empresa em questão, foram

divulgadas informações quantitativas da melhoria de processos como a redistribuição da porcentagem de tempo relativa às atividades dos projetos (gerência, análise, codificação) e também a diminuição do retrabalho (de 44% a 7,3%) no período de 3 anos.

:: Iniciativa de melhoria de processos - Objetivos: Aumentar a capacidade das organizações através do uso adequado de técnicas de engenharia de software para aumentar a qualidade dos produtos de software e, assim, aumentar a competitividade organizacional.

Na primeira fase do projeto o foco era na implantação de processos com foco na ISO 12207. A 2a-fase teve o foco na implantação de CMMI nível 2 e MPS.BR nível F (níveis equivalentes). Na fase 3 o foco foi a implantação de CMMI nível 3 e MPS.BR níveis C. Todas elas com o uso das ferramentas do Taba.

:: Organizações apoiadas - Descrição: 24 empresas já utilizaram as ferramentas, são pequenas e médias empresas no Rio de Janeiro em 4 fases do projeto denominado Qualisoft. O foco do artigo é uma pequena empresa em especial, a BL Informática.

:: Organizações apoiadas - Localização: Brasil

Título: The long-term effects of an EPG/ER in a small software organisation

Autores: Kurniawati, Felicia Jeffery, Ross

Veículo: Proceedings of the Australian Software Engineering Conference, ASWEC

Ano: 2004 **Volume:** 2004 **Número:** **Páginas:** 128-136 **Núm. Conferência:** 2004

Abstract:

This paper presents an assessment of the usage of an improved Electronic Process Guide/Experience Repository (EPG/ER) tool about a year after its installation in a small-to-medium software development company (SME). The current study, conducted over 21 weeks, reveals the long-term effects of EPG/ER usage in the company. The findings not only validate results from previously published preliminary evaluations on the effectiveness of the EPG/ER as a Software Process Improvement (SPI) tool but also demonstrate the ability of the EPG/ER to be incrementally improved to suit the changing needs of the company. Our study shows that tangible benefits from the

use of the EPG/ER can not only be realized quickly but also, more importantly, that the tool remains useful with many more benefits accruing over time.

Classificação: Sim **url:** <http://dx.doi.org/10.1109/ASWEC.2004.1290465>

1º Filtro: C11-08 - Podem ser selecionadas publicações que apresentam apoio ferramental para execução de processos em uma organização.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

:: Tamanho da Organização - Média

:: Agregação da Organização - Individual

:: Abrangência do Apoio ferramental - Gerência de Conhecimento / Repositório de Experiências

:: Abrangência do Apoio ferramental - Guia eletrônico de processos

:: Modelos e Normas Mencionados - ISO/IEC 12207

:: Agregação das Ferramentas - Ferramenta(s) integrada(s)

:: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

:: Geral - Objetivo: Mostrar os benefícios tangíveis do uso de uma ferramenta de EPG/ER (guia eletrônico de processos / repositório de experiências) em uma média organização australiana.

:: Geral - Escopo revisão da literatura: Não relevante.

:: Apoio ferramental - Uso em contexto real: São utilizadas durante a execução de projetos de software pelos participantes do projeto como fonte de conhecimento.

:: Apoio ferramental - Descrição: Divulgação de processos da organização (EPG). Baseada no framework WAGNER.

Repositório de experiências (ER).

:: Iniciativa de melhoria de processos - Resultados obtidos: O sistema de EPG/ER foi utilizado durante 21 semanas e algumas métricas foram coletadas para mostrar a quantidade de acessos, as contribuições realizadas e os itens de conhecimento recuperados.

:: Iniciativa de melhoria de processos - Objetivos: Divulgar os processos da organização, aumentando o seu uso e melhorando a institucionalização. Criação de um repositório de experiências. Aumento da memória organizacional.

:: Organizações apoiadas - Descrição: Allette Systems (desenvolve aplicações web para setores de finanças e negócios).

:: Organizações apoiadas - Localização: Austrália

Título: The management of software processes with software process improvement tool based on ISO 15504

Autores: Choi, Yoonjung Lee, EunSeok Ha, Sujung

Veículo: The 7th International Conference on Advanced Communication Technology, ICACT
2005

Ano: 2005 **Volume:** 2 **Número:** **Páginas:** 933-936 **Núm. Conferência:** 2

Abstract:

Software processes are slowly penetrating the software development industry. The fact that software process has to be applied throughout the whole lifecycle of software development is particularly true for successful projects. Therefore, SPI(Software Process Improvement) is an ongoing activity throughout project duration. In this paper, we suggest process management system which is needed to assist SPI activities. To enhance the quality of software product, the proposed system supports internal assessors to conduct the gap analysis compliant with ISO IS 15504. Later, it has evolved as a process mentoring tool for developers, as well as SPI tool for software development organizations.

Classificação: Sim **url:**

1º Filtro: CI1-03 - Podem ser selecionadas publicações que descrevam um protótipo de ambiente desde que seja acompanhado de uma justificativa das características deste ambiente em relação a um programa de

melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

As figuras não podem ser visualizadas corretamente por problemas na conversão do PDF.

Critérios de Análise:

- :: Tamanho da Organização - Pequena
- :: Tamanho da Organização - Média
- :: Abrangência do Apoio ferramental - Avaliação da Iniciativa de Melhoria
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de protótipo em um ambiente simulado

Itens de Análise:

- :: Geral - Objetivo: Propor um sistema de gerência de processos com o objetivo de apoiar atividades de SPI.
- :: Geral - Escopo revisão da literatura: Apenas se precisar justificar os problemas causados pela ausência de gerência de processos de software. Segundo os autores, isto causa problemas gerais durante o desenvolvimento de software.
- :: Apoio ferramental - Uso em contexto real: As ferramentas foram utilizadas em ambiente simulado.
- :: Apoio ferramental - Descrição: Num primeiro momento foi feita uma versão com o objetivo de apoiar a realização de "gap analysis" com base na norma ISO/IEC 15504. Depois a ferramenta foi evoluída para funcionar como uma ferramenta para mentoring para os desenvolvedores.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Segundo os autores, os participantes do estudo ganharam confiança em sua habilidade de desempenhar os papéis no grupo do projeto.
- :: Iniciativa de melhoria de processos - Objetivos: Foi feito um uso simulado com o objetivo de aplicar a norma ISO/IEC 15504 em um curso de graduação.
- :: Organizações apoiadas - Descrição: As ferramentas foram utilizadas em ambiente simulado com 3 grupos de alunos de um curso de graduação (alunos de tempo integral, alunos de tempo parcial que trabalham em SME e alunos de tempo parcial ligados a projetos de pesquisa).
- :: Organizações apoiadas - Localização: Não fica claro.

Título: The use and effects of an electronic process guide and experience repository: a longitudinal study

Autores: Kurniawati, Felicia Jeffery, Ross

Veículo: Information and Software Technology

Anc: **Volume:** 48 **Número:** 7 **Páginas:** 566-577 **Núm. Conferência:**

Abstract:

This paper presents a consolidated view of two evaluations on the use of an electronic process guide and experience repository within a small software development company. The use and effects of the tool were studied over a period of one and a half years, first for 6 months and then 1 year after its installation, for another 5 months. The tool was used regularly and in a consistent manner in both studies but declining usage was observed in the second study. The repository remained used to retrieve mostly examples and templates but the number of retrievals of anecdotal experiences, such as lessons learned had noticeably increased. Similar benefits such as time saving and improved documentation quality were observed in both studies, with additional benefits in the second study like improved project planning and cost estimation, and easier negotiation and traceability of altered or new system requirements with clients. The initial load that users experienced in learning to use the tool was not observed in the second study. The results show that tangible benefits can be realised quickly and continued to be experienced, leading to users having higher morale and more confidence in executing their tasks. © 2005 Elsevier B.V. All rights reserved.

Classificação: Sim **url:** <http://dx.doi.org/10.1016/j.infsof.2005.06.002>

1º Filtro: CI1-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: CI2 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de

processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Critérios de Análise:

- :: Tamanho da Organização - Pequena
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Gerência de Conhecimento / Repositório de Experiências
- :: Abrangência do Apoio ferramental - Guia eletrônico de processos
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - RUP
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Apresentar um estudo de caso sobre o uso de guia eletrônico de processos e de um repositório de experiências em uma pequena empresa de desenvolvimento de software.
- :: Geral - Escopo revisão da literatura: Pode ser útil pra falar de guias eletrônico de processos ou repositório de experiências.
- :: Apoio ferramental - Uso em contexto real: O uso das ferramentas foi monitorado em dois períodos, um de 26 semanas e outro de 21 semanas.
- :: Apoio ferramental - Descrição: Guia eletrônico de processos com repositório de experiências.
- :: Iniciativa de melhoria de processos - Resultados obtidos: A ferramenta foi utilizada regularmente e de maneira consistente. Dentre os benefícios identificados estão a melhora no planejamento de projetos e na documentação do projeto e o aumento da satisfação dos usuários.
- O estudo serve para apoiar a evidência de que "user guidance" é um importante atributo para a eficiência de repositórios de experiências e que os processos provêm uma estrutura natural e lógica para guiar a execução do trabalho das pessoas.
- :: Iniciativa de melhoria de processos - Objetivos: Uso de um guia eletrônico de processos (EPG) e repositório de experiências (ER) durante a execução de projetos de software.
- :: Organizações apoiadas - Descrição: Allette Systems
- :: Organizações apoiadas - Localização: Austrália

Título: Toward computational support for software process improvement activities

Autores: Sakamoto, Keishi Nakakoji, Kumiyo Takagi, Yasunari Niihara, Naoki

Veículo: Proceedings - International Conference on Software Engineering

Ano: 1998 **Volume:** **Número:** **Páginas:** 22-31 **Núm. Conferência:**

Abstract:

Software organizations and projects need guidance on 'how' to improve software process, not just guidelines on 'what' to improve. Several surveys demonstrate that the Capability Maturity Model (CMM) and ISO-9000 only provide the latter. This paper reports our in-depth analysis on a seventeen-month effort in software process improvement (SPI) at OMRON Corporation. The goal of the analysis was to identify issues and challenges of SPI and to design a step-wise practical method to avoid such problems. Major problems we have found include the lack of shared goal among stakeholders, insufficient understanding of the current progress of SPI efforts, and underutilization of a large amount of complex information generated during SPI. We present the method for software organizations and projects for dealing with the problems, and argue for a knowledge-based SPI support system based on the method.

Classificação: Sim **url:** <http://dx.doi.org/10.1109/ICSE.1998.671099>

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

Apesar do protótipo apresenta não ser funcional ele está bem fundamentado nos problemas e lições aprendidas discutidas no artigo, por isso, o artigo foi considerado para a revisão.

Critérios de Análise:

- :: Tamanho da Organização - Corporação
- :: Agregação da Organização - Individual
- :: Abrangência do Apoio ferramental - Execução da Iniciativa de Melhoria
- :: Abrangência do Apoio ferramental - Gerência de Conhecimento / Repositório de Experiências
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - Série ISO 9000
- :: Agregação das Ferramentas - Ferramenta(s) integrada(s)
- :: Uso do Apoio Ferramental - Descrição de protótipo criado com base em uma experiência prática

Itens de Análise:

- :: Geral - Objetivo: O objetivo do artigo é descrever a iniciativa de melhoria de processos da OMRON Corporation que durou 17 meses. Os autores apresentam os principais problemas da iniciativa e como eles foram solucionados. Também descrevem um protótipo de ferramenta criado com base nesta experiência.
- :: Geral - Escopo revisão da literatura: A discussão sobre problemas de SPI é interessante, assim como os problemas apontados por uma corporação na execução de sua iniciativa de melhoria de processos.

::Apoio ferramental - Uso em contexto real: O protótipo apresentado não foi utilizado, entretanto, sua definição foi fortemente baseada na observação dos problemas encontrados durante a iniciativa de melhoria de processos e parece ser bem fundamentado e consistente. Ele procura suprir várias necessidades identificadas ao longo da iniciativa de melhoria.

::Apoio ferramental - Descrição: Foi criado um protótipo de ambiente denominado PIASS (Process Improvement Activities Support Systems) com ferramentas para apoiar a iniciativa de melhoria de processos na avaliação (assessment) de processos, gerência de conhecimento e gerência das atividades de SPI. Este ambiente possui funções próprias para membros do SEPG, desenvolvedores e gerentes da iniciativa de SPI.

:: Iniciativa de melhoria de processos - Resultados obtidos: Houve uma redução de 30% do esforço em homem-mês no projeto alvo da iniciativa de melhoria.

Além disso, foi percebido uma falta de uma abordagem para gerência de conhecimento pelos membros do SEPG, não foi percebido pela alta gerência a relação entre os esforços de SPI e os benefícios obtidos. Apesar dos problemas reportados Entretanto, a análise realizada pelos envolvidos na iniciativa de SPI gerou a criação de um protótipo de ambiente para apoiar as atividades de SPI na corporação.

:: Iniciativa de melhoria de processos - Objetivos: Os objetivos de SPI foi melhorar o esforço dos projetos (medida via homem/mês).

:: Organizações apoiadas - Descrição: Social Systems Business Group da OMRON Corporation presente em 3 cidades japoneses. Produz sistemas mecânicos (como POS e ATM) com software embarcado com interface com usuários.

:: Organizações apoiadas - Localização: Japão

Título: Uma Abordagem para Melhoria Contínua do Processo de Desenvolvimento de Software

Autores: Malheiros, V., Paim, F. R., Guzzo, H., Mendonça Neto, M. G.

Veículo: V Simpósio Brasileiro de Qualidade de Software, SBQS'2006

Ano: **Volume:** **Número:** **Páginas:** 218-230 **Núm. Conferência:**

Abstract:

Classificação: Sim **url:**

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crterios de Análise:

:: Tamanho da Organização - Corporação

:: Agregação da Organização - Individual

:: Abrangência do Apoio ferramental - Avaliação da Iniciativa de Melhoria

:: Modelos e Normas Mencionados - CMM e CMMI

:: Modelos e Normas Mencionados - PMBOK

:: Modelos e Normas Mencionados - RUP

:: Agregação das Ferramentas - Ferramenta(s) isolada(s)

:: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

::Geral - Objetivo: Descrever uma abordagem e ferramenta apoiar a melhoria de processos de software na organização.

::Geral - Escopo revisão da literatura: Exemplo de melhoria de processos de software em uma corporação. A figura 1 (página 220) exemplifica a estrutura do SEPG.

::Apoio ferramental - Uso em contexto real: A ferramnetta é utilizada pelas diversas unidades de gestão da organização e por todos os envolvidos com os processos para a solicitação e acompanhamento de melhorias no processo padrão.

::Apoio ferramental - Descrição: O objetivo da ferramenta GM-PSDS é viabilizar a análise, controle e monitoração de oportunidades de melhoria nos processos de software, vinculando-as aos ganhos obtidos para o processo, numa abordagem de trabalho cooperativo.

:: Iniciativa de melhoria de processos - Resultados obtidos: A ferramenta tem se mostrado efetiva para a execução e acompanhamento da melhoria nos processos e, através dela, é possível obter dados sobre as melhorias como, por exemplo, quantas melhorias foram abertas, aprovadas, implantação por regional ou unidade de gestão.

:: Iniciativa de melhoria de processos - Objetivos: Apoio ao trabalhado do SEPG durante a implantação de CMMI níveis 2 e 3 na empresa. O programa de melhoria de processos visa garantir treinamento, recursos e fundos para incorporação do PSDS (o processo de desenvolvimento da empresa) pelas unidades de desenvolvimento, e mantê-lo alinhado com as boas práticas de mercado. O programa trata o processo de desenvolvimento de forma corporativa, auferindo-lhe a independência necessária à formulação das políticas e procedimentos e à padronização das práticas executadas pelas unidades do SERPRO que desenvolvem software.

:: Organizações apoiadas - Descrição: Serpro (o artigo não descreve, mas deve-se olhar o artigo de Tavares et al. No SBQS 2002 sobre o Serpro).

:: Organizações apoiadas - Localização: Várias unidades no Brasil (o artigo não descreve, mas deve-se olhar o

artigo de Tavares et al. No SBQS 2002 sobre o Serpro).

Título: Understanding the use of an electronic process guide

Autores: Scott, Louise Carvalho, Lucila Jeffery, Ross D'Ambra, John Becker-Kornstaedt, Ulrike

Veículo: Information and Software Technology

Ano: **Volume:** 44 **Número:** 10 **Páginas:** 601-616 **Núm. Conferência:**

Abstract:

This paper presents a case study of the installation and use of an electronic process guide within a small-to-medium software development company. The purpose of the study is to better understand how software engineers use this technology so that it can be improved and better used to support software process improvement. In the study the EPG was used to guide new processes in a software improvement programme. The use of the EPG was studied over a period of 8 months with data collected through access logs, by questionnaires and by interviews. The results show that the improvement programme was successful in improving project documentation, project management and the company's relationship with its customers. The EPG contributed to the improvement programme by providing support for the creation of templates for key project documentation, assisting with project planning and estimation and providing a forum for discussion of process and work practices. The biggest improvements that could be made to the EPG would be to provide better navigation tools including a graphical overview of the process, provide tailoring facilities, include examples and experience and link to a project management tool. © 2002 Elsevier Science B.V. All rights reserved.

Classificação: Sim **url:** [http://dx.doi.org/10.1016/S0950-5849\(02\)00080-0](http://dx.doi.org/10.1016/S0950-5849(02)00080-0)

1º Filtro: C11-11 - Podem ser selecionadas publicações que apresentam apoio ferramental para melhoria de processos de software.

2º Filtro: C12 +SPI +Fer +Uso - Só podem ser selecionadas publicações que descrevam uma iniciativa de melhoria de processos de software, descrevam o apoio ferramental e dê indícios que a iniciativa de melhoria de processos é real e que o apoio ferramental foi utilizado na prática.

Comentários Gerais:

-

Crterios de Análise:

- :: Tamanho da Organização - Pequena
- :: Abrangência do Apoio ferramental - Guia eletrônico de processos
- :: Modelos e Normas Mencionados - CMM e CMMI
- :: Modelos e Normas Mencionados - ISO/IEC 12207
- :: Modelos e Normas Mencionados - ISO/IEC 15504 / SPICE
- :: Modelos e Normas Mencionados - GQM
- :: Modelos e Normas Mencionados - XP
- :: Modelos e Normas Mencionados - IEEE 1233
- :: Modelos e Normas Mencionados - RUP
- :: Modelos e Normas Mencionados - IMPACT
- :: Agregação das Ferramentas - Ferramenta(s) isolada(s)
- :: Uso do Apoio Ferramental - Uso de apoio ferramental funcional em uma organização

Itens de Análise:

- :: Geral - Objetivo: Apresentar um estudo de caso sobre o uso de guia eletrônico de processos em uma pequena empresa de desenvolvimento de software.
- :: Geral - Escopo revisão da literatura: Pode ser útil pra falar de guias eletrônico de processos.
- :: Apoio ferramental - Uso em contexto real: O maior uso do EPG foi em prover informação para a criação de templates para o desenvolvimento de documentos chave e definir a lista de tarefas dos projetos. O EPG também foi utilizado como um fórum de discussão sobre os templates de documentos. Segundo os autores, no entanto, o EPG não foi utilizado para guiar a execução do processo durante os projetos ou para prover acesso conveniente aos templates (se utilizava os documentos de projetos passados como base para novas elaborações).
- :: Apoio ferramental - Descrição: EPG sobre PageSeeder.
- :: Iniciativa de melhoria de processos - Resultados obtidos: Os participantes do survey reportaram um melhora na documentação, uma melhora no entendimento do processo e das tarefas, melhor elaboração de planos dos projetos, melhor estimativa dos projetos, aumento na satisfação dos funcionários e melhor relação com os clientes.
- :: Iniciativa de melhoria de processos - Objetivos: Uso de um guia eletrônico de processos (EPG) durante a execução de projetos de software.
- :: Organizações apoiadas - Descrição: Allette Systems
- :: Organizações apoiadas - Localização: Austrália

5.2 Dados das Publicações Seleccionadas a Partir das Palavras-Chave

A Tabela 2 apresenta as publicações seleccionadas pela execução da busca pelas palavras-chave. Para cada publicação são apresentados título, autores e dados gerais da publicação. Além disso, é indicado se a publicação passou no primeiro filtro e o critério (conforme lista apresentada na seção 2.6) utilizado para a inclusão ou exclusão durante esta primeira análise. São informadas, também, as máquinas de busca a publicação foi encontrada.

Tabela 2 – Artigos Seleccionados pelas Palavras-Chave

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
1		Liao, Li Leung, Hareton K. N. Qu, Yuzhong	Proceedings of the Eighth IASTED International Conference on Software Engineering and Applications, pp.696-702, Cambridge, MA, United States (2004)	Não	CE1-15		X	X	
2		Murray, F.	IFIP Int. Fed. Inf. Process., pp.109-115, Piercom Limited, Shannon, Ireland	Não	CE1-18		X		
3	4CC: A framework for managing software product development	Rautiainen, Kristian Lassenius, Casper Sulonen, Reijo	EMJ - Engineering Management Journal, pp.27-32, n. 2	Não	CE1-18		X	X	
4	A beginner's look at process improvement documentation	Starbuck, R.A.	CrossTalk, pp.18-20, n. 3, MetaVista Consulting Group	Não	CE1-17		X		
5	A Continuous Improvement Model in ImPProS	Bezerra Oliveira, S.R. Lins de Vasconcel, A.M.	Computer Software and Applications Conference, 2006. COMPSAC '06. 30th Annual International, vol. 2, pp.370-371 (2006)	Sim	CI1-11	X			
6	A Dynamic Integrated Framework for Software Process Improvement	Ruiz, M. Ramos, I. Toro, M.	Softw. Qual. J., pp.181-194, n. 2, Department of Computer Languages, Esc. Tec. Sup. de Ing. Info., University of Seville, Spain	Sim	CI1-07		X		
7	A fast method for analysing and improving complex software processes	Hobday, M. Brady, T.	R D Manage., pp.1-20, n. 1, Ctr. for Res. in Innov. Management, University of Brighton, United Kingdom	Sim	CI1-11		X		
8	A framework for assessing the use of third-party software quality assurance standards to meet FDA medical device software process control guidelines	Bovee, M.W. Paul, D.L. Nelson, K.M.	IEEE Transactions on Engineering Management, pp.465-478, n. 4	Não	CE1-18		X	X	
9	A framework for assisting the design of effective software process improvement implementation strategies	Niazi, Mahmood Wilson, David Zowghi, Didar	Journal of Systems and Software, pp.204-222, n. 2	Não	CE1-14		X	X	
10	A framework for coping with process evolution	Nejmeh, Brian A. Riddle, William E.	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.302-316. 3840 NCS, Beijing, China (2006)	Sim	CI1-05			X	
11	A framework for evaluation and prediction of software process improvement success	Wilson, D.N. Hall, T. Baddoo, N.	Journal of Systems and Software, pp.135-142, n. 2	Sim	CI1-11		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
12	A framework for the effective adoption of software development methodologies	Woo, Fleming Mikusauskas, Romas Bartlett, Dean Law, Rob	Proceedings of the Annual Southeast Conference, pp.198-203, 2006, Melbourne, FL, United States (2006)	Não	CE1-18		X	X	
13	A longitudinal study of software process management in Taiwan's top companies	Li, E.Y. Chen, H.-G. Lee, T.-S.	Total Qual. Manage. Bus. Excellence, pp.571-590, n. 5, Chinese University of Hong Kong, Shatin, Hong Kong	Não	CE1-14		X		
14	A low-overhead method for software process appraisal	Wilkie, F.G. Mc Caffery, F. McFall, D. Lester, Neil Wilkinson, Emmanuel	Software Process Improvement and Practice, pp.339-349	Sim	CI1-11		X	X	
15	A meta-model for requirements engineering in system family context for software process improvement using CMMI	Ceron, Rodrigo Duenas, Juan C. Serrano, Enrique Capilla, Rafael	Lecture Notes in Computer Science, pp.173-188, 3547, Oulu, Finland (2005)	Não	CE1-16		X	X	
16	A method to draw lessons from project postmortem databases	Schalcken, Joost Brinkkemper, Sjaak Van Vliet, Hans	Software Process Improvement and Practice, pp.35-46, n. 1	Não	CE1-18		X	X	
17	A methodology and its support environment for benchmark-based adaptable software process improvement	Wu, Minghui Ying, Jing Yu, Chunyan	Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, pp.5183-5188, 6, The Hague, Netherlands (2004)	Sim	CI1-10	X	X	X	
18	A process improvement framework and a supporting software oriented to chinese small organizations	Gong, Bo He, Xingui Liu, Weihong	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.277-286, 3840 NCS, Beijing, China (2006)	Sim	CI1-06		X	X	
19	A process improvement framework for XP based SMEs	Ramachandran, Muthu	Lecture Notes in Computer Science, vol. 3556, pp.202-205, 3556, Sheffield, United Kingdom (2005)	Sim	CI1-06		X	X	
20	A process management tool supporting component-based process development and hierarchical management mechanism	Wang, Yasha Li, Dongni He, Xiaoyang	Proceedings - Fifth International Conference on Computer and Information Technology, CIT 2005, pp.906-910, 2005, Shanghai, China (2005)	Sim	CI1-08		X	X	
21	A Process-Agent construction method for software process modeling in SoftPM	Wang, Qing Xiao, Junchao Li, Mingshu Nisar, M. Wasif Yuan, Rong Zhang, Lei	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.204-213, 3966 NCS, Shanghai, China (2006)	Não	CE1-12		X	X	
22	A reflective case study of software process improvement for a small-scale project	Jo, Jung-Hee Choi, Ho-Jin	Proceedings - Fourth Annual ACIS International Conference on Computer and Information Science, ICIS 2005, pp.314-319, 2005, Jeju Island, South Korea (2005)	Não	CE1-13	X	X	X	
23	A software process tailoring system focusing to quantitative management plans	Hikichi, Kazumasa Fushida, Kyohei Iida, Hajimu Matsumoto, Ken'ichi	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.441-446, 4034 NCS, Amsterdam, Netherlands (2006)	Sim	CI1-08		X	X	
24	A study on metrics for supporting the software process improvement based on SPICE	Hwang, Sun-Myung Kim, Hye-Mee	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.71-80, 3647 NCS, Los Angeles, CA, United States (2005)	Não	CE1-13		X	X	
25	A tentative framework for managing software product development in small companies	Rautiainen, K. Lassenius, C. Vahaniitty, J. Pyhajarvi, M. Vanhanen, J.	System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on, vol. , pp.3409-3417 (2002)	Sim	CI1-06	X			
26	A three level framework for process support: The MOWAHS approach	Wang, Alf Inge Sorensen, Carl-Fredrik Conradi, Reidar	Proceedings of the International Conference on Software Engineering Research and Practice, SERP'04, pp.10-16, 1, Las Vegas, NV, United States (2004)	Não	CE1-12		X	X	
27	A tool to improve the software process quality in a R and D center using PSP	Etxaniz, Inaki	WSEAS Transactions on Information Science and Applications, pp.763-770, n. 4	Sim	CI1-11		X	X	
28	A user story driven software development method	Gong, Bo Tian, Li-Yun Li, Zhi	Beijing Gongye Daxue Xuebao / Journal of Beijing University of Technology, pp.43-47, n. SUPPL	Não	CE1-18		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
29	A value-based software process framework	Boehm, Barry Jain, Apurva	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.1-10, 3966 NCS, Shanghai, China (2006)	Não	CE1-18		X	X	
30	Abordagem em Níveis para Avaliação e Melhoria de Processos de Software	Campos, F. B., Albuquerque, A. B., Andrade, J. M., Silva Filho, R. C., Rocha, A. C	V Simpósio Brasileiro de Qualidade de Software, SBQS'2006, pp.100-114 (2006)	Não	CE1-13				X
31	Active measurement model for software process control and improvement	Wang, Qing Li, Ming-Shu Liu, Xia	Ruan Jian Xue Bao/Journal of Software, pp.407-418, n. 3	Não	CE1-18		X	X	
32	Adept: A unified assessment method for small software companies	McCaffery, Fergal Taylor, Philip S. Coleman, Gerry	IEEE Software, pp.24-31, n. 1	Não	CE1-14	X	X	X	
33	Adopting the SW-CMM in a small IT organization	Guerrero, F. Eterovic, Y.	Software, IEEE, pp.29-35, n. 4	Não	CE1-13	X			
34	Aggregating viewpoints for strategic software process improvement - A method and a case study	Karlstrom, D. Runeson, P. Wohlin, C.	IEE Proceedings: Software, pp.143-152, n. 5	Não	CE1-13	X	X	X	
35	An autonomic framework for quantitative software process improvement	Tianfield, H.	Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on, vol. , pp.446-450 (2003)	Sim	CI1-11	X			
36	An empirical investigation on factors affecting software developer acceptance and utilization of electronic process guides	Dyba, Tore Moe, Nils B. Mikkelsen, Edda M.	Proceedings - International Software Metrics Symposium, pp.220-231, Chicago, IL, United States (2004)	Não	CE1-15		X	X	
37	An empirical study of software process in practice	Coleman, Gerry	Proceedings of the Annual Hawaii International Conference on System Sciences, pp.315-, Big Island, HI, United States (2005)	Não	CE1-14		X	X	
38	An empirical study on the utility of formal routines to transfer knowledge and experience	Conradi, R. Dyba?, T.	8th European Engineering Conference (ESEC) and 9th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE-9), vol. , pp.268-276, Vienna (2001)	Não	CE1-14		X		
39	An evaluation of CMMI process areas for small- To medium-sized software development organisations	Wilkie, F.G. McFall, D. McCaffery, F.	Software Process Improvement and Practice, pp.189-201, n. 2	Não	CE1-13		X	X	
40	An evolutionary approach to enterprise process collaborative modeling using intelligent software agents	Tan, Wenan Shen, Weiming Zhao, Jianming Hao, Qi	Proceedings - 2006 10th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2006, pp.1268-1273, Nanjing, China (2006)	Não	CE1-12			X	
41	Application of poor quality indicator model in an emergent software organization	Demirors, O. Guceglioglu, A.S.	Euromicro Conference, 2001. Proceedings. 27th, vol. , pp.225-232 (2001)	Não	CE1-13	X			
42	Applying a framework for the improvement of software process maturity	Canfora, Gerardo Garcia, Felix Piattini, Mario Ruiz, Francisco Visaggio, C.A.	Software - Practice and Experience, vol. 36, pp.283-304, n. 3	Sim	CI1-07		X	X	
43	Applying and adjusting a software process improvement model in practice: the use of the IDEAL model in a small software enterprise	Kautz, Karlheinz Hansen, Henrik Westergaard Thaysen, Kim	Proceedings - International Conference on Software Engineering, pp.626-633, Limerick, Ireland (2000)	Sim	CI1-07		X	X	
44	Applying CMM project planning practices to diverse environments	Johnson, D.L. Brodman, J.G.	Software, IEEE, pp.40-47, n. 4	Não	CE1-13		X		
45	Applying lessons learned from software process assessments to ABET accreditation	Collofello, James	Proceedings - Frontiers in Education Conference, FIE, pp.3-24-3-26-, 1, Savannah, GA, United States (2004)	Não	CE1-13		X	X	
46	Applying Little-JIL to describe Process-Agent knowledge in SoftPM	Xiao, Junchao Osterweil, Leon J. Zhang, Lei Wise, Alexander Wang, Qing	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.214-221, 3966 NCS, Shanghai, China (2006)	Não	CE1-12		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
47	Approaching software process improvement to organizations	Amescua, Antonio Garcia, Javier Sanchez-Segura, Maria-Isabel Medina-Dominguez, Fuensanta	WSEAS Transactions on Computers, pp.507-514, n. 3	Sim	CI1-11		X	X	
48	Assessment of a software process assessment process	Makinen, T. Varkoi, T. Jaakkola, H.	Management of Engineering and Technology, 2001. PICMET '01. Portland International Conference on, vol. 1, pp.437vol.1- (2001)	Não	CE1-13	X			
49	Automated support of software quality improvement	Leung, H.K.N. Li, L. Yuzhong, Q.	Int. J. Qual. Reliab. Manage., pp.230-243, n. 3, Department of Computer Science and Engineering, Southeast University, Nanjing, China	Sim	CI1-02		X		
50	Automatic verification of static policies on software process models	Reis, R.Q. Lima Reis, C.A. Schlebbe, H. Nunes, D.J.	Ann. Softw. Eng., pp.197-234, n. 1-4, Institute of Informatics, Fed. Univ. of Rio Grande do Sul, Campus do Vale, Bloco IV, Porto Alegre, RS, 91501-970, Brazil	Não	CE1-12		X		
51	BBN construction for software process tailoring	Tseng, Wan-Hui Fan, Chin-Feng	IEICE Transactions on Information and Systems, pp.648-655, n. 3	Não	CE1-12			X	
52	Benchmarking Kappa: interrater agreement in software process assessments	El Emam, Khaled	Empirical Software Engineering, pp.113-133, n. 2	Não	CE1-13		X	X	
53	Best practice approaches in know-how and technology transfer methods for manufacturing SMEs	Enzenhofer, W. Chroust, G.	Euromicro Conference, 2001. Proceedings. 27th, vol. , pp.279-286 (2001)	Sim	CI1-06	X			
54	BOOTSTRAP 3.0 - A SPICE Conformant Software Process Assessment Methodology	Kuvaja, P.	Softw. Qual. J., pp.7-19, n. 1, Dept. of Info. Processing Sciences, INFOTECH Research Group, University of Oulu, Oulu, Finland	Não	CE1-04		X		
55	Business-oriented software process improvement based on CMM using QFD	Liu, Xiaoqing Sun, Yan Kane, Gautam Kyoya, Yuji Noguchi, Kunio	Software Process Improvement and Practice, pp.573-589, n. 6	Não	CE1-13		X	X	
56	Case study in innovative process improvement: Code synthesis from formal specifications	Garbett, P. Parkes, J.P. Shackleton, M. Anderson, S.	Microprocessors and Microsystems, pp.417-424, n. 7	Sim	CI1-07		X	X	
57	CASE STUDY: A practical approach for SPI in large Spanish companies	Guzman, Javier Garcia Garcia, Roman Lopez-Cortijo De Seco, Antonio Amescua Agustin, Gonzalo Cuevas	Software Process Improvement and Practice, pp.261-268, n. 3	Sim	CI1-09		X	X	
58	Case-base maintenance for CCBR-based process evolution	Weber, B. Reichert, M. Wild, W.	8th European Conference on Case-Based Reasoning, ECCBR 2006, vol. 4106 LNAI, pp.106-120, 4106 LNAI, Fethiye (2006)	Não	CE1-12		X		
59	CMM e Comprometimento: Um estudo de caso na implantação do nível 2	Scheible, A., Bastos, A. V.	IV Simpósio Brasileiro de Qualidade de Software, SBQS'2005, pp.255-269 (2005)	Não	CE1-08				X
60	CMM Implementation and Organizational Learning: Findings from a Case Study Analysis	Bellini, E. Io Storto, C.	Technology Management for the Global Future, 2006. PICMET 2006, vol. 3, pp.1256-1271 (2006)	Não	CE1-13	X			
61	CMMI framework in small business environment - A case study	Shinde, Mino Shiva, Sajjan G.	Proceedings of the International Conference on Software Engineering Research and Practice, SERP'04, pp.823-826, 2, Las Vegas, NV, United States (2004)	Não	CE1-17		X	X	
62	Conceptual evaluation of methods for engineering situational ISD methods	Leppanen, Mauri	Software Process Improvement and Practice, pp.539-555, n. 5	Não	CE1-18		X	X	
63	Confluence of six sigma, simulation and software development	Mahanti, R. Antony, J.	Manage. Audit. J., pp.739-762, n. 7, Division of Management, Caledonian Business School, Glasgow Caledonian University, Glasgow, United Kingdom	Sim	CI1-11		X		
64	Creating a dual-agility method: The value of method engineering	Henderson-Sellers, B. Serour, M.K.	Journal of Database Management, pp.1-23, n. 4	Não	CE1-13		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
65	Current practices of measuring quality in finnish software engineering industry	Soini, Jari Tenhunen, Vesa Tukiainen, Markku	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.100-110, 4257 NCS, Joensuu, Finland (2006)	Não	CE1-13		X	X	
66	Database implementation for a software process assessment model	Makinen, T. Varkoi, T. Jaakkola, H.	Management of Engineering and Technology, 1999. Technology and Innovation Management. PICMET '99. Portland International Conference on, vol. 1, pp.148vol.1- (1999)	Não	CE1-15	X			
67	Defining a requirements process improvement model	Beecham, S. Hall, T. Rainer, A.	Softw. Qual. J., pp.247-279, n. 3, Department of Computer Science, University of Hertfordshire, College Lane, Hatfield A10 9AB, United Kingdom	Não	CE1-13		X		
68	Design of opportunity tree for organization's process strategy decision-making based on SPICE assessment experience	Song, Ki Won Kim, Haeng Kon Lee, Kyung Whan	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.873-882, 3983 NCS, Glasgow, United Kingdom (2006)	Não	CE1-13		X	X	
69	Design of opportunity tree framework for effective process improvement based on quantitative project performance	Song, Ki-Won Lee, Kyung-Whan	Proceedings - Third ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2005, pp.360-367, 2005, Mount Pleasant, MI, United States (2005)	Não	CE1-14	X	X	X	
70	Development and evaluation of software process improvement methods	Komi-Sirvio, Seija	VTT Publications, pp.175-, n. 535	Não	CE1-20		X	X	
71	Development system security process of ISO/IEC TR 15504 and security considerations for software process improvement	Lee, Eun-Ser Lee, Malrey	Lecture Notes in Computer Science, pp.363-372, n. II, 3481, Singapore (2005)	Não	CE1-04		X	X	
72	Diagnóstico, Definição e Melhoria do Processo de Software: um Estudo de Caso	Tavares, D. P. D., Fabbri, S. C. P. F., Sanches, R.	I Simpósio Brasileiro de Qualidade de Software, SBQS'2002, pp.189-197	Sim	CI1-11				X
73	DuoTracker: Tool Support for Software Defect Data Collection and Analysis	Akinwale, O. Dascalu, S. Karam, M.	Software Engineering Advances, International Conference on, vol. , pp.22-22 (2006)	Sim	CI1-08	X			
74	Eclipse-based management system for process innovation & methodology enhancement	Choi, Yoonjung Ha, Su-Jung Kim, Jin-Sam	Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference, vol. 1, pp.5pp.- (2006)	Não	CE1-12	X	X	X	
75	Effective elements of integrated software development process supported platform	Fang, Min Ying, Jing Wu, Minghui	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3168 NCS, pp.368-377, 3168 NCS, Xiamen, China (2005)	Sim	CI1-08			X	
76	Eliciting software process models with the E&sup>3</sup> language	Jaccheri, Maria Letizia Picco, Gian Pietro Lago, Patricia	ACM Transactions on Software Engineering and Methodology, pp.368-410, n. 4	Não	CE1-12			X	
77	Empirical investigation of innovation diffusion in a software process	Tortorella, M. Visaggio, G.	Int. J. Software Engineer. Knowledge Engineer., pp.595-621, n. 5, DIB - Department of Informatica, University of Bari, Via Orabona 4, 70126 Bari, Italy	Não	CE1-18		X		
78	Enabling software process improvement in agile software development teams and organisations	Salo, Outi	VTT Publications, pp.1-149, n. 618	Não	CE1-14			X	
79	Enterprise-oriented software development environments to support software products and processes quality improvement	Montoni, Mariano Santos, Gleison Villela, Karina Rocha, Ana Regina Travassos, Guilherme H. Figueiredo, Savio Mafra, Somulo Albuquerque, Adriano Mian, Paula	Lecture Notes in Computer Science, vol. 3547, pp.370-384, 3547, Oulu, Finland (2005)	Sim	CI1-08		X	X	
80	e-R&D - Effectively managing process diversity	Ebert, C. De Man, J.	Ann. Softw. Eng., pp.73-91, n. 1-4, Alcatel, 54 rue La Boetie, 75008 Paris, France	Sim	CI1-09		X		

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
81	European experiences with software process improvement	O'Hara, F.	Software Engineering, 2000. Proceedings of the 2000 International Conference on, vol. , pp.635-640 (2000)	Não	CE1-13	X			
82	Evaluation of a metrics framework for product and process integrity	McKeown, Karen A. McGuire, Eugene G.	Proceedings of the Hawaii International Conference on System Sciences, pp.106-, Maui, USA (2000)	Não	CE1-13	X	X	X	
83	Evolution of a software maintenance organization from cost center to service center	Smit, S. de With, P.H.N. van Dijk, G.-J.	Software Maintenance, 2003. ICSM 2003. Proceedings. International Conference on, vol. , pp.209-212 (2003)	Não	CE1-13		X		
84	Experiences from the pilot operation and commissioning phase of a SCM process improvement program	Nattinen, M. Rahikkala, T. Valimaki, A.	EUROMICRO Conference, 1999. Proceedings. 25th, vol. 2, pp.185-192vol.2 (1999)	Sim	CI1-08	X			
85	Experiences with the TSP technology insertion	Trechter, R. Hirmanpour, I.	CrossTalk, pp.13-16, n. 3, AMS, Inc., 421 Seventh ST NE, Atlanta, GA 30308	Não	CE1-13		X		
86	FMESP: Framework for the modeling and evaluation of software processes	Garcia, Felix Piattini, Mario Ruiz, Francisco Canfora, Gerardo Visaggio, Corrado A.	Journal of Systems Architecture, vol. 52, pp.627-639	Sim	CI1-08		X	X	
87	Frameworks of Integration Repository for Software Process Improvement using SOA	Park, Eun-Ju Kim, Haeng-Kon Lee, Roger Y.	Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on, vol. , pp.200-206 (2007)	Sim	CI1-08	X			
88	From compliance to business success: Improving outsourcing service controls by adopting external regulatory requirements	Biro, Miklos Deak, Csilla Ivanyos, Janos Messnarz, Richard	Software Process Improvement and Practice, pp.239-249, n. 3	Não	CE1-13		X	X	
89	From VSC attributes and characteristics to SCM challenges	Rahikkala, T. Seppanen, V.	Euromicro Conference, 2000. Proceedings of the 26th, vol. 2, pp.308-316vol.2 (2000)	Não	CE1-16	X			
90	Global software development projects in one of the biggest companies in Latvia: Is geographical distribution a problem?	Smite, Darja	Software Process Improvement and Practice, pp.61-76, n. 1	Não	CE1-14		X	X	
91	Global software processes definition in a distributed environment	Vanzin, Mariangela Ribeiro, Marcelo Blois Prikladnicki, Rafael Ceccato, Ilmari Antunes, Dante	29th Annual IEEE/NASA Goddard Software Engineering Workshop - Tutorial Notes, pp.57-65, 2005, Greenbelt, MD, United States (2005)	Não	CE1-18			X	
92	Guided support for collaborative modeling, enactment and simulation of software development processes	Fernandez, Alejandro Garzaldeen, Badie Grutzner, Ines Munch, Jurgen	Software Process Improvement and Practice, pp.95-106, n. 2	Sim	CI1-11		X	X	
93	How software process automation affects software evolution: A longitudinal empirical analysis	Barry, Evelyn J. Kemerer, Chris F. Slaughter, Sandra A.	Journal of Software Maintenance and Evolution, pp.1-31, n. 1	Sim	CI1-02		X	X	
94	Identifying high performance ERP projects	Stensrud, Erik Myrtevit, Ingunn	IEEE Transactions on Software Engineering, pp.398-416, n. 5	Não	CE1-18		X	X	
95	Implantando CMM Nível 2: A Estratégia SERPRO	Tavares, H. C., Paim, F. R. S., Carvalho, A. E.	I Simpósio Brasileiro de Qualidade de Software, SBQS'2002, pp.181-188	Sim	CI1-09				X
96	Implementing software process improvement: two cases of technology transfer	Kautz, Karlheinz Nielsen, Peter Axel	Proceedings of the Hawaii International Conference on System Sciences, pp.188-, Maui, USA (2000)	Sim	CI1-07		X	X	
97	Improve by improving software process improvers	Bo?rjesson, A.	Int. J. Bus. Inf. Syst., pp.310-338, n. 3, Ericsson AB, IT University of Gothenburg, Lindholmospiren 11, 417 56 Gothenburg, Sweden	Não	CE1-13		X		
98	Improving software organizations: Agility challenges and implications	Bo?rjesson, A. Mathiassen, L.	Inf. Technol. People, pp.359-382, n. 4, Georgia State University, Atlanta, GA, United States	Não	CE1-13		X		

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
99	Improving the operations and efficiency of processing, transport and storage of heat sensitive materials using a novo wireless remote monitoring and Control System	Zeidler, Gideon	AIChE Annual Meeting, Conference Proceedings, pp.12515-12541, Cincinnati, OH, United States (2005)	Não	CE1-18		X	X	
100	Improvisation in small software organizations	Dyba, T.	IEEE Software, pp.82-87, n. 5, SINTEF Telecom and Informatics, S.P. Andersens vei 15, N-7465 Trondheim, Norway	Não	CE1-13		X		
101	Integrated model for information systems migration: an evidence from a manufacturing company	Huang, Ming-Shang Lin, Hsin-Hui Lin, Fu-Ren	Guoli Bingdon Keji Daxue Xuebao/ Bulletin of National Pingtung University of Science and Technology, pp.143-160, n. 2	Não	CE1-18			X	
102	Integrated process and knowledge management for product definition, development and delivery	Koenig, S.	Software: Science, Technology and Engineering, 2003. SwSTE '03. Proceedings. IEEE International Conference on, vol. , pp.133-141 (2003)	Sim	CI1-08	X			
103	Integrating goal-oriented measurement in industrial software engineering: Industrial experiences with and additions to the Goal/Question/Metric method (GQM)	Van Solingen, R. Berghout, E.	International Software Metrics Symposium, Proceedings, pp.246-258, London (2001)	Não	CE1-18		X	X	
104	Integrating knowledge and processes in a learning organization	Levine, L.	IEEE Engineering Management Review, pp.33-44, n. 1	Não	CE1-19		X	X	
105	Integrating software process assessment models using a process meta model	Lepasaar, Marion Makinen, Timo	IEEE International Engineering Management Conference, pp.224-229, 1, Cambridge, United Kingdom (2002)	Sim	CI1-11	X	X	X	
106	Interaction modelling in federated process-centered environments	Piccinelli, Giacomo	HP Laboratories Technical Report, pp.1-15, n. 98-54	Não	CE1-12			X	
107	Key success factors for implementing software process improvement: A maturity-based analysis	Rainer, Austen Hall, Tracy	Journal of Systems and Software, pp.71-84, n. 2	Não	CE1-14		X	X	
108	Knowledge creation through a project review process in software production	Kess, P. Haapasalo, H.	Int J Prod Econ, pp.49-55, n. 1, Department of Industrial Engineering, University of Oulu, P.O. Box 4610, FIN-90014 University of Oulu, Finland	Sim	CI1-07		X		
109	Knowledge management in medium-sized software consulting companies	Dingsoyr, Torgeir	Empirical Software Engineering, pp.383-386, n. 4	Não	CE1-13		X	X	
110	Laboratory for real-time and embedded systems	Soklic, Milan E.	ASEE Annual Conference Proceedings, pp.1057-1065, Montreal, Que., Canada (2002)	Não	CE1-18		X	X	
111	Lessons learned in framework-based software process improvement	Jalote, P.	Software Engineering Conference, 2002. Ninth Asia-Pacific, vol. , pp.261-265 (2002)	Não	CE1-13	X			
112	Limited software warranties	Voas, J.M.	Engineering of Computer Based Systems, 2000. (ECBS 2000) Proceedings. Seventh IEEE International Conference and Workshop on the, vol. , pp.56-61 (2000)	Não	CE1-18	X			
113	Making sense of measurement for small organizations	Kautz, Karlheinz	IEEE Software, pp.14-20, n. 2	Não	CE1-13			X	
114	Making the software factory work: Lessons from a decade of experience	Sly, H.P. Herbsleb, J.D. Mockus, A. Krishnan, M. Tucker, G.T.	International Software Metrics Symposium, Proceedings, pp.317-326, London (2001)	Não	CE1-13	X	X	X	
115	Management of continuous software process improvement	Varkoi, Timo	IEEE International Engineering Management Conference, pp.334-337, 1, Cambridge, United Kingdom (2002)	Sim	CI1-07		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
116	Managing risk in software process improvement: AN action research approach	Iversen, Jakob H. Mathiassen, Lars Nielsen, Peter Axel	MIS Quarterly: Management Information Systems, pp.395-434, n. 3	Não	CE1-20		X	X	
117	Managing software process measurement: A metamodel-based approach	Garcia, F. Serrano, M. Cruz-Lemus, J. Ruiz, F. Piattini, M.	Information Sciences, pp.2570-2586, n. 12	Sim	CI1-11		X	X	
118	Measurement based continuous assessment of software engineering processes	Jarvinen, J.	VTT Publications, pp.93-, n. 426	Sim	CI1-07		X	X	
119	Measuring and improving software process in China	Wang, Qing Li, Mingshu	2005 International Symposium on Empirical Software Engineering, ISESE 2005, pp.183-192, Queensland, Australia (2005)	Sim	CI1-11	X	X	X	
120	Measuring the effectiveness of introducing new methods in the software development process	Winokur, M. Grinman, A. Yosha, I. Gallant, R.	Euromicro Conference, 1998. Proceedings. 24th, vol. 2, pp.800-807vol.2 (1998)	Sim	CI1-06	X			
121	Metamodeling and measurement for the software process improvement	Garcia, F. Ruiz, F. Piattini, M.	Computer Systems and Applications, 2003. Book of Abstracts. ACS/IEEE International Conference on, vol. , pp.43- (2003)	Não	CE1-19	X			
122	Method for software quality planning, control, and evaluation	Boegh, Jorgen Depanfilis, Stefano Kitchenham, Barbara Pasquini, Alberto	IEEE Software, pp.69-77, n. 2	Não	CE1-18		X	X	
123	Methodological issues in a CMM level 4 implementation	Antoniol, Giuliano Gradara, Sara Venturi, Gabriele	Software Process Improvement and Practice, pp.33-50, n. 1	Não	CE1-13		X	X	
124	Model merging based approach for modeling the CMM implementation process	Li, Juan Yuan, Feng Li, Ming-Shu Wang, Qing	Jisuanji Xuebao/Chinese Journal of Computers, pp.54-65, n. 1	Não	CE1-12		X	X	
125	Modeling a CMM implementation method with SPEM	Elizondo, Javier Sales, Rubens Becerra, Jorge	Proceedings of the International Conference on Software Engineering Research and Practice, SERP'04, pp.335-341, 1, Las Vegas, NV, United States (2004)	Não	CE1-19		X	X	
126	Modeling and simulating software acquisition process architectures	Choi, S.J. Scacchi, W.	Journal of Systems and Software, pp.343-354, n. 3	Não	CE1-12		X	X	
127	Modelling & supporting the ECSS software process	Gonzalez, Luis von Schoultz, Fredrick Stragapede, Alessandra	European Space Agency, (Special Publication) ESA SP, pp.95-101, n. 447, 447 (1999)	Não	CE1-16			X	
128	Modelling the likelihood of software process improvement: An exploratory study	El-Emam, K. Goldenson, D. Mccurley, J. Herbsleb, J.	Empirical Software Engineering, pp.207-229, n. 3	Não	CE1-14		X	X	
129	Motivators of Software Process Improvement: An analysis of practitioners' views	Baddoo, Nathan Hall, Tracy	Journal of Systems and Software, pp.85-96, n. 2	Não	CE1-14		X	X	
130	MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings	Montoni, Mariano Santos, Gleison Rocha, Ana Regina Weber, Kival C. Araujo, Eratostenes E.R. de	Software Quality, 2007. WoSQ'07: ICSE Workshops 2007. Fifth International Workshop on, vol. , pp.4-4 (2007)	Sim	CI1-06	X			
131	New model of improving quality management in china software company	Luo, X. Zhan, J. Mao, M.	J. Comput. Inf. Syst., pp.175-177, n. 1, Comp. Sci. Dept., Guangdong Polytech. Normal Univ., Guangzhou 510665, China	Não	CE1-14		X		
132	Open standard development platforms for distributed sensor networks	Merrill, William Sohrabi, Katayoun Girod, Lewis Elson, Jeremy Newberg, Fredric Kaiser, William	Proceedings of SPIE - The International Society for Optical Engineering, pp.327-337, 4743, Orlando, FL, United States (2002)	Não	CE1-18		X	X	
133	Operação do SEPG na Motorola Brasil	Lima Filho, J. M. S., Florencio, A. L. A., Costa, M. C. C., Brunetto, C.	I Simpósio Brasileiro de Qualidade de Software, SBQS'2002, pp.206-212 (2002)	Sim	CI1-09				X
134	OWPL: A Gradual Approach for Software Process Improvement In SMEs	Alexandre, Simon Renault, Alain Habra, Naji	Proceedings - 32nd Euromicro Conference on Software Engineering and Advanced Software Engineering and Advanced Applications, 2006. SEAA '06. 32nd EUROMICRO Conference on, vol. , pp.328-335 (2006)	Sim	CI1-06	X	X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
135	Packaging software process improvement issues: A method and a case study	Gorschek, Tony Wohlin, Claes	Software - Practice and Experience, pp.1311-1344, n. 14	Não	CE1-13		X	X	
136	Participative process introduction: Three case studies from the indiGo project	Decker, B. Rech, J. Althoff, K.-D. Klotz, A. Leopold, E. Voss, A.	J. Univers. Comput. Sci., pp.186-204, n. 3, Fraunhofer AIS	Não	CE1-12		X		
137	Personal quality management with the personal software process	Koch, A.S.	Software Engineering Education Conference, Proceedings, pp.43-, Charlotte, NC (2001)	Não	CE1-20	X	X	X	
138	Practical approach to development of SPI activities in a large organization Toshiba's SPI history since 2000	Ogasawara, Hideto Ishikawa, Takashi Moriya, Tetsuro	Proceedings - International Conference on Software Engineering, vol. 2006, pp.595-599, 2006, Shanghai, China (2006)	Sim	CI1-09		X	X	
139	Practical development of software configuration management for embedded systems	Taramaa, Jorma	Publications - Technical Research Centre of Finland, pp.147-, n. 366	Não	CE1-18			X	
140	Practical software process improvement - the IMPACT project	Scott, L. Jeffery, R. Carvalho, L. D'Ambra, J. Rutherford, P.	Software Engineering Conference, 2001. Proceedings. 2001 Australian, vol. , pp.182-189 (2001)	Sim	CI1-06	X			
141	Practice of CMM requirement management	Zhou, Zhi-Hao	Wuhan Ligong Daxue Xuebao/Journal of Wuhan University of Technology, pp.152-155	Não	CE1-13		X	X	
142	Prescription, description, reflection: The shape of the software process improvement field	Hansen, Bo Rose, Jeremy Tjornehoj, Gitte	International Journal of Information Management, pp.457-472, n. 6	Não	CE1-13		X	X	
143	Problems in measuring effectiveness in software process improvement: A longitudinal study of organizational change at Danske Data	Iversen, Jakob Ngwenyama, Ojelanki	International Journal of Information Management, pp.30-43, n. 1	Não	CE1-14		X	X	
144	Proceedings of the 1998 7th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE		Proceedings of the Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE, pp.385-, Stanford, CA, USA (1998)	Não	CE1-03			X	
145	Proceedings of the Thirty-First Hawaii International Conference on System Sciences		System Sciences, 1998., Proceedings of the Thirty-First Hawaii International Conference on, vol. 2, pp.- (1998)	Não	CE1-03	X			
146	Process change management in switching	Ebert, C.	Alcatel Telecommunications Review, pp.6-8, n. 1st Quarter	Sim	CI1-09		X	X	
147	Process improvement competencies for IS professionals: a survey of perceived needs	McGuire, Eugene G. Randall, Kim A.	Proceedings of the ACM SIGCPR Conference, pp.1-8, Boston, MA, USA (1998)	Não	CE1-08			X	
148	Process improvement for small firms: An evaluation of the RAPID assessment-based method	Cater-Steel, A. Toleman, M. Rout, T.	Inf Software Technol, pp.323-334, n. 5, Software Quality Institute, Griffith University, Brisbane, QLD, Australia	Não	CE1-13		X		
149	Process improvement priorities in small software companies	Varkoi, T. Makinen, T. Jaakkola, H.	Management of Engineering and Technology, 1999. Technology and Innovation Management. PICMET '99. Portland International Conference on, vol. 1, pp.555vol.1- (1999)	Não	CE1-13	X			
150	Process-based software engineering: Building the infrastructures	Wang, Y. Bryant, A.	Ann. Softw. Eng., pp.9-37, n. 1-4, School of Information Management, Leeds Metropolitan University, Beckett Park, Leeds LS6 3QS, United Kingdom	Não	CE1-15		X		
151	Product Focused Software Process Improvement: 6th International Conference, PROFES 2005. Proceedings		Lecture Notes in Computer Science, pp.596-, 3547, Oulu, Finland (2005)	Não	CE1-03		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
152	Product-Focused Software Process Improvement - 7th International Conference, PROFES 2006, Proceedings		Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.485-, 4034 NCS, Amsterdam, Netherlands (2006)	Não	CE1-03		X	X	
153	Project assessments: supporting commitment, participation, and learning in software process improvement	Arent, Jesper Iversen, Jakob H. Andersen, Carsten V. Bang, Stig	Proceedings of the Hawaii International Conference on System Sciences, pp.106-, Maui, USA (2000)	Não	CE1-13	X	X	X	
154	Project management and executing integration technology	Cao, Jian Zhang, Shen-Sheng Mou, Yu-Jie Zhang, Jian-Hong	Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS, pp.37-43	Não	CE1-12			X	
155	Project web and electronic process guide as software process improvement	Moe, Nils Brede Dingsoyr, Torgeir Nilsen, Ken Rune Villmones, Nils Jakob	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.175-186, 3792 NCS, Budapest, Hungary (2005)	Sim	CI1-11		X	X	
156	Quality management in systems development: An organizational system perspective	Ravichandran, T. Rai, Arun	MIS Quarterly: Management Information Systems, pp.381-416, n. 3	Não	CE1-13		X	X	
157	Quality through Managed Improvement and Measurement (QMIM): Towards a Phased Development and Implementation of a Quality Management System for a Software Company	Balla, K. Bemelmans, T. Kusters, R. Trienekens, J.	Softw. Qual. J., pp.177-193, n. 3, Technical University Budapest, Hungary	Não	CE1-13		X		
158	RAMALA: A knowledge base for software process improvement	Garcia, Javier Rimawi, Yaser Sanchez, Maria Isabel Amescua, Antonio	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3792 NCS, pp.106-117, 3792 NCS, Budapest, Hungary (2005)	Sim	CI1-06		X	X	
159	Reference model for software process improvement: A Brazilian experience	Rocha, Ana Regina Montoni, Mariano Santos, Gleison Mafra, Somulo Figueiredo, Savio Albuquerque, Adriano Mian, Paula	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3792 NCS, pp.130-141, 3792 NCS, Budapest, Hungary (2005)	Sim	CI1-06		X	X	
160	Requirement process establishment and improvement: From the viewpoint of cybernetics	Hong, Xu Sawyer, Pete Sommerville, Ian	Proceedings - International Computer Software and Applications Conference, pp.89-92, 2, Edinburgh, Scotland, United Kingdom (2005)	Não	CE1-13	X	X	X	
161	Requirements for a knowledge management framework to be used in software intensive organizations	Martinez, Paloma Amescua, Antonio Garcia, Javier Cuadra, Dolores Llorens, Juan Fuentes, J.M. Martin, Diego Cuevas, Gonzalo Calvo-Manzano, Jose Antonio Feliu, Tomas San	Proceedings of the 2005 IEEE International Conference on Information Reuse and Integration, IRI - 2005, pp.554-559, 2005, Las Vegas, NV, United States (2005)	Sim	CI1-04		X	X	
162	Requirements for a software process improvement support and learning environment	Varkoi, Timo Makinen, Timo Jaakkola, Hannu	PICMET, pp.433-434, Portland, OR, United States (2002)	Não	CE1-14	X			
163	Research into long-term improvements in small-to medium-sized organisations using SPICE as a framework for standards	Sanders, Marty Richardson, Ita	Software Process Improvement and Practice, pp.351-359	Sim	CI1-11		X	X	
164	Research on CMMI-based software process metrics	Xu, Ruzhi Xue, Yunjiao Nie, Peiyao Zhang, Yuan Li, Desheng	First International Multi- Symposiums on Computer and Computational Sciences, IMSCCS'06, pp.391-397, 2, Hangzhou, Zhejiang, China (2006)	Sim	CI1-11		X	X	
165	Research on measurement models and fundamental metrics in project management	Zhang, Hong-Yan Xiang, Hui-Ru	Beijing Gongye Daxue Xuebao / Journal of Beijing University of Technology, pp.1-6, n. SUPPL	Não	CE1-13			X	
166	Research on the phenomenon of software drift in software processes	Ma, Yutao Chen, Jianxun Wu, Jianghua	International Workshop on Principles of Software Evolution (IWPSE)Principles of Software Evolution, Eighth International Workshop on, vol. , pp.195-198 (2005)	Não	CE1-13		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
167	Role of software process improvement into total quality management: An industrial experience	Della Volpe, Renato L. Nobre, Farley S.M. Pessoa, Marcelo S.P. Spinola, Mauro	IEEE International Engineering Management Conference, pp.29-34, Albuquerque, NM, USA (2000)	Não	CE1-13		X	X	
168	Secure access of products in a process environment	Chou, S.-C. Lai, C.-W.	IEICE Trans Inf Syst, pp.197-203, n. 2, SerComm Corporation, Taipei, Taiwan	Não	CE1-12		X		
169	Selection priority of process areas based on CMMI continuous representation	Huang, Sun-Jen Han, Wen-Ming	Information and Management, pp.297-307, n. 3	Não	CE1-17		X	X	
170	Software defect report and tracking system in an Intranet	Monteiro, A. Almeida, A.B. Goulao, M. Abreu, F.B. Sousa, P.	Proceedings of the Euromicro Conference on Software Maintenance and Reengineering, CSMR, pp.198-201, Amsterdam, Neth (1999)	Sim	CI1-08		X	X	
171	Software engineering in the small: Practical software engineering and management	Moitra, Deependra	Computer, pp.39-40, n. 10	Não	CE1-20		X	X	
172	Software maintenance process analysis using discrete-event simulation	Podnar, I. Mikac, B.	Software Maintenance and Reengineering, 2001. Fifth European Conference on, vol. , pp.192-195 (2001)	Não	CE1-18		X		
173	Software process commonality analysis	Ocampo, Alexis Bella, Fabio Munch, Jurgen	Software Process Improvement and Practice, pp.273-285, n. 3	Sim	CI1-12		X	X	
174	Software Process Improvement - 12th European Conference, EuroSPI 2005, Proceedings		Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.220-, 3792 NCS, Budapest, Hungary (2005)	Não	CE1-03		X	X	
175	Software Process Improvement - 13th European Conference, EuroSPI 2006, Proceedings		Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.225-, 4257 NCS, Joensuu, Finland (2006)	Não	CE1-03		X	X	
176	Software process improvement as emergent change: A structural analysis	Allison, I. Merali, Y.	Information and Software Technology, pp.668-681, n. 6	Não	CE1-13		X	X	
177	Software process improvement at ABB	Larsson, S.B.M. Kolb, P.	ABB Review, pp.10-14, n. 3	Não	CE1-18		X	X	
178	Software process improvement by example (SPIE)	Brebner, P.C.	Software Engineering: Education and Practice, 1998. Proceedings. 1998 International Conference, vol. , pp.88-95 (1998)	Não	CE1-19	X			
179	Software process improvement in Europe: Potential of the new V-Modell XT and research issues	Biffi, Stefan Winkler, Dietmar Hohn, Reinhard Wetzl, Herbert	Software Process Improvement and Practice, pp.229-238, n. 3	Não	CE1-13		X	X	
180	Software process improvement motivators: An analysis using multidimensional scaling	Baddoo, Nathan Hall, Tracy	Empirical Software Engineering, pp.93-114, n. 2	Não	CE1-14		X	X	
181	Software process improvement problems in twelve software companies: An empirical analysis	Beecham, Sarah Hall, Tracy Rainer, Austen	Empirical Software Engineering, pp.7-42, n. 1	Sim	CI1-12		X	X	
182	Software process improvement strategy for enterprise information systems development	Fung, Richard Y.K. Tam, W.T. Ip, Andrew W.H. Lau, Henry C.W.	International Journal of Information Technology and Management, pp.225-241, n. 2-3	Sim	CI1-07		X	X	
183	Software Process Improvement Support System: SPIS	Fukuyama, Shun-ichi Miyamura, Shuu-ichi Takagi, Hideo Tanaka, Ryoji	IEICE Transactions on Information and Systems, vol. E83-D, pp.747-756, n. 4	Sim	CI1-07		X	X	
184	Software process improvement: Blueprints versus recipes	Aaen, Ivan	IEEE Software, pp.86-93, n. 5	Sim	CI1-12		X	X	
185	Software process management	Bhattacharyya, Arundhati	Defence Science Journal, pp.403-414, n. 4	Não	CE1-17		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
186	Software process management: Practices in China	Wang, Qing Li, Mingshu	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.317-331, 3840 NCS, Beijing, China (2006)	Sim	CI1-06			X	
187	Software process: Applying industrial-strength methods in engineering education	Sebern, M.J.	2005 ASEE Annual Conference and Exposition: The Changing Landscape of Engineering and Technology Education in a Global World, vol. , pp.12829-12846, Portland, OR (2005)	Não	CE1-04		X		
188	Software Processes Improvement and Specifications for Embedded Systems	Jun, Dong Rui, Lin Yi-min, He	Software Engineering Research, Management & Applications, 2007. SERA 2007. 5th ACIS International Conference on, vol. , pp.13-18 (2007)	Sim	CI1-03	X			
189	Software quality management and software process model	Mao, Mingzhi Luo, Xiaonan	Journal of Information and Computation Science, pp.203-207, n. 3	Não	CE1-13		X	X	
190	SPI agility: How to navigate improvement projects	Aaen, Ivan Borjesson, Anna Mathiassen, Lars	Software Process Improvement and Practice, pp.267-281, n. 3	Não	CE1-14		X	X	
191	SPI methodology for virtual organizations	Martins, Paula Ventura Da Silva, Alberto Rodrigues	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.824-833, 4277 NCS - I, Montpellier, France (2006)	Não	CE1-15		X	X	
192	Standard based software process assessments in small companies	Gresse Von Wangenheim, Christiane Varkoi, Timo Salviano, Clenio F.	Software Process Improvement and Practice, pp.329-335, n. 3	Sim	CI1-06		X	X	
193	State of the art and future of research in software process improvement	Serrano, M.A.	Proceedings of the 28th Annual International Computer Software and Applications Conference, COMPSAC 2004, vol. 1, pp.239-, 1, Hong Kong, China (2004)	Não	CE1-20		X		
194	Strategies for personal process improvement a comparison	O'Connor, Rory Coleman, Gerry	Proceedings of the ACM Symposium on Applied Computing, pp.1036-1040, Madrid, Spain (2002)	Não	CE1-14		X	X	
195	Study on workflow simulation for software process improvement	Lu, Chun-Xia Li, Wen-Li Li, Xu	Xitong Fangzhen Xuebao / Journal of System Simulation, pp.2501-2506, n. 11	Sim	CI1-08		X	X	
196	Successful process implementation	Borjesson, A. Mathiassen, L.	Software, IEEE, pp.36-44, n. 4	Não	CE1-13	X			
197	Suitability of requirements prioritization methods for market-driven software product development	Lehtola, Laura Kauppinen, Marjo	Software Process Improvement and Practice, pp.7-19, n. 1	Não	CE1-18		X	X	
198	Supporting software development in virtual enterprises	Noll, J. Scacchi, W.	J. Digit. Inf., pp.-, n. 4, ATRIUM Laboratory, University of Southern California	Não	CE1-12		X		
199	Supporting virtual software projects on the Web	Alho, Kari Sulonen, Reijo	Proceedings of the Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, WET ICE, pp.10-14, Stanford, CA, USA (1998)	Não	CE1-12			X	
200	Taba workstation: Supporting software process deployment based on CMMI and MR-MPS.BR	Montoni, Mariano Santos, Gleison Rocha, Ana Regina Figueiredo, Savio Cabrai, Reinaldo Barcellos, Rafael Barreto, Ahilton Scares, Andrea Cerdeiral, Cristina Lupo, Peter	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 4034 NCS, pp.249-262, 4034 NCS, Amsterdam, Netherlands (2006)	Sim	CI1-06		X	X	
201	Taba workstation: Supporting software process improvement initiatives based on software standards and maturity models	Ferreira, Analia Irigoyen Ferreiro Santos, Gleison Cerqueira, Roberta Montoni, Mariano Barreto, Ahilton Rocha, Ana Regina Figueiredo, Savio Barreto, Andrea Filho, Reinaldo C. Silva Lupo, Peter Cerdeiral, Cristina	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 4257 NCS, pp.207-218, 4257 NCS, Joensuu, Finland (2006)	Sim	CI1-07		X	X	

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
202	Tackling the complexity of requirements engineering process improvement by partitioning the improvement task	Nikula, Uolevi Sajaniemi, Jorma	Proceedings of the Australian Software Engineering Conference, ASWEC, pp.48-57, 2005, Brisbane, Australia (2005)	Sim	CI1-07	X	X	X	
203	Technical controlling and software process improvement	Ebert, C.	J Syst Software, pp.25-39, n. 1, Alcatel Switching Systems Division, Fr.-Wellesplein 1, B-2018, Antwerpen, Belgium	Sim	CI1-09		X		
204	The AV-8B team learns synergy of EVM and TSP accelerates software process improvement	Pracchia, L.	CrossTalk, pp.20-22, n. 1, NAWCWD 41K300D (L. Pracchia), Bldg. 1494, I Administration Circle, China Lake, CA 93555	Sim	CI1-11		X		
205	The impact of institutional forces on software metrics programs	Gopal, Anandasivam Mukhopadhyay, Tridas Krishnan, M.S.	IEEE Transactions on Software Engineering, pp.679-694, n. 8	Não	CE1-18	X	X	X	
206	The Knowledge Based Software Process Improvement Program: A Rational Analysis	K, Alagarsamy S, Justus K, Iyakutti	Software Engineering Advances, 2007. ICSEA 2007. International Conference on, vol. , pp.61-61 (2007)	Sim	CI1-11	X			
207	The long-term effects of an EPG/ER in a small software organisation	Kurniawati, Felicia Jeffery, Ross	Proceedings of the Australian Software Engineering Conference, ASWEC, pp.128-136, 2004, Melbourne, Vic., Australia (2004)	Sim	CI1-08	X	X	X	
208	The management of software processes with software process improvement tool based on ISO 15504	Choi, Yoonjung Lee, EunSeok Ha, Sujung	The 7th International Conference on Advanced Communication Technology, ICACT 2005, vol. 2, pp.933-936, 2, Phoenix Park, South Korea (2005)	Sim	CI1-08		X	X	
209	The role of awareness support in collaborative improvement of software processes	de Araujo, R.M. da Silva Borges, M.R.	String Processing and Information Retrieval Symposium, 1999 and International Workshop on Groupware, vol. , pp.343-347 (1999)	Sim	CI1-11	X			
210	The role of knowledge management supporters in software development companies	Feher, Peter Gabor, Andras	Software Process Improvement and Practice, pp.251-260, n. 3	Não	CE1-18		X	X	
211	The role of software process improvement into total quality management: an industrial experience	Della Volpe, R.L. Nobre, F.S.M. Pessoa, M.S.P. Spinola, M.	Engineering Management Society, 2000. Proceedings of the 2000 IEEE, vol. , pp.29-34 (2000)	Não	CE1-13	X			
212	The use and effects of an electronic process guide and experience repository: a longitudinal study	Kurniawati, Felicia Jeffery, Ross	Information and Software Technology, pp.566-577, n. 7	Sim	CI1-06		X	X	
213	The use of an electronic process guide in a medium-sized software development company	Moe, Nils Brede Dyba, Tore	Software Process Improvement and Practice, pp.21-34, n. 1	Não	CE1-15		X	X	
214	Tolerating deviations in process support systems via flexible enactment of process models	Cugola, Gianpaolo	IEEE Transactions on Software Engineering, pp.982-1001	Não	CE1-18	X		X	
215	Toward computational support for software process improvement activities	Sakamoto, Keishi Nakakoji, Kumiyo Takagi, Yasunari Niihara, Naoki	Proceedings - International Conference on Software Engineering, pp.22-31, Kyoto, Jpn (1998)	Sim	CI1-09	X		X	
216	Towards a metrics based verification and validation maturity model	Jacobs, J.C. Trienekens, J.J.M.	Software Technology and Engineering Practice, 2002. STEP 2002. Proceedings. 10th International Workshop on, vol. , pp.123-128 (2002)	Não	CE1-13	X			
217	Towards virtual software configuration management. A case study	Rahikkala, T.	VTT Publ., pp.-, n. 409, VTT Electronics	Não	CE1-12		X		
218	Training experts in the fundamentals: an experience in providing software engineers with the basics of software process improvement	Ovalle, N.K. Egdorf, H.W.	Software Engineering Education and Training, 2003. (CSEE&T 2003). Proceedings. 16th Conference on, vol. , pp.339- (2003)	Não	CE1-13	X			
219	Uma Abordagem de Garantia da Qualidade de Processos e Produtos de Software com Apoio de Gerência de Conhecimento na Estação Taba	Montoni, M., Santos, G., Figueiredo, S., Silva Filho, R. C., Barcelos, R., Barreto, A., Barreto, A., Cerdeiral, C., Lupo, P., Rocha, A. R.	V Simpósio Brasileiro de Qualidade de Software, SBQS'2006, pp.87-99 (2005)	Não	CE1-16				X

#	Título	Autores	Dados	1º Filtro	Critério	Local Busca			
						IEEE	Scopus	Compendex	Manual
220	Uma Abordagem para Melhoria Contínua do Processo de Desenvolvimento de Software	Malheiros, V., Paim, F. R., Guzzo, H., Mendonça Neto, M. G.	V Simpósio Brasileiro de Qualidade de Software, SBQS'2006, pp.218-230	Sim	CI1-09				X
221	Understanding the implementation of software process improvement innovations in software organizations	Kautz, K. Nielsen, P.A.	Inf. Syst. J., pp.3-22, n. 1, Department of Computer Science, Aalborg University, Frederik Bajiers Vej 7e, DK-9220 Aalborg, Denmark	Não	CE1-13		X		
222	Understanding the use of an electronic process guide	Scott, Louise Carvalho, Lucila Jeffery, Ross D'Ambra, John Becker-Kornstaedt, Ulrike	Information and Software Technology, pp.601-616, n. 10	Sim	CI1-11		X	X	
223	Unifying the Software Process Spectrum - International Software Process Workshop, SPW 2005, Revised Selected Papers		Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.534-, 3840 NCS, Beijing, China (2006)	Não	CE1-03		X	X	
224	Using academic courses for empirical validation of software development processes	Ciolkowski, Marcus Muthig, Dirk Rech, Jorg	Conference Proceedings of the EUROMICRO, pp.354-361, 30, Rennes, France (2004)	Não	CE1-18	X	X	X	
225	Using grounded theory to understand software process improvement: A study of Irish software product companies	Coleman, Gerry O'Connor, Rory	Information and Software Technology, pp.654-667, n. 6	Sim	CI1-12		X	X	
226	Using knowledge management to improve software process performance in a CMM level 3 organization	De Almeida Falbo, Ricardo Borges, Ligia S. Mota Valente, Fabio Feu Rosa	Proceedings - Fourth International Conference on Quality Software, QSIC 2004, pp.162-169, Braunschweig, Germany (2004)	Sim	CI1-07	X	X	X	
227	Using the Internet to communicate software metrics in a large organization	Jokikyyny, Tony	Conference Record / IEEE Global Telecommunications Conference, vol. 3, pp.1947-1953, 3, Rio de Janeiro, Braz (1999)	Sim	CI1-08	X	X	X	
228	Valmont: A language for workflow programming	Chan, Daniel K.C. Leung, Karl R.P.H.	Proceedings of the Hawaii International Conference on System Sciences, pp.744-753, 7, Big Island, HI, USA (1998)	Não	CE1-12	X		X	
229	Version Web: A tool for open source software development support	Junqueira, Daniel Carnio De Fortes, Renata Pontin M.	Proceedings - WebMedia and LA-Web 2004, pp.65-67, Ribeirao Preto-SP, Brazil (2004)	Não	CE1-12		X	X	
230	Virtual software engineering laboratories in support of trade-off analyses	Munch, J. Pfahl, D. Rus, I.	Softw. Qual. J., pp.407-428, n. 4, IEEE Computer Society	Não	CE1-18		X		
231	What elements of XP are being adopted by industry practitioners?	Livermore, J.A.	IEEE SoutheastCon 2006, vol. 2006, pp.149-152, 2006, Memphis, TN (2006)	Não	CE1-14		X		
232	What to expect from software experience exploitation	Schneider, K.	J. Univers. Comput. Sci., pp.570-580, n. 6, DaimlerChrysler AG, Research Center Ulm, P.O. Box 2360, 89013 Ulm, Germany	Não	CE1-13		X		
233	Worldwide survey of base process activities towards software engineering process excellence	Wang, Yingxu King, Graham Dorling, Alec Patel, Dilip Court, Ian Staples, Geoff Ross, Margaret	Proceedings - International Conference on Software Engineering, pp.439-442, Kyoto, Jpn (1998)	Não	CE1-14	X		X	

5.3 Dados das Publicações Seleccionadas no Primeiro Filtro

A Tabela 3 apresenta as publicações seleccionadas após aplicação do primeiro filtro de selecção. Para cada publicação são apresentados título, autores e dados gerais da publicação. Além disso, é indicado se a publicação estava disponível para análise, uma revisão do critério do primeiro filtro após a leitura do artigo, se a publicação passou no segundo filtro e o critério (conforme lista apresentada na secção 2.6) utilizado para a inclusão ou exclusão durante esta segunda análise.

Tabela 3 – Artigos Lidos Após Aplicação do Primeiro Filtro

#	Título	Autores	Dados	Disponibilidade	Critério 1º Filtro (Rev.)	2º Filtro	Critério 2º Filtro
1	A Continuous Improvement Model in ImPProS	Bezerra Oliveira, S.R. Lins de Vasconcel, A.M.	Computer Software and Applications Conference, 2006. COMPSAC '06. 30th Annual International, vol. 2, pp.370-371 (2006)	X	CE1-19	Não	CS2 +SPI +Fer -Uso
2	A Dynamic Integrated Framework for Software Process Improvement	Ruiz, M. Ramos, I. Toro, M.	Softw. Qual. J., pp.181-194, n. 2, Department of Computer Languages, Esc. Tec. Sup. de Ing. Info., University of Seville, Spain	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
3	A fast method for analysing and improving complex software processes	Hobday, M. Brady, T.	R D Manage., pp.1-20, n. 1, Ctr. for Res. in Innov. Management, University of Brighton, United Kingdom	X	CE1-18	Não	CS2 -SPI -Fer -Uso
4	A framework for coping with process evolution	Nejmeh, Brian A. Riddle, William E.	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.302-316, 3840 NCS, Beijing, China (2006)	-		Não	
5	A framework for evaluation and prediction of software process improvement success	Wilson, D.N. Hall, T. Baddoo, N.	Journal of Systems and Software, pp.135-142, n. 2	X	CE1-14	Não	CS2 -SPI -Fer -Uso
6	A low-overhead method for software process appraisal	Wilkie, F.G. Mc Caffery, F. McFall, D. Lester, Neil Wilkinson, Emmanuel	Software Process Improvement and Practice, pp.339-349	-		Não	
7	A methodology and its support environment for benchmark-based adaptable software process improvement	Wu, Minghui Ying, Jing Yu, Chunyan	Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, pp.5183-5188, 6, The Hague, Netherlands (2004)	X	CE1-19	Não	CS2 +SPI +Fer -Uso
8	A process improvement framework and a supporting software oriented to chinese small organizations	Gong, Bo He, Xingui Liu, Weihong	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.277-286, 3840 NCS, Beijing, China (2006)	X	CI1-06	Sim	CI2 +SPI +Fer +Uso
9	A process improvement framework for XP based	Ramachandran, Muthu	Lecture Notes in Computer Science, vol. 3556, pp.202-205, 3556,	X	CE1-	Não	CS2 +SPI +Fer -Uso

#	Título	Autores	Dados	Disponibilidade	Critério 1º Filtro (Rev.)	2º Filtro	Critério 2º Filtro
	SMEs		Sheffield, United Kingdom (2005)		12		
10	A process management tool supporting component-based process development and hierarchical management mechanism	Wang, Yasha Li, Dongni He, Xiaoyang	Proceedings - Fifth International Conference on Computer and Information Technology, CIT 2005, pp.906-910, 2005, Shanghai, China (2005)	X		Não	
11	A software process tailoring system focusing to quantitative management plans	Hikichi, Kazumasa Fushida, Kyohei Iida, Hajimu Matsumoto, Ken'ichi	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.441-446, 4034 NCS, Amsterdam, Netherlands (2006)	X	CE1-12	Não	CS2 -SPI +Fer -Uso
12	A tentative framework for managing software product development in small companies	Rautiainen, K. Lassenius, C. Vahaniitty, J. Pyhajarvi, M. Vanhanen, J.	System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on, vol. , pp.3409-3417 (2002)	X	CE1-13	Não	CS2 +SPI -Fer +Uso
13	A tool to improve the software process quality in a R and D center using PSP	Etxanziz, Inaki	WSEAS Transactions on Information Science and Applications, pp.763-770, n. 4	-		Não	
14	An autonomic framework for quantitative software process improvement	Tianfield, H.	Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on, vol. , pp.446-450 (2003)	X	CE1-12	Não	CS2 -SPI +Fer -Uso
15	Applying a framework for the improvement of software process maturity	Canfora, Gerardo Garcia, Felix Piattini, Mario Ruiz, Francisco Visaggio, C.A.	Software - Practice and Experience, vol. 36, pp.283-304, n. 3	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
16	Applying and adjusting a software process improvement model in practice: the use of the IDEAL model in a small software enterprise	Kautz, Karlheinz Hansen, Henrik Westergaard Thaysen, Kim	Proceedings - International Conference on Software Engineering, pp.626-633, Limerick, Ireland (2000)	X	CE1-13	Não	CS2 +SPI -Fer +Uso
17	Approaching software process improvement to organizations	Amescua, Antonio Garcia, Javier Sanchez-Segura, Maria-Isabel Medina-Dominguez, Fuensanta	WSEAS Transactions on Computers, pp.507-514, n. 3	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
18	Automated support of software quality improvement	Leung, H.K.N. Li, L. Yuzhong, Q.	Int. J. Qual. Reliab. Manage., pp.230-243, n. 3, Department of Computer Science and Engineering, Southeast University, Nanjing, China	X	CE1-15	Não	CS2 -SPI -Fer -Uso
19	Best practice approaches in know-how and technology transfer methods for manufacturing SMEs	Enzenhofer, W. Chroust, G.	Euromicro Conference, 2001. Proceedings. 27th, vol. , pp.279-286 (2001)	X	CE1-13	Não	CS2 +SPI +Fer -Uso
20	Case study in innovative process improvement: Code synthesis from formal specifications	Garbett, P. Parkes, J.P. Shackleton, M. Anderson, S.	Microprocessors and Microsystems, pp.417-424, n. 7	-		Não	
21	CASE STUDY: A practical approach for SPI in large Spanish companies	Guzman, Javier Garcia Garcia, Roman Lopez-Cortijo De Seco, Antonio Amescua Agustin, Gonzalo Cuevas	Software Process Improvement and Practice, pp.261-268, n. 3	-	CE1-13	Não	CS2 +SPI -Fer +Uso
22	Confluence of six sigma, simulation and software development	Mahanti, R. Antony, J.	Manage. Audit. J., pp.739-762, n. 7, Division of Management, Caledonian Business School, Glasgow Caledonian University, Glasgow, United Kingdom	X	CE1-18	Não	CS2 -SPI -Fer -Uso
23	Diagnóstico, Definição e Melhoria do Processo de Software: um Estudo de Caso	Tavares, D. P. D., Fabbri, S. C. P. F., Sanches, R.	I Simpósio Brasileiro de Qualidade de Software, SBQS'2002, pp.189-197	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
24	DuoTracker: Tool Support for Software Defect Data Collection and Analysis	Akinwale, O. Dascalu, S. Karam, M.	Software Engineering Advances, International Conference on, vol. , pp.22-22 (2006)	X	CE1-12	Não	CS2 -SPI +Fer -Uso

#	Título	Autores	Dados	Disponibilidade	Critério 1º Filtro (Rev.)	2º Filtro	Critério 2º Filtro
25	Effective elements of integrated software development process supported platform	Fang, Min Ying, Jing Wu, Minghui	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3168 NCS, pp.368-377, 3168 NCS, Xiamen, China (2005)	-		Não	
26	Enterprise-oriented software development environments to support software products and processes quality improvement	Montoni, Mariano Santos, Gleison Villela, Karina Rocha, Ana Regina Travassos, Guilherme H. Figueiredo, Savio Mafra, Somulo Albuquerque, Adriano Mian, Paula	Lecture Notes in Computer Science, vol. 3547, pp.370-384, 3547, Oulu, Finland (2005)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
27	e-R&D - Effectively managing process diversity	Ebert, C. De Man, J.	Ann. Softw. Eng., pp.73-91, n. 1-4, Alcatel, 54 rue La Boetie, 75008 Paris, France	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
28	Experiences from the pilot operation and commissioning phase of a SCM process improvement program	Nattinen, M. Rahikkala, T. Valimaki, A.	EUROMICRO Conference, 1999. Proceedings. 25th, vol. 2, pp.185-192vol.2 (1999)	X	CE1-14	Não	CS2 +SPI +Fer -Uso
29	FMESP: Framework for the modeling and evaluation of software processes	Garcia, Felix Piattini, Mario Ruiz, Francisco Canfora, Gerardo Visaggio, Corrado A.	Journal of Systems Architecture, vol. 52, pp.627-639	X	CE1-12	Não	CS2 +SPI +Fer -Uso
30	Frameworks of Integration Repository for Software Process Improvement using SOA	Park, Eun-Ju Kim, Haeng-Kon Lee, Roger Y.	Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on, vol. , pp.200-206 (2007)	X	CE1-12	Não	CS2 -SPI +Fer -Uso
31	Guided support for collaborative modeling, enactment and simulation of software development processes	Fernandez, Alejandro Garzaldeen, Badie Grutzner, Ines Munch, Jurgen	Software Process Improvement and Practice, pp.95-106, n. 2	X	CE1-12	Não	CS2 -SPI +Fer +Uso
32	How software process automation affects software evolution: A longitudinal empirical analysis	Barry, Evelyn J. Kemerer, Chris F. Slaughter, Sandra A.	Journal of Software Maintenance and Evolution, pp.1-31, n. 1	X	CE1-18	Não	CS2 -SPI -Fer -Uso
33	Implantando CMM Nível 2: A Estratégia SERPRO	Tavares, H. C., Paim, F. R. S., Carvalho, A. E.	I Simpósio Brasileiro de Qualidade de Software, SBQS'2002, pp.181-188	X	CI1-09	Sim	CI2 +SPI +Fer +Uso
34	Implementing software process improvement: two cases of technology transfer	Kautz, Karlheinz Nielsen, Peter Axel	Proceedings of the Hawaii International Conference on System Sciences, pp.188-, Maui, USA (2000)	X	CE1-18	Não	CS2 -SPI -Fer +Uso
35	Integrated process and knowledge management for product definition, development and delivery	Koenig, S.	Software: Science, Technology and Engineering, 2003. SwSTE '03. Proceedings. IEEE International Conference on, vol. , pp.133-141 (2003)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
36	Integrating software process assessment models using a process meta model	Lepasaar, Marion Makinen, Timo	IEEE International Engineering Management Conference, pp.224-229, 1, Cambridge, United Kingdom (2002)	X	CE1-04	Não	CS2 -SPI -Fer -Uso
37	Knowledge creation through a project review process in software production	Kess, P. Haapasalo, H.	Int J Prod Econ, pp.49-55, n. 1, Department of Industrial Engineering, University of Oulu, P.O. Box 4610, FIN-90014 University of Oulu, Finland	X	CE1-13	Não	CS2 +SPI -Fer +Uso
38	Management of continuous software process improvement	Varkoi, Timo	IEEE International Engineering Management Conference, pp.334-337, 1, Cambridge, United Kingdom (2002)	X	CE1-13	Não	CS2 +SPI -Fer +Uso
39	Managing software process measurement: A metamodel-based approach	Garcia, F. Serrano, M. Cruz-Lemus, J. Ruiz, F. Piattini, M.	Information Sciences, pp.2570-2586, n. 12	X	CE1-12	Não	CS2 -SPI +Fer -Uso

#	Título	Autores	Dados	Disponibilidade	Critério 1º Filtro (Rev.)	2º Filtro	Critério 2º Filtro
40	Measurement based continuous assessment of software engineering processes	Jarvinen, J.	VTT Publications, pp.93-, n. 426	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
41	Measuring and improving software process in China	Wang, Qing Li, Mingshu	2005 International Symposium on Empirical Software Engineering, ISESE 2005, pp.183-192, Queensland, Australia (2005)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
42	Measuring the effectiveness of introducing new methods in the software development process	Winokur, M. Grinman, A. Yosha, I. Gallant, R.	Euromicro Conference, 1998. Proceedings. 24th, vol. 2, pp.800-807vol.2 (1998)	X	CE1-07	Não	CS2 -SPI +Fer +Uso
43	MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings	Montoni, Mariano Santos, Gleison Rocha, Ana Regina Weber, Kival C. Araujo, Eratostenes E.R. de	Software Quality, 2007. WoSQ'07: ICSE Workshops 2007. Fifth International Workshop on, vol. , pp.4-4 (2007)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
44	Operação do SEPG na Motorola Brasil	Lima Filho, J. M. S., Florencio, A. L. A., Costa, M. C. C., Brunetto, C.	I Simpósio Brasileiro de Qualidade de Software, SBQS'2002, pp.206-212 (2002)	X	CE1-13	Não	CS2 +SPI -Fer +Uso
45	OWPL: A Gradual Approach for Software Process Improvement In SMEs	Alexandre, Simon Renault, Alain Habra, Naji	Proceedings - 32nd Euromicro Conference on Software Engineering and Advanced Software Engineering and Advanced Applications, 2006. SEAA '06. 32nd EUROMICRO Conference on, vol. , pp.328-335 (2006)	X	CE1-13	Não	CS2 -SPI -Fer -Uso
46	Practical approach to development of SPI activities in a large organization Toshiba's SPI history since 2000	Ogasawara, Hideto Ishikawa, Takashi Moriya, Tetsuro	Proceedings - International Conference on Software Engineering, vol. 2006, pp.595-599, 2006, Shanghai, China (2006)	X	CI1-09	Sim	CI2 +SPI +Fer +Uso
47	Practical software process improvement - the IMPACT project	Scott, L. Jeffery, R. Carvalho, L. D'Ambra, J. Rutherford, P.	Software Engineering Conference, 2001. Proceedings. 2001 Australian, vol. , pp.182-189 (2001)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
48	Process change management in switching	Ebert, C.	Alcatel Telecommunications Review, pp.6-8, n. 1st Quarter	-		Não	
49	Project web and electronic process guide as software process improvement	Moe, Nils Brede Dingsoyr, Torgeir Nilsen, Ken Rune Villmones, Nils Jakob	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.175-186, 3792 NCS, Budapest, Hungary (2005)	X	CI1-12	Sim	CI2 +SPI +Fer +Uso
50	RAMALA: A knowledge base for software process improvement	Garcia, Javier Rimawi, Yaser Sanchez, Maria Isabel Amescua, Antonio	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3792 NCS, pp.106-117, 3792 NCS, Budapest, Hungary (2005)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
51	Reference model for software process improvement: A Brazilian experience	Rocha, Ana Regina Montoni, Mariano Santos, Gleison Mafra, Somulo Figueiredo, Savio Albuquerque, Adriano Mian, Paula	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 3792 NCS, pp.130-141, 3792 NCS, Budapest, Hungary (2005)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
52	Requirements for a knowledge management framework to be used in software intensive organizations	Martínez, Paloma Amescua, Antonio Garcia, Javier Cuadra, Dolores Llorens, Juan Fuentes, J.M. Martín, Diego Cuevas, Gonzalo Calvo-Manzano, Jose Antonio Feliu, Tomas San	Proceedings of the 2005 IEEE International Conference on Information Reuse and Integration, IRI - 2005, pp.554-559, 2005, Las Vegas, NV, United States (2005)	X	CE1-12	Não	CS2 -SPI +Fer -Uso
53	Research into long-term improvements in small- to medium-sized organisations using SPICE as a framework for standards	Sanders, Marty Richardson, Ita	Software Process Improvement and Practice, pp.351-359	-		Não	

#	Título	Autores	Dados	Disponibilidade	Critério 1º Filtro (Rev.)	2º Filtro	Critério 2º Filtro
54	Research on CMMI-based software process metrics	Xu, Ruzhi Xue, Yunjiao Nie, Peiyao Zhang, Yuan Li, Desheng	First International Multi- Symposiums on Computer and Computational Sciences, IMSCCS'06, pp.391-397, 2, Hangzhou, Zhejiang, China (2006)	X	CE1-04	Não	CS2 -SPI +Fer -Uso
55	Software defect report and tracking system in an Intranet	Monteiro, A. Almeida, A.B. Goulao, M. Abreu, F.B. Sousa, P.	Proceedings of the Euromicro Conference on Software Maintenance and Reengineering, CSMR, pp.198-201, Amsterdam, Neth (1999)	-		Não	
56	Software process commonality analysis	Ocampo, Alexis Bella, Fabio Munch, Jurgen	Software Process Improvement and Practice, pp.273-285, n. 3	-		Não	
57	Software process improvement problems in twelve software companies: An empirical analysis	Beecham, Sarah Hall, Tracy Rainer, Austen	Empirical Software Engineering, pp.7-42, n. 1	X	CE1-17	Não	CS2 -SPI -Fer -Uso
58	Software process improvement strategy for enterprise information systems development	Fung, Richard Y.K. Tam, W.T. Ip, Andrew W.H. Lau, Henry C.W.	International Journal of Information Technology and Management, pp.225-241, n. 2-3	-		Não	
59	Software Process Improvement Support System: SPIS	Fukuyama, Shun-ichi Miyamura, Shuu-ichi Takagi, Hideo Tanaka, Ryoji	IEICE Transactions on Information and Systems, vol. E83-D, pp.747-756, n. 4	-		Não	
60	Software process improvement: Blueprints versus recipes	Aaen, Ivan	IEEE Software, pp.86-93, n. 5	X	CE1-14	Não	CS2 -SPI -Fer -Uso
61	Software process management: Practices in China	Wang, Qing Li, Mingshu	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp.317-331, 3840 NCS, Beijing, China (2006)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
62	Software Processes Improvement and Specifications for Embedded Systems	Jun, Dong Rui, Lin Yi-min, He	Software Engineering Research, Management & Applications, 2007. SERA 2007. 5th ACIS International Conference on, vol. , pp.13-18 (2007)	X	CI1-08	Sim	CI2 +SPI +Fer +Uso
63	Standard based software process assessments in small companies	Gresse Von Wangenheim, Christiane Varkoi, Timo Salviano, Clenio F.	Software Process Improvement and Practice, pp.329-335, n. 3	-	CE1-20	Não	CS2 -SPI -Fer -Uso
64	Study on workflow simulation for software process improvement	Lu, Chun-Xia Li, Wen-Li Li, Xu	Xitong Fangzhen Xuebao / Journal of System Simulation, pp.2501-2506, n. 11	-		Não	
65	Taba workstation: Supporting software process deployment based on CMMI and MR-MPS.BR	Montoni, Mariano Santos, Gleison Rocha, Ana Regina Figueiredo, Savio Cabrai, Reinaldo Barcellos, Rafael Barreto, Ahilton Scares, Andrea Cerdeiral, Cristina Lupo, Peter	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 4034 NCS, pp.249-262, 4034 NCS, Amsterdam, Netherlands (2006)	X	CI1-06	Sim	CI2 +SPI +Fer +Uso
66	Taba workstation: Supporting software process improvement initiatives based on software standards and maturity models	Ferreira, Analia Irigoyen Ferreira Santos, Gleison Cerqueira, Roberta Montoni, Mariano Barreto, Ahilton Rocha, Ana Regina Figueiredo, Savio Barreto, Andrea Filho, Reinaldo C. Silva Lupo, Peter Cerdeiral, Cristina	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 4257 NCS, pp.207-218, 4257 NCS, Joensuu, Finland (2006)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
67	Tackling the complexity of requirements engineering process improvement by partitioning the improvement task	Nikula, Uolevi Sajaniemi, Jorma	Proceedings of the Australian Software Engineering Conference, ASWEC, pp.48-57, 2005, Brisbane, Australia (2005)	X	CE1-16	Não	CS2 +SPI -Fer +Uso

#	Título	Autores	Dados	Disponibilidade	Critério 1º Filtro (Rev.)	2º Filtro	Critério 2º Filtro
68	Technical controlling and software process improvement	Ebert, C.	J Syst Software, pp.25-39, n. 1, Alcatel Switching Systems Division, Fr.-Wellesplein 1, B-2018, Antwerpen, Belgium	X	CE1-18	Não	CS2 -SPI -Fer -Uso
69	The AV-8B team learns synergy of EVM and TSP accelerates software process improvement	Pracchia, L.	CrossTalk, pp.20-22, n. 1, NAWCWD 41K300D (L. Pracchia), Bldg. 1494, I Administration Circle, China Lake, CA 93555	X	CE1-13	Não	CS2 +SPI -Fer +Uso
70	The Knowledge Based Software Process Improvement Program: A Rational Analysis	K, Alagarsamy S, Justus K, Iyakutti	Software Engineering Advances, 2007. ICSEA 2007. International Conference on, vol. , pp.61-61 (2007)	X	CE1-13	Não	CS2 +SPI -Fer +Uso
71	The long-term effects of an EPG/ER in a small software organisation	Kurniawati, Felicia Jeffery, Ross	Proceedings of the Australian Software Engineering Conference, ASWEC, pp.128-136, 2004, Melbourne, Vic., Australia (2004)	X	CI1-08	Sim	CI2 +SPI +Fer +Uso
72	The management of software processes with software process improvement tool based on ISO 15504	Choi, Yoonjung Lee, EunSeok Ha, Sujung	The 7th International Conference on Advanced Communication Technology, ICACT 2005, vol. 2, pp.933-936, 2, Phoenix Park, South Korea (2005)	X	CI1-03	Sim	CI2 +SPI +Fer +Uso
73	The role of awareness support in collaborative improvement of software processes	de Araujo, R.M. da Silva Borges, M.R.	String Processing and Information Retrieval Symposium, 1999 and International Workshop on Groupware, vol. , pp.343-347 (1999)	X	CE1-12	Não	CS2 -SPI +Fer -Uso
74	The use and effects of an electronic process guide and experience repository: a longitudinal study	Kurniawati, Felicia Jeffery, Ross	Information and Software Technology, pp.566-577, n. 7	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
75	Toward computational support for software process improvement activities	Sakamoto, Keishi Nakakoji, Kumiyo Takagi, Yasunari Niihara, Naoki	Proceedings - International Conference on Software Engineering, pp.22-31, Kyoto, Jpn (1998)	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
76	Uma Abordagem para Melhoria Contínua do Processo de Desenvolvimento de Software	Malheiros, V., Paim, F. R., Guzzo, H., Mendonça Neto, M. G.	V Simpósio Brasileiro de Qualidade de Software, SBQS'2006, pp.218-230	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
77	Understanding the use of an electronic process guide	Scott, Louise Carvalho, Lucila Jeffery, Ross D'Ambra, John Becker-Kornstaedt, Ulrike	Information and Software Technology, pp.601-616, n. 10	X	CI1-11	Sim	CI2 +SPI +Fer +Uso
78	Using grounded theory to understand software process improvement: A study of Irish software product companies	Coleman, Gerry O'Connor, Rory	Information and Software Technology, pp.654-667, n. 6	X	CE1-14	Não	CS2 -SPI -Fer -Uso
79	Using knowledge management to improve software process performance in a CMM level 3 organization	De Almeida Falbo, Ricardo Borges, Ligia S. Mota Valente, Fabio Feu Rosa	Proceedings - Fourth International Conference on Quality Software, QSIC 2004, pp.162-169, Braunschweig, Germany (2004)	X	CE1-12	Não	CS2 +SPI +Fer -Uso
80	Using the Internet to communicate software metrics in a large organization	Jokikyyny, Tony	Conference Record / IEEE Global Telecommunications Conference, vol. 3, pp.1947-1953, 3, Rio de Janeiro, Braz (1999)	X	CE1-12	Não	CS2 -SPI +Fer -Uso

Referências Bibliográficas

- BIOLCHINI, J., MIAN, P.G., NATALI, A.C., *et al.*, 2005, *Systematic Review in Software Engineering: Relevance and Utility*, Technical Report ES-679/05, Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- DIESTE, O., PADUA, A.G., 2007, "Developing Search Strategies for Detecting Relevant Experiments for Systematic Reviews". In: *Proceedings of the First International Symposium on Empirical Software Engineering and Measurement*.
- KITCHENHAM, B., 2004, *Procedures for Performing Systematic Reviews*, Technical Report, Department of Computer Science Keele University, Keele.
- MAFRA, S., 2006, *Definição de uma Técnica de Leitura Baseada em Perspectiva (OO-PBR) Apoiada por Estudos Experimentais*, Dissertação de M. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- MAFRA, S., TRAVASSOS, G.H., 2006, *Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software*, Relatório Técnico RT-ES 687/06, Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- MONTONI, M., 2007, *Uma Abordagem para Condução de Iniciativas de Melhoria de Processos de Software*, Exame de Qualificação, COPPE, UFRJ, Rio de Janeiro, Brasil.
- SILVA FILHO, R.C., 2006, *Uma Abordagem para Avaliação de Propostas de Melhoria em Processos de Software*, Dissertação de M.Sc., COPPE, UFRJ, Rio de Janeiro, RJ, Brasil.

ANEXO IV - ONTOLOGIA DE CORPORAÇÃO

Este anexo descreve de forma completa a ontologia de Corporação citada no Capítulo V desta tese. A ontologia aqui apresentada é uma evolução da ontologia de Organização proposta por VILLELA (2004), que, por sua vez, é uma evolução da ontologia de Processo de Software definida por FALBO (1998). Esta ontologia descreve conceitos, propriedades, relações e restrições que contemplam os diferentes aspectos de um modelo organizacional. Os conceitos e relações criados e/ou alterados em decorrência desta tese são aqueles descritos no Capítulo V, todos os demais são iguais aos descritos por VILLELA (2004).

1 Introdução

A ontologia de organização tem como propósito fornecer um vocabulário comum que possa ser utilizado para representar conhecimento útil para os desenvolvedores de software sobre as organizações envolvidas em um projeto de software.

Em termos de definição do escopo da ontologia de corporação, as seguintes questões gerais de competência¹ (QG) foram formuladas:

- QG1: Como a organização é percebida em seu ambiente?
- QG2: Como a organização está estruturada?
- QG3: Quem são as pessoas que fazem parte da organização?
- QG4: Como a organização se comporta?
- QG5: Qual é a distribuição de autoridade e responsabilidade na organização?
- QG6: Como as competências desejadas e possuídas encontram-se distribuídas na organização?
- QG7: Quais são os objetivos estabelecidos para a organização?
- QG8: Como têm sido conduzidos os projetos da organização?

¹ Questões de competência são questões que a ontologia deve possibilitar que sejam respondidas (USCHOLD E GRUNINGER, 1996).

2 Definição e Formalização da Ontologia

A definição da ontologia de organização por VILLELA (2004) evoluiu a ontologia de processo de software definida por FALBO (1998) e foi baseada em pesquisa bibliográfica sobre as ontologias relacionadas, elaboradas no contexto do projeto TOVE (*TOronto Virtual Enterprise*) (FOX *et al.*, 1993; FADEL *et al.*, 1994; GRUNINGER e FOX, 1994; FOX *et al.*, 1996; FOX e GRUNINGER, 1998), do projeto *Enterprise (USCHOLD et al., 1998)* e por COTA (COTA, 2002), em pesquisa bibliográfica na área de Administração de Empresas (MEGGINSON *et al.*, 1986a; MEGGINSON *et al.*, 1986b; CHIAVENATO, 1998c; CHIAVENATO, 1998b; CHIAVENATO, 1998a) e em entrevistas com especialistas da mesma área. As evoluções identificadas nesta tese foram definidas com base na experiência do grupo de Engenharia de Software da COPPE/UFRJ, em duas ontologias desenvolvidas por BERTOLLO (2006) e RUY (2006) sobre processo de software e organização, respectivamente, e em definições comumente aceitas pela comunidade relacionadas a processos de software (por exemplo, (ISO/IEC, 2003; CHRISISS *et al.*, 2006; SOFTEX, 2007)).

Inicialmente, as questões gerais de competência (identificadas pela sigla QG) foram refinadas, dando origem às questões de competência (identificadas pela sigla Q) listadas a seguir, que representam a especificação de requisitos da ontologia.

QG1: Como a organização é percebida em seu ambiente?

- Q1.1: Qual é a missão da organização?
- Q1.2: Quais são os domínios de conhecimento nos quais a organização atua?
- Q1.3: Quais são os artefatos/serviços oferecidos pela organização?
- Q1.4: Quais são as organizações clientes da organização?

QG2: Como a organização está estruturada?

- Q2.1: Quais são os cargos existentes na organização?
- Q2.2: Como a organização é decomposta em unidades organizacionais?
- Q2.3: Como uma unidade organizacional é decomposta em posições?
- Q2.4: Quais são as comissões existentes na organização?
- Q2.5: Quais são as relações de hierarquia entre duas organizações

QG3: Quem são as pessoas que fazem parte da organização?

- Q3.1: Quais são as pessoas que possuem um determinado cargo na organização?
- Q3.2: Quais são as pessoas que ocupam uma determinada posição na organização?

Q3.3: Quais são as pessoas que integram uma determinada comissão na organização?

QG4: Como a organização se comporta?

Q4.1: Quais são os processos executados na organização?

Q4.2: Quais são as normas seguidas por um determinado processo?

Q4.3: Como um determinado processo é decomposto em atividades?

Q4.4: Quais são os recursos e insumos necessários para executar uma determinada atividade?

Q4.5: Quais são os produtos resultantes da execução de uma determinada atividade?

Q4.6: Quais são as pessoas que estão alocadas para a execução de uma atividade?

Q4.7: Quais procedimentos podem ser utilizados para executar uma determinada atividade?

Q4.8: Quais são as ferramentas de software que automatizam um determinado procedimento?

QG5: Qual é a distribuição de autoridade e responsabilidade na organização?

Q5.1: A quais unidades organizacionais uma determinada unidade organizacional se reporta?

Q5.2: A quais unidades organizacionais uma determinada unidade organizacional assessora ou presta serviço?

Q5.3: A quais posições uma determinada posição se reporta?

Q5.4: A quais posições uma determinada posição assessora ou presta serviço?

Q5.5: Quais são as atividades que são atribuição de uma determinada posição ou cargo?

QG6: Como as competências desejadas e possuídas encontram-se distribuídas na organização?

Q6.1: Quais pessoas na organização possuem uma determinada competência?

Q6.2: Em quais atividades da organização uma determinada competência é requerida?

Q6.3: Em quais cargos da organização uma determinada competência é requerida?

Q6.4: Em quais posições da organização uma determinada competência é requerida?

Q6.5: A que domínio de conhecimento pertence um conhecimento?

QG7: Quais são os objetivos estabelecidos para a organização?

Q7.1: Quais são os objetivos da organização?

Q7.2: Quais são os objetivos de uma determinada unidade organizacional?

Q7.3: Quais são os objetivos de uma determinada posição?

Q7.4: Como um determinado objetivo é decomposto?

Q7.5: Qual é a prioridade de um determinado objetivo?

QG8: Como têm sido conduzidos os projetos da organização?

Q8.1: Quais são os projetos conduzidos pela organização?

Q8.2: Quais são as outras organizações envolvidas em um determinado projeto?

Q8.3: Qual processo orienta a execução de um determinado projeto?

Q8.4: Qual é a equipe de um determinado projeto?

Para iniciar a captura da ontologia, termos e frases potencialmente relevantes foram identificados e atribuídos a áreas de trabalho, que tiveram sua semântica definida e deram origem a subontologias. A Figura 1 mostra as subontologias obtidas e os relacionamentos entre elas.

Para cada subontologia, foram elaborados modelos utilizando o subconjunto da UML proposto por MIAN (MIAN, 2003) para descrição de ontologias. Além disso, os conceitos, relações e restrições foram descritos em linguagem natural e exemplificados. Na formalização, foi utilizada lógica de primeira ordem, definindo-se as constantes, predicados e axiomas. A avaliação da ontologia ocorreu ao longo de todo o processo. A seguir, são apresentadas a definição e formalização de cada subontologia.

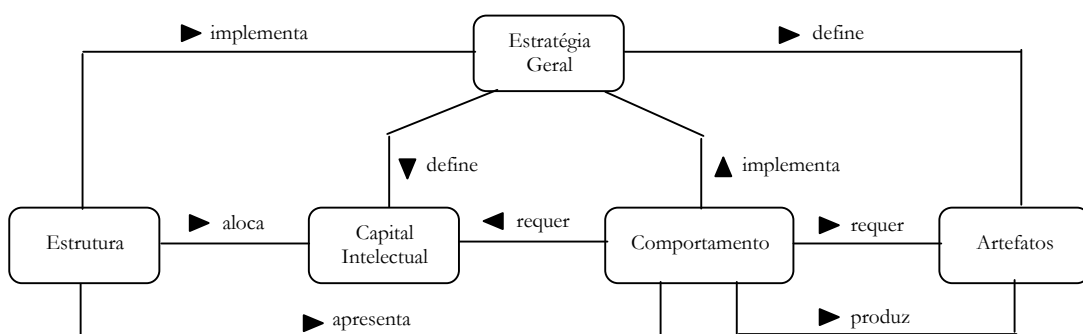


Figura 1 – Subontologias da Ontologia de Corporação

2.1 Subontologia de Capital Intelectual

A subontologia de Capital Intelectual estabelece o vocabulário necessário para descrever o capital intelectual de uma organização, respondendo às questões de competência Q6.1 e Q6.5. Uma visão geral de seus conceitos pode ser vista na Figura 2. Os seguintes aspectos são tratados pela subontologia: taxonomia de competência, interação entre experiência e conhecimento, disponibilidade de competências e decomposição de domínio de conhecimento.

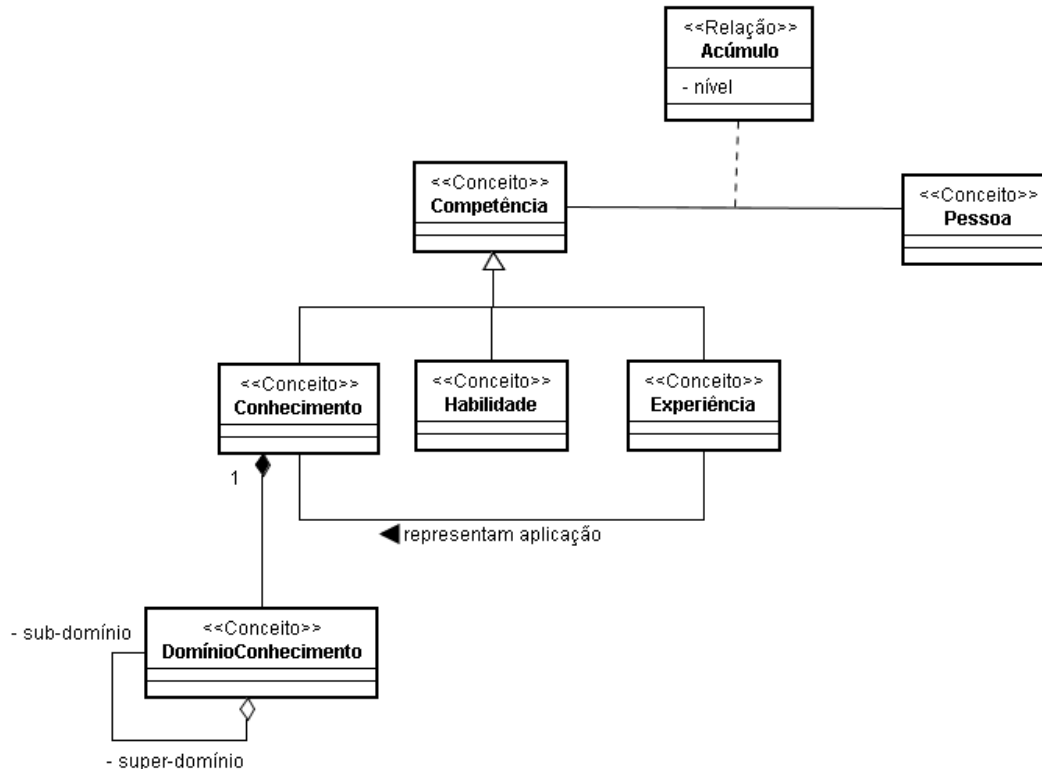


Figura 2 – Subontologia de Capital Intelectual

2.1.1 Taxonomia de Competência

Competências tornam as pessoas capazes de executar atividades que envolvem algum grau de dificuldade. A taxonomia de competência define como competências podem ser classificadas quanto a sua natureza, de acordo com o propósito da ontologia.

Conhecimentos são apropriações de objetos pelo pensamento através de definição, percepção clara, análise, apreensão completa ou outra forma de apropriação (HOLANDA, 1999). Exemplos de conhecimento são: conhecimento sobre análise orientada a objeto, conhecimento sobre arquiteturas cliente-servidor, conhecimento sobre as teorias de administração e conhecimento sobre as normas de tráfego aéreo.

Habilidades são aptidões natas ou adquiridas não associadas a uma atividade ou domínio de conhecimento específico. Exemplos de habilidades são: liderança, habilidade de negociação, habilidade de argumentação e habilidade de expor idéias em público.

Experiências são adquiridas através da prática, ou seja, através da execução de atividades. Exemplos de experiências são: experiência na definição de arquiteturas cliente-servidor, experiência na administração de aeroportos e experiência na assistência emergencial de pacientes com problemas cardíacos.

2.1.2 Interação entre Experiência e Conhecimento

A interação entre experiência e conhecimento também é tratada pela subontologia. Experiências geralmente são relevantes por representarem a aplicação de conhecimentos na prática. Por exemplo, a experiência na definição de arquiteturas cliente-servidor representa a aplicação do conhecimento sobre este tipo de arquitetura na prática. No entanto, existem experiências que são relevantes por outros motivos, tais como a experiência na elaboração de projetos visuais de *sites* para a Internet e a experiência na criação de peças publicitárias, que são relevantes pela criatividade envolvida. Além disso, experiências podem ser descritas de forma tão genérica, que pode não ser de interesse especificar o que a experiência significa em termos de aplicação de conhecimentos. Desta forma, não foi estabelecida nenhuma restrição para a relação proposta.

2.1.3 Disponibilidade de Competências

A subontologia trata do aspecto da disponibilidade de competências na organização ao relacionar os conceitos **Pessoa** e **Competência** através do predicado **Acúmulo**.

Pessoas são fundamentais para o funcionamento de uma organização, atuando na execução de atividades necessárias ao cumprimento da missão da organização.

Ao longo de sua trajetória profissional, as pessoas acumulam competências, disponibilizando-as para a organização em que trabalham. As competências possuídas pelos profissionais de uma organização são de grande importância para os próprios profissionais, pois são utilizadas para estabelecer o seu papel e o seu valor na organização, e também para a organização, pois representam o seu capital intelectual.

2.1.4 Decomposição de Domínio de Conhecimento

O último aspecto tratado pela subontologia é a decomposição de um domínio de conhecimento.

Um *domínio de conhecimento* é um conjunto de conhecimentos reunidos de acordo com a homogeneidade de conteúdo. Exemplos de domínios de conhecimento são: cardiologia, aviação comercial e extração de petróleo.

Além disso, domínios de conhecimentos mais amplos, chamados *super-domínios*, podem ser compostos a partir de outros domínios de conhecimento, chamados *subdomínios*, incorporando os seus conhecimentos. Por exemplo, o super-domínio de Computação é composto dos subdomínios de Engenharia de Software, Redes de Computadores, Banco de Dados, entre outros.

Os predicados *domínioelementar(d)* e *macrodomínio(d)* indicam, respectivamente, que *d* é um domínio que não pode ser decomposto em outros domínios de conhecimento, e que *d* é um domínio que não faz parte de nenhum outro domínio de conhecimento. Estes predicados são definidos em termos dos conceitos de subdomínio e super-domínio, como mostram as sentenças abaixo.

$$(\forall d) (\text{domínioelementar}(d) \leftrightarrow \neg (\exists d_1) (\text{subdomínio}(d_1, d))) \quad (\text{A1})$$

$$(\forall d) (\text{macrodomínio}(d) \leftrightarrow \neg (\exists d_1) (\text{superdomínio}(d_1, d))) \quad (\text{A2})$$

Para garantir a integridade do modelo, de forma que um domínio de conhecimento seja composto ou de um conjunto de subdomínios ou de um conjunto de conhecimentos, foi definido o axioma de consolidação abaixo, indicando que apenas domínios elementares são decompostos em termos de conhecimentos.

$$(\forall c, d) (\text{domínio_composição}(c, d) \rightarrow \text{domínioelementar}(d)) \quad (\text{A3})$$

Para representar que o conteúdo de um domínio de conhecimento é dado pelo conteúdo de seu subdomínios (se é um super-domínio) ou pelo conjunto de conhecimentos que o compõem (se é um domínio elementar), foi definido o predicado *domínio_conteúdo(c,d)*, que indica que o conhecimento *c* faz parte do conteúdo definido pelo domínio de conhecimento *d*. Para isto, foi definido o seguinte axioma:

$$(\forall c, d) (\text{domínio_conteúdo}(c, d) \leftrightarrow \text{domínio_composição}(c, d) \vee (\exists d_k) (\text{subdomínio}(d_k, d) \wedge \text{domínio_conteúdo}(c, d_k))) \quad (\text{A4})$$

2.2 Subontologia de Estrutura

A subontologia de Estrutura estabelece o vocabulário necessário para descrever como a organização está estruturada, respondendo às questões de competência Q2.1 a Q2.5, Q3.1 a Q3.3, Q5.1 a Q5.5, Q6.3, Q6.4 e Q7.1 a Q7.5. Com base nestas questões de competência, foi verificada a necessidade de se tratar os seguintes aspectos: decomposição da organização, distribuição de autoridade e responsabilidade entre unidades organizacionais, decomposição de unidade organizacional, distribuição de autoridade e responsabilidade entre posições, especificação de cargo e posição, preenchimento de vagas, formação de equipe e definição de objetivos.

2.2.1 Decomposição da Organização

Organizações são grupos de pessoas trabalhando em conjunto para o cumprimento de uma missão, onde a missão de uma organização define o seu propósito dentro do sistema econômico ou social. Segundo MEGGINSON *et al.* (1986b), a divisão do trabalho é o princípio base das organizações, permitindo que grupos de pessoas, trabalhando em conjunto de modo cooperativo e coordenado, possam realizar mais do que cada uma agindo de forma independente. Existem diversas maneiras de estruturar uma organização de forma a estabelecer a divisão de trabalho (MEGGINSON *et al.*, 1986b), mas os componentes básicos são os cargos, as unidades organizacionais e as comissões. A Figura 3 aborda o aspecto da decomposição da organização.

Uma **corporação** pode ser definida como um tipo de organização composta de outras organizações (por exemplo, subsidiárias ou filiais) com certo grau de independência, mas subordinadas, de alguma forma, aos interesses e regras da corporação. O que difere uma corporação de uma organização comum é o fato de haver uma relação de hierarquia entre as organizações que compõem a corporação e o comando central da corporação. Esta relação de hierarquia pode estar configurada de diversas formas, por exemplo, ser uma relação apenas administrativa, sendo as organizações unidades independentes com ramo de negócio particulares, ou então ser uma relação de comando e gerência das atividades fins da organização, estando as organizações subordinadas a uma linha mestra definida pela corporação. Neste segundo caso, é comum, também, que estas organizações tenham ramos de atuação semelhantes e compartilhem alguns nichos de mercado, mas em contextos/localidades diferentes.

Os axiomas do relacionamento entre Organização e Corporação não serão descritos formalmente por serem derivados diretamente do modelo. De forma geral, *toda corporação é uma organização* e toda *corporação é composta de organizações*. Além disso, uma *organização é uma organização independente se não está associada a uma corporação* e, de forma similar, uma *organização é subordinada a uma corporação quando está associada a uma corporação*.

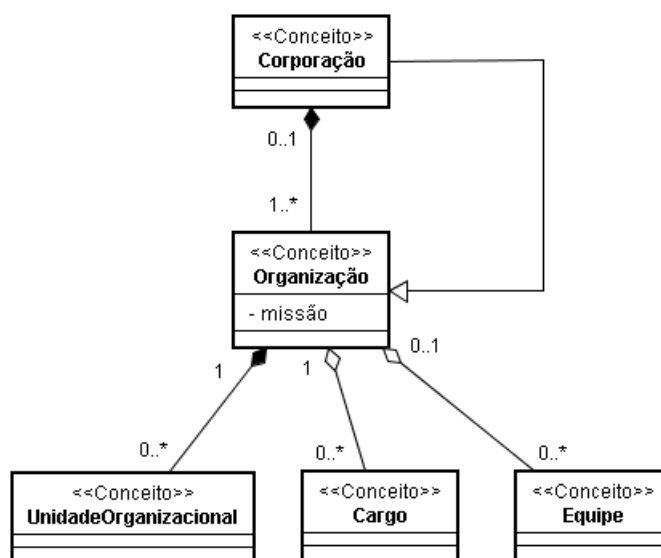


Figura 3 – Decomposição da Organização

Unidades Organizacionais são agrupamentos de componentes da organização (por exemplo: atividades e pessoas) de acordo com a homogeneidade de conteúdo² para que a organização possa ser econômica e eficiente (baseado em (CHIAVENATO, 1998b)). Exemplos de unidades organizacionais são: divisão farmacêutica, departamento de antibióticos, agência em Brasília e seção de contas corrente.

Equipes são agrupamentos de pessoas com finalidade determinada e, normalmente, por período de tempo limitado. O conceito de Equipe, no contexto desta subontologia, tem o objetivo de possibilitar a representação de comissões. Segundo CHIAVENATO (1998c), comissões se distinguem das unidades organizacionais por geralmente terem finalidade que abrange várias unidades organizacionais (assuntos interdepartamentais),

² A Teoria Geral de Administração prevê agrupamento por funções (por exemplo: produção, contabilidade, vendas e serviço ao cliente), por produtos ou serviços, por localização geográfica, por clientela, por processos, por projetos, ressaltando que estes tipos de agrupamento podem ser insuficientes para resolver todos os casos e que a maioria das organizações utiliza mais de um tipo de agrupamento (MEGGINSON *et al.*, 1986b; CHIAVENATO, 1998c).

terem participantes que pertencem a diferentes unidades organizacionais e diferentes níveis hierárquicos, não terem posição definida na estrutura organizacional, funcionarem esporádica ou intermitentemente durante certos dias ou em determinadas horas, e durarem enquanto não atingirem seu objetivo ou não realizarem a tarefa para a qual foram criadas. Como será visto na subontologia de Comportamento, o conceito de Equipe também pode estar relacionado a projetos da organização. Sendo assim, o conceito de **Comissão** foi definido como sendo uma equipe que compõe a organização sem estar relacionada a um projeto. Segundo MEGGINSON *et al.* (1986b), alguns exemplos de comissões são comissões orçamentárias, conselhos administrativos e comissão de planejamento de um novo produto.

O axioma de consolidação que reflete a restrição implícita na definição de comissão será apresentado posteriormente (axioma A39), pois depende de predicados que extrapolam o escopo desta subontologia.

Cargos especificam o conjunto de atividades a serem executadas pelas pessoas que os ocupam, suas responsabilidades, competências desejadas, além das condições de trabalho oferecidas, mas não determinam as unidades organizacionais específicas às quais essas pessoas serão alocadas. Não foi adotado o termo Função para este conceito, pois este termo tem outro significado no contexto da Administração, conforme pode ser observado na nota da página anterior.

2.2.2 Distribuição de Autoridade e Responsabilidade entre Unidades Organizacionais

A Figura 4 mostra como a distribuição de autoridade e responsabilidade entre unidades organizacionais é tratada na subontologia.

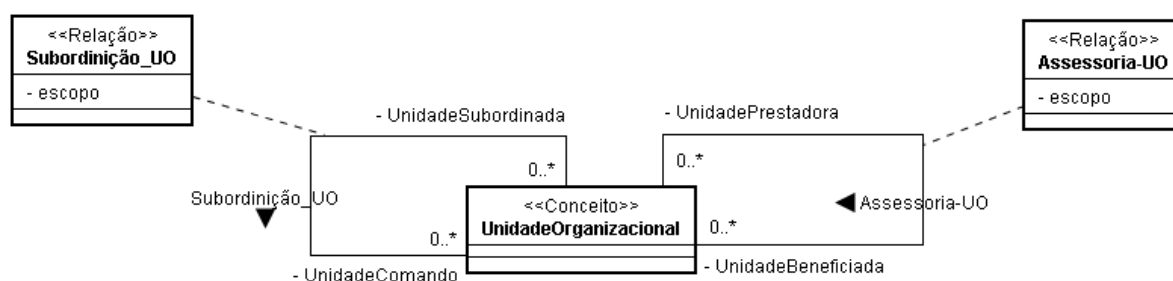


Figura 4 – Distribuição de Autoridade e Responsabilidade

O modelo proposto permite: 1) a autoridade linear ou única, em que cada unidade subordinada reporta-se única e exclusivamente a uma única unidade superior; 2) a

autoridade funcional ou dividida, em que cada unidade subordinada reporta-se a várias unidades superiores, porém reporta-se a cada uma delas somente nos assuntos da especialidade de cada uma (ou seja, dentro de um escopo); e 3) a fusão das abordagens anteriores com predomínio da primeira, que é chamada linha-*staff*, em que cada unidade organizacional reporta-se a uma única unidade superior (linha), porém recebe assessoria e serviços especializados de várias unidades organizacionais (*staff*), que possuem autoridade relativa e parcial estabelecida mediante uma relação funcional e de consultoria (CHIAVENATO, 1998c). A Figura 5 fornece uma representação esquemática dessas relações.

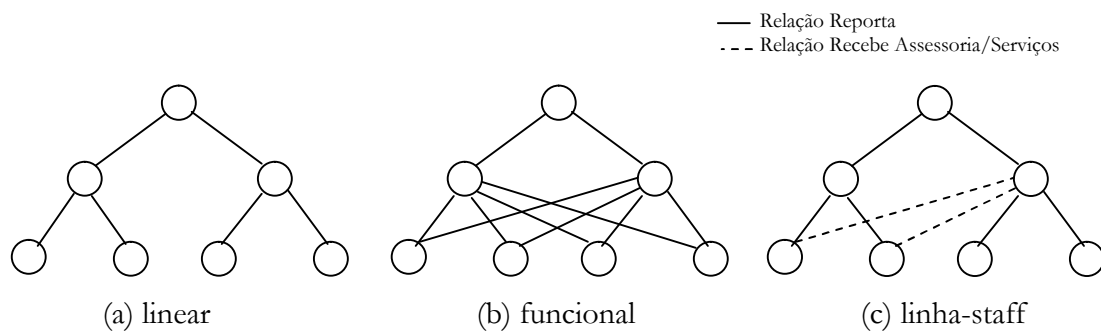


Figura 5 – Esquema das Estruturas adotadas nas Organizações

Um exemplo de unidade organizacional que tem relação de assessoria e prestação de serviços com outras unidades em organizações que desenvolvem e mantêm software é a Gerência da Qualidade, cuja atividade não é decidir, comandar, executar, mas sim pensar, planejar, sugerir, recomendar, assessorar e prestar serviços especializados relacionados à qualidade.

Para formalizar as relações representadas na Figura 4, foram definidos o predicado *subordinação_UO*(u_1, u_2, s), que indica que a unidade organizacional u_1 se reporta à unidade organizacional u_2 dentro do escopo definido por s , e o predicado *assessoria_UO*(u_1, u_2, s), que indica que a unidade organizacional u_1 assessoria ou presta serviço à unidade organizacional u_2 dentro do escopo definido por s . Também foram definidos os predicados *UOsubordinada*(u_1, u_2), *UOcomando*(u_2, u_1), *UObeneficiada*(u_1, u_2), *UOprestadora*(u_2, u_1), indicando os papéis representados pelas unidades nas duas relações acima.

2.2.3 Decomposição de Unidade Organizacional

A Figura 6 apresenta o modelo referente à decomposição de unidade organizacional.

Posições são o resultado não só da alocação de cargos a unidades organizacionais, mas, principalmente, da definição da estrutura necessária para que a unidade organizacional funcione de forma eficiente e econômica, especificando atividades, responsabilidades e competências adicionais, em conformidade com a finalidade da unidade organizacional específica. Em uma universidade federal, por exemplo, departamentos são unidades organizacionais e professor assistente, professor adjunto e professor titular são alguns dos cargos existentes. Ao serem alocados a um departamento (por concurso ou por empréstimo entre departamentos), os professores passam a ocupar posições no departamento. Uma das posições, por exemplo, é a de chefe de departamento, que tem como propósito coordenar as ações dos professores e demais funcionários alocados ao departamento. Chefe de departamento em uma universidade não é um cargo, mas sim uma posição que é ocupada por um professor por um período de tempo.

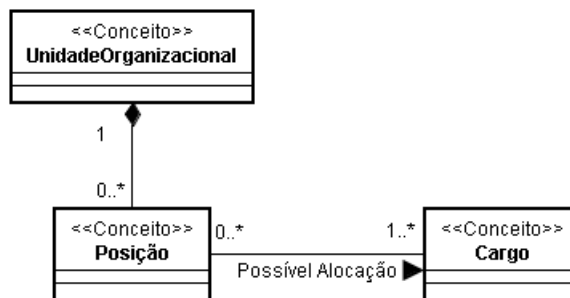


Figura 6 – Decomposição de Unidade Organizacional

O predicado *possívelalocação(p,c)* indica que a posição *p* representa a possibilidade de alocação de pessoas com o cargo *c* na unidade à qual *p* pertence, ou seja, que a posição *p* pode ser ocupada por pessoas que possuem o cargo *c*.

2.2.4 Distribuição de Autoridade e Responsabilidade entre Posições

O aspecto da distribuição de autoridade e responsabilidade entre posições é capturado pelas relações descritas no modelo da Figura 7, que contemplam as mesmas estruturas discutidas no nível de unidades organizacionais, pois estas estruturas são aplicáveis em ambos os níveis, conforme discutido por CHIAVENATO (1998c). Isto

implica que predicados e axiomas similares aos definidos para formalizar as relações da Figura 4 foram definidos.

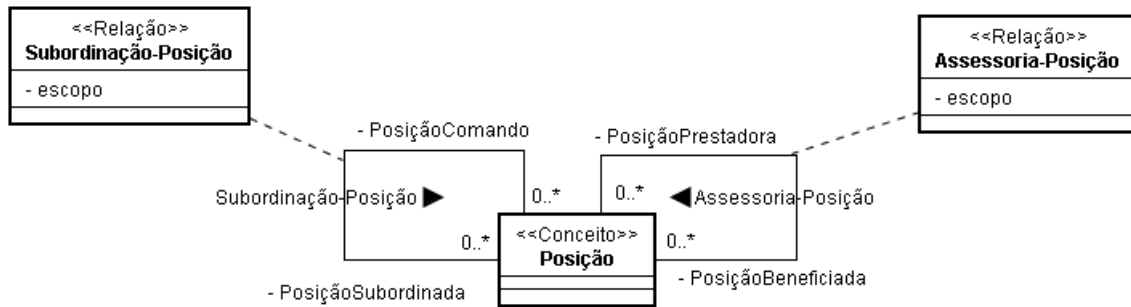


Figura 7 – Distribuição de Autoridade e Responsabilidade

2.2.5 Especificação de Cargos e Posições

A Figura 8 ilustra como a especificação de cargos e posições é abordada na subontologia.

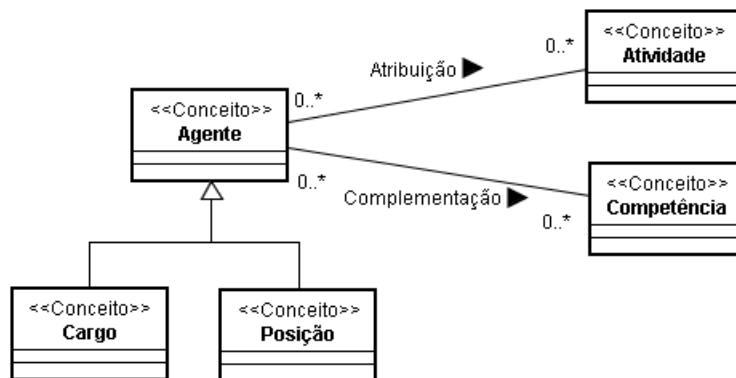


Figura 8 – Especificação de Cargos e Posições

Tanto cargos quanto posições especificam competências desejadas, atividades a serem executadas e responsabilidades a serem assumidas pelas pessoas que fazem parte da organização. Desta forma, foi introduzido o conceito de *Agente* na subontologia, que é uma especificação de um perfil necessário para que a organização cumpra a sua missão através da execução de atividades. Um agente pode representar um cargo ou uma posição.

Os predicados *atribuição(a,g)* e *complementação(c,g)* foram definidos, para indicar, respectivamente, que *a* é uma atividade a ser executada pelo agente *g* (ou seja, que lhe foi atribuída e lhe caracteriza), e que o perfil definido pelo agente *g* especifica que a competência *c* é desejada, em complementação às competências requeridas pelas atividades que lhe foram atribuídas. Os axiomas necessários para derivar as competências desejadas

em um agente serão apresentados posteriormente (axioma A19), pois dependem de predicados que extrapolam o escopo desta subontologia.

2.2.6 Preenchimento de Vagas e Formação de Equipes

O preenchimento das vagas em uma organização é representado pela alocação de pessoas às posições existentes e a consequente ocupação dos possíveis cargos, considerando, para isso, as competências possuídas pelas pessoas e as competências requeridas pelas posições e cargos. A Figura 9 aborda este aspecto da subontologia e o aspecto da formação de equipe.

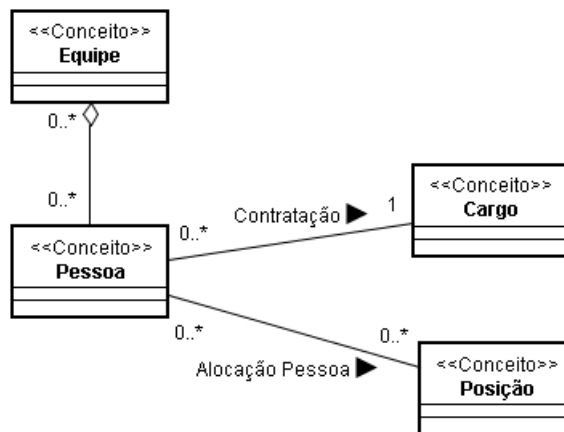


Figura 9 – Preenchimento de Vagas e Formação de Equipe

Para formalização das relações representadas na Figura 9, foram definidos os predicados *contratação(p,c)* e *alocação_pessoa(p,s)*, que indicam, respectivamente, que a pessoa *p* foi contratada para exercer o cargo *c* e que a pessoa *p* foi alocada à posição *s*, cabendo ressaltar que a relação de contratação representa qualquer forma de ingresso de uma pessoa em uma organização, incluindo contratos temporários e de sociedade, e que pessoas podem não estar alocadas a uma posição, uma vez que micro e pequenas empresas podem não ser decompostas em unidades organizacionais.

Uma observação a ser feita é que não foram definidos axiomas para restringir o preenchimento de vagas em uma organização com relação ao cargo necessário para ocupar uma posição ou as competências necessárias para ocupar um cargo ou posição, pois estas restrições nem sempre são obedecidas nas organizações.

2.2.7 Definição de Objetivos

A definição de objetivos é o último aspecto tratado por esta subontologia (Figura 10).

Objetivos são enunciados escritos sobre os resultados a serem alcançados num período de tempo determinado, onde os resultados devem ser, o quanto possível, quantificáveis (CHIAVENATO, 1998a).

Apesar de alguns autores preferirem diferenciar o conceito de Objetivo do conceito de Meta, MEGGINSON *et al.* (1986a) defende que é muito mais comum não diferenciá-los. Tanto as metas como os objetivos fornecem orientação e refletem as condições desejadas para melhoria do desempenho global da organização. Enquanto os objetivos são mais amplos, as metas são mais específicas, sendo mais adequadas para orientar as tomadas de decisão e atividades cotidianas da organização. A relação de composição de objetivos estabelece que um objetivo, chamado **super-objetivo**, pode ser decomposto em outros objetivos, chamados **subobjetivos**, o que permite que os objetivos iniciais, estabelecidos em termos gerais, possam ser detalhados até estarem adequados para orientar as tomadas de decisão e atividades cotidianas da organização. Sendo assim, não foi considerado necessário diferenciar estes conceitos no contexto desta ontologia.

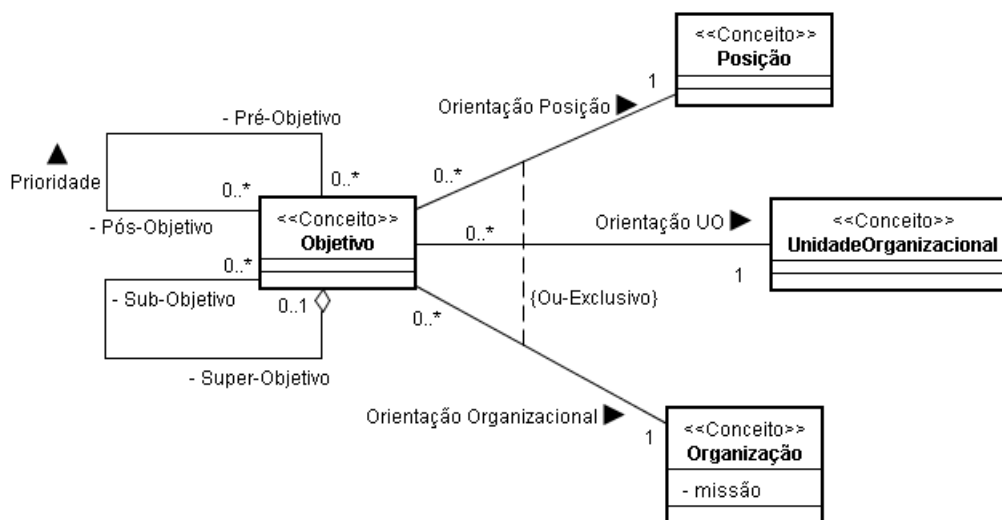


Figura 10 – Definição de Objetivos

Objetivos podem ser estabelecidos para a organização como um todo, para suas unidades organizacionais, e uma unidade organizacional pode, ainda, estabelecer objetivos a serem alcançados pelas suas posições, de forma que os objetivos da unidade organizacional sejam alcançados (MEGGINSON *et al.*, 1986a; CHIAVENATO, 1998a). Objetivos organizacionais podem estar relacionados à posição competitiva no mercado, índices de eficiência, taxa de dividendos, inovação e criatividade nos produtos, responsabilidade pública e social, entre outros. Exemplos de objetivos organizacionais são: “Administrar a empresa de modo que ela permaneça como líder mundial no campo de abrasivos”, “Manter e construir a reputação da empresa como empresa responsável”.

A maioria das organizações possui múltiplos objetivos. Unidades organizacionais e posições também podem possuir múltiplos objetivos. A existência de múltiplos objetivos pode acarretar conflitos entre os mesmos, sendo necessário, portanto, estabelecer o equilíbrio ótimo entre eles (MEGGINSON *et al.*, 1986a; CHIAVENATO, 1998a). Além da redefinição dos objetivos, uma forma de equilibrá-los é estabelecer prioridades entre eles. Segundo CHIAVENATO (1998a), objetivos precisam ser escalonados em ordem gradativa de prioridade. A relação de prioridade entre objetivos estabelece que um objetivo, chamado *pré-objetivo*, é mais importante que um outro objetivo, chamado *pós-objetivo*, o que permite a solução dos possíveis conflitos.

Para formalizar as relações mencionadas, foram definidos: 1) o predicado *subobjetivo*(t_1, t_2), que indica que o objetivo t_1 é um subobjetivo do (ou é um meio de atingir o) objetivo t_2 , e o correspondente *superobjetivo*(t_2, t_1); 2) o predicado *preobjetivo*(t_1, t_2), que indica que t_1 é um pré-objetivo de (ou deve ser priorizado em relação a) t_2 , e o correspondente *posobjetivo*(t_2, t_1); e 3) os predicados *orientação_org*(t, r), *orientação_UO*(t, u) e *orientação_pos*(t, p), que indicam, respectivamente, que o objetivo t orienta a execução das atividades da organização r , da unidade u e da posição p .

Deve-se observar, ainda, que as restrições que estabelecem a relação de prioridade sempre envolvem objetivos do mesmo nível (axiomas A6 e A7), sendo que, no nível de organização (axioma A5), a restrição é ainda mais forte e os objetivos relacionados devem pertencer a uma mesma organização, pois objetivos de organizações diferentes são completamente independentes entre si. Já as restrições sobre a relação de composição (axiomas A8 a A10) estabelecem que esta relação sempre envolve objetivos que orientam as atividades do mesmo elemento da estrutura organizacional, quer este seja a organização, uma unidade organizacional ou uma posição.

$$(\forall t_1, t_2, r) (\text{orientação_org}(t_1, r) \wedge \text{preobjetivo}(t_1, t_2) \rightarrow \text{orientação_org}(t_2, r)) \quad (\text{A5})$$

$$(\forall t_1, t_2, u_1) (\text{orientação_UO}(t_1, u_1) \wedge \text{preobjetivo}(t_1, t_2) \rightarrow (\exists u_2) (\text{orientação_UO}(t_2, u_2))) \quad (\text{A6})$$

$$(\forall t_1, t_2, p_1) (\text{orientação_pos}(t_1, p_1) \wedge \text{preobjetivo}(t_1, t_2) \rightarrow (\exists p_2) (\text{orientação_pos}(t_2, p_2))) \quad (\text{A7})$$

$$(\forall t_1, t_2, r) (\text{orientação_org}(t_1, r) \wedge \text{subobjetivo}(t_1, t_2) \rightarrow \text{orientação_org}(t_2, r)) \quad (\text{A8})$$

$$(\forall t_1, t_2, u) (\text{orientação_UO}(t_1, u) \wedge \text{subobjetivo}(t_1, t_2) \rightarrow \text{orientação_UO}(t_2, u)) \quad (\text{A9})$$

$$(\forall t_1, t_2, p) (\text{orientação_pos}(t_1, p) \wedge \text{subobjetivo}(t_1, t_2) \rightarrow \text{orientação_pos}(t_2, p)) \quad (\text{A10})$$

De forma similar à composição de domínio de conhecimento, foram definidos os predicados *objetivoelementar(t)* e *macroobjetivo(t)*.

2.3 Subontologia de Artefatos

Artefato é qualquer elemento produzido pelo homem e não por causas naturais, podendo exercer diferentes papéis em uma organização, tais como o de insumo ou produto de uma atividade. A subontologia de Artefatos, que pode ser vista na Figura 11, agrupa os conceitos e relações que definem os artefatos em termos de sua natureza e composição. Considerada de forma isolada, a subontologia não responde a nenhuma questão de competência, mas é fundamental para que as questões Q1.5, Q4.4 e Q4.5 sejam respondidas.

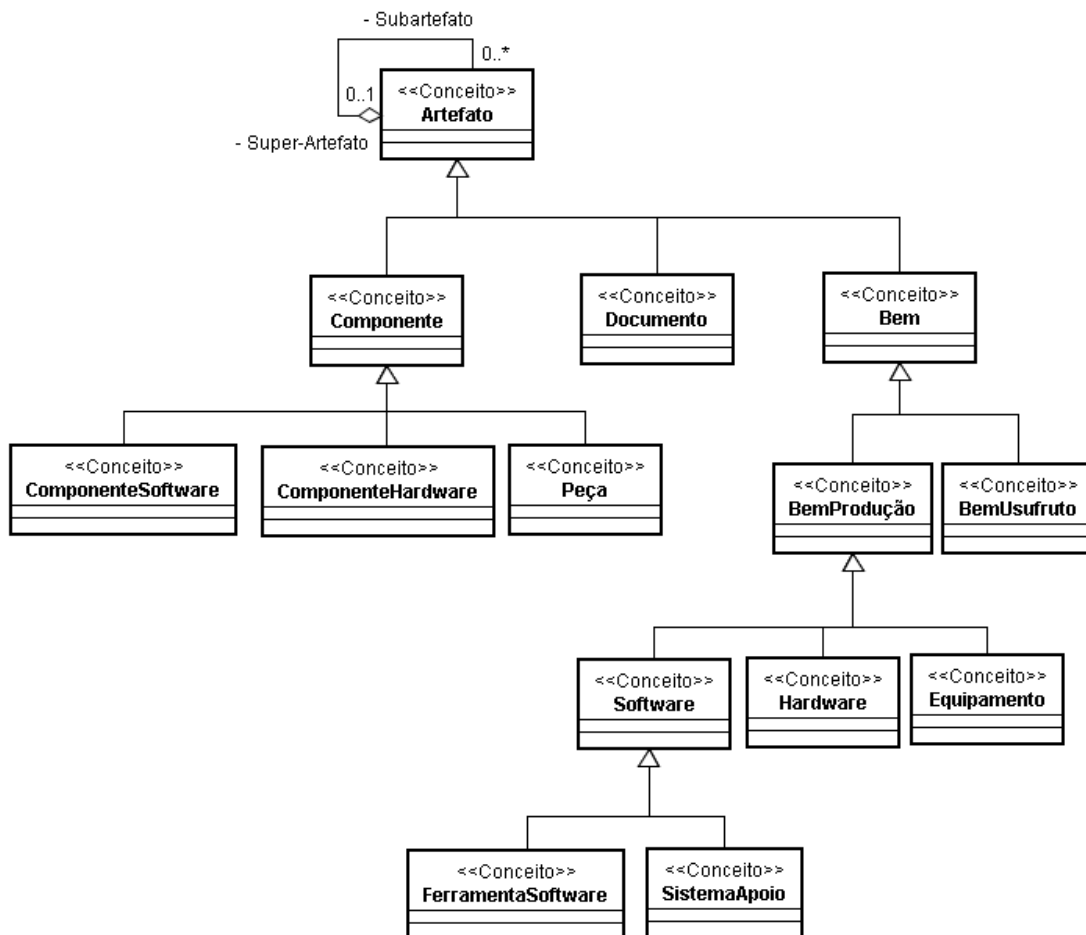


Figura 11 – Subontologia de Artefatos

2.3.1 Decomposição de Artefato

A relação de composição entre artefatos estabelece que um artefato, chamado *super-artefato*, pode ser decomposto em outros artefatos, chamados *subartefatos*. De forma similar à composição de domínio de conhecimento, foram definidos os predicados *subartefato(s₁,s₂)*, *superartefato(s₂,s₁)*, *artefatoelementar(s)* e *macroartefato(s)*.

2.3.2 Taxonomia de Artefato

A taxonomia de artefato define como um artefato pode ser classificado quanto a sua natureza, de acordo com o propósito da ontologia.

Documentos são artefatos escritos ou que podem ser impressos, cuja função é fornecer informação, conhecimento ou prova³. Exemplos de documentos são especificações de requisitos, a norma ISO/IEC 12207 (ISO/IEC, 2008), documentos de identificação pessoal, ordens de serviço e de pagamento.

Bens são artefatos concluídos, no sentido de não participarem da composição de outros artefatos, além de não serem documentos. Em outras palavras, bens são artefatos que já passaram por todas as transformações necessárias para que incorporassem suas funcionalidades e que, conseqüentemente, fazem parte do seu processo de produção. Bens podem ser classificados em bens de usufruto e bens de produção.

Bens de Usufruto são bens dos quais se pode usufruir as funcionalidades, que não são voltadas para a criação ou transformação de artefatos. Exemplos de bens de usufruto são jogos eletrônicos, aparelhos de som e carros.

Bens de Produção são bens cujas funcionalidades apóiam a criação ou a transformação de artefatos. Bens de produção podem ser classificados em software, hardware e equipamentos.

Software é um bem de produção representado por um conjunto de instruções e dados que, utilizado em conjunto com um hardware, é capaz de executar ou apoiar a execução de atividades. Artefatos de Software⁴ podem ser classificados em ferramentas de software e sistemas de apoio, que são definidos conforme proposto por FALBO (1998).

Ferramentas de Software são artefatos de software utilizados para (semi-)

³ Definição estabelecida com base na definição fornecida em (MICHAELIS, 1998).

⁴ O termo “Artefatos de software” é utilizado como plural do termo “Software”, assim como o termo “Artefatos de hardware” será utilizado como plural do termo “Hardware”.

automatizar procedimentos. Exemplos de ferramentas de software que (semi-) automatizam procedimentos de desenvolvimento e manutenção de software são ferramentas de programação, sistemas gerenciadores de bancos de dados, ferramentas para coleta automática de métricas, geradores de casos de teste e sistemas de controle de versões. Para organizações com outro tipo de negócio, os exemplos de ferramentas de software são sistemas de controle de caldeiras, ferramentas de CAD (*Design Assisted by Computer*) e sistemas de conta corrente. Além disso, existem as ferramentas de software de propósito geral, tais como editores de texto, editores de formulário e editores de figuras.

Sistemas de Apoio são artefatos de software que não (semi-) automatizam procedimentos, mas, por serem necessários à operação adequada do hardware ou de outros artefatos de software, são requeridos para a execução de atividades da organização. Exemplos de sistemas de apoio são sistemas operacionais e sistemas de gerenciamento de redes.

De volta aos tipos de bens de produção, **hardware** é um bem de produção representado por um computador, um de seus periféricos ou uma máquina qualquer operada com o auxílio de um software, e **equipamento** é um bem de produção, sem ser um software ou hardware, requerido para a execução de uma atividade, podendo ser uma máquina ou instrumento qualquer.

Componentes são artefatos que são utilizados na composição de outros artefatos, mas que não são documentos. Componentes podem ser classificados em componentes de software, componentes de hardware e peças.

Componentes de Software são constituídos por conjuntos de instruções e dados, podendo ser produzidos em uma atividade de desenvolvimento de software ou ser obtidos de uma biblioteca de componentes. Exemplos de componentes de software são funções, subprogramas, classes, *frameworks*, padrões gerativos.

Componentes de Hardware são componentes especificamente utilizados na composição de artefatos de hardware. Exemplos são: unidades de entrada e saída, circuitos integrados e discos.

Peças são componentes de propósito geral, que podem entrar na composição de equipamentos ou de artefatos de hardware. Exemplos são: parafusos, correias, manivelas e motores.

Para formalizar a taxonomia de artefato, foram definidos os seguintes predicados: **componente(s,t)**, **documento(s)**, **bem(s)**, **bemprodução(s)**, **bemusufruto(s)**, **software(s)**, **hardware(s)**, **equipamento(s)**, **ferramentasoft(s)**, **sistemaapoio(s)**, sendo que

componente(s,t) indica que *s* é um componente do tipo *t*, onde *t* assume um dos seguintes valores $\{CompSoftware, CompHardware, Peça\}$. Os demais foram definidos de forma similar à definição de predicados para a taxonomia de competência. Além disso, foram formalizados os seguintes axiomas:

- Bens são, por definição, macro-artefatos (axioma A11);
- Um componente só pode ser composto por outros componentes do mesmo tipo (axioma A12);
- Um documento só pode ser composto por outros documentos (axioma A13);
- Um software só pode ser composto de componentes de software ou de documentos (axioma A14);
- Um equipamento só pode ser composto de peças e documentos (axioma A15);
- Se um artefato é software e tem subartefatos, então um dos subartefatos é um componente de software (axioma A16);
- Se um artefato é hardware e tem subartefatos, então um dos subartefatos é um componente de hardware (axioma A17);
- Se um artefato é equipamento e tem subartefatos, então um dos subartefatos é uma peça (axioma A18).

$$(\forall s) (bem(s) \rightarrow macroartefato(s)) \quad (A11)$$

$$(\forall s_1, s_2, t_1, t_2) (componente(s_1, t_1) \wedge componente(s_2, t_2) \wedge subartefato(s_1, s_2) \rightarrow t_1=t_2) \quad (A12)$$

$$(\forall s_1, s_2) (documento(s_2) \wedge subartefato(s_1, s_2) \rightarrow documento(s_1)) \quad (A13)$$

$$(\forall s_1, s_2) (software(s_2) \wedge subartefato(s_1, s_2) \rightarrow componente(s_1, CompSoftware) \vee documento(s_1)) \quad (A14)$$

$$(\forall s_1, s_2) (equipamento(s_2) \wedge subartefato(s_1, s_2) \rightarrow componente(s_1, Peça) \vee documento(s_1)) \quad (A15)$$

$$(\forall s) (software(s) \wedge (\exists s_1, \dots, s_n) (subartefato(s_1, s) \wedge \dots \wedge subartefato(s_n, s)) \rightarrow (\exists s_k) (subartefato(s_k, s) \wedge componente(s_k, CompSoftware))) \quad (A16)$$

$$(\forall s) (hardware(s) \wedge (\exists s_1, \dots, s_n) (subartefato(s_1, s) \wedge \dots \wedge subartefato(s_n, s)) \rightarrow (\exists s_k) (subartefato(s_k, s) \wedge componente(s_k, CompHardware))) \quad (A17)$$

$$(\forall s) (equipamento(s) \wedge (\exists s_1, \dots, s_n) (subartefato(s_1, s) \wedge \dots \wedge subartefato(s_n, s)) \rightarrow (\exists s_k) (subartefato(s_k, s) \wedge componente(s_k, Peça))) \quad (A18)$$

2.4 Subontologia de Comportamento

A subontologia de Comportamento responde às questões de competência Q4.1 a Q4.8, Q6.2 e Q8.1 a Q8.4, estando fortemente baseada na ontologia definida por FALBO (1998), sendo que os conceitos foram, em geral, definidos de forma mais genérica para poderem ser utilizados para descrever o comportamento de diferentes organizações. Os aspectos abordados pela subontologia são: atividade como ação de transformação, taxonomia de atividade, decomposição de processo e de atividade, adoção de procedimento, taxonomia de procedimento, método como procedimento sistemático, automatização de procedimento, processos definidos na organização e normas associadas, além de projetos da organização.

2.4.1 Atividade como Ação de Transformação

Uma *atividade* é uma ação de transformação que pode requerer competências para a sua execução e, ao ser executada, fazer uso de bens de produção (recursos), consumir matérias-primas e artefatos de entrada (insumos), além de produzir artefatos de saída (produtos). A Figura 12 ilustra a parte da subontologia que captura o aspecto da atividade como ação de transformação.

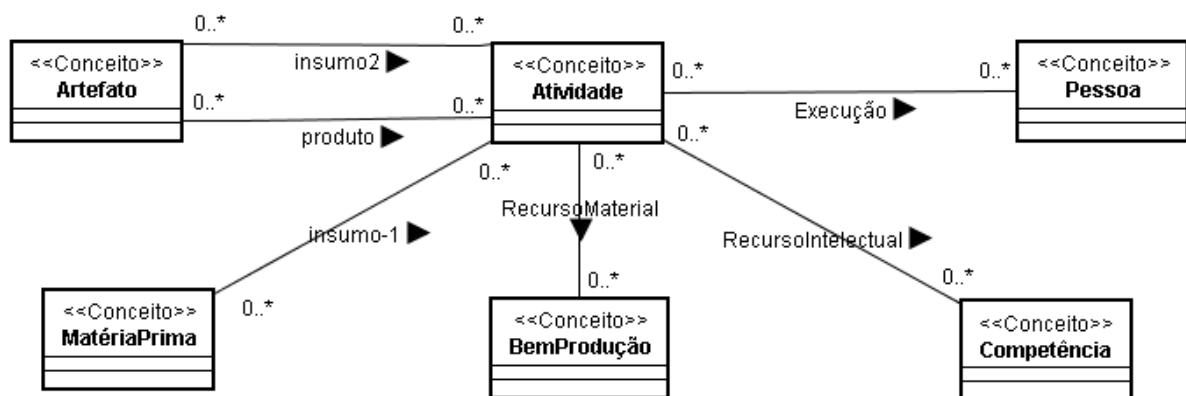


Figura 12 – Subontologia de Comportamento - Atividade como Ação de Transformação

Uma *matéria-prima* é uma matéria bruta ou pouco elaborada utilizada como insumo em uma atividade, no sentido de ser um objeto de transformação desta atividade⁵.

Para formalizar as relações ilustradas na Figura 12 foram definidos os predicados *insumo-1(m,t)*, que indica que a matéria-prima *m* é um insumo para a atividade *t*; *insumo-*

⁵ Definição estabelecida com base na definição fornecida em (MICHAELIS, 1998).

$2(s,t)$, que indica que o artefato s é um insumo para a atividade t ; $produto(s,t)$, que indica que o artefato s é um produto da atividade t ; $recursomaterial(b,t)$, que indica que o bem de produção b é um recurso material para a atividade t ; $recursointelectual(c,t)$, que indica que a competência c é um recurso intelectual para a atividade t ; além de $execução(p,t)$, que indica que a pessoa p é a responsável pela execução da atividade t .

Nenhuma restrição foi estabelecida para a relação de execução de atividades por pessoas, pois nem sempre é desejado estabelecer quem exatamente é responsável pela execução de uma atividade.

Uma vez definida a relação de competência requerida por uma atividade (*recurso intelectual*), foi possível especificar as competências de um agente em termos das competências requeridas nas atividades que são de sua responsabilidade, o que é formalizado através do seguinte axioma:

$$(\forall c,g) (competência_agente(c,g) \leftrightarrow (\exists t) (atribuição(t,g) \wedge recursointelectual(c,t)) \vee complementação(c,g)) \quad (A19)$$

Além disso, os axiomas abaixo foram formalizados para representar que uma posição, quando relacionada a apenas um cargo através da relação *possívelalocação*, herda as atividades e as competências requeridas para o cargo, podendo acrescentar competências e atividades específicas.

$$(\forall c,p) ((\exists r) (possívelalocação(p,r) \wedge competência_agente(c,r)) \rightarrow competência_agente(c,p)) \quad (A20)$$

$$(\forall p,t) ((\exists r) (possívelalocação(p,r) \wedge atribuição(t,r)) \rightarrow atribuição(t,p)) \quad (A21)$$

2.4.2 Taxonomia de Atividade

A taxonomia de atividade define, de acordo com o propósito da ontologia, como uma atividade pode ser classificada quanto ao papel que desempenha em relação à missão da organização (Figura 13) e quanto a sua natureza (Figura 14).

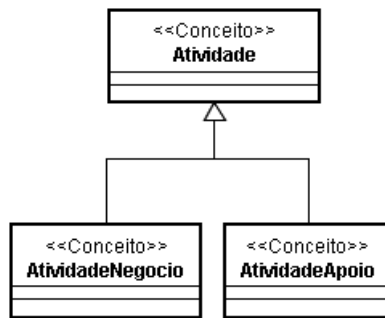


Figura 13 – Taxonomia de Atividade quanto ao Papel

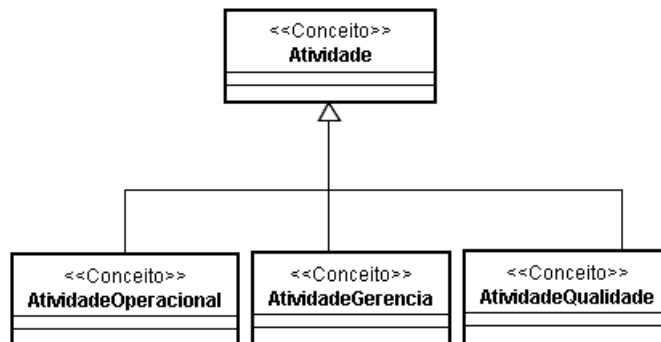


Figura 14 – Taxonomia de Atividade quanto à Natureza

Atividades de negócio são atividades essenciais para que a organização cumpra a sua missão e ofereça os artefatos ou serviços a que se propõe. Estas atividades são também chamadas de atividades-fins ou atividades de linha (CHIAVENATO, 1998c).

Atividades de apoio são atividades que buscam oferecer melhores condições para a execução das atividades de negócio. Estas atividades são também chamadas de atividades-meio ou atividades de *staff* (CHIAVENATO, 1998c).

Atividades de gerência são atividades de negócio ou de apoio relacionadas ao planejamento, organização, coordenação e acompanhamento gerencial de outras atividades.

Atividade de avaliação da qualidade são atividades de apoio relacionadas com a garantia da qualidade dos artefatos ou serviços oferecidos pela organização ou dos processos utilizados para oferecer estes artefatos ou serviços.

Atividades operacionais são atividades de negócio ou de apoio responsáveis pelo funcionamento da organização, não sendo, portanto, atividades relacionadas com a garantia da qualidade ou com gerência.

De forma similar à adotada na definição de predicados para a taxonomia de competência, foram definidos os predicados *atvnegocio(t)*, *atvapoio(t)*, *atvgerencia(t)*,

$atvqualidade(t)$ e $atvoperacional(t)$ para formalizar os tipos de atividade. O seguinte axioma de consolidação reflete o fato de uma atividade de avaliação da qualidade ser, por definição, uma atividade de apoio.

$$(\forall t) (atvqualidade(t) \rightarrow atvapoio(t)) \quad (A22)$$

2.4.3 Decomposição de Processo e Atividade

Um *processo* é um conjunto de atividades estruturadas e destinadas a resultar em um artefato ou serviço de valor para a organização ou para um determinado cliente ou mercado. Implica em uma ordenação específica das atividades com começo, fim, insumos e produtos claramente identificados. A Figura 15 ilustra como processos e atividades são decompostos.

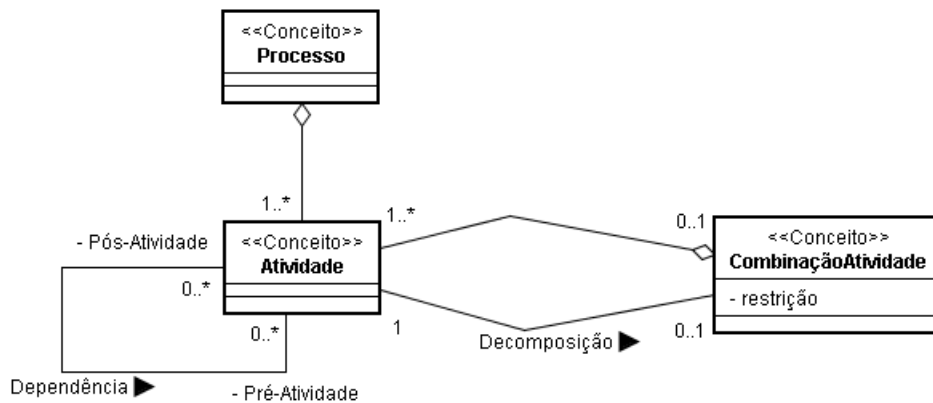


Figura 15 – Decomposição de Processo e de Atividade

Uma *combinação de atividade* é um agrupamento de atividades que permite representar a decomposição de uma atividade em um conjunto de atividades e, inclusive, estabelecer uma possível restrição a ser considerada na execução das atividades agrupadas. Uma restrição pode determinar que as atividades que compõem a combinação devem ser todas obrigatoriamente executadas ou que elas são mutuamente exclusivas.

Uma atividade é dita uma *pré-atividade* de outra atividade, se precisa ser executada para que a outra, chamada de *pós-atividade*, também o seja.

Para formalizar as relações ilustradas na Figura 15, foram definidos os predicados $preatividade(t_1, t_2)$ e $posatividade(t_2, t_1)$, indicando que t_1 é uma pré-atividade de (antecede) t_2 ou, de forma inversa, que t_2 é uma pós-atividade de (sucede) t_1 . Também foram definidos os predicados $combinação_atividade(c, r)$, que indica que c é uma

combinação de atividades que obedece à restrição r , onde r assume um dos seguintes valores $\{Obrigatoriedade, Exclusividade, Nenhuma\}$; ***decomposição(t,c)***, que indica que a atividade t é uma atividade definida por uma combinação de atividades c ; e, finalmente, ***combatv_composição(t,c)***, que indica que a atividade t faz parte da combinação de atividades c .

O axioma a seguir estabelece como o predicado ***preatividade(t₁,t₂)*** é definido em termos dos predicados que representam os insumos e produtos dessas atividades.

$$(\forall t_1, t_2) (\text{preatividade}(t_1, t_2) \leftrightarrow (\exists s) (\text{produto}(s, t_1) \wedge (\text{insumo-1}(s, t_2) \vee \text{insumo-2}(s, t_2)))) \quad (\text{A23})$$

Também foram definidos os conceitos de ***super-atividade*** e ***subatividade***, onde uma super-atividade é uma atividade que não pode ser executada diretamente, isto é, ela é composta de outras atividades menores e sua execução se dá, na realidade, através da execução dessas subatividades. A partir destes conceitos, foram definidos os predicados ***subatividade(t₁,t₂)***, ***superatividade(t₂,t₁)***, ***macroatividade(t)*** e ***atividadeelementar(t)***, os axioma de A24 a A27.

$$(\forall t_1, t_2) (\text{subatividade}(t_1, t_2) \leftrightarrow (\exists c) (\text{combinação_atividade}(c, *) \wedge \text{decomposição}(t_2, c) \wedge \text{combatv_composição}(t_1, c))) \quad (\text{A24})$$

onde o asterisco (*) indica que não importa o valor atribuído para este argumento⁶.

Os insumos e produtos de uma atividade composta são estabelecidos em termos dos insumos e produtos de suas subatividades, sendo obtidos através dos seguintes axiomas:

$$(\forall t, t_1, \dots, t_n, s) (\text{subatividade}(t_1, t) \wedge \dots \wedge \text{subatividade}(t_n, t) \wedge \text{insumo-1}(s, t_i) \wedge ((\neg \exists t_k) \text{produto}(s, t_k)) \rightarrow \text{insumo-1}(s, t)) \quad (\text{A25})$$

$$(\forall t, t_1, \dots, t_n, s) (\text{subatividade}(t_1, t) \wedge \dots \wedge \text{subatividade}(t_n, t) \wedge \text{insumo-2}(s, t_i) \wedge ((\neg \exists t_k) \text{produto}(s, t_k)) \rightarrow \text{insumo-2}(s, t)) \quad (\text{A26})$$

$$(\forall t, t_1, \dots, t_n, s) (\text{subatividade}(t_1, t) \wedge \dots \wedge \text{subatividade}(t_n, t) \wedge \text{produto}(s, t_i) \wedge ((\neg \exists t_k) (\text{insumo-2}(s, t_k)) \rightarrow \text{produto}(s, t)) \quad (\text{A27})$$

onde t_i e $t_k \in \{t_1, t_2, \dots, t_n\}$.

Por fim, o axioma abaixo estabelece a dependência entre os recursos de uma subatividade e de sua super-atividade.

⁶ Sempre que, em um axioma, um dos argumentos de um predicado puder assumir um valor arbitrário, este argumento será representado com um asterisco (*).

$$(\forall t_1, t, b) (\text{recursomaterial}(b, t_1) \wedge \text{subatividade}(t_1, t) \rightarrow \text{recursomaterial}(b, t)) \quad (\text{A28})$$

$$(\forall t_1, t, c) (\text{recursointelectual}(c, t_1) \wedge \text{subatividade}(t_1, t) \rightarrow \text{recursointelectual}(c, t)) \quad (\text{A29})$$

2.4.4 Subtipos de Processo

Uma importante classe de processo a ser identificada e definida é a de *processos organizacionais* (ou *processos de negócio*), que são aqueles processos utilizados por uma organização para formalizar e sistematizar as atividades que devem ser executadas para garantir que a organização se mantenha econômica e eficiente. Dentre estes processos, destacam-se os *processos de software*, que são aqueles processos utilizados por uma organização ou projeto para planejar, gerenciar, executar, controlar e monitorar suas atividades relacionadas a software (ISO/IEC-15504, 2003). A relação entre *processo* e seus subtipos pode ser vista através da Figura 16:

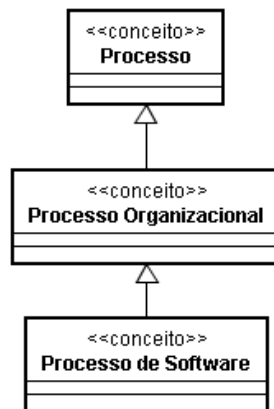


Figura 16 – Subtipos de processos segundo a natureza do processo

De acordo com o seu nível de abstração um *processo* pode ser um *processo padrão*, *processo especializado* ou *processo definido para o projeto*.

Um *processo padrão* é um conjunto de definições de processos básicos que guiam todos os processos na organização (ISO/IEC, 2003), geralmente são definidos num nível organizacional e contém os ativos de processo básicos que devem fazer parte de qualquer processo de projeto de uma organização (BERTOLLO, 2006). Essas definições de processos cobrem os elementos de processo fundamentais (e os seus inter-relacionamentos) que devem ser incorporados dentro dos processos definidos que são implementados nos projetos pela organização. Um processo padrão estabelece consistência

entre as atividades através da organização e é desejável para estabilidade e melhoria de longo prazo. O conjunto de processos-padrão da organização descreve os elementos de processo fundamentais que será parte dos processos definidos para o projeto. Também descreve os relacionamentos (por exemplo: seqüência e interfaces) entre esses elementos do processo (SOFTEX, 2007).

Um *processo especializado* é um processo padrão adaptado a partir de outro processo padrão ao detalhar algumas das atividades do processo base com o objetivo de detalhar aspectos específicos seguindo um critério pré-determinado. Tais critérios podem incluir, mas não se limitam a detalhes específicos do paradigma, métodos de desenvolvimento, características do desenvolvimento de software na organização, como experiência dos desenvolvedores e tipos de software desenvolvidos, ou outras particularidades a critério da organização, como características de tamanho ou distribuição geográfica dos projetos a que se aplicam (adaptado de (VILLELA *et al.*, 2004) e (BERTOLLO, 2006)). Todos os ativos de processo definidos para o processo padrão tornam-se parte do processo especializado, contudo, novos ativos podem ser incluídos (BERTOLLO, 2006)⁷.

Exemplos de paradigma são o estruturado, o orientado a objetos e o baseado em conhecimento. Os processos especializados são importantes para apoiar o desenvolvimento e a manutenção de software na organização, refletindo o conhecimento e a experiência obtidos sobre o uso do paradigma e dos métodos. A depender do tipo de software, pode haver a necessidade de incluir atividades específicas para o tipo de software e/ou de especializar atividades já previstas (VILLELA *et al.*, 2004).

Um *processo definido para o projeto* (também chamado de *processo instanciado* ou, simplesmente, *processo definido*) é um processo que é gerenciado (planejado, monitorado e ajustado) e é adaptado de um conjunto de processos padrão de acordo com os guias de adaptação da organização (ISO/IEC-15504, 2003) e considerando as particularidades do projeto. Pode ou não ser definido a partir de uma adaptação de um **processo padrão**, neste caso será composto pelos ativos do processo padrão e novos ativos podem ser incluídos considerando características do projeto, como complexidade, tamanho, experiência da equipe etc. (BERTOLLO, 2006).

⁷ BERTOLLO (2006) define conceitos na ontologia de Processo de Software específicos a critérios utilizados durante a especialização de processos. Nesta revisão da ontologia tais conceitos não foram incluídos, pois a definição utilizada deixa em aberto os tipos de critérios a serem utilizados para a especialização.

Outros axiomas não serão descritos formalmente por serem derivados diretamente do modelo que pode ser visto na Figura 17. De forma geral *todo processo padrão é um processo*, *todo processo especializado é um processo*, *todo processo definido é um processo* e *todo processo especializado é um processo padrão*. Além disso, *todo processo especializado é originado de um processo padrão ou de outro processo especializado* e *um processo definido pode ser originado de um processo especializado ou de um processo padrão*. Todas as demais relações entre **processo** e os outros conceitos da ontologia continuam válidos, exceto a relação original entre **processo** e **projeto** que foi transformada numa relação específica entre **processo definido para o projeto** e **projeto**. Dessa forma, *um projeto pode ser executado com base num processo definido para o projeto*.

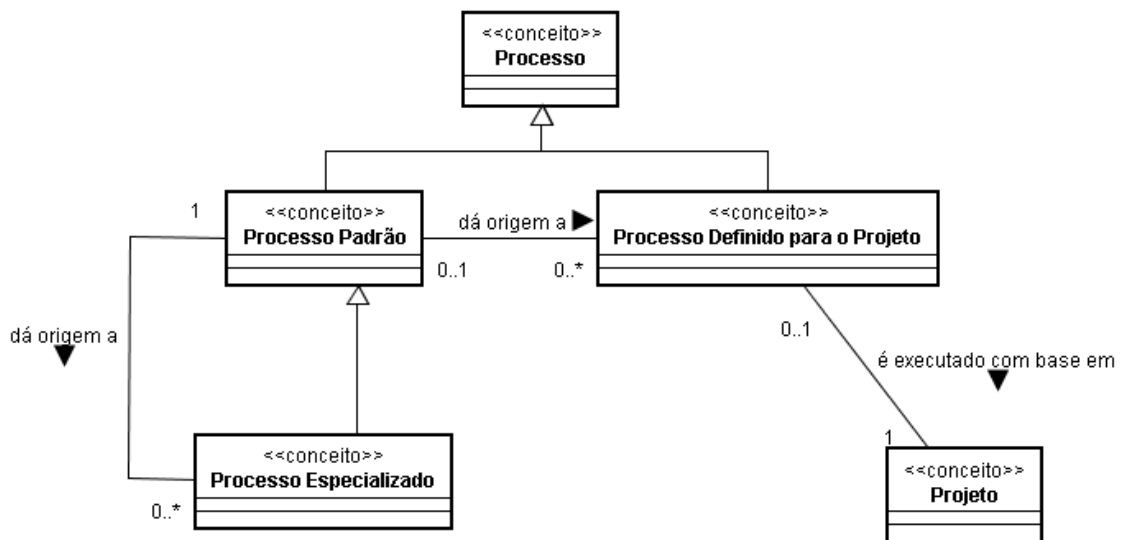


Figura 17 – Conceitos derivados de Processo

2.4.5 Adoção de Procedimento

Procedimentos são condutas bem estabelecidas e ordenadas para a execução de atividades (FALBO, 1998). O aspecto da adoção de procedimento é ilustrado na Figura 21.

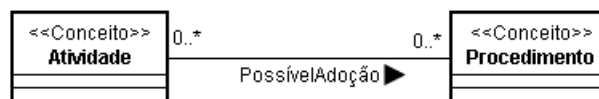


Figura 18 – Adoção de Procedimento

Para formalizar o aspecto da adoção de procedimento, foi definido o predicado *possiveladoção(p,t)*, que indica que o procedimento *p* pode ser adotado para a execução da atividade *t*.

2.4.6 Taxonomia de Procedimento

A taxonomia de procedimento (Figura 19) define como um procedimento pode ser classificado quanto a sua natureza, de acordo com o propósito da ontologia.

Métodos são procedimentos sistemáticos, definindo passos (subatividades) e heurísticas para a execução de uma ou mais atividades. Métodos podem ser classificados, de acordo com o tipo de atividades que podem apoiar, em *métodos de gerência*, *métodos de avaliação da qualidade* e *métodos operacionais*.

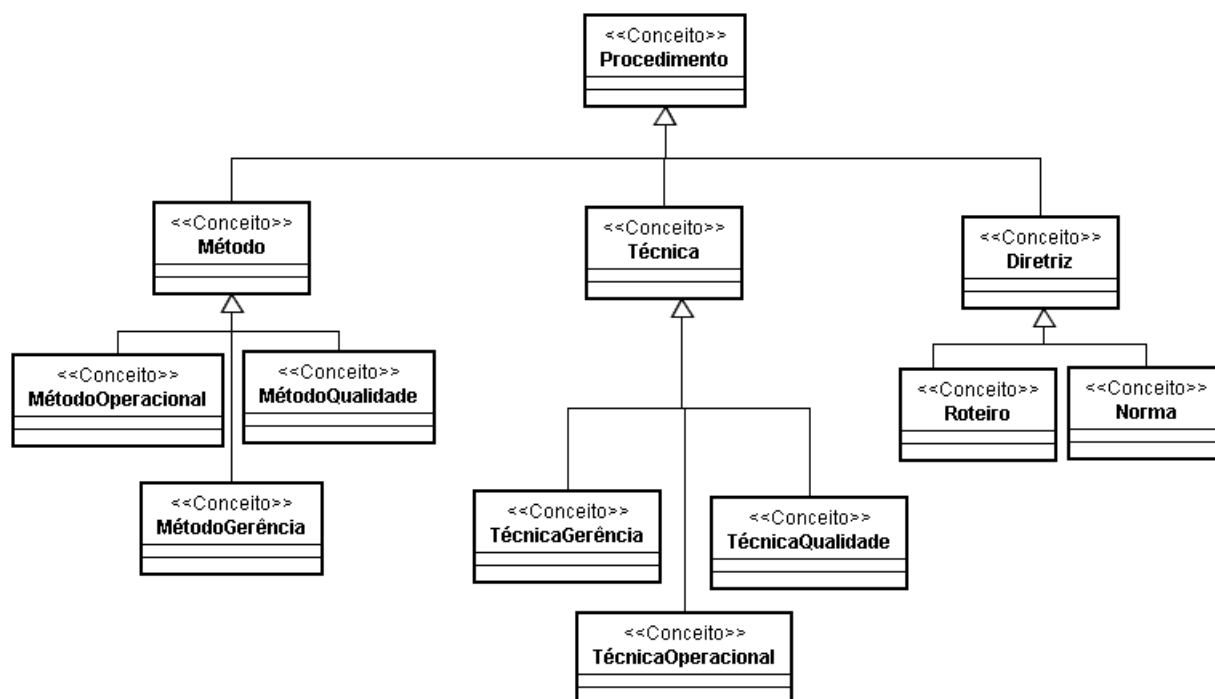


Figura 19 – Taxonomia de Procedimento

Técnicas são procedimentos que descrevem apenas aspectos gerais para a execução de uma atividade, não descrevendo como executá-la em termos de subatividades. Técnicas podem ser classificadas, de acordo com o tipo de atividades que podem apoiar, em *técnicas de gerência*, *técnicas de avaliação da qualidade* e *técnicas operacionais*.

Diretriz são procedimentos que visam estabelecer um padrão para a execução de atividades. Diretrizes podem ser classificadas em roteiros e normas.

Roteiros são diretrizes para a elaboração de documentos, estabelecendo os principais tópicos a serem abordados. Roteiros, portanto, só podem ser utilizados em atividades que produzam documentos.

Normas são diretrizes que visam estabelecer padrões para a execução ou avaliação de atividades ou processos, o que inclui a elaboração de documentos.

Para formalizar o conceito de procedimento e seus vários tipos, foram definidos os seguintes predicados: *procedimento(p)*, *método(m)*, *técnica(t)*, *diretriz(d)*, *metgerência(m)*, *metqualidade(m)*, *metoperacional(m)*, *tecgerência(t)*, *tecqualidade(t)*, *tecoperacional(t)*, *roteiro(r)* e *norma(n)*.

Como pode ser observado, há uma forte relação entre tipos de procedimentos e tipos de atividades. Roteiros só podem ser adotados por atividades que produzam documentos (axioma A30); métodos e técnicas operacionais, por atividades operacionais (axioma A31); métodos e técnicas de gerência, por atividades de gerência (axioma A32); e métodos e técnicas de avaliação da qualidade, por atividades de avaliação da qualidade (axioma A33).

$$(\forall r, t) (\text{roteiro}(r) \wedge \text{possíveladoção}(r, t) \rightarrow (\exists s) (\text{documento}(s) \wedge \text{produto}(s, t))) \quad (\text{A30})$$

$$(\forall p, t) ((\text{metoperacional}(p) \vee \text{tecoperacional}(p)) \wedge \text{possíveladoção}(p, t) \rightarrow \text{atvoperacional}(t)) \quad (\text{A31})$$

$$(\forall p, t) ((\text{metgerência}(p) \vee \text{tecgerência}(p)) \wedge \text{possíveladoção}(p, t) \rightarrow \text{atvgerência}(t)) \quad (\text{A32})$$

$$(\forall p, t) ((\text{metqualidade}(p) \vee \text{tecqualidade}(p)) \wedge \text{possíveladoção}(p, t) \rightarrow \text{atvqualidade}(t)) \quad (\text{A33})$$

2.4.7 Método como Procedimento Sistemático

Um método, como procedimento sistemático, descreve uma ou mais atividades em termos de outras atividades, o que está representado na Figura 20.

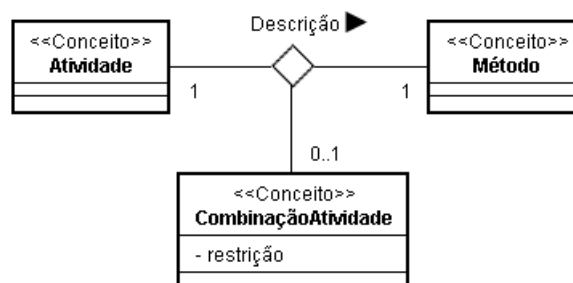


Figura 20 – Método como Procedimento Sistemático

Para formalizar o modelo da Figura 20, foi definido o predicado *descrição(m,t,c)*, indicando que o método *m* descreve a atividade *t* através da combinação de atividades *c*. Além disso, os seguintes axiomas de consolidação devem ser observados:

$$(\forall c,m,t) (descrição(m,t,c) \rightarrow possíveladoção(m,t) \wedge combinação_atividade(c,*)) \quad (A34)$$

$$(\forall c,m,t,t_1) (combatv_composição(t_1,c) \wedge descrição(m,t,c) \rightarrow t_1 \neq t \wedge \neg superatividade(t_1,t)) \quad (A35)$$

Sendo assim, um método não pode ser adotado por atividades elementares, assim como técnicas só podem ser adotadas por atividades elementares, uma vez que não descrevem como a atividade deve ser decomposta. Os axiomas, no entanto, devem considerar que o propósito da ontologia é descrever e esta descrição pode estar incompleta, ou seja, um método pode não estar especificado. Os axiomas a seguir refletem estas considerações.

$$(\forall m,a) (método(m) \wedge (\exists c) descrição(m,a,c) \rightarrow \neg atividadeelementar(a)) \quad (A36)$$

$$(\forall a,t) (técnica(t) \wedge possíveladoção(t,a) \rightarrow atividadeelementar(a)) \quad (A37)$$

2.4.8 Automatização de Procedimento

A automatização de procedimento é o aspecto tratado na Figura 21.

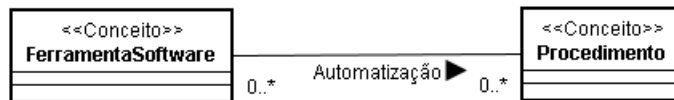


Figura 21 – Automatização de Procedimento

Para formalizar o aspecto da automatização de procedimento, foi definido o predicado *automatização(f,p)*, que indica que a ferramenta de software *f* pode ser utilizada para (semi-)automatizar o procedimento *p*. O axioma de consolidação abaixo também foi definido para formalizar a restrição de que uma ferramenta de software só pode ser requerida por uma atividade, se for capaz de (semi-)automatizar um procedimento passível de adoção pela atividade.

$$(\forall t,f) ((ferramentasoft(f) \wedge recursomaterial(f,t)) \rightarrow (\exists p) (possíveladoção(p,t) \wedge automatização(f,p))) \quad (A38)$$

2.4.9 Processos definidos na Organização e Normas Associadas

Os processos definidos na organização e normas associadas são os aspectos modelados na Figura 22.

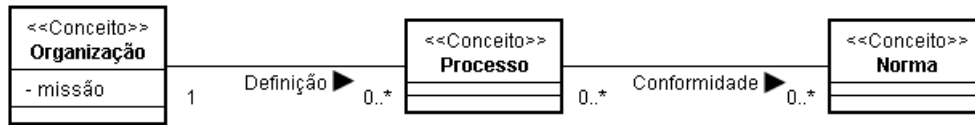


Figura 22 – Processos definidos na Organização e Normas Associadas

Para formalizar as relações da Figura 22, foram definidos os predicados *definição(p,r)*, indicando que *p* é um processo da organização *r*; *conformidade(p,n)*, indicando que o processo *p* está em conformidade com a norma *n*.

2.4.10 Projetos da Organização

Os projetos da organização são o último aspecto tratado nesta subontologia (Figura 23).

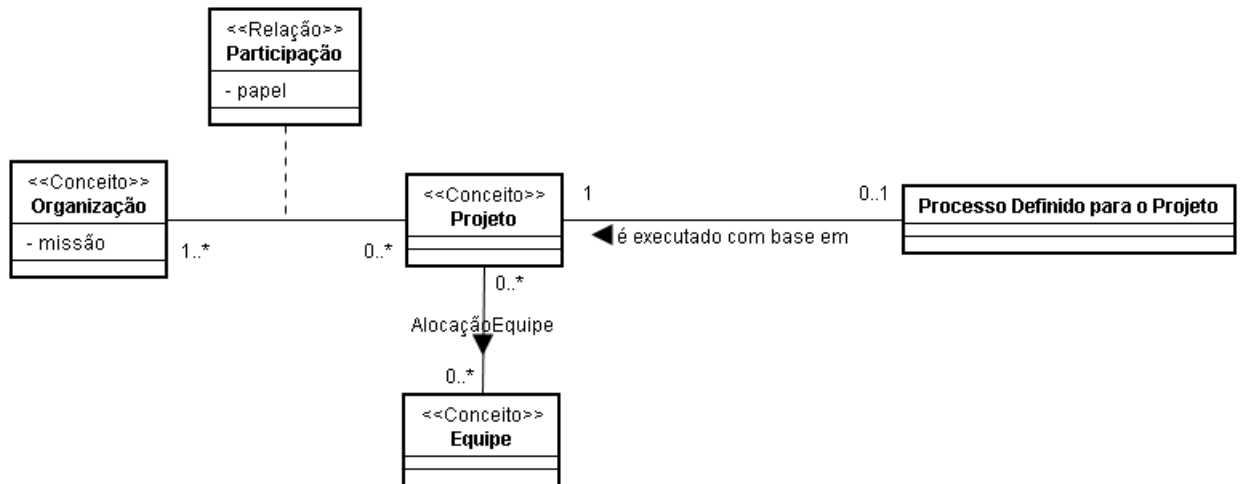


Figura 23 – Projetos da Organização

Projetos são empreendimentos que são conduzidos de acordo com determinado esquema, normalmente expresso em um plano do projeto, que define, entre outros, o processo para execução do projeto e a equipe ou as equipes de projeto.

Para formalizar as relações ilustradas na Figura 23, foram definidos os predicados *participação(r,p,l)*, que indica que a organização *r* participa do projeto *p* exercendo o papel *l*, onde *l* assume um dos seguintes valores {*Cliente, OrgSoftware, Parceiro, Fornecedor*},

alocação_equipe(q,p), que indica que a equipe *q* é responsável pela execução de atividades no (ou seja, está alocada ao) projeto *p*; e *execução(p,c)*, que indica que o projeto *p* é executado segundo o processo *c*.

Uma vez definido o predicado *alocação_equipe(q,p)*, o axioma de consolidação que reflete o conceito de comissão pôde ser, enfim, definido.

$$(\forall q) (equipe(q) \wedge (\neg \exists p) \text{alocação_equipe}(q,p) \leftrightarrow comissão(q)) \quad (\text{A39})$$

2.4.11 Ativos de Processo

Um *ativo de processo* pode ser definido como algo que a organização considere útil para atingir os objetivos do processo, por exemplo, políticas, processos definidos, lições aprendidas, roteiros (*templates*) de documentos, padrões, material de treinamento (CHRISSIS *et al.*, 2006). A identificação de um ativo de processo é dependente do contexto em que está inserido, pois um mesmo item (por exemplo, um plano de projeto ou manual) pode ser considerado ativo de processo para uma organização e não para outra. Em geral, a definição de um ativo de processo se refere a um item material relacionado a um processo, dessa forma, uma pessoa não poderia ser considerada um ativo de processo, mas uma vez que um conhecimento desta pessoa estiver externalizado e explícito tal conhecimento poderia ser considerado um ativo de processo.

Este termo foi definido na ontologia na forma de um conceito derivado expresso por meio de axiomas que indicam quando determinadas entidades da ontologia devem ser consideradas ativos de processo⁸. Dentre os conceitos definidos na ontologia de Organização foram identificados cinco que podem ser considerados *ativos de processo*: *processo*, *documento*, *ferramenta de software*, *atividade* e *procedimento*. Assim, são considerados ativos de processo *todo processo de uma organização, toda atividade que faça parte de um processo, toda ferramenta que seja recurso material para a execução de uma atividade que faça parte de um processo, todo documento que seja insumo de uma atividade que faça*

⁸ Outra possibilidade para a definição de *ativo de processo* seria modelá-lo como um papel desempenhado por outros conceitos da ontologia. GUIZZARDI (2005) apresenta um padrão para esta modelagem, entretanto, considerou-se que a modelagem dessa forma tornaria a formalização da ontologia mais complexa. Dessa forma, visando à simplificação da modelagem, e esperando que com isso haja uma maior facilidade de disseminação de seu conhecimento, a modelagem de *ativo de processo* como conceito derivado foi escolhida.

2.5.1 Domínios de Atuação

O predicado *atuação(r,d)* foi definido para formalizar a relação que descreve os domínios de atuação de uma organização. Esta relação indica que a organização *r* atua no domínio de conhecimento *d*, ou seja, que possui capital intelectual relativo ao domínio de conhecimento e executa atividades que requerem conhecimentos que pertencem a este domínio.

2.5.2 Artefatos e Serviços oferecidos pela Organização

Também é abordado o aspecto dos artefatos/serviços oferecidos pela organização. Para formalizar as relações envolvidas, foram definidos o predicado *prestação(r,s)*, que indica que a organização *r* presta o serviço *s* para terceiros, e o predicado *fornecimento(r,s)*, que indica que a organização *r* fornece o artefato *s* para terceiros.

Um *serviço* é oferecido para satisfazer a necessidade ou desejo de um determinado cliente ou mercado, sendo não tangível e abstrato, em contraste com o artefato, que é um elemento tangível.

2.5.3 Relação com Organizações Clientes

A subontologia também descreve as relações da organização com suas organizações clientes.

Um *acordo comercial* é um instrumento que estabelece uma relação de negócio entre duas organizações, que atuam, respectivamente, como fornecedora e cliente.

Para formalizar as relações, foram definidos os predicados: *envolvimento_cliente(c,r)*, que indica que a organização *c* participa do acordo comercial *r* como organização cliente; *envolvimento_fornec(f,r)*, que indica que a organização *f* participa do acordo comercial *r* como fornecedora de artefatos e/ou serviços; *objeto-1(s,r)*, que indica que o serviço *s* é objeto do acordo comercial *r*, sendo um serviço prestado à organização cliente; e *objeto-2(s,r)*, que indica que o artefato *s* é objeto do acordo comercial *r*, sendo fornecido à organização cliente.

Os axiomas abaixo completam a formalização.

$$(\forall c, f)(\text{cliente}(c, f) \leftrightarrow (\exists r) (\text{envolvimento_cliente}(c, r) \wedge \text{envolvimento_fornec}(f, r))) \quad (\text{A45})$$

$$(\forall f, r, s) ((\text{envolvimento_fornec}(f, r) \wedge \text{objeto-1}(s, r)) \rightarrow \text{prestação}(f, s)) \quad (\text{A46})$$

$$(\forall f, r, s) ((\text{envolvimento_fornec}(f, r) \wedge \text{objeto-2}(s, r)) \rightarrow \text{fornecimento}(f, s)) \quad (\text{A47})$$

$$(\forall c, r, s) ((\text{envolvimento_cliente}(c, r) \wedge \text{objeto-1}(s, r)) \rightarrow \text{demanda_serviço}(c, s)) \quad (\text{A48})$$

$$(\forall c, r, s) ((\text{envolvimento_cliente}(c, r) \wedge \text{objeto-2}(s, r)) \rightarrow \text{demanda_artefato}(c, s)) \quad (\text{A49})$$

Referências Bibliográficas

- BERTOLLO, G., 2006, *Definição de Processos em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.
- CHIAVENATO, I., 1998a, "Administração por Objetivos (APO)", *Introdução à Teoria Geral da Administração*, 5a ed., capítulo 10, Editora Campus.
- CHIAVENATO, I., 1998b, "Decorrências da Abordagem Neoclássica: Departamentalização", *Introdução à Teoria Geral da Administração*, 5a ed., capítulo 9, Editora Campus.
- CHIAVENATO, I., 1998c, "Decorrências da Abordagem Neoclássica: Tipos de Organização", *Introdução à Teoria Geral da Administração*, 5a ed., capítulo 8, Editora Campus.
- CHRISIS, M.B., KONRAD, M., SHRUM, S., 2006, *CMMI (Second Edition): Guidelines for Process Integration and Product Improvement*, Addison Wesley Professional.
- COTA, R., 2002, *Modelagem Computacional para Gestão Empresarial*, Dissertação de M. Sc., Universidade Federal do Espírito Santo, Vitória, Brasil.
- FADEL, F.G., FOX, M.S., GRUNINGER, M., 1994, "A Generic Enterprise Resource Ontology". In: *Proceedings of the Third Workshop on Enabling Technologies – Infrastructures for Collaborative Enterprises*, West Virginia University, USA.
- FALBO, R., 1998, *Integração de Conhecimento em um Ambiente de Desenvolvimento de Software*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- FOX, M.S., BARBUCEANU, M., GRUNINGER, M., 1996, "An Organisation Ontology for Enterprise Modelling: Preliminary Concepts for Linking Structure and Behaviour", *Computers in Industry*, v. 29 (1996), pp. 123-134.

- FOX, M.S., CHIONGLO, J.F., FADEL, F.G., 1993, "A Common-Sense Model of the Enterprise". In: *Proceedings of the 2nd Industrial Engineering Research Conference*, pp. 425-429, Norcross, USA.
- FOX, M.S., GRUNINGER, M., 1998, "Enterprise Modelling", *AI Magazine*, v. 9, n. 3 (Fall 1998), pp. 109-121.
- GRUNINGER, M., FOX, M.S., 1994, "An Activity Ontology for Enterprise Modelling". In: *Proceedings of the Third Workshop on Enabling Technologies - Infrastructures for Collaborative Enterprises*, West Virginia University, USA.
- GUIZZARDI, G., 2005, *Ontological Foundations for Structural Conceptual Models*, PhD Thesis, Departamento de Informática, Centre for Telematics and Information Technology, Enschede, The Netherlands.
- HOLANDA, A.B., 1999, *Novo Aurélio Século XXI - O Dicionário da Língua Portuguesa*, 5a ed., Nova Fronteira.
- ISO/IEC-15504, 2003, "Information Technology – Software Process Assessment", *Parts 1-9, The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 15504.
- ISO/IEC, 2003, "Information Technology – Software Process Assessment", *Parts 1-9, The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 15504.
- ISO/IEC, 2008, "Systems and software engineering – Software life cycle processes", *The International Organization for Standardization and the International Electrotechnical Commission*, v. ISO/IEC 12207:2008.
- MEGGINSON, L., MOSLEY, D., PIETRI, P., 1986a, "Estabelecimento de Objetivos e Metas Organizacionais", *Administração: Conceitos e Aplicações*, 5a ed., capítulo 6, Editora Harbra.
- MEGGINSON, L., MOSLEY, D., PIETRI, P., 1986b, "Fundamentos do Modelo Organizacional", *Administração: Conceitos e Aplicações*, 5a ed., capítulo 8, Editora Harbra.
- MIAN, P.G., 2003, *ODEd: Uma Ferramenta de Apoio ao Desenvolvimento de Ontologias em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.
- MICHAELIS, 1998, *Michaelis Moderno Dicionário da Língua Portuguesa*, Editora Melhoramentos.

- RUY, F.B., 2006, *Semântica em um Ambiente de Desenvolvimento de Software*, Dissertação de M. Sc., Departamento de Informática, UFES, Vitória, Brasil.
- SOFTEX, 2007, "MPS.BR – Guia Geral".
- USCHOLD, M., GRUNINGER, M., 1996, "Ontologies: principles, methods and applications", *The Knowledge Engineering Review*, v. 11, n. 2 (1996), pp. 93-136.
- USCHOLD, M., KING, M., MORALEE, S., *et al.*, 1998, "The Enterprise Ontology", *The Knowledge Engineering Review, Special Issue on Putting Ontologies to Use (eds. Mike Uschold e Austin Tate)*, v. 13, n. 1 (1998).
- VILLELA, K., 2004, *Definição e Construção de Ambientes de Desenvolvimento de Software Orientados à Organização*, Tese de D. Sc., COPPE, UFRJ, Rio de Janeiro, Brasil.
- VILLELA, K., SANTOS, G., MONTONI, M., *et al.*, 2004, "Definição de Processos em Ambientes de Desenvolvimento de Software Orientados a Organização". In: *III Simpósio Brasileiro de Qualidade de Software - SBQS 2004*, pp. 4-18, Brasília - DF.

ANEXO V - ESPECIFICAÇÃO DE REQUISITOS

Este anexo apresenta com mais detalhes o levantamento de requisitos inicial realizado decorrente das necessidades de implementação desta tese. O objetivo deste levantamento é analisar as necessidades do cliente e os modelos dos processos de negócio que originaram as necessidades de implementação referentes à criação dos Ambientes de Engenharia de Software Orientados a Corporação. 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 negócio. Parte deste levantamento foi descrito no Capítulo V desta tese. O conteúdo deste anexo não representa toda a especificação e modelagem elaborada para a implementação. Outros diagramas e documentos foram gerados dentro do contexto do processo de desenvolvimento em uso na Estação Taba, mas não serão apresentados.

1 Necessidades do Cliente

A necessidade- A definição dos Ambientes de Engenharia de Software Orientados a Corporação depende da construção de uma infra-estrutura básica que contemple a definição de ambientes e de suas funcionalidades básicas, incluindo os processos a serem utilizados ao longo das iniciativas de melhoria de processos de software das corporações e organizações. A necessidade deste trabalho de doutorado é implementar tal infra-estrutura básica.

Afeta - Usuários da Estação Taba, incluindo equipe de implementadores da consultoria e usuários em corporações e organizações em que as ferramentas poderão ser utilizadas.

O seu impacto é - A alteração na forma com o que as iniciativas de melhoria de processos de software serão tratadas nas corporações e organizações.

Benefícios com a solução - Aumentar o controle e eficiência das iniciativas de melhoria de processos de software em corporações e organizações.

2 Modelo de Domínio

Para auxiliar o entendimento dos itens que compõem este levantamento de requisitos, foram gerados modelos de domínio para cada um dos componentes identificados.

2.1 Gerência de Ambientes

O modelo de domínio associado com a Gerência de Ambientes é exibido na Figura 1.

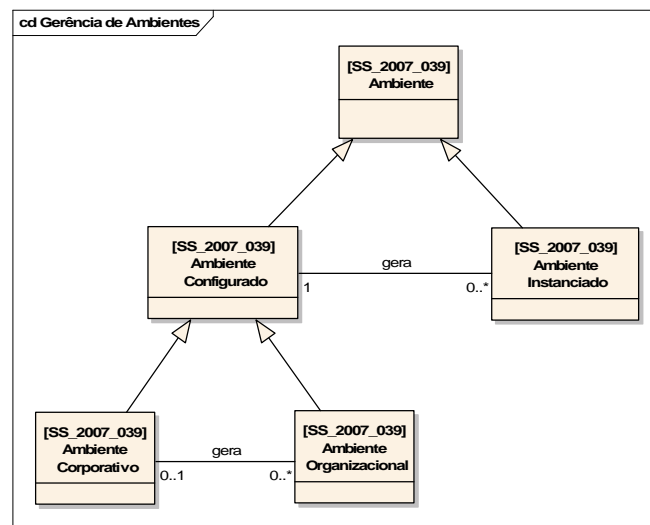


Figura 1 – Diagrama de domínio da Gerência de Ambientes

2.2 Gerência de Ativos de Processos

O modelo de domínio associado com a Gerência de Ativos de Processos é exibido na Figura 2.

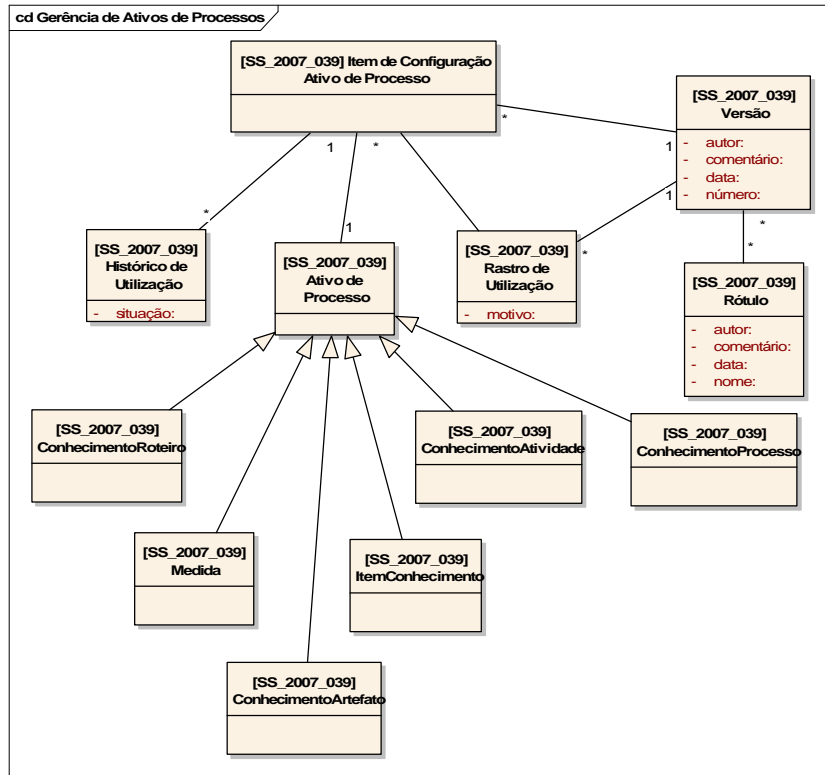


Figura 2 – Diagrama de domínio da Gerência de Ativos de Processos

2.3 Gerência de Organização

O modelo de domínio associado com a Gerência de Organização é exibido na Figura 3.

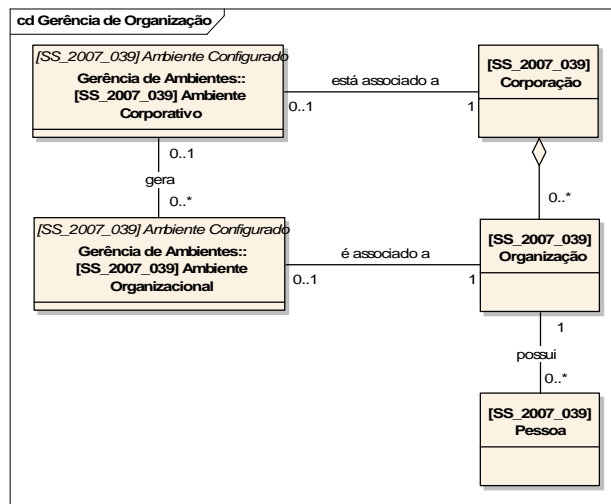


Figura 3 – Diagrama de domínio da Gerência de Organização

2.4 Gerência de Processos

O modelo de domínio associado com a Gerência de Processos é exibido na Figura 4.

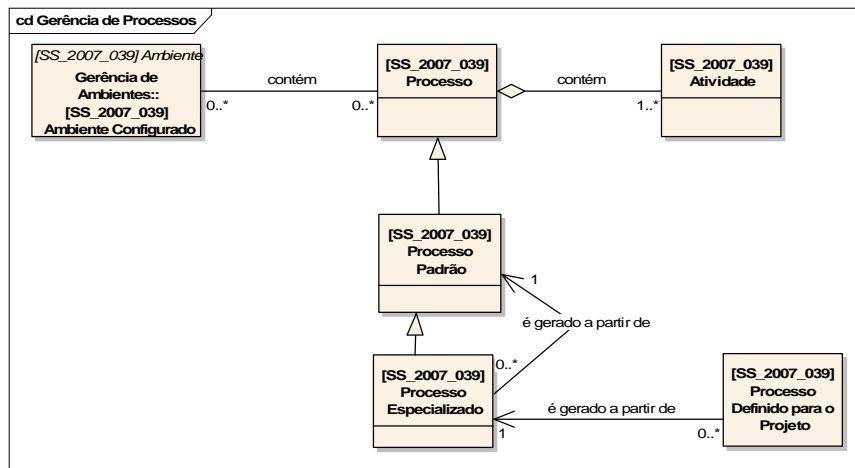


Figura 4 – Diagrama de domínio da Gerência de Processos

2.5 Gerência de Execução de Processos

O modelo de domínio associado com a Gerência de Execução de Processos é exibido na Figura 5.

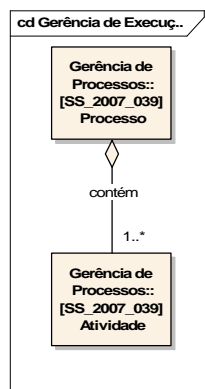


Figura 5 – Diagrama de domínio da Gerência de Execução de Processos

3 Requisitos do Cliente

É preciso implementar a estrutura básica dos Ambientes de Engenharia de Software Orientados a Corporação (AESCorp). A motivação, contexto e descrição dos AESCorp pode ser visto no texto da Tese de Doutorado de Gleison Santos (ainda em andamento). 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 da Estação Taba, foram levantados os requisitos do cliente descritos nas subseções a seguir. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela 20 – Requisitos de Produto.

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.

3.1 Gerência de Ambientes

A Tabela 1 apresenta os requisitos de cliente associados com a Gerência de Ambientes. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela 20.

Tabela 1 – Requisitos de Cliente da Gerência de Ambientes

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
RC.001	RP.009	Permitir a definição de Ambientes Corporativos a partir do Meta-ambiente.	X		
RC.002	RP.009	Permitir a definição de Ambientes Organizacionais a partir de um Ambiente Corporativo.	X		
RC.003	RP.014	Permitir a definição de Ambientes de Projeto a partir de um Ambiente Organizacional.	X		
RC.010	RP.015	Definir as funcionalidades dos Ambientes Corporativos necessárias para apoiar a iniciativa de melhoria de processos na corporação.		X	
RC.011	RP.015	Definir as funcionalidades dos Ambientes Organizacionais necessárias para apoiar a iniciativa de melhoria de processos na corporação.		X	
RC.012	RP.015	Definir as funcionalidades dos Ambientes de Projeto necessárias para apoiar a execução de processos de software.		X	

3.2 Gerência de Ativos de Processos

A Tabela 2 apresenta os requisitos de cliente associados com a Gerência de Ativos de Processos. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela 20.

Tabela 2 – Requisitos de Cliente da Gerência de Ativos de Processos

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
RC.013	RP.021	Possibilitar a definição de uma política de gerência de ativos de processos na Corporação (envolvendo definição, evolução, armazenamento, divulgação, garantia da qualidade e gerência de configuração).			X
RC.014	RP.021	Possibilitar a definição de uma política de gerência de ativos de processos na organização com base na política corporativa.			X
RC.015	RP.021	Possibilitar a definição dos ativos de processos a serem utilizados na corporação.	X		
RC.016	RP.021	Possibilitar a definição dos ativos de processo a serem utilizados em uma organização.	X		
RC.017	RP.021 RP.003	Possibilitar a evolução dos ativos de processo de acordo com as políticas definidas, incluindo o controle pela corporação da evolução dos ativos de processo das organizações.		X	
RC.018	RP.021	Possibilitar o compartilhamento de ativos de processo dentro da corporação e suas organizações.		X	
RC.019	RP.021 RP.003	Possibilitar a existência de diferentes versões de um mesmo ativo de processo na corporação e suas organizações, além de	X		

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
		possibilitar a uma corporação, a utilização de diferentes versões de um mesmo ativo de processo conforme necessidades do programa de melhoria de processos.			
RC.020	RP.021	Possibilitar à corporação o acompanhamento/ monitoração da utilização dos ativos de processo corporativos pelas organizações.		X	
RC.025	RP.021	Separar logicamente a base de Ativos de Processo da base de Versões de Ativos de Processo para garantir uma solução menos intrusiva possível à infra-estrutura atual da Estação Taba.	X		

3.3 Gerência de Organização

A Tabela 3 apresenta os requisitos de cliente associados com a Gerência de Organização. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela 20.

Tabela 3 – Requisitos de Cliente da Gerência de Organização

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
RC.021	RP.011	Possibilitar a descrição da estrutura organizacional das organizações.		X	
RC.022	RP.011	Possibilitar a descrição da estrutura organizacional das corporações.		X	
RC.023	RP.012	Possibilitar a descrição dos profissionais das organizações.		X	
RC.024	RP.012	Possibilitar a descrição dos profissionais das corporações.		X	

3.4 Gerência de Processos

A Tabela 4 apresenta os requisitos de cliente associados com a Gerência de Processos. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela 20.

Tabela 4 – Requisitos de Cliente da Gerência de Processos

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
RC.004	RP.008	Permitir a definição de Processos Padrão para Corporações e Organizações.	X		
RC.005	RP.008	Permitir a definição de Processos Especializados para Corporações e Organizações.	X		
RC.006	RP.008	Permitir a definição de Processos Definidos para o Projeto.	X		
RC.007	RP.008	Permitir a alteração de Processos Padrão nas Corporações e Organizações.		X	
RC.008	RP.008	Permitir a alteração de Processos Especializados nas Corporações e Organizações.		X	
RC.009	RP.008	Permitir a alteração de Processos Definidos para o Projeto a partir da evolução dos processos da Organização.		X	

3.5 Gerência de Execução de Processos

Tabela 5 apresenta os requisitos de cliente associados com a Gerência de Execução de Processos. Para cada Requisito de Cliente (ID RC) foram identificados os Requisitos do Produto (ID RP) afetados ou atendidos, descritos na Tabela 20.

Tabela 5 – Requisitos de Cliente da Gerência de Execução de Processos

ID RC	ID RP	Descrição	Prioridade		
			A	M	B
RC.026	RP.015	Possibilitar a execução de processos de software associado a um projeto.	X		
RC.027	RP.015	Possibilitar a execução de processos de software associado a uma ferramenta.	X		

4 Requisitos Funcionais de Software

O GEPS desenvolve e mantém um ambiente centrado em processo chamado Estação Taba composto de ferramentas de apoio a definição, execução e melhoria de processos de software.

Para atender o serviço solicitado, será necessário adaptar características de funcionalidade, segurança, desempenho e usabilidade da Estação Taba e das ferramentas. Para atender as necessidades do cliente levantadas, foram identificados os requisitos funcionais que devem ser implementados na Estação Taba e nas ferramentas identificadas. Esses requisitos estão descritos nas subseções 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.

4.1 Gerência de Ambientes

A Tabela 6 apresenta os requisitos funcionais associados com a Gerência de Ambientes. Para cada Requisito de Funcional (ID RF) foram identificados os Requisitos de Cliente (ID RC) afetados ou atendidos.

Tabela 6 – Requisitos Funcionais – Gerência de Ambientes

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
RF.027	RC.001 RC.002	Definir as características do ambiente relevantes para a configuração (por exemplo, nome, descrição, organização/corporação a que se destina, tipos de software e paradigmas utilizados etc.).	X		
RF.028	RC.001 RC.002	Identificar os processos que devem compor a base inicial de ativos de processo do ambiente.	X		
RF.029	RC.001 RC.002	Adaptar o processo escolhido para uso no ambiente configurado.	X		
RF.030	RC.001 RC.002	Identificar ferramentas que devem compor o ambiente configurado corporativo ou organizacional.		X	
RF.031	RC.001 RC.002	Disponibilizar o ambiente configurado para uso da organização ou da corporação de acordo com as características e processos definidos.	X		
RF.032	RC.003	Definir as características do ambiente relevantes para a instânciação (por exemplo, nome, descrição, detalhes do projeto, tipos de software e paradigmas utilizados etc.).	X		
RF.033	RC.003	Disponibilizar o ambiente para uso em um projeto.	X		
RF.034	RC.010	Identificar funcionalidades presentes nos ambientes corporativo,			X

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
	RC.011 RC.012	organizacional e do projeto.			
RF.035	RC.010 RC.011 RC.012	Identificar perfis de acesso às funcionalidades disponíveis no ambiente.			X
RF.041	RC.003	Identificar o processo que deve ser utilizado para execução do projeto.	X		
RF.042	RC.003	Adaptar o processo para o projeto seguindo as regras de adaptação e o modelo de ciclo de vida a ser adotado.	X		

4.2 Gerência de Ativos de Processos

A Tabela 7 apresenta os requisitos funcionais associados com a Gerência de Ativos de Processos. Para cada Requisito de Funcional (ID RF) foram identificados os Requisitos de Cliente (ID RC) afetados ou atendidos.

Tabela 7 – Requisitos Funcionais – Gerência de Ativos de Processos

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
*	RC.013	Possibilitar a definição de uma política de gerência de ativos de processos na Corporação (envolvendo definição, evolução, armazenamento, divulgação, garantia da qualidade e gerência de configuração).			X
*	RC.014	Possibilitar a definição de uma política de gerência de ativos de processos na organização com base na política corporativa.			X
RF.010	RC.015 RC.016 RC.018 RC.025	Possibilitar a criação de uma base de ativos de processo a partir de outra base (por exemplo, da corporação) ou uma base nova.	X		
RF.011	RC.015 RC.016	Identificar as diferentes categorias de ativos de processo que podem ser gerenciados pela Estação Taba.	X		
RF.012	RC.015 RC.016	Identificar dentre as instâncias dos diferentes tipos de ativos de processo aqueles que serão efetivamente tratados como itens de ativo de processo para efeitos de gerência de configuração.	X		
RF.013	RC.025 RC.019	Implementar um mecanismo de controle de versão para os ativos de processo.	X		
RF.014	RC.025 RC.019	Implementar um mecanismo de controle de alteração para os ativos de processo para registrar os pedidos de alteração, análise do pedido, alteração do ativo e aceitação do ativo alterado.	X		
RF.015	RC.015	Implementar mecanismo que possibilite o controle de acesso aos ativos de processo cadastrados.	X		
RF.016	RC.019	Possibilitar a criação de uma linha base para os ativos de processo.		X	
RF.017	RC.019	Adicionar itens novos ao controle de ativos de processo (similar ao comando <i>add</i> do CVS).	X		
RF.018	RC.019	Remover itens já existentes do controle de ativos de processo (similar ao comando <i>remove</i> do CVS).	X		
RF.019	RC.019	Recuperar itens armazenados no controle de ativos de processo (similar ao comando <i>checkout</i> do CVS).	X		
RF.020	RC.019	Fazer atualizações dos itens de trabalho a partir das versões atuais destes itens no controle de ativos de processo (similar ao comando <i>update</i> do CVS).	X		
RF.021	RC.019	Fazer submissões de atualizações em itens de trabalho para o controle de ativos de processo (similar ao comando <i>commit</i> do CVS).	X		
RF.022	RC.019	Possuir controle de desenvolvimento em paralelo de diferentes versões dos ativos de processo via criação de ramos (<i>branches</i>).		X	

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
RF.023	RC.019	Manter o histórico de evolução de versão do item de ativo de processo, de utilização do item de ativo de processo em outros contextos (por exemplo, nas demais organizações) e de evolução da estrutura do item de ativo de processo mesmo para objetos em desuso nos ambientes ou removidos da base de dados.	X		
RF.024	RC.020	Possuir mecanismo para monitorar utilização dos ativos de processo através da disponibilização de um relatório.		X	
RF.025	RC.017 RC.018	Possibilitar que uma organização obtenha versões de ativos de processo presente em outras bases de organizações ou da base corporativa. Possibilitar que uma corporação disponibilize novas versões de ativos de processo para as organizações.	X		
RF.026	RC.020	Possibilitar o acesso para leitura do conteúdo de outras bases de ativos de processo. Pode ser útil para que a corporação veja a evolução realizada pelas organizações ou para uma organização conseguir identificar oportunidades de melhoria na sua própria base de ativos de processo.		X	

(*) Não implementado(s) computacionalmente.

4.3 Gerência de Organização

A Tabela 8 apresenta os requisitos funcionais associados com a Gerência de Organização. Para cada Requisito de Funcional (ID RF) foram identificados os Requisitos de Cliente (ID RC) afetados ou atendidos.

Tabela 8 – Requisitos Funcionais – Gerência de Organização

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
RF.001	RC.022	Definir os dados básicos sobre uma corporação, como nome, missão, pessoas associadas e organizações relacionadas.	X		
RF.002	RC.021 RC.022	Definir os dados básicos sobre uma organização, como nome, missão e pessoas associadas.	X		
RF.003	RC.023 RC.024	Possibilitar a definição de uma pessoa, incluindo seus dados pessoais, endereço e competências.	X		
RF.004	RC.021 RC.022	Definir os detalhes de uma unidade organizacional incluindo nome, descrição e relação com outras unidades.	X		
RF.005	RC.021 RC.022	Definir os detalhes de uma unidade organizacional incluindo nome, descrição, relação com outras posições e competências requeridas/desejadas.	X		
RF.006	RC.021 RC.022	Exibir um relatório contendo os dados sobre a estrutura da organização e das pessoas que estão alocadas à estrutura organizacional.		X	
RF.007	RC.023 RC.024	Possibilitar o cadastro dos itens que comporão as habilidades, conhecimentos e experiências.			X
RF.008	RC.023 RC.024	Cadastrar escalas de valores que podem ser utilizadas para ser associadas a diversos outros itens na Estação Taba.			X
RF.009	RC.023 RC.024	Possibilitar o cadastro dos itens que comporão a formação acadêmica de um profissional (por exemplo, instituição de ensino e especialidades).			X

4.4 Gerência de Processos

A Tabela 9 apresenta os requisitos funcionais associados com a Gerência de Processos. Para cada Requisito de Funcional (ID RF) foram identificados os Requisitos de Cliente (ID RC) afetados ou atendidos.

Tabela 9 – Requisitos Funcionais – Gerência de Processos

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
RF.036	RC.004 RC.005	Definir um processo padrão contendo a descrição de todas as suas atividades e informações relacionadas.	X		
RF.037	RC.004 RC.005	Identificar característica do processo que permitam indicar seu contexto de aplicabilidade.	X		
RF.038	RC.004 RC.005	Incluir atividades já existentes na base de ativos de processo ou em processos já existentes.	X		
RF.039	RC.004 RC.005	Definir as regras de adaptação que podem ser aplicadas ao processo padrão em questão.		X	
RF.040	RC.005	Adaptar processo existente para gerar processo especializado.	X		
RF.041	RC.006	Identificar o processo que deve ser utilizado para execução do projeto.	X		
RF.042	RC.006	Adaptar o processo para o projeto seguindo as regras de adaptação e o modelo de ciclo de vida a ser adotado.		X	
RF.043	RC.007 RC.008	Editar o conteúdo do processo padrão.	X		
RF.044	RC.007 RC.008	Alterar os processos padrão em uso na organização devido a uma alteração dos processos corporativos.	X		
RF.045	RC.009	Editar o processo definido para o projeto devido a necessidades do projeto.	X		
RF.046	RC.009	Alterar processos definidos para o Projeto devido a evolução do processo padrão da organização.		X	

4.5 Gerência de Execução de Processos

A Tabela 10 apresenta os requisitos funcionais associados com a Gerência de Execução de Processos. Para cada Requisito de Funcional (ID RF) foram identificados os Requisitos de Cliente (ID RC) afetados ou atendidos.

Tabela 10 – Requisitos Funcionais – Gerência de Execução de Processos

ID RF	ID RC	Descrição	Prioridade		
			A	M	B
RF.047	RC.026	Possibilitar o planejamento da execução do processo de software associado a um ambiente de projeto.	X		
RF.048	RC.026	Executar o processo de software associado com um ambiente para o projeto.	X		
RF.049	RC.026	Possibilitar a monitoração da execução do processo associado a um ambiente de projeto.	X		
RF.050	RC.026	Possibilitar a visualização e recuperação de artefatos de outros projetos em execução ou já finalizados.	X		
RF.051	RC.026	Possibilitar a disponibilização de artefatos de outros projetos em execução ou já finalizados.	X		
RF.052	RC.026	Possibilitar a gerencia dos documentos produzidos ao longo da execução do processo associado a um ambiente do projeto.	X		
RF.053	RC.027	Executar processo associado com o funcionamento das ferramentas orientadas a processo da Estação Taba.	X		
RF.054	RC.026	Exibir o processo associado com o projeto do ambiente de projeto de forma a possibilitar a sua execução.	X		
RF.055	RC.027	Exibir o processo associado com o funcionamento das ferramentas orientadas a processo da Estação Taba de forma a possibilitar a sua execução.	X		

5 Caso de Uso e Processos de Apoio

Para cada um dos módulos identificados foram identificados casos de uso e definido um conjunto de processos de apoio para guiar a execução das ferramentas a serem construídas. 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.

5.1 Gerência de Ambientes

A Gerência de Ambientes será apoiada por cinco diferentes processos: Definir Ambientes Configurados (que atende pelos Ambientes Corporativos e Organizacionais), Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais, Definir Ambientes de Projetos e Alterar Ambientes de Projetos.

A Figura 6 apresenta o processo de apoio Definir Ambientes Configurados.

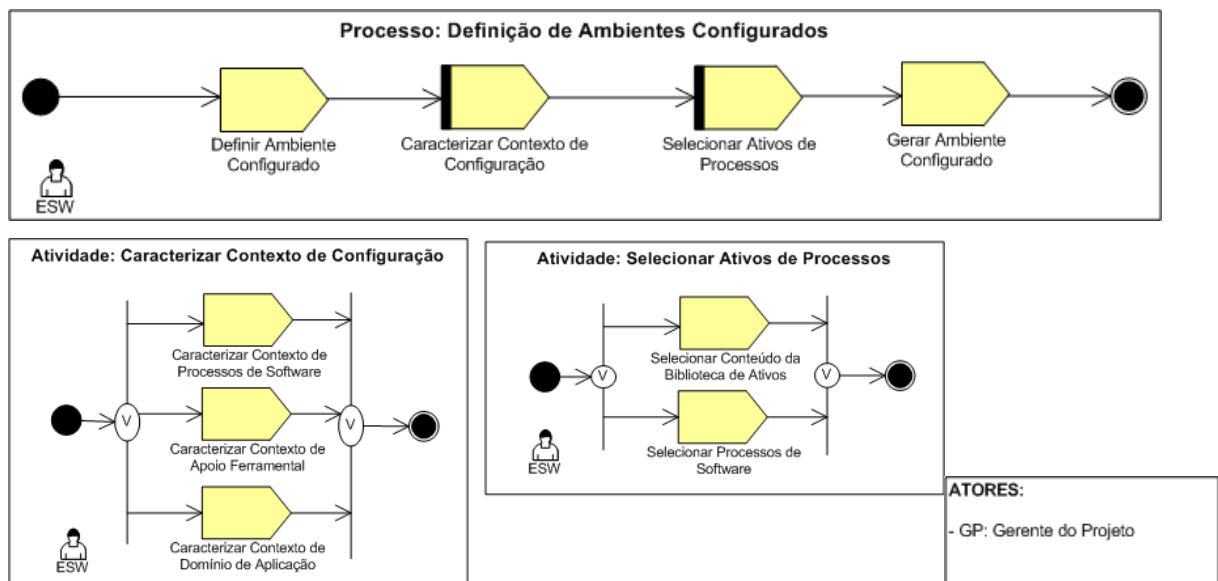


Figura 6 – Processo Definir Ambientes Configurados

As atividades do processo Definir Ambientes Configurados serão atendidas pelos casos de uso identificados Tabela 11. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 11 – Casos de uso do processo Definir Ambientes Configurados

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.001	RF.027	Caracterizar Ambiente			X
UC.002	RF.027	Caracterizar Contexto de Configuração			X
UC.003	RF.028	Selecionar Processos			X
UC.004	RF.028	Selecionar Ativos de Processos			X
UC.008	RF.029	Exibir relatórios (ambientes corporativos)			X
UC.067	RF.030 RF.031 RF.034 RF.025	Gerar ambiente configurado			X

A Figura 7 apresenta o processo de apoio Definir Ambientes de Projetos.

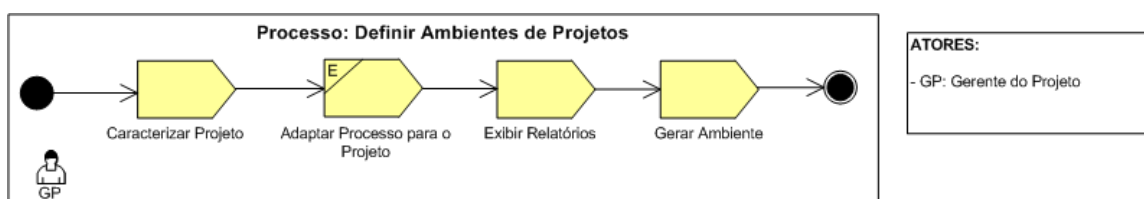


Figura 7 – Processos Definir Ambientes de Projetos

As atividades do processo Definir Ambientes de Projetos serão atendidas pelos casos de uso identificados Tabela 13. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 12 – Casos de uso do processo Definir Ambientes de Projetos

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.061	RF.032 RF.041	Caracterizar projeto			X
UC.062	RF.041 RF.042	Adaptar processo para o projeto			X
UC.063	RF.033	Gerar ambiente do projeto			X
UC.064	RF.033	Exibir relatórios (ambiente do projeto)			X

A Figura 8 apresenta os processos de apoio Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais e Alterar Ambientes de Projetos.

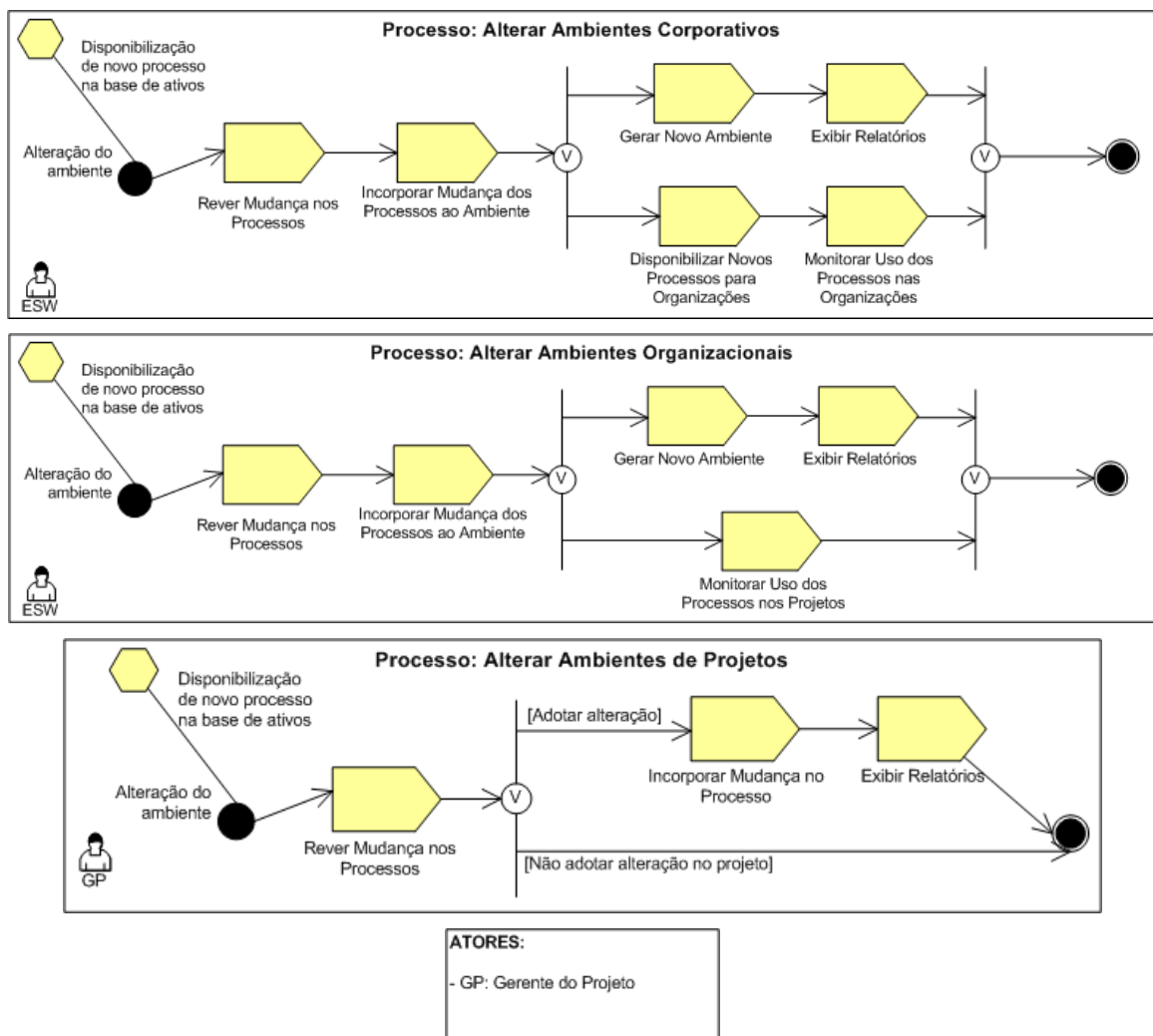


Figura 8 – Processos Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais e Alterar Ambientes de Projetos

As atividades dos processos Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais e Alterar Ambientes de Projetos serão atendidas pelos casos de uso identificados Tabela 13. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 13 – Casos de uso dos processos Alterar Ambientes Corporativos, Alterar Ambientes Organizacionais e Alterar Ambientes de Projetos

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.056	RF.029	Rever mudança nos processos			X
UC.057	RF.031	Incorporar mudança nos processos do ambiente		X	
UC.058	RF.031	Gerar novo ambiente			X
UC.059	RF.031	Disponibilizar novos processos			X
UC.060	RF.031	Monitorar uso dos processos	X		
UC.065	RF.042	Rever mudança nos processos		X	
UC.066	RF.041	Incorporar mudança nos processos		X	

5.2 Gerência de Ativos de Processos

A Gerência de Ativos de Processos será apoiada por dois diferentes processos: Definir Base de Ativos de Processos e Gerenciar Base de Ativos de Processos.

A Figura 9 apresenta o processo de apoio Definir Base de Ativos de Processos.

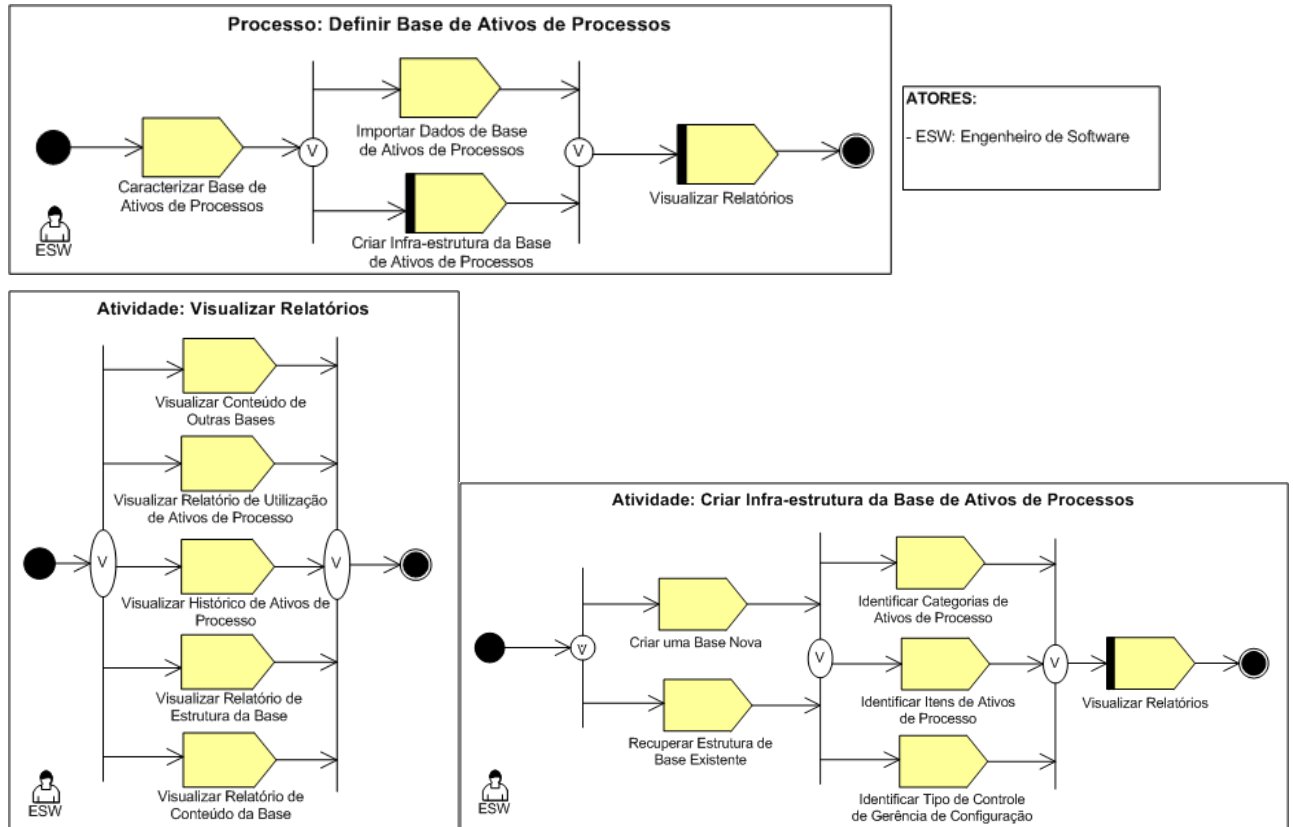


Figura 9 – Processo Definir Base de Ativos de Processos

As atividades do processo Definir Base de Ativos de Processos serão atendidas pelos casos de uso identificados Tabela 14. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 14 – Casos de uso do processo Definir Base de Ativos de Processos

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.035	RF.011	Caracterizar base de ativos de processos			X
UC.036	RF.025	Importar dados de base de ativos de processos			X
UC.037	RF.023 RF.024 RF.026	Visualizar relatórios			X
UC.038	RF.025	Recuperar estrutura de base existente			X
UC.039	RF.010	Criar nova base			X
UC.040	RF.011	Identificar categorias de ativos de processos			X
UC.041	RF.012	Identificar itens de ativos de processos			X
UC.042	RF.012 RF.015	Identificar tipo de controle de ativos de processo			X

A Figura 10 apresenta o processo de apoio Gerenciar Base de Ativos de Processos.

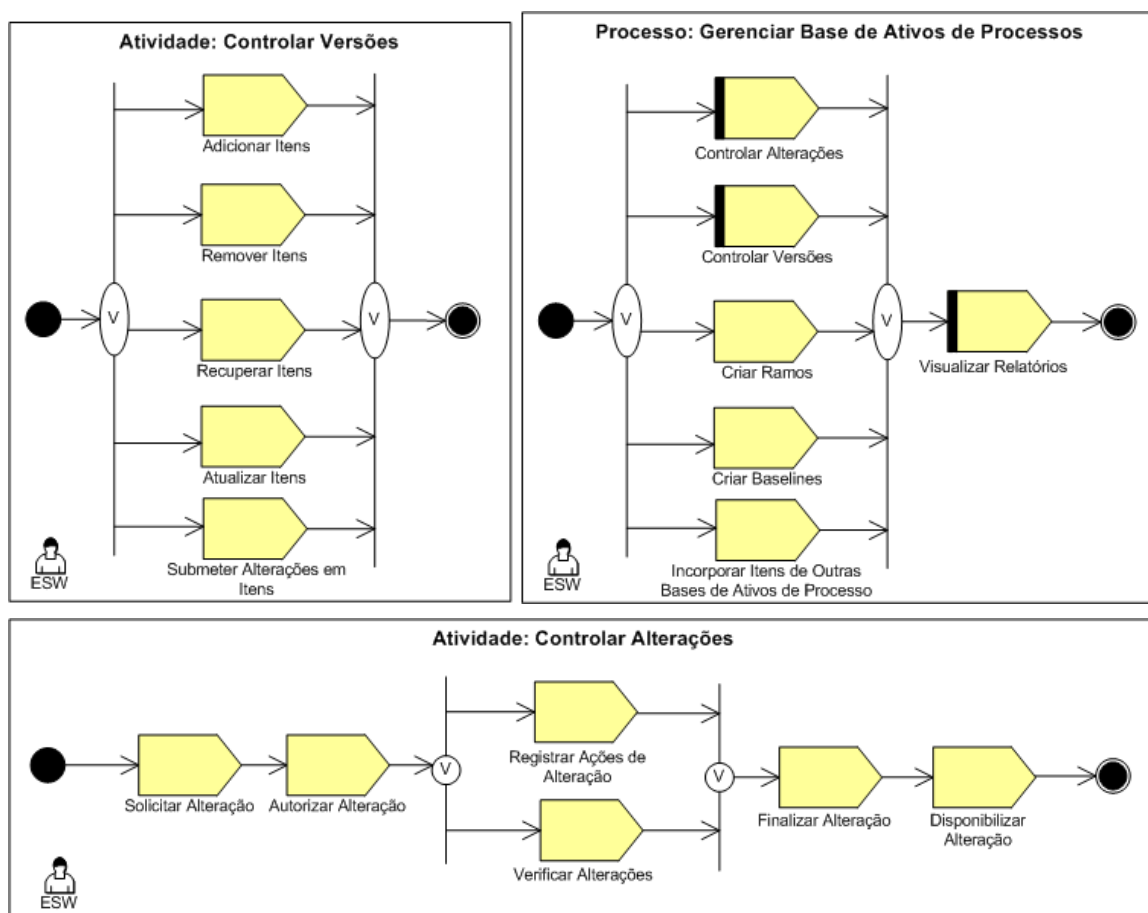


Figura 10 – Processo Gerenciar Base de Ativos de Processos

As atividades do processo Gerenciar Base de Ativos de Processos serão atendidas pelos casos de uso identificados Tabela 15. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 15 – Casos de uso do processo Gerenciar Base de Ativos de Processos

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.043	RF.013 RF.017	Adicionar itens			X
UC.044	RF.013 RF.018	Remover itens			X
UC.045	RF.013 RF.019	Recuperar itens			X
UC.046	RF.013 RF.020	Atualizar itens			X
UC.047	RF.021 RF.013	Submeter alterações nos itens			X
UC.048	RF.022	Criar ramos			X
UC.049	RF.016	Criar baselines			X
UC.050	RF.014	Solicitar alteração			X
UC.051	RF.014	Autorizar alteração			X
UC.052	RF.014	Registrar ações de alteração			X
UC.053	RF.014	Verificar alterações			X
UC.053	RF.014	Finalizar alterações			X
UC.054	RF.014	Disponibilizar alterações			X

5.3 Gerência de Organização

A Gerência de Organização será apoiada por três diferentes processos: Descrever Corporação, Descrever Organização e Manter Cadastros.

A Figura 11 e a Figura 12 apresenta os processos de apoio Descrever Corporação e Descrever Organização.

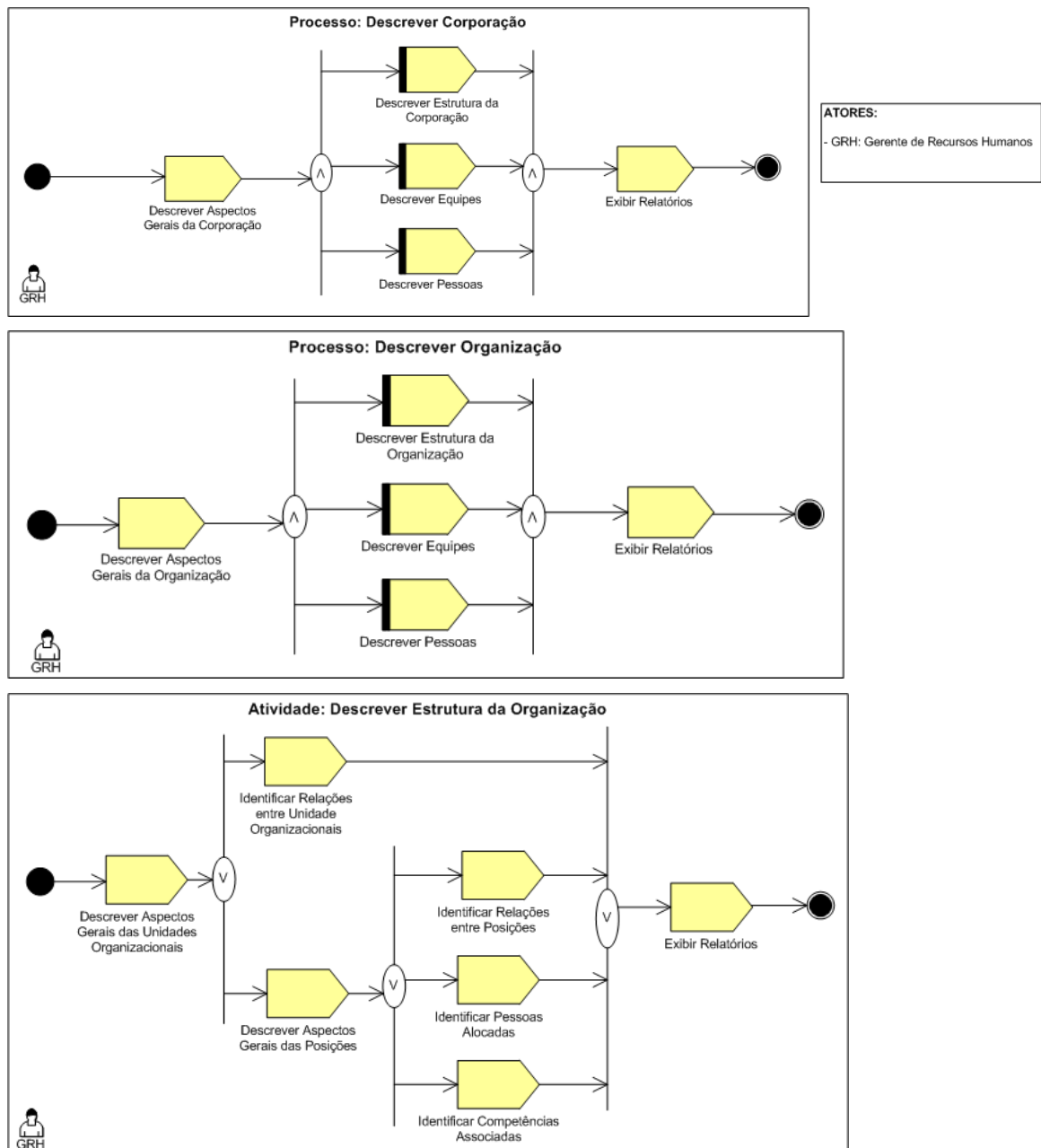


Figura 11 – Processos Descrever Corporação e Descrever Organização - 1

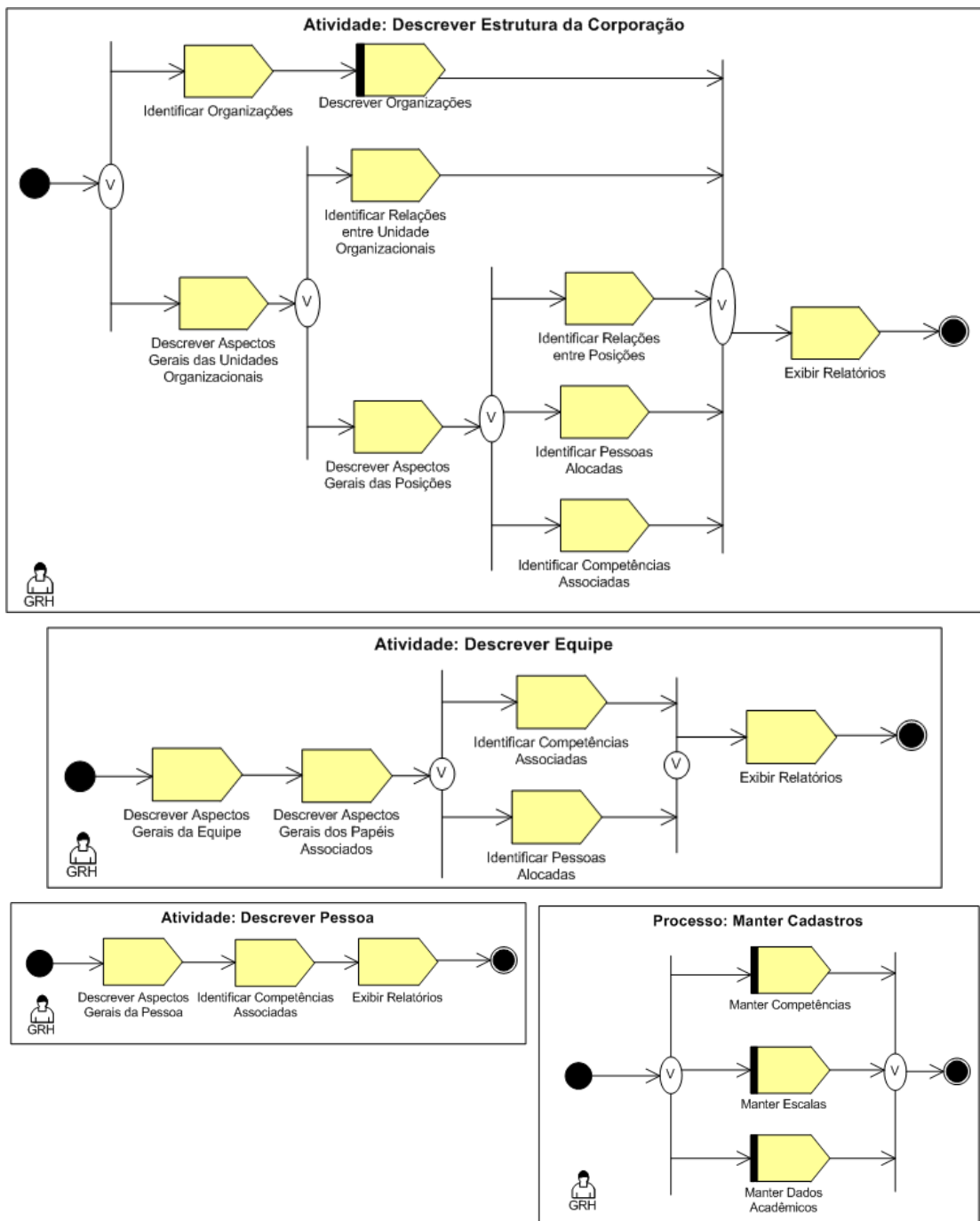


Figura 12 – Processos Descrever Corporação e Descrever Organização - 2

As atividades dos processos Descrever Corporação e Descrever Organização serão atendidas pelos casos de uso identificados Tabela 16. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 16 – Casos de uso dos processos Descrever Corporação e Descrever Organização

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.023	RF.001	Descrever aspectos gerais das corporações			X
UC.024	RF.001	Identificar organizações da corporação			X
UC.025	RF.001 RF.002	Descrever aspectos gerais da organização / corporação			X
UC.026	RF.006	Exibir relatórios		X	
UC.027	RF.002	Descrever estrutura organizacional			X
UC.028	RF.004	Descrever unidades organizacionais			X
UC.029	RF.003	Descrever equipes	X		
UC.030	RF.003	Descrever pessoas		X	
UC.031	RF.005	Descrever posições		X	

A Figura 13 apresenta o processo de apoio Manter Cadastros.

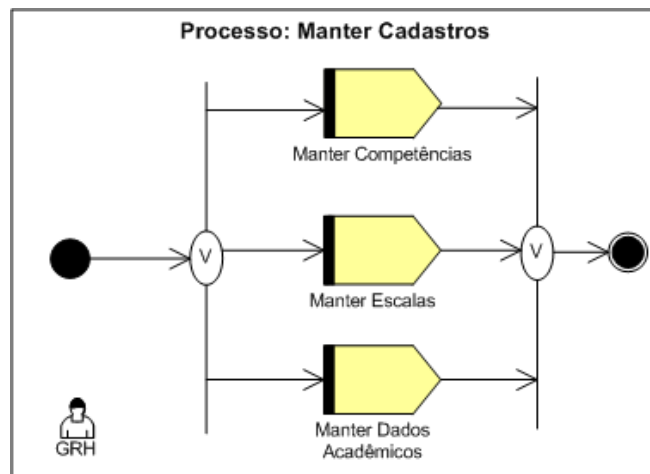


Figura 13 – Processo Manter Cadastros

As atividades do processo Manter Cadastros serão atendidas pelos casos de uso identificados Tabela 17. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 17 – Casos de uso do processo Manter Cadastros

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.032	RF.007	Manter Competências			X
UC.033	RF.008	Manter Escalas			X
UC.034	RF.009	Manter dados acadêmicos			X

5.4 Gerência de Processos

A Gerência de Processos será apoiada por dois diferentes processos: Definir Processos (que inclui os subprocessos Definir Processo Padrão e Definir Processo Especializado) e Adaptar Processo para Projeto (que pode é utilizado na forma de uma atividade por processos da Gerência de Ambientes).

A Figura 14 apresenta o processo de apoio Definir Processos e seus subprocessos.

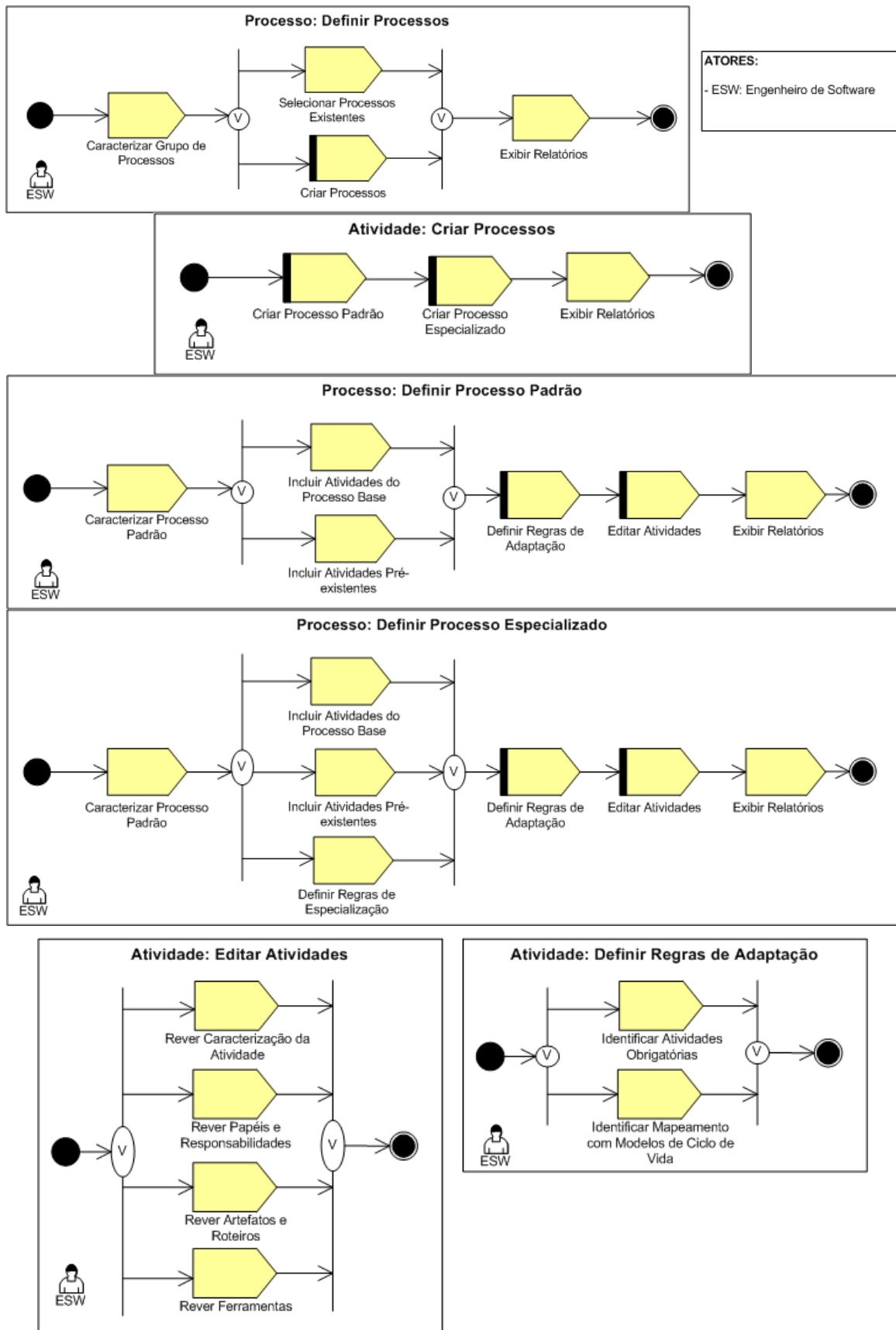


Figura 14 – Processo Definir Processos e seus subprocessos

As atividades do processo Definir Base de Ativos de Processos serão atendidas pelos casos de uso identificados Tabela 18. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 18 – Casos de uso do processo Definir Processos e seus subprocessos

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.009	RF.036	Caracterizar grupo de processos			X
UC.010	RF.036	Selecionar processos existentes			X
UC.011	RF.036	Exibir relatórios (processo padrão)			X
UC.012	RF.037	Caracterizar processo padrão			X
UC.013	RF.038	Incluir atividades do processo base			X
UC.014	RF.038	Incluir atividades pré-existentes			X
UC.015	RF.040	Definir regras de especialização			X
UC.016	RF.036 RF.043 RF.044 RF.042 RF.045	Editar atividades			X
UC.017	RF.039	Definir regras de adaptação			X
UC.018	RF.039	Identificar atividades obrigatórias			X
UC.019	RF.039	Identificar mapeamento com modelos de ciclo de vida			X

A Figura 15 apresenta o processo de apoio Adaptar Processo para o Projeto.



Figura 15 – Processo Adaptar Processo para o Projeto

As atividades do processo Adaptar Processo para o Projeto serão atendidas pelos casos de uso identificados Tabela 19. Para cada Caso de Uso (ID UC) foram identificados os Requisitos Funcionais (ID RF) relacionados.

Tabela 19 – Casos de uso do processo Adaptar Processo para o Projeto

ID UC	ID RF	Descrição	Complexidade		
			A	M	B
UC.020	RF.041	Identificar processo para o projeto		X	
UC.021	RF.042	Mapear processo ao modelo de ciclo de vida		X	
UC.022	RF.041	Exibir relatórios (processo do projeto)		X	

6 Requisitos de Produto da Estação Taba

A Tabela 20 apresenta os requisitos de produto da Estação Taba. Estes requisitos são aqueles mencionados na Tabela 5.1 do Capítulo 5 desta tese.

Tabela 20 – Requisitos de Produto da Estação Taba

ID RP	Descrição
RP.001	Possuir interface consistente.
RP.002	Possuir um modelo comum de armazenamento de dados.
RP.003	Apoiar o controle de versões e a gerência de configuração.
RP.004	Possuir conhecimento sobre processo de software e abordagens de desenvolvimento.
RP.005	Possuir mecanismo de integração de ferramentas.
RP.006	Permitir a descrição de tarefas.
RP.007	Apoiar a utilização de Teorias de Domínio.
RP.008	Apoiar a definição de processos de software.
RP.009	Gerar Ambientes Configurados para o contexto desejado.
RP.010	Apoiar a gerência de conhecimento nos ambientes.
RP.011	Permitir a descrição da estrutura organizacional.
RP.012	Permitir a descrição dos profissionais envolvidos.
RP.013	Permitir a descrição dos processos organizacionais.
RP.014	Gerar ambiente para projetos específicos.
RP.015	Apoiar a execução e a gerência de processos de software.
RP.016	Possuir apoio para a avaliação do produto.
RP.017	Possuir apoio para a avaliação do processo.
RP.018	Apoiar a definição de arquiteturas de referência.
RP.019	Apoiar o trabalho cooperativo.
RP.020	Possuir apoio para a melhoria contínua do processo.
RP.021	Apoiar a Gerência dos Ativos de Processo.

Além disso, a Estação Taba possui requisitos de arquitetura que também devem ser respeitados e apoiados, conforme pode ser visto na Tabela 21 – Requisitos da Arquitetura do Produto da Estação Taba. Estes requisitos são aqueles mencionados na Tabela 5.3 do Capítulo 5 desta tese.

Tabela 21 – Requisitos da Arquitetura do Produto da Estação Taba

ID RAP	Descrição
RAP.001	Possuir uma arquitetura bem definida, utilizando componentização para facilitar o desenvolvimento, e que minimize o esforço e custo de construção das funcionalidades e ferramentas.
RAP.002	Permitir a utilização do sistema em diferentes locais sem a necessidade de instalação prévia.
RAP.003	Ser construído usando tecnologias escaláveis, robustas e bem aceitas no mercado, de preferência gratuitas, evitando a utilização de tecnologias incipientes, instáveis e/ou não confiáveis.
RAP.004	Possuir uma interface desacoplada, de forma a permitir sua evolução sem impacto nos demais elementos da aplicação.
RAP.005	Possuir uma integração de dados entre as ferramentas evitando a duplicidade de entrada de informações.
RAP.006	Possibilitar a utilização do sistema de qualquer lugar através de interface web.
RAP.007	Manter compatibilidade com a infra-estrutura atual da Estação Taba.
RAP.008	Possibilitar o acesso concorrente de múltiplos usuários ao sistema nas mesmas ferramentas e ambientes.
RAP.009	Possuir um tempo de resposta aceitável para as ações do usuário.
RAP.010	Possuir facilidades de uso para a inclusão e edição das informações a serem digitadas.
RAP.011	Possuir uma interface amigável, mantendo o padrão da interface atual da Estação Taba, mas adicionando outras facilidades.
RAP.012	Possuir suporte à internacionalização dos elementos de interface e mensagens para os usuários.
RAP.013	Ter uma interface padrão entre as ferramentas.
RAP.014	Possuir independência de sistema operacional do cliente ou do servidor. Além disso,

ID RAP	Descrição
	possibilitar a utilização de diferentes navegadores.
RAP.015	Poder ser utilizado com diferentes tipos de banco de dados.
RAP.016	Possuir dispositivo para atualização automática de versões.
RAP.017	Possuir dispositivo para envio automático de exceções e ocorrências de falha.
RAP.018	Facilitar a manutenção das funcionalidades do ambiente, permitindo a evolução dos elementos de forma independente sem impacto nos demais elementos do ambiente.
RAP.019	Armazenar o log das ações tomadas pelos usuários tanto para fins de auditoria quanto para auxiliar na identificação de erros nas ferramentas.
RAP.020	Possibilitar a definição de diferentes perfis de controle de acesso para o acesso às ferramentas e às funcionalidades de determinadas ferramentas mais críticas de acordo com o papel nos projetos.