

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

ALESSANDRA DAHMER

Um Modelo para Processo de Curso

Tese apresentada como requisito parcial para a
obtenção do grau de Doutor em Ciência da
Computação

Prof. Dr. Daltro José Nunes
Orientador

Prof. Dr^a. Rosa Maria Viccari
Co-orientadora

Porto Alegre, março de 2006.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Dahmer, Alessandra

Um Modelo para Processo de Curso / Alessandra Dahmer –
Porto Alegre: Programa de Pós-Graduação em Computação, 2006.

132 f.:il.

Tese (doutorado) – Universidade Federal do Rio Grande do Sul.
Programa de Pós-Graduação em Computação. Porto Alegre, BR –
RS, 2006. Orientador: Daltro José Nunes; Co-orientador: Rosa
Maria Viccari.

1.Educação a Distância. 2.Processo de Curso 3. Processo de
Software. I. Nunes, Daltro José. II. Viccari, Rosa Maria. III.
Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. José Carlos Ferraz Hennemann

Vice-Reitor: Prof. Pedro Cezar Dutra Fonseca

Pró-Reitora de Pós-Graduação: Profa. Valquíria Linck Bassani

Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux

Coordenador do PPGC: Prof. Flávio Rech Wagner

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

Para Helena, Helio, Lincoln e Xuxa

AGRADECIMENTOS

Primeiramente, a Deus. Sem Ele, nada seria possível!

Obrigada aos meus pais, Helena e Helio, duas pessoas maravilhosas a quem devo tudo que sou, por terem se sacrificado para que eu pudesse chegar até aqui. Por terem sempre me apoiado, me dado força, confiado em mim, muito obrigado! Eu amo vocês dois! Nunca se esqueçam disto.

Agradeço também ao meu marido, Lincoln. Sempre acreditando em mim, até quando eu mesma duvidei, sempre me alegrando com suas brincadeiras, sempre me estimulando a seguir em frente. Obrigada, meu amor. Eu te amo! Muito.

E não podia esquecer da minha irmãzinha, Xuxa, que quebra um monte de galhos para mim. Obrigada, mana! E também obrigada ao Sid, meu cunhadinho querido, por sua infinita paciência.

Agradeço, muito, à minha madrinha, Dinda Dolores, por ter me recebido em sua casa como filha, desde que vim para Porto Alegre fazer graduação. Sem o teu apoio e do Dindo, Dindinha, eu não teria chegado até aqui. Obrigada a vocês dois.

Obrigada também aos meus padrinhos Ribeiro, Ruth, César, Gladis, Rosângela e Zé. Sou afortunada por ter tantos padrinhos maravilhosos.

Nunca poderia deixar de agradecer as minhas avós. Duas não estão mais aqui, mas sei que elas estão me vendo agora e abençoando esta minha conquista. Obrigada, Vó Irma e Vó Rosa! E, na falta delas, outra Irma me adotou como neta e devo um agradecimento a ela também, obrigada Vó.

Obrigada ao meu orientador, Prof. Daltro, pela atenção e orientações sempre tão valiosas. Aprendi muito com o senhor!

Agradeço também a minha co-orientadora, Prof. Rosa, pelas palavras gentis e carinhosas, incentivo e atenção.

Também preciso agradecer à minha sogra, Lúcia, que mesmo de longe sempre telefonava para me incentivar. Muito obrigada!

E a minha querida cunhada que está quase terminando sua dissertação e, em breve, ficará tão feliz quanto estou hoje. Obrigada, Jalva.

Aos meus amigos Carla e Rodrigo, por essa amizade tão sincera e verdadeira que sempre é um esteio para mim nas horas difíceis. E nas alegres também! Apesar da distância, me sinto sempre ao lado de vocês e da Amandinha. Muito obrigada!

Obrigada à Simone, sempre quebrando os meus galhos em cima da hora. E ao Xande, Lucas e Pedro também!

Aos meus tantos amigos da UFRGS, UNISC e La Salle. Seria difícil enumerar todos, mas os represento agradecendo à Daniela, Paulo e Marcos que tanto contribuíram para que este trabalho fosse finalizado.

Obrigada à Tati e a Rita por resolverem sempre os problemas para mim, na UNISC.

Obrigada também ao pessoal do PROSOFT. Sempre presentes às minhas várias prévias.

Aos professores Tiarajú e Navaux, dos quais já fui orientanda e que continuaram sempre a me encorajar a seguir em frente.

Aos funcionários da UFRGS, sempre tão queridos e amigos: Beatriz, Silvania, Ida, Luis Otávio, Elisiene, Eliane, Jorge e tantos outros que tanto prezo.

Gostaria de agradecer também à Prof. Lúcia Giraffa por toda a força que me deu na elaboração deste texto. Suas sugestões e comentários foram fundamentais para que eu chegasse a este resultado.

Enfim, obrigada a todos aqueles que de alguma forma contribuíram para que eu chegasse até aqui. Se esqueci de citar alguém, me perdoe, mas você sabe que eu estou muito grata pelo seu apoio. Ele me ajudou a chegar até aqui!!!!

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS	9
LISTA DE FIGURAS.....	10
LISTA DE TABELAS	11
RESUMO.....	12
ABSTRACT.....	13
1 INTRODUÇÃO	14
1.1 Contextualização.....	14
1.2 Motivação	15
1.3 Organização do Texto	17
2 METODOLOGIA DE PESQUISA.....	18
2.1 Delimitação do Tema.....	18
2.2 Questão de Pesquisa	19
2.3 Hipótese	19
2.4 Objetivos.....	19
2.4.1 Geral	19
2.4.2 Específicos.....	19
2.5 Fundamentação Teórica	20
2.6 Método de Pesquisa	20
2.7 Avaliação e Análise dos Dados	21
2.8 Considerações sobre o Capítulo	21
3 EDUCAÇÃO A DISTÂNCIA.....	22
3.1 Cursos a Distância	22
3.2 Estrutura de um Curso em EAD.....	23
3.3 Caracterização da EAD	24
3.4 Gerência de Cursos a Distância.....	26
3.5 Processo de Curso.....	27
3.6 O Papel do Docente na EAD.....	27
3.7 Considerações sobre o Capítulo	30
4 TECNOLOGIA DE PROCESSO DE SOFTWARE	31
4.1 Conceituação	31
4.1.1 Processo de software	32

4.2	Ambientes de Desenvolvimento de Software Orientados a Processos	33
4.3	Domínios de Processo de Software	34
4.4	Ciclo de Vida de Processos de Software	35
4.5	Modelagem de Processo de Software	36
4.5.1	Execução de modelos de processo.....	37
4.5.2	Reutilização de processos de software	38
4.6	Tecnologia de Processo de Software no Projeto PROSOFT	38
4.7	Considerações sobre o Capítulo	39
5	O AMBIENTE APSEE	41
5.1	Classes PROSOFT	41
5.1.1	Classe APSEE	42
5.1.2	Classe APSEE-Types	42
5.1.3	Classe Processes	43
5.2	Arquitetura do APSEE	44
5.2.1	Meta-modelo APSEE	44
5.2.1.1	Processos de Software.....	45
5.2.1.2	Templates.....	46
5.2.1.3	Hierarquia de tipos (APSEE-Types).....	46
5.2.1.4	Organização	46
5.2.1.5	Artefatos.....	47
5.2.1.6	Conhecimento sobre o processo (ProcessKnowledge)	47
5.2.1.7	Ferramentas.....	48
5.2.1.8	Políticas.....	48
5.2.2	Mecanismo de gerência de processos.....	49
5.2.2.1	APSEE-PML.....	49
5.2.2.2	APSEE-Planner.....	49
5.2.2.3	Mecanismo de execução de processos.....	49
5.2.2.4	APSEE-Reuse	50
5.2.3	Interação com o Usuário.....	50
5.3	Implementação do Modelo APSEE em PROSOFT-Java	52
5.3.1	Ambiente PROSOFT-Java	52
5.3.2	Ambiente APSEE	53
5.3.3	Editor de processos de software	53
5.3.4	Mecanismo de execução de processos do APSEE	55
5.3.5	Editor e interpretador de políticas de instanciação.....	57
5.4	Considerações sobre o Capítulo	57
6	UM MODELO PARA PROCESSO DE CURSO	58
6.1	Requisitos do Modelo	58
6.2	Trabalhos Relacionados	60
6.2.1	Hyper-Automaton.....	60
6.2.2	EduQNet.....	61
6.2.3	Modelos baseados em técnicas de workflow.....	62
6.2.4	Arquitetura baseada em agentes	64
6.2.5	Um Modelo Baseado em Projetos	65
6.2.6	Abordagem ISD.....	66
6.2.7	Análise Comparativa	67
6.3	Modelo Proposto	67
6.3.1	Atividades.....	68
6.3.1.1	Análise de Requisitos.....	69

6.3.1.2	Projeto do Curso	69
6.3.1.3	Instanciação	70
6.3.1.4	Execução do Curso	71
6.3.1.5	Avaliação	71
6.3.1.6	Aquisição de Conhecimento	72
6.3.2	Artefatos	72
6.3.3	Participantes	73
6.4	Vantagens e Limitações do Modelo.....	74
6.5	Considerações sobre o Capítulo	75
7	AVALIAÇÃO DO MODELO	76
7.1	O Protótipo PRO-EAD.....	76
7.1.1	Classes do PRO-EAD.....	77
7.1.2	Arquitetura do PRO-EAD	78
7.1.2.1	Meta-modelo.....	78
7.1.2.2	Mecanismo de Gerência de Processos	79
7.1.2.3	Nível de Interação com o Usuário	81
7.1.3	Funcionamento	81
7.1.3.1	Definição dos Requisitos	81
7.1.3.2	Projeto do Curso	82
7.1.3.3	Instanciação	83
7.1.3.4	Execução.....	84
7.2	Metodologia de Avaliação.....	87
7.2.1	Seleção da Amostra	87
7.2.2	Questionário	87
7.2.3	Aplicação do instrumento.....	88
7.2.4	Análise dos Dados	89
7.3	Considerações sobre o Capítulo	93
8	CONCLUSÃO.....	95
8.1	Contribuições	95
8.2	Análise dos Resultados	96
8.3	Questões em aberto e perspectivas para o futuro.....	97
	REFERÊNCIAS.....	98
	ANEXO A DIAGRAMAS DE PACOTES UML	114
	ANEXO B INSTRUMENTO DE AVALIAÇÃO UTILIZADO NO ESTUDO DE CASO	116
	ANEXO C CURSOS MODELADOS PELOS DOCENTES PARTICIPANTES DA AVALIAÇÃO DO PRO-EAD.....	119
	APÊNDICE A AVALIAÇÃO DE CURSOS A DISTÂNCIA.....	122

LISTA DE ABREVIATURAS E SIGLAS

ADS	Ambiente de Desenvolvimento de Software
APSEE	Ambiente de Desenvolvimento de Software do grupo PROSOFT
ATOs	Ambiente de Tratamento de Objetos
APSEE-PML	Linguagem de Modelagem de Processos do Ambiente APSEE
CMM	<i>Capability Maturity Model</i>
COPLE	<i>Cooperative Project Learning Environment</i>
CSCL	<i>Computer-Supported Collaborative Learning</i> – Aprendizagem Colaborativa Suportada por Computador
EAD	Educação a Distância
GQM	<i>Goal, Question, Metric</i>
HTML	<i>Hyper Text Markup Language</i>
IHEP	<i>Institute for Higher Education Policy</i>
ISD	<i>Instrucional Systems Design</i>
ISO	<i>International Standard Organization</i>
LPCEL	<i>Learning Processes Composition and Execution Languages</i> – Linguagens para Composição e Execução de Processos de Aprendizagem
PML	<i>Process Model Language</i> – Linguagem para Modelagem de Processos
PRO-EAD	Processo de Educação a Distância
PROSOFT	Ambiente para Desenvolvimento de Software
PSEE	<i>Process-Centered Software Engineering Environment</i> - Ambiente de desenvolvimento de software centrado em processo
TABA	Ambiente de Desenvolvimento de Software Orientado a Processo
UFRGS	Universidade Federal do Rio Grande do Sul
WF	<i>Workflow</i>

LISTA DE FIGURAS

Figura 3.1: Exemplo de estrutura de um curso a distância.....	23
Figura 4.1: Arquitetura conceitual.....	33
Figura 4.2: Domínios de processos de software	35
Figura 5.1: Exemplos dos construtores de classes no PROSOFT.	41
Figura 5.2: Classe APSEE.....	42
Figura 5.3: Classes APSEE-Types e Types no APSEE	43
Figura 5.4: Exemplo de hierarquia de tipos de recursos.	43
Figura 5.5: Classes APSEE, <i>Processes</i> e <i>ProcessModel</i>	44
Figura 5.6: Visão geral do modelo APSEE.....	45
Figura 5.7: Visão funcional de processos de software no ambiente APSEE	51
Figura 5.8: APSEE-Monitor: visão informacional.....	52
Figura 5.9: Editor de processos do APSEE	54
Figura 5.10: Formulário para definição de detalhes de uma atividade.....	54
Figura 5.11: Formulário para definição de agentes em uma atividade.....	55
Figura 5.12: Formulário para seleção de processos para execução.....	56
Figura 5.13: Agenda do desenvolvedor.....	56
Figura 5.14: Editor de políticas do APSEE.....	57
Figura 6.1: Curso modelado como Máquina de Moore.....	61
Figura 6.2: Autoria de um curso em EAD.....	63
Figura 6.3: Arquitetura para ambientes de aprendizagem.....	64
Figura 6.4: Abordagem ISD	66
Figura 6.5: Modelo de Processo de Curso.....	68
Figura 6.6: Diagrama de Casos de Uso	73
Figura 7.1: Classe ProEadAPSEE	77
Figura 7.2: Classe ProEad	77
Figura 7.3: Classe ProEadProcesses.....	78
Figura 7.4: Arquitetura do PRO-EAD.....	79
Figura 7.5: Símbolos da Linguagem APSEE-PML.....	80
Figura 7.7: Tela inicial do PRO-EAD	81
Figura 7.8: Definição dos processos de curso no PRO-EAD.....	82
Figura 7.9: Modelagem dos processos de curso no PRO-EAD.....	82
Figura 7.10: Instanciação de processos de curso no PRO-EAD.....	83
Figura 7.11: Formulário para definição de Agentes	83
Figura 7.12: Visão global de um processo de curso não iniciado	84
Figura 7.13: Visão global de um processo de curso em execução	85
Figura 7.14: Visão do aluno de um processo em execução.....	85
Figura 7.15: Visualização do curso pelo aluno, com atividade pronta para execução ...	86
Figura 7.16: Visão do curso pelo aluno, com uma atividade ativada e outra em pausa.	86

LISTA DE TABELAS

Tabela 6.1: Requisitos para Gerência de Cursos a Distância	58
Tabela 6.2: Análise dos Trabalhos Relacionados	67
Tabela 7.1: Análise dos Questionários de Avaliação	89
Tabela 8.1: Quadro Comparativo com os Trabalhos Relacionados	96

RESUMO

A Educação a Distância (EAD) vem recebendo atenção dos pesquisadores de várias áreas, na busca de modelos e ferramentas que possam aumentar a eficiência desta modalidade de educação. Mas, ferramentas tecnologicamente avançadas não são suficientes para isso. A atuação do docente é fundamental para o sucesso de um curso a distância.

O problema é que falta preparo a muitos professores que atuam em EAD, para planejar os cursos, estimar recursos e organizar o conteúdo. Além da dificuldade na criação dos cursos, os professores também enfrentam o problema de avaliar os cursos já oferecidos.

Esta tese apresenta um modelo para gerência de cursos a distância. O modelo de “Processo de Curso”, nomenclatura proposta neste trabalho, engloba o projeto, criação, execução e avaliação de cursos a distância. O modelo proposto pretende ser uma alternativa de solução para a seguinte questão de pesquisa:

Que elementos um modelo, embasado pela Engenharia de Software, precisa conter para representar as atividades envolvidas na gerência de cursos a distância?

A definição desse modelo foi fundamentada em duas áreas distintas: A Ciência da Computação e a Informática na Educação, mais especificamente na Engenharia de Software e na Educação a Distância. Como a tese baseia-se na analogia de Processo de Software e Processo de Curso e, por isso, o estudo da área de Tecnologia de Processo de Software foi de fundamental importância.

O modelo de Processo de Curso é constituído pelas atividades que compõem um curso a distância (projeto, execução, avaliação e outras), os agentes que realizam essas atividades, produtos gerados e recursos necessários para a realização da atividade. Um dos destaques dessa abordagem é a possibilidade de reutilização de cursos anteriores, utilizando conceitos herdados da Engenharia de Software.

A comprovação da viabilidade de implementação do modelo foi realizada através da implementação, no ambiente PROSOFT-APSEE, de um protótipo para gerência de cursos a distância. Para avaliar o modelo e o protótipo, foram selecionados professores especialistas que modelaram cursos no PRO-EAD e responderam a um questionário de avaliação

Acredita-se que as contribuições deste trabalho tragam avanços significativos na busca de métodos e ferramentas que venham a auxiliar os professores na criação de cursos a distância com mais qualidade.

Palavras-Chave: Educação a Distância, Processo de Curso, Processo de Software.

A Course Process Model

ABSTRACT

Researchers in many academic areas have been devoting increasing attention to Distance Learning (DL), aiming at designing models and tools that can enhance the efficiency of this kind of instruction. However, technologically advanced tools alone are not enough to achieve that goal; the instructor's performance is also crucial to the success of this type of enterprise.

Currently, the greatest challenge lies on overcoming the lack of expertise on the part of the professionals teaching Distance Learning courses, specially with regard to course planning, cost estimates and content organization. Besides the difficulties implied in designing the courses, educators also face the problem of evaluating the ongoing courses.

The present study presents a model for managing Distance Learning courses, named "Course Process", which comprises the project, creation, implementation and assessment of Distance Learning courses. The proposed model is expected to provide answers to the following question:

Which elements are required for a model to represent the activities involved in managing Distance Learning courses?

The definition of this model finds its grounds on two distinct fields: Computer Science and Education, more specifically Software Engineering and Distance Learning. The study of Software Process Technology was of paramount importance to the elaboration of the present thesis, for it is based on the analogy of Software Process and Course Process.

The Course Process model is constituted by the activities that comprise a Distance Learning course (project, implementation, assessment and others), the agents that perform these activities, their outcomes and the resources required to their accomplishment. One of the strengths of the proposed approach is the possibility of reusing prior courses, applying concepts inherited from the Software Engineering.

The viability of the model was validated by means of the implementation, in the PROSOFT-APSEE environment, of a prototype to the management of Distance Learning courses. Teachers with previous experience in distance learning were invited to designer courses at the PRO-EAD and to answer a questionnaire so as to evaluate both the model and the prototype.

It is believed that the contributions of this study provide significant advances in the search for helpful methods and tools to educators engaged in creating more qualified Distance Learning courses.

Keywords: Distance Learning, Course Process, Software Process

1 INTRODUÇÃO

A educação a distância vem crescendo nos últimos anos de uma forma vertiginosa devido, principalmente, à utilização da Internet. Muitos cursos a distância, via Internet, têm sido oferecidos por instituições no mundo todo. Embora esta modalidade de educação venha demonstrando ser muito promissora, muitos desses cursos têm repetido as velhas fórmulas do ensino tradicional, tornando-se experiências mal sucedidas de ensino a distância.

Estes cursos virtuais¹ mal sucedidos são, na maioria das vezes, mal projetados e gerenciados. Falta preparo aos professores, em sua grande maioria, para planejar os cursos, estimar recursos e organizar o conteúdo para esta nova modalidade de educação.

Neste contexto, a busca por modelos e ferramentas que possam melhorar a qualidade dos cursos em Educação a Distância (EAD) tem sido objeto de pesquisa em vários grupos. Apesar de todos os esforços, muitas questões ainda estão em aberto.

Esta tese situa-se na área de Engenharia de Software aplicada na Educação a Distância. Pela característica interdisciplinar do tema, o levantamento bibliográfico abordou duas áreas: Ciência da Computação e Informática na Educação. Apesar da amplitude deste contexto, o trabalho está focado nos estudos da gerência de cursos a distância e na área de Tecnologia de Processo de Software, cujos conceitos foram empregados na definição de um modelo para Processo de Curso.

1.1 Contextualização

O enfoque pedagógico da nova perspectiva de EAD fundamenta-se nos princípios do construtivismo, tendo como premissa fundamental a idéia de que o indivíduo é agente ativo de seu próprio conhecimento (STRUCHINER et al., 1998).

É importante ressaltar que o paradigma educacional construtivista, cujo eixo central é a organização das interações professor-aluno tendo o aprendiz como centro do processo, requer uma atenção maior na elaboração do material que é disponibilizado a ele (MOORE; KEARSLEY, 1996; ROSENBERG, 2002; HORTON, 2000; RODRIGUES, 1998). Enquanto em uma sala de aula tradicional (presencial) o professor usa a criatividade para manter a atenção do aluno, no ambiente virtual isto é um desafio. Explorar e oferecer alternativas para resolver esta questão e outras envolvendo a avaliação e organização de materiais tem sido objeto de estudo de pesquisadores do PPGCC da UFRGS (DAHMER et al., 2001; DAHMER; VICCARI;

¹ Esta tese está focada na educação a distância via Internet. Por isso, o termo “cursos virtuais” é considerado sinônimo de cursos a distância.

NUNES, 2005; SIZILIO; EDELWEISS, 2001) e de outros grupos (HANNEY; WATSON, 2005; TSUKAHARA; SEKI; OKAMOTO, 2005; TORRES; DODERO; ZARRAONANDIA, 2005; KARAMPIPERIS; SAMPSON, 2005; HASEGAWA; OCHIMIZU, 2005).

Além de desconsiderar aspectos mais subjetivos, muitos dos cursos não consideram aspectos bem mais simples, tais como: estimar mal as atividades propostas, propor tarefas pouco estimulantes, apresentar o assunto de forma confusa e desconexa, entre outros (DAHMER; GASPARY; FROZZA, 1999). Esses fatores levam à perda de interesse e contribuem para o baixo rendimento do aluno (GAGNÉ, 1999).

Outra questão importante refere-se à avaliação destes cursos. Muitas referências apresentam estratégias para avaliação dos alunos em educação a distância (MCCORMACK, 1998; FUKS et al., 2003; DAHMER; GASPARY; FROZZA, 1999; BERCHT, 1997). Poucas, porém abordam a questão da avaliação dos cursos (WESSON, 2002; JONSON, 2005; AEDO; DIAZ, 2000; HORTON, 2001) e da reutilização dos resultados obtidos nos mesmos para a elaboração de novos (BENIGNO; TRENTIN, 2000; SANTORO; BORGES; SANTOS, 2002).

Para possibilitar o domínio do processo de desenvolvimento e gerência dos cursos a distância e para que o curso funcione como planejado é necessário que exista uma definição do modelo do processo de criação, execução e avaliação de cursos. Esta representação do curso permitirá a análise, compreensão e automatização do seu processo.

Este trabalho apresenta uma proposta de um modelo para processos de cursos a distância, onde se propõe a utilização das experiências prévias de avaliação de cursos já ministrados como apoio na criação de novos cursos.

O foco principal deste trabalho refere-se a este contexto. Definir um modelo para as etapas envolvidas em um curso a distância. Para isso, é proposta uma analogia entre processo de curso e processo de software. Da área de Engenharia de Software foram resgatados conceitos que embasam o modelo, como as definições de ciclo de vida de processo de software (DERNIAME; KABA; WASTELL, 1999) e de fábrica de experiências (BASILI, 1994).

1.2 Motivação

A Internet proporciona vários recursos para disponibilização de material didático, comunicação e cooperação. Estes recursos oferecem diversas possibilidades para educação a distância, permitindo a construção do conhecimento de uma forma dinâmica e personalizada, avançando em relação a outros recursos utilizados anteriormente nesta modalidade de educação (TESSAROLLO, 2000).

Entretanto, criar cursos na Internet que utilizem esses recursos eficientemente é uma tarefa muito complexa e que exige bastante empenho por parte dos professores.

O principal papel do professor na EAD é planejar o curso e encorajar uma participação efetiva dos alunos (CASTRO, 2002). Sob o ponto de vista da educação este papel não é novo. O desafio é fazer isto de forma virtual, sem contar com a interação presencial e com os fatores intervenientes do contato pessoal.

Em aulas a distância não há espaço para imprevisto ou imprevistos, daí a importância de planejar muito antes de oferecer um curso virtual. A criação de um curso a distância

envolve alguns passos a serem seguidos, como o planejamento didático das atividades educacionais e dos conteúdos a serem abordados, o público ao qual será dirigido o mesmo e o objetivo do curso em questão.

Assim, novos desafios surgem aos educadores. Eles precisam estar preparados para esta nova dinâmica e, muitas vezes, esses docentes não possuem a experiência necessária para isso. Não estão aptos para explorar ao máximo as potencialidades na criação e gerência de cursos a distância. Foi com o propósito de dar apoio a estes professores que surgiu a motivação para este trabalho.

Outros fatores que motivaram este trabalho foram:

- Existem diversos trabalhos que propõem estruturas para representação de cursos (MACHADO, J., 2000; SIZILIO, 2000; JUNG, 2001; ANDRADE, 2003; KRAMER et al., 1999; SANTORO; BORGES; SANTOS, 2002; KUNDE; SOUTO; OLIVEIRA, 2001; TORRES, 2005). Rapchan et al. (2002) propõem um modelo de qualidade de processo para cursos a distância e Tavares et al. (2001) apresentam uma abordagem orientada a processos para ambientes de aprendizagem. Entretanto, não foi encontrada nenhuma referência que fizesse a analogia entre processo de curso e processo de software. Existem muitos pontos análogos entre as duas áreas e esta semelhança precisa ser explorada;
- Um modelo que utilize as experiências passadas para auxiliar na criação de novos cursos pode trazer maior produtividade ao professor, pois, com a tarefa de planejamento simplificada, o professor pode dedicar-se ao acompanhamento individual dos alunos, ao incentivo da criatividade dos alunos e à promoção de debates, desenvolvimento e análise de experimentos e implementações dos conteúdos (MACHADO, J., 2000);
- A questão de avaliação de cursos a distância ainda é considerada um desafio, pela dificuldade em saber o que é importante e válido avaliar, em como definir critérios para realizar a avaliação (RODRIGUES, 1998; BENIGNO; TRENTIN, 2000; HORTON, 2001; MCCORMACK, 1998; SOONG et al., 2001, MOURA, 2002; JONSON, 2005). Um modelo que destaque a etapa de avaliação dos cursos pode contribuir para o desenvolvimento da área.

Portanto, a partir destes pontos, decidiu-se focalizar o trabalho na elaboração de um modelo para gerência de cursos a distância baseado na analogia entre Processo de Curso e Processo de Software.

Um modelo de Processo de Software é uma descrição formal do processo de software e tem como função indicar quais os passos a serem executados, quem os executará, como, quando, onde e o porquê de sua realização (LONCHAMP, 1993). Para isso, o modelo é constituído por atividades que precisam ser realizadas, agentes que executam as atividades, produtos e recursos necessários para a realização da atividade (SOUSA, 2003).

Fundamentada por esses conceitos, esta tese propõe um modelo para Processo Curso, descrito no capítulo 6. Esse modelo descreve as etapas que compõem o processo, os atores envolvidos, recursos utilizados e produtos gerados. Espera-se, com este modelo, minimizar as dificuldades encontradas pelos professores, onde o próprio docente possa planejar seus cursos através da recuperação de experiências de cursos anteriores.

Para realizar a avaliação do modelo, foi implementado um protótipo que demonstra a viabilidade de concretização do modelo em um ambiente de desenvolvimento de software orientado a processos. O ambiente utilizado foi o APSEE (LIMA REIS, 2003), ferramenta desenvolvida no âmbito do projeto PROSOFT. Maiores detalhes sobre os ambientes PROSOFT e o APSEE podem ser encontrados na seção 4.6 e no capítulo 5.

1.3 Organização do Texto

Este texto foi organizado em 8 capítulos.

O capítulo 1 apresentou a contextualização e motivações da tese.

O capítulo 2 descreve como foi conduzida a pesquisa para desenvolvimento desta tese. O tema, problema de pesquisa, objetivos e método utilizado no trabalho são detalhados.

O capítulo 3 aborda a Educação a Distância, com ênfase nos aspectos referentes às características e etapas de um curso a distância. Além disso, são discutidas as mudanças que ocorrem no papel do professor que passa a atuar em EAD.

O capítulo 4 apresenta alguns conceitos sobre Tecnologia de Processo de Software. Esta área foi utilizada como base para o modelo proposto e os conceitos apresentados são de extrema importância para compreensão da proposta.

O capítulo 5 descreve o ambiente APSEE-PROSOFT, utilizado na implementação do protótipo.

O capítulo 6 apresenta a contribuição principal desta tese, referente ao processo de um curso para EAD. Primeiramente, alguns trabalhos relacionados são discutidos e depois o modelo de Processo de Curso é descrito com detalhes.

No capítulo 7 o protótipo PRO-EAD, desenvolvido para auxiliar na avaliação do modelo proposto, é apresentado. Também é descrita a avaliação realizada com professores especialistas.

O capítulo 8 descreve as conclusões do trabalho. As contribuições e limitações do trabalho são detalhadas.

Finalmente, as referências bibliográficas utilizadas para realização deste trabalho são apresentadas.

2 METODOLOGIA DE PESQUISA

Pesquisa é o conjunto de procedimentos sistemáticos, baseado no raciocínio lógico, que tem por objetivo encontrar soluções para problemas propostos, mediante a utilização de métodos científicos (ANDRADE, 2003).

A pesquisa científica pode ser classificada quanto à natureza, aos objetivos e aos procedimentos (JUNG, 2004).

De acordo com essa definição, a pesquisa realizada nesta tese classifica-se como fundamental, explicativa e realizada através de estudo de caso:

- Fundamental: teve por finalidade analisar, explicar e descrever um sistema real através de um modelo que poderá ser utilizado, posteriormente, em pesquisas aplicadas ou tecnológicas.
- Explicativa: objetivou criar uma teoria aceitável a respeito de um fenômeno, assim como definir um modelo teórico que descrevesse esse fenômeno.
- Estudo de caso: foi definido um procedimento de pesquisa que investigou o fenômeno dentro de um contexto real.

Um trabalho de pesquisa passa por várias etapas que estão interligadas, visando enquadrar o assunto em diversas etapas metodológicas: escolha do tema, delimitação do tema, justificativa da escolha, formulação do problema, formulação das hipóteses, definição dos objetivos, levantamento de dados, definição do método de pesquisa, análise dos dados, discussão dos resultados, conclusão (FACHIN, 2001; CERVO; BERVIAN, 2002; SANTOS, 2000).

2.1 Delimitação do Tema

O tema de uma pesquisa é o assunto que se deseja desenvolver e que mereça ser investigado cientificamente (LAKATOS; MARCONI, 2001). Delimitar a pesquisa é essencial para estabelecer limites para a investigação.

O tema desta tese é a gerência de cursos em Educação a Distância.

Como apresentado na introdução deste trabalho, a gerência de cursos a distância é uma tarefa complexa que demanda um planejamento detalhado e pesquisas que possam avançar na busca de modelos eficientes de gerência em EAD, justificando, assim, sua relevância para centrar este trabalho.

2.2 Questão de Pesquisa

A questão ou problema de pesquisa é uma dificuldade, teórico ou prática, no conhecimento de alguma coisa de real importância, para o qual se deve encontrar uma solução. Enquanto o tema de uma pesquisa é uma proposição abrangente, a formulação da questão de pesquisa é mais específica, indica exatamente a dificuldade que se pretende resolver (LAKATOS; MARCONI, 2001).

Assim, após a identificação da área e da problemática envolvendo a gerência de cursos em EAD, e considerando a riqueza de modelos e técnicas disponíveis na área de Engenharia de Software, foi elaborada a seguinte questão de pesquisa:

Que elementos um modelo, embasado pela Engenharia de Software, precisa conter para representar as atividades envolvidas na gerência de cursos a distância?

2.3 Hipótese

Hipótese é uma proposição que se faz na tentativa de verificar uma resposta existente para determinado problema (LAKATOS; MARCONI, 2001).

Relacionada a esta questão, a seguinte hipótese foi formulada:

Um modelo que descreve as atividades de gerência de cursos a distância deve oferecer elementos análogos aos que compõem o modelo do ciclo de vida de um processo de software, e pode ser implementado em um ambiente de desenvolvimento de software orientado a processos;

A hipótese foi formulada a partir da observação de semelhanças entre o processo de cursos a distância e de software. Portanto, conforme Parra Filho (2001), é denominada hipótese por analogia.

A hipótese deve passar por dois processos importantes: formulação e teste (FACHIN, 2001). A etapa de teste da hipótese, formulada neste trabalho, foi realizada através da definição de um modelo baseado na hipótese e da posterior avaliação deste modelo.

2.4 Objetivos

Após delimitar a questão e formular a hipótese de pesquisa, os objetivos deste trabalho foram definidos.

2.4.1 Geral

O objetivo principal deste trabalho é apresentar uma proposta de modelo para processo de curso, visando auxiliar o professor, que atua em EAD, a criar cursos a distância de qualidade.

2.4.2 Específicos

Para atingir este objetivo geral, foram observados os seguintes objetivos específicos:

- Definir uma proposta de estrutura para cursos em EAD, para a qual o modelo será proposto;

- Analisar modelos existentes para cursos a distância a fim de identificar os elementos comuns;
- Identificar, através do estudo da área de Tecnologia de Processo de Software, os conceitos relevantes para a definição do modelo proposto;
- Definir o modelo para gerência de cursos em EAD e representá-lo;
- Selecionar um ambiente de gerência de processo de software para concretização do modelo;
- Implementar um protótipo para avaliar o modelo proposto;
- Avaliar as potencialidades e limitações do modelo proposto através de um estudo de caso.

2.5 Fundamentação Teórica

Como o tema da pesquisa é a gerência de cursos a distância e a hipótese de pesquisa se relaciona à área de Tecnologia de Processo de Software, a fundamentação teórica deste trabalho está focada em duas áreas:

- Engenharia de Software, mais especificamente o estudo do processo de desenvolvimento de software;
- Educação a Distância.

O estudo da educação a distância centrou-se na organização de cursos e atividades envolvidas neste processo. Já na Engenharia de Software, a análise do processo de desenvolvimento de software foi centrada nas etapas e agentes envolvidos.

Assim, para fundamentar o modelo proposto, este trabalho apresenta estudos sobre os principais conceitos de educação a distância e tecnologia de processo de software. Descreve também, o ambiente APSEE-PROSOFT, escolhido como plataforma para implementação do protótipo utilizado para avaliar o modelo proposto.

2.6 Método de Pesquisa

Método de pesquisa é o caminho a ser trilhado pelos pesquisadores em busca do conhecimento (ANDRADE, 2003).

O método de pesquisa utilizado nesta tese foi o método por analogia que permite concluir, de certas semelhanças observadas, outras não observadas *a priori* (PARRA FILHO; SANTOS, 2001).

O tipo de analogia utilizada foi da natureza às qualidades, pois a partir da semelhança dos processos de curso e de software foram definidas as características das etapas e componentes do processo de curso.

Esse método de pesquisa não leva a uma certeza, mas a uma provável conclusão. A verificação do raciocínio analógico pode ser feita por demonstração.

O capítulo 6, que descreve o modelo proposto para processo de curso, apresenta esta demonstração. Através da descrição das etapas do modelo, é demonstrado que não existem diferenças, entre os dois processos definidos como análogos, que possam

enfraquecer esta premissa. Os estudos realizados demonstram apenas as semelhanças entre os dois processos.

2.7 Avaliação e Análise dos Dados

Para realizar a avaliação do modelo foi implementado um protótipo, o PRO-EAD, e realizado um estudo de caso, visando identificar as contribuições do modelo e suas limitações.

Segundo Freimut et al. (2002), um estudo de caso é um tipo de estudo observacional, ou seja, é realizado pela observação de projeto ou atividade em andamento. Durante a realização de um estudo de caso, diferentes procedimentos de coleta de dados podem ser utilizados (questionários, entrevistas, etc) o que favorece a obtenção de dados qualitativos e/ou quantitativos.

A diferença entre estudo de caso e experimentos é que nos experimentos as variáveis de estudo podem ser controladas, por outro lado, um estudo de caso ocorre em um ambiente típico (i.e., ambiente real onde ocorre o problema sob estudo) não sendo possível o controle e manipulação das variáveis (YIN, 1994; WOHLIN; HÖST; HENNINGSSON, 2003).

O estudo de caso, nesta tese, foi implementado através da utilização do PRO-EAD, por professores com experiência em Educação a Distância e conhecimentos de Engenharia de Software. Após esta etapa, foi aplicado um instrumento de coleta de dados (questionário) para avaliar o modelo e o protótipo.

Após o recebimento dos questionários preenchidos pelos sujeitos selecionados, foi realizada a análise dos dados obtidos.

Esta etapa da pesquisa e os resultados obtidos serão descritos com maiores detalhes e discutidos no capítulo 7 desta tese.

2.8 Considerações sobre o Capítulo

Este capítulo apresentou a metodologia de pesquisa desta tese e descreveu as etapas seguidas para o desenvolvimento da pesquisa,

É importante destacar a analogia entre processo de curso e processo de software, o método de pesquisa que guiou o desenvolvimento do trabalho e que norteou a formulação da hipótese de pesquisa. Também foi descrito o estudo de caso, realizado para testar essa hipótese.

Como o tema desta tese é a gerência de cursos a distância, o próximo capítulo apresenta alguns conceitos fundamentais para o modelo proposto neste trabalho.

3 EDUCAÇÃO A DISTÂNCIA

Este capítulo visa apresentar alguns dos conceitos básicos da Educação a Distância, principal área de concentração desta tese. Como este trabalho propõe um modelo para gerência de cursos em EAD, torna-se necessário, primeiramente, caracterizar e definir a estrutura desses cursos.

3.1 Cursos a Distância

Com o surgimento dos computadores e da internet, surgiu um novo meio capaz de implementar a educação a distância, mais poderoso do que os anteriores. A Internet permite a comunicação entre alunos e professores de uma forma ágil (SILVA, 2001). Esta comunicação bidirecional ainda não é possível com o rádio e a televisão e é ineficiente no ensino por correspondência, devido ao tempo que se passa entre o envio e o recebimento das cartas (CUNHA; KIENBAUM; OLIVEIRA, 2000).

O uso da Internet como meio de comunicação possibilitou que várias pessoas acessassem salas de aulas virtuais, bibliotecas on-line num espaço compartilhado. Novas possibilidades de transmissão de informação e interação entre professores e alunos se tornaram viáveis. A EAD através da Internet mistura ferramentas de comunicação assíncronas e as síncronas.

- Ferramentas de comunicação assíncronas: permitem uma aprendizagem do tipo auto-estudo, através de recursos que podem se adequar aos horários disponíveis que o aprendiz possui para estudo. Nesta categoria encontram-se o correio eletrônico, lista de discussão e fórum.
- Ferramentas de comunicação síncrona: essa modalidade de comunicação oferece ambientes para discussão em tempo real, através de salas de aula virtuais utilizando uma combinação de métodos. Dentre essas ferramentas pode-se citar: bate-papo (chat) e videoconferência.

O acompanhamento da comunicação, realizada através dessas ferramentas, não é uma tarefa simples. Vários trabalhos propõem estratégias e mecanismos para monitoração e avaliação da comunicação em EAD (ARAÚJO; LUCENA FILHO, 2005; CINELLI et al., 2005; SILVA; FEIJÓ, 2002; OTSUKA; ROCHA, 2002)

A intensidade da comunicação pode tornar os programas de EAD mais ou menos distantes de seus alunos, devendo ser dirigida com o maior empenho para que a distância entre os participantes do processo tenha menor significado e influência.

Para facilitar a execução de cursos através da web, são utilizados ambientes computacionais de apoio à educação a distância como, por exemplo: WEBCT², Teleduc³, Aulanet⁴, Moodle⁵ (SANTOS; MELLO; SEGRE, 2003). Esses sistemas permitem uma maior interação aluno/professor, um melhor controle pelo professor e uma gestão adequada no contexto da organização.

As principais características de um bom ambiente para EAD são: interface com boa usabilidade (facilidade de uso e aprendizado); disponibilidade de um conjunto de ferramentas que facilitem o aprendizado, a comunicação e a colaboração (listas, chats, e-mail, glossário, entre outras); e um conjunto de ferramentas administrativas objetivando controle acadêmico (ZAMBALDE; ALVES; SANTOS, 2001).

3.2 Estrutura de um Curso em EAD

Existem diversos formatos que podem ser adotados em cursos a distância (MACHADO, J., 2000; SIZILIO, 2000).

Neste trabalho será adotado um conceito baseado no trabalho de Machado, L. (2003). Um curso será entendido como uma seqüência de atividades relacionadas (Figura 3.1). Essas atividades podem ser:

- Módulos de conteúdo: estas atividades compreendem módulos de exposição de conteúdos. Nestes módulos, pressupõe-se o acesso a páginas *web* ou outros tipos de documentos, especialmente confeccionados para apresentação do conteúdo focado pelo módulo.
- Atividades de produção coletiva ou individual: atividades que produzem como resultado documentos ou outros formatos programados. Podem ser atividades individuais ou coletivas.

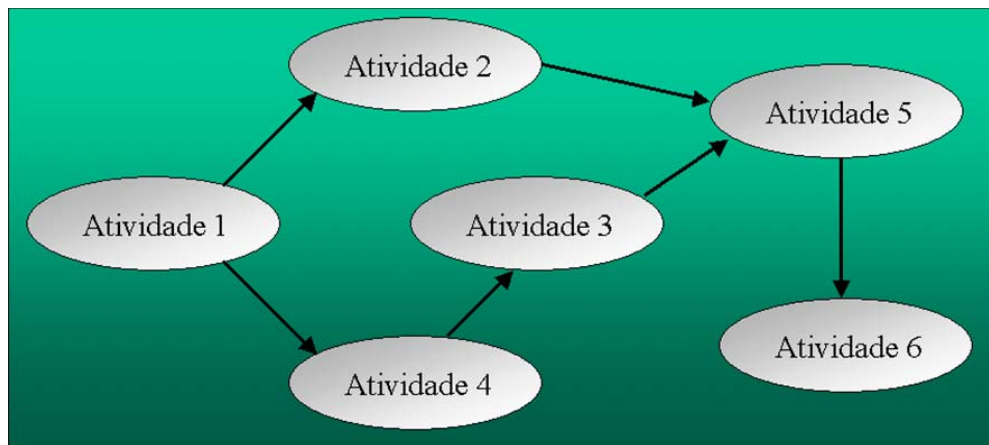


Figura 3.1: Exemplo de estrutura de um curso a distância

² <http://www.webct.com>

³ <http://teleduc.nied.unicamp.br>

⁴ <http://www.eduweb.com.br>

⁵ <http://www.moodle.org>

- Atividades Interativas: constituem as atividades que pressupõem o uso de recursos computacionais para efetivar comunicações síncronas ou assíncronas (chat, fórum de discussão, etc). As atividades interativas são coletivas por natureza. Como afirma Demo (2004), entre outros, um curso em EAD, requer uma mescla inteligente de presença física e virtual. Seguindo esta premissa, no modelo proposto, considera-se que momentos presenciais também podem integrar um curso, sendo classificados como atividades interativas síncronas, sem necessidade de alteração no modelo do curso.
- Avaliações: um curso pode ter diversas avaliações. É importante ressaltar que essas avaliações podem ser virtuais ou presenciais. Por exemplo, segundo a legislação brasileira, as avaliações em disciplinas de cursos de graduação realizadas na modalidade a distância devem ser presenciais.

As atividades que compõem um curso estão relacionadas, de acordo com a relação de dependência que existe entre elas. Neste trabalho, esta dependência define quando uma atividade pode ser liberada para execução.

3.3 Caracterização da EAD

Caracterizada pelo distanciamento físico entre educador e aluno, a educação a distância está em constante crescimento, o que implica em novas tecnologias desenvolvidas para acompanhar este desenvolvimento (SANTOS, 2002). Primeiramente vinculada à iniciativa de alguns professores, já na segunda metade do século XIX, a EAD começa a existir institucionalmente. No século XX, o que se observou foi um contínuo movimento de consolidação e expansão da educação a distância. Sob o aspecto quantitativo ampliou-se o número de países, de instituições, de cursos, e de alunos envolvidos com esta modalidade de educação (LOBO NETO, 1998).

Moore e Kearsley (1996) identificam na evolução da EAD, 3 gerações: estudo por correspondência; cursos através de televisão, rádio e telefone; e EAD baseada em redes de computadores. É importante ressaltar que não ocorreu a substituição de uma alternativa pela outra, as novas alternativas foram sendo incorporadas, ajustando as anteriores e criando novos modelos. Até hoje um grande percentual de cursos a distância são conduzidos por correspondência.

A terceira geração de cursos a distância, foco deste trabalho, utiliza o computador e a Internet, que viabiliza mecanismos para os estudantes se comunicarem de forma síncrona (salas de *chat*) e assíncrona (fóruns de discussão, correio eletrônico) (MCISAAC; RALSTON, 1997). A Internet trouxe grandes benefícios para Educação a Distância, pois permite a troca de informação de forma muito eficiente e rápida, não existindo problemas de localização geográfica, oferecendo ótimas alternativas ao sistema de ensino presencial.

Descrições detalhadas da evolução Educação a Distância no mundo podem ser encontrados em diversos trabalhos (ALVES, 1994; MOORE; KEARSLEY, 1996; MATTELART, 1994; LANDIM apud RODRIGUES, 1998; NUNES, 1992; HOLMBERG, 1981; PRETI, 1996; ARETIO, 1994; BELLONI, 2001; PETERS, 2003). A história da EAD no Brasil também pode ser consultada em diversas publicações (PIMENTEL, 1995; NISKIER, 1993; ALVES, 1994; NUNES, 1992; ALONSO, 1996; MATTELART, 1994; PRETI, 1996; KRAMER, 1999; TAJRA, 2002; SILVA, 2003; ELIASQUEVICI, 2004).

Diversos autores apresentam definições para Educação a Distância (MOORE; KEARSLEY, 1996; HOLMBERG, 1981; PETERS, 1973; PRETI, 1996; ALBES; NOVA, 2003). De um modo geral, as definições são descritivas, definindo a EAD a partir do ensino convencional da sala de aula. O termo comum entre elas é a distância, entendida em termos de espaço. A separação entre professores e alunos não está explícita em algumas das definições, mas constitui-se em fator determinante no processo de ensino e aprendizagem a distância.

Peters apud Nunes (1992), em 1973, utilizou conceitos da economia e da sociologia industrial para definir EAD. Ele incluiu a separação entre professor e estudante e o uso de meios técnicos como elementos essenciais do EAD, mas foi além, examinando os modos de organização dos sistemas e sua estrutura didática. Segundo ele, a EAD podia ser melhor entendida a partir de princípios que regem a produção industrial, especialmente os de produtividade, divisão do trabalho e produção de massa. Hoje Peters já trabalha com conceitos mais atuais, afirmando que a Educação a Distância informatizada permite acompanhar as mudanças sociais da sociedade “pós-industrial” (PETERS, 2003).

Nesses conceitos mais modernos já são considerados como elementos essenciais em EAD o uso de meios tecnológicos e a existência de uma estrutura organizacional. Alguns outros autores citam a segmentação do ensino em duas etapas: preparação e desenvolvimento da aula (BELLONI, 2001) e a necessidade de uma cuidadosa organização e existência de tutoria, propiciando uma aprendizagem independente e flexível para os alunos (ARETIO, 2001).

Com uma definição bem objetiva, Moore e Kearsley (1996) afirmam que o conceito fundamental da Educação a Distância é simples: alunos e professores estão separados pela distância e algumas vezes também pelo tempo.

Baseando-se nessas diversas definições, Aretio (2001) cita os elementos que são essenciais para uma definição clara da EAD:

- Separação aluno-professor: professor e aluno encontram-se separados no espaço e, na maior parte das vezes, no tempo;
- Organização de apoio-tutoria: a tutoria em Educação a Distância possibilita um acompanhamento do andamento do curso e das dificuldades encontradas pelos alunos;
- Aprendizagem independente e flexível: Através de um planejamento cuidadoso do processo ensino-aprendizado, a tecnologia de EAD possibilita o trabalho independente e a individualização da aprendizagem;
- Comunicação bidirecional: A EAD busca uma atividade educativa bidirecional, com feedback entre professor e aluno;
- Enfoque tecnológico: A educação é otimizada pela tecnologia quando vista sob uma concepção científica, sistemática e globalizadora. Faz-se necessário um planejamento sistemático.
- Comunicação massiva: A transmissão/recepção de mensagens é facilitada pelas novas tecnologias da informação e pelos modernos meios de comunicação, propiciando o aproveitamento destas mensagens por grande número de pessoas, dispersas geograficamente. Embora a comunicação massiva seja uma possibilidade da EAD, esta modalidade pode estar direcionada, também, a

minorias e, inclusive, a um só aluno. Os autores atuais inclusive desaconselham a massificação da educação a distância.

Enfim, a EAD se apresenta na esfera pedagógica como mais uma opção metodológica. Ela traz consigo características próprias que impõem a necessidade de novas aprendizagens por parte de quem a planeja, desenvolve e avalia, implicando, inclusive, na necessidade de que seja construída uma nova maneira de compreender o processo de ensino-aprendizagem (LEITE; SILVA, 1998).

3.4 Gerência de Cursos a Distância

Como apresentado na seção anterior, a Internet oferece vários recursos para acesso a informações, comunicação e cooperação. Entretanto, gerenciar cursos na Internet que utilizem esses recursos eficientemente e de forma integrada é uma tarefa muito complexa e que exige bastante empenho dos professores. São necessários conhecimentos que vão além do conteúdo e da maneira de conduzir o curso. A criação de um curso a distância envolve alguns passos a serem seguidos, como o planejamento didático das atividades educacionais e das aulas que o curso abordará, a quem será dirigido e qual será o objetivo do curso.

Vários autores apresentam sugestões para elaboração e gerência de cursos a distância (PRETI, 1996; WILLIS, 1996; MOORE; KEARSLEY, 1996; PORTER, 1997; ROSENBERG, 2002; HORTON, 2000). Estas abordagens sugerem analisar detalhadamente, durante o planejamento, o público alvo, objetivos a serem atingidos e conteúdo a ser desenvolvido no curso. Além disso, os recursos disponíveis devem ser avaliados.

Segundo Rodrigues (1998), a gerência de um curso a distância envolve quatro etapas: planejamento, produção dos materiais, implementação e avaliação.

- Planejamento: envolve a definição dos objetivos, do tema; estruturação do conteúdo em módulos; definição da linguagem; programação das atividades dos alunos, com cronograma de realização; definição da forma de avaliação dos alunos e do curso e das estratégias de suporte aos alunos (RODRIGUES, 1998). A etapa de planejamento é de suma importância para a EAD (BERGE, 2001). Incorporar, durante o curso, etapas ou recursos que não tenham sido previstos no planejamento poderá ser difícil ou inviável, principalmente se envolver alterações significativas no orçamento e/ou cronograma (MOORE; KEARSLEY, 1996). É preciso projetar, antecipar, prever, antever; é preciso definir em que condições e com que instrumentos vão se estabelecer as relações educativas; quem são as pessoas envolvidas e até que ponto seus perfis podem interferir no resultado pretendido (ANDRADE, 1997). É importante ressaltar que os professores que estão iniciando na área de EAD, geralmente não estão familiarizados e não possuem um conhecimento prévio sobre o planejamento e organização de um curso a distância.
- Produção de materiais didáticos: a produção dos materiais que serão utilizados no curso e a previsão dos possíveis encontros presenciais é tarefa que demanda cuidado e tempo. A produção de materiais para cursos a distância normalmente é realizada por grupos multidisciplinares (MOORE; KEARSLEY, 1996). A distribuição dos materiais também é fundamental ao sucesso do curso

(RODRIGUES, 1998). Os materiais precisam estar disponíveis a todos os alunos com antecedência suficiente para que possam ser acessados com tempo.

- **Implementação:** esta etapa compreende o curso propriamente dito: as aulas, os trabalhos, os seminários, as interações, as aplicações dos testes. Tudo o que foi planejado e preparado é colocado em prática. A partir desta etapa a participação dos alunos influencia a condução do curso.
- **Avaliação:** esta etapa visa analisar se o curso atingiu seus objetivos. Na avaliação é verificado se tudo o que foi planejado foi implementado. Além disso, é necessário avaliar se o objetivo pretendido (a aprendizagem dos alunos) foi atingido. A avaliação pode considerar dados coletados durante a implementação do curso e outros, obtidos através de questionários respondidos por alunos e professores. A etapa de avaliação é fundamental para a qualificação dos programas de EAD e nos últimos anos tem recebido atenção de vários pesquisadores da área (AEDO; DIAZ, 2000; SOONG et al., 2001; BONAMY; CHARLIER; SAUNDERS, 2001; MOURA, 2002; WESSON, 2002; JONSON, 2005).

3.5 Processo de Curso

Educação a distância implica grande habilidade de organização e **controle processual**, em particular quando os estudantes se dispersam pelo país, também em lugares de difícil acesso eletrônico ou a materiais adequados de pesquisa (DEMO, 2004).

Neste trabalho, as diversas etapas envolvidas na criação, execução e avaliação de cursos a distância foram denominadas “Processo de Curso”. Este processo abrange todas as etapas descritas nas seções anteriores, além de outras etapas incluídas nesta proposta.

O modelo deste processo é apresentado com detalhes no capítulo 5 e é uma das contribuições deste trabalho.

Como este modelo foi desenvolvido focando as necessidades de um professor que atua em EAD, a próxima seção aborda as funções do docente na Educação a Distância.

3.6 O Papel do Docente na EAD

Diferentemente do senso comum, o trabalho e a função do professor não desaparecem com a utilização da *Web* na EAD, mas precisam ser transformadas para esta nova modalidade. Educar hoje é mais complexo porque a sociedade e as competências necessárias também o são (SILVA, 2005).

Segundo Ferreira (1998), três elementos possuem importância fundamental para eficiência de qualquer programa de educação a distância:

- **projeto:** métodos de ensino tradicionais não podem ser utilizados na EAD. Novos projetos de aprendizado precisam ser desenvolvidos para tirar proveito dos benefícios trazidos pelas novas tecnologias;
- **tecnologia:** a forma de utilização da tecnologia disponível pode ser o motivo do sucesso ou do fracasso;

- apoio: fornecer auxílio aos participantes pode incentivá-los a desenvolver ou expandir caminhos eficientes e efetivos.

Estes três elementos se relacionam, principalmente, à atuação do professor na EAD. Com isso, as responsabilidades do professor são inúmeras.

Segundo Almeida (1998), um programa de EAD eficiente consiste no esforço integrado de alunos e professores. Atingir as necessidades do aluno é o ponto principal do ensino a distância eficiente e o objetivo para o qual todos os esforços feitos devem ser focados (PALLOF; PRAT, 2004). Mas o sucesso de qualquer programa de educação a distância reside nas aptidões do professor. Na EAD o professor precisa ter muitas aptidões que envolvem questões pedagógicas e tecnológicas.

Nessas abordagens fica clara a importância da atuação do professor na busca pela eficiência em EAD. Os professores que nunca atuaram em programas de educação a distância muitas vezes receiam em atuar nesta modalidade por não conhecê-la e não possuírem experiência na criação de cursos virtuais.

Este trabalho é direcionado para as necessidades dos professores que não estão familiarizados com os métodos e as tecnologias disponíveis, buscando servir de apoio para o desenvolvimento de cursos a distância.

O professor precisa concentrar suas ações na coordenação das atividades, dirigindo o aprendizado, esclarecendo ou gerando questões. Também cabe ao professor desenvolver conteúdos que sejam estimulantes para o aprendizado. Esta situação coloca o professor não mais como um transmissor de conhecimentos, mas como um elemento coordenador da aquisição de conhecimentos pelo aluno (MACHADO, J., 2000).

O professor em EAD também precisa estar sempre atualizado tecnologicamente para produzir cursos de qualidade, disponibilizando o conteúdo da melhor maneira e para isso se utilizando das diversas opções tecnológicas oferecidas. Para que seja possível realizar atividades pedagógicas através da Internet é necessário que os especialistas da educação preencham uma série de requisitos no que concerne ao uso de ferramentas ligadas à informática. Além do empenho a longo prazo, é preciso ultrapassar obstáculos técnicos e assimilar uma série de informações (HEIDE; STILBORNE, 2000).

Os docentes precisam ensinar e aprender utilizando esses novos espaços virtuais de aprendizagem (MAIA, 2001). Eles precisam saber integrar as tecnologias de forma inovadora (MORAN; MASETTO; BEHRENS, 2000).

Essa mudança na postura do professor que atua a distância, em relação ao professor presencial, é sempre questionada por todos aqueles que desejam trabalhar neste novo espaço suportado pelas novas tecnologias.

Num primeiro momento, novas tecnologias são uma novidade para o professor e isto requer adaptação em termos operacionais (GRINSPUN, 1999). O professor precisa aprender a operar os equipamentos, trabalhar com software e assimilar conceitos e vocabulário próprios de uma nova área.

Além disso, o uso dessas tecnologias leva a novas experiências em um sentido mais profundo. O mundo da comunicação mediada por computador funciona em espaço e tempo diversos da comunicação de oralidade primária e da cultura escrita (AZÊVEDO, 1999).

Os professores precisam saber “como fazer” EAD. Ensinar a distância é muito diferente de ensinar presencialmente, mesmo para professores com larga experiência em ensino. São necessárias diferentes habilidades de apresentação da informação e de planejamento, desenvolvimento e avaliação de estratégias de ensino nas quais professor e aluno estejam distantes fisicamente. Além do mais, é necessário dominar a tecnologia disponível (LEITE; SILVA, 1998). O educador que tiver intimidade e se sentir confortável com as tecnologias e suas aplicações educacionais, poderá transmitir naturalmente a seus alunos a aplicabilidade e as vantagens de sua utilização na construção do conhecimento (CHAVES, 2005).

Antes de adotar qualquer modelo para a EAD, deve-se ter a preocupação com o preparo e a adaptação do professor. Porque ele terá que aprender a lidar com novas tecnologias. E é com esse aprendizado que o professor poderá aplicar eficientemente as tecnologias, sabendo avaliar qual o melhor momento de dispor delas e com que finalidade. Além disso, muitas vezes os professores precisam se adaptar às características de tempo e realidade do aprendiz (RODRIGUEZ et al, 1999).

Segundo Leite e Silva (1998), o professor que se propõe a ensinar em sistemas de EAD deve refletir sobre alguns aspectos fundamentais:

- Contexto de ensino: o ambiente de aprendizagem assume nova configuração devido à separação física entre os participantes do processo. O professor precisa reconhecer essa mudança no ambiente e sua influência no contexto, trabalhando com as potencialidades do ambiente e adaptando-as às limitações impostas pela abordagem educacional;
- Alunos: em programas de EAD eles visualizam a aprendizagem de maneira diversa do ensino presencial, portanto têm uma perspectiva diferente. O professor precisa estar atento aos obstáculos psicológicos, sociais e técnicos a serem enfrentados pelos alunos de cursos a distância.
- Métodos: os educadores, em EAD, precisam ser cuidadosos em não reaplicar métodos tradicionais de ensino presencial, pois precisam reconhecer que eles não podem ser simplesmente utilizados na EAD. Há necessidade de serem exploradas estratégias alternativas de ensino, contextualizadas no ambiente de EAD. Os métodos de ensino de EAD devem buscar reduzir a distância interpessoal, promovendo a interação e facilitando a aprendizagem.

Almeida (1997) também destaca alguns aspectos que devem ser considerados pelo docente que busca qualidade no processo de educar a distância:

- Aperfeiçoar a organização: a representação do conteúdo a ser disponibilizado ao aluno usuário da tecnologia de EAD deve ser desenvolvida e adaptada conforme as necessidades individuais de cada aluno.
- Suprir as necessidades dos alunos: tanto em termos de conteúdo apresentado, quanto em termos de métodos de aprendizagem, exige-se um esforço adicional por parte do professor para que o processo de aprendizagem seja realizado de forma eficiente para os diferentes perfis de alunos existentes.
- Usar métodos de ensino eficientes: a eficiência, bem como o sucesso, associado à Educação a Distância depende, em geral, do constante aperfeiçoamento das metodologias de ensino aplicadas.

- Investir e apostar na interação e no “feedback”: permite a identificação das necessidades individuais de cada aluno e realizar discussões ricas em sugestões propostas pelos alunos.

Por tudo isso, o trabalho do professor em EAD não se torna mais leve do que no ensino presencial. Ao contrário, o controle do conteúdo didático e o acompanhamento do aprendizado exigem um esforço expressivo, em todas as etapas do processo. Também, a elaboração da nova formatação do conteúdo didático utilizado pode ser uma tarefa bastante árdua. O desafio fundamental para os pesquisadores da área de EAD se torna, então, intensificar esforços no desenvolvimento de recursos que façam uso mais efetivo do novo paradigma apresentado pela evolução da WWW e de suas tecnologias relacionadas (BORONI; GOOSEY apud MACHADO, J., 2000).

Enfim, a tarefa de preparar cursos para disponibilizar na *Web* é complexa, pois exige um grande empenho por parte do professor. O desenvolvimento de aulas para EAD, para alguns professores é uma tarefa muito difícil de realizar, porque eles têm pouco ou nenhum conhecimento sobre Informática.

Este trabalho busca auxiliar o docente na sua adaptação a novas tecnologias. O modelo proposto (descrito no capítulo 5) busca oferecer ao professor um conjunto de etapas que o oriente a gerenciar seu curso em abordagem de EAD.

3.7 Considerações sobre o Capítulo

Este capítulo apresentou alguns conceitos sobre Educação a Distância que foram utilizados na definição do processo e na conseqüente proposta de um modelo para cursos em EAD.

Além disso, as considerações sobre as atribuições de um professor que atua em educação a distância, apresentadas na seção 1.4, demonstram a necessidade de pesquisas sobre modelos e ferramentas que possam facilitar a atuação do professor. Este trabalho busca alcançar este objetivo através do modelo proposto.

Neste trabalho é proposta a analogia entre processo de curso a distância e processo de software. Para que essa analogia seja melhor compreendida, o próximo capítulo aborda a área de Tecnologia de Processo de Software.

4 TECNOLOGIA DE PROCESSO DE SOFTWARE

Dentre as muitas áreas que constituem a Ciência da Computação, está a Engenharia de Software, a qual trata dos aspectos tecnológicos e gerenciais do processo de desenvolvimento de software (REIS, 2002).

O desenvolvimento de software vem evoluindo muito nos últimos anos, mas ainda são freqüentes os problemas, prejuízos financeiros e acidentes causados por falhas em sistemas (FUGGETTA, 2000). Para minimizar estes problemas surgiram diversos estudos sobre a concepção de técnicas e métodos para entender e melhorar a qualidade do desenvolvimento do software. Neste contexto, a tecnologia de processo de software tem se destacado por ressaltar uma forte relação entre a qualidade do software e a qualidade do processo de desenvolvimento (SOUSA, 2003).

A tecnologia de processo de software surgiu em meados da década de 1980 e tem suas origens relacionadas com as primeiras propostas de um ciclo de vida para o software, isto é, a definição de um conjunto bem definido de estágios que forneçam uma referência para o acompanhamento do desenvolvimento (LIMA REIS, 2003).

Esta área de pesquisa representou um importante passo em direção à melhoria da qualidade do software através de mecanismos que proporcionam o gerenciamento automatizado do desenvolvimento de software. Diversas teorias, conceitos, formalismos, metodologias e ferramentas surgiram nesse contexto, enfatizando a descrição formal do modelo de processo de software, para que possa ser automatizado por um ambiente integrado de desenvolvimento de software. Os modelos de processos de software descrevem o conhecimento de uma organização e, portanto, modelos que descrevem experiências bem sucedidas devem ser continuamente disseminados para reutilização em diferentes projetos (REIS, 2002).

Este capítulo apresenta os principais conceitos envolvidos na tecnologia de processo de software. Estes conceitos fundamentaram o modelo para um processo de curso, apresentado nesta tese.

4.1 Conceituação

A tecnologia de processos de desenvolvimento de software é uma importante área de estudo e pesquisas na Engenharia de Software. Ela envolve a construção de ferramentas e ambientes para modelagem, execução, simulação e evolução de processos de desenvolvimento de software. Tais ambientes são denominados ambientes de

desenvolvimento de software (ADS) orientados ao processo ou PSEEs (*Process-Centered Software Engineering Environments*) (LIMA, 1998).

A tecnologia de processo de software pode ser definida como um conjunto de políticas, estruturas organizacionais, tecnologias e atividades, necessárias para o entendimento, desenvolvimento e manutenção de um produto de software (FUGGETA, 2000). Alguns dos principais artigos da área podem ser encontrados em (GARG, 1996), aspectos mais recentes são abordados em (DERNIAME; KABA; WASTELL, 1999), (GRUHN, 2002) e (ARBAOUI et al., 2002).

É importante ressaltar que, devido a grande quantidade de pesquisas em tecnologia de processo de software, existem muitas definições e terminologias nesta área (LIMA REIS, 2003). Alguns esforços no sentido de padronizar estas definições são apresentados em (DOWSON; NEJMEH; RIDDLE, 1991; FEILER; HUMPHREY, 1993; LONCHAMP, 1993). Além disso, existem diversos paradigmas de modelagem de processo de software, bem como de execução de tais modelos. Este trabalho utiliza os conceitos descritos abaixo, adotadas pelo grupo PROSOFT, em trabalhos anteriores (REIS, 2002; LIMA REIS, 2003; SOUSA, 2003).

4.1.1 Processo de software

O processo de desenvolvimento de software (processo de software) pode ser compreendido como o conjunto de todas as atividades necessárias para transformar os requisitos do usuário em software (HUMPHREY, 1989; OSTERWEIL, 1987). Essas atividades estão relacionadas com conjuntos de artefatos, pessoas, recursos, estruturas organizacionais e restrições (LONCHAMP, 1993; DOWSON; NEJMEH; RIDDLE, 1991).

O processo que tem como objetivo alcançar um produto de software é representado a partir de um modelo de processos de software (SOUSA, 2003).

Já um modelo de processo de software é uma descrição formal do processo de software e tem como função indicar quais os passos a serem executados, quem os executará, como, quando, onde e o porquê de sua realização (LONCHAMP, 1993).

Um modelo é constituído por atividades (análise, projeto, revisão, etc) que precisam ser realizadas, agentes (humanos ou não) que realizam as atividades, produtos (diagramas de análise, algoritmos, manuais, etc), e recursos (software, equipamentos técnicos, etc), necessários para a realização da atividade (SOUSA, 2003).

Na representação de um modelo de processo de software são usadas linguagens de modelagem de processos de software ou PML (*Process Model Language*). As PMLs possuem algumas características específicas, que as distanciam das linguagens de propósito geral. Pode-se destacar, por exemplo, a necessidade de suportar atividades que necessitam da interação humana para serem concluídas (GIMENES, 1994).

Diversas classificações são encontradas na literatura com o objetivo de facilitar o estudo das linguagens de processo de software. O leitor interessado em maiores detalhes sobre PMLs pode conferir as diferentes abordagens presentes na literatura em: (REIS, 1999; GARG; JAZAYERI, 1996; AMBRIOLA; CONRADI; FUGGETTA, 1997).

4.2 Ambientes de Desenvolvimento de Software Orientados a Processos

Como já foi dito, um ADS que permite a modelagem e execução de processos de software é denominado ADS orientado ao processo (GIMENES, 1994) ou PSEE. Os PSEEs suportam a criação e exploração de modelos de processos de software. Nestes ambientes o processo para produzir produtos de software pode ser definido explicitamente pelo usuário através de uma PML (SOUSA, 2003).

Além disso, um PSEE deve auxiliar o usuário na monitoração do processo, automatizando algumas partes e direcionando o processo de execução (GARG; JAZAYERI, 1996).

Um PSEE deve possuir alto nível de integração entre suas ferramentas e ser capaz de executar um modelo de processo de software através da coordenação dos agentes na execução de suas tarefas, gerência de alocação de recursos, coleta de métricas, execução automática de algumas atividades e mudança do processo durante sua execução (LIMA, 2000).

Assim, é possível identificar uma arquitetura de alto nível para PSEEs, dividindo-o em três camadas conceituais (Figura 4.1).

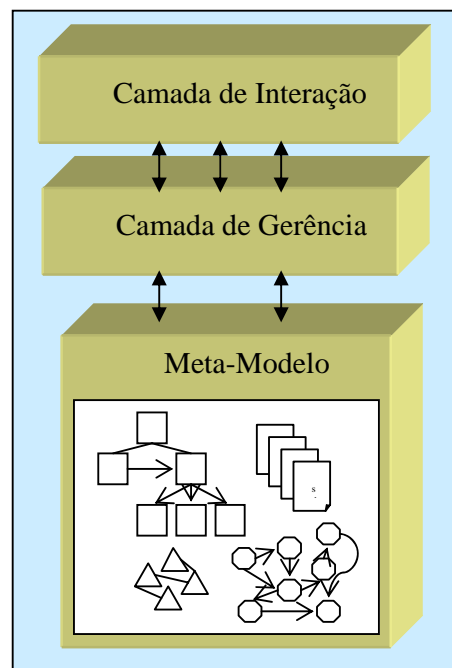


Figura 4.1: Arquitetura conceitual (SOUSA, 2003)

A camada de interação apresenta os mecanismos de interação com os usuários: ferramentas para a visualização e representação do modelo de processos, ferramentas de groupware, etc. A camada de gerência oferece mecanismos para controle e execução de processos: máquina de processos, regras de execução, etc. E o meta-modelo corresponde ao repositório das definições semânticas da linguagem de modelagem aplicadas aos componentes do ambiente (entidades como recursos, ferramentas, agentes, políticas, etc.) (SOUSA, 2003).

O uso de PSEEs para automatizar a gerência dos processos traz inúmeros benefícios. Entre eles destaca-se: melhor comunicação entre as pessoas envolvidas; realização de algumas ações automáticas; fornecimento de informações sobre o andamento do processo quando necessário; possibilidade de reutilização de processo de software e a coleta automática de métricas (LIMA REIS, 2003).

Diversos PSEEs são apresentados na literatura. Pode-se destacar: MARVEL (KAISER; FELLER; POPOVICH, 1988), EPOS (CONRADI, 1994), SPADE (BANDINELLI et al, 1994), ENDEAVORS (TAYLOR; BOLCER, 1996) e DYNAMITE (HEIMANN; KRAPP; WESTFECHTEL, 1997). No país, algumas propostas foram desenvolvidas no âmbito de ambientes de modelagem e gerência de processos de software. Dentre elas, pode-se citar o ambiente ExpSEE (GIMENES, 2000) que permite “definição precisa e instanciação de atividades do processo de software em relação a artefatos, atores (agentes) e ferramentas utilizadas por atividades”; e a Estação TABA (TRAVASSOS, 1994) que é um ambiente de desenvolvimento de software que apóia a execução das atividades a serem desempenhadas em um processo de software, através de um conjunto de ferramentas integradas e repositórios contendo informações adquiridas durante a execução do processo (ROCHA; TRAVASSOS, 2005). Outros exemplos podem ser encontrados em (ASSISTANT-GDPA, 2003; POHL et al., 1999; LIMA, 1998).

4.3 Domínios de Processo de Software

O envolvimento humano na execução do processo adiciona a imprevisibilidade, pois não há como determinar a ausência de futuros atrasos, e ainda é necessário permitir alterações nos processos em execução para adicionar uma nova característica não prevista (SOUSA, 2003). Esta característica do processo pode gerar discrepâncias entre as definições e a execução real do processo.

Assim, existem três diferentes domínios de processos de software, apontados em (DOWSON; FERNSTRÖM, 1994):

- o domínio das definições de processo (*definition*): contém modelos de processos ou fragmentos desses modelos, expressos de forma prescritiva em alguma notação;
- o domínio da realização do processo (*performance*): engloba as atividades ou ações conduzidas por pessoas e agentes não-humanos (programas);
- e o domínio da execução do processo (*enactment*): preocupa-se com o que é necessário em um ambiente de desenvolvimento de software para suportar a execução de uma definição de processo; isto é essencialmente um mecanismo de interpretação do modelo de processos por uma máquina de processos.

Na figura 4.2 são ilustrados os domínios de processo e seus relacionamentos.

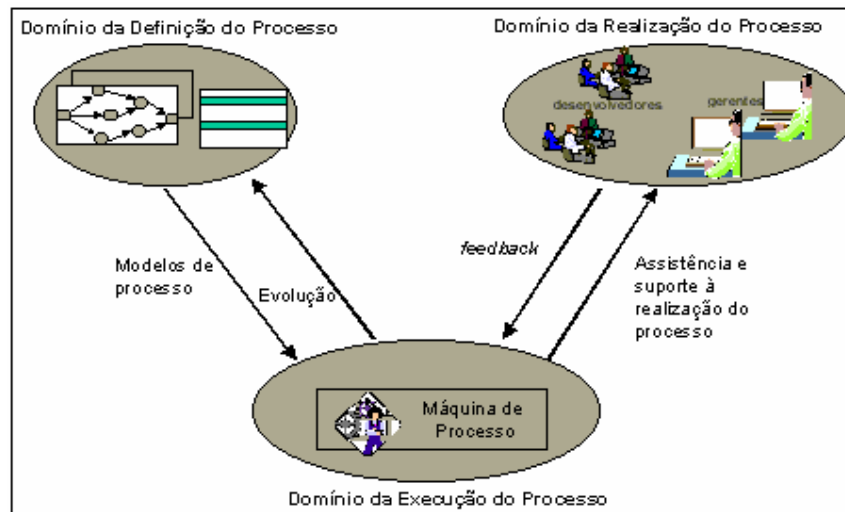


Figura 4.2: Domínios de processos de software (SOUSA, 2003)

4.4 Ciclo de Vida de Processos de Software

Assim como para produtos de software, existe um ciclo de vida para processos de software. As atividades deste ciclo são chamadas de meta-atividades, e o ciclo de vida é chamado de meta-processo de software (DERNIAME; KABA; WASTELL, 1999).

O ciclo de vida de um processo de software é composto pelas seguintes etapas (REIS, 2002; LIMA REIS, 2003):

- Provisão de Tecnologia: engloba o fornecimento de tecnologia de suporte a produção de software e de modelos de processo;
- Análise de Requisitos do Processo: visa descrever os requisitos para descrição inicial do processo (envolvendo detalhes do processo e/ou do produto a ser desenvolvido);
- Projeto do Processo: provê a arquitetura geral e detalhada do processo. As linguagens de modelagem de processos são utilizadas nesta etapa. Esta etapa resulta em um modelo de processo abstrato;
- Instanciação do Processo: modifica a especificação do processo produzida pela atividade anterior acrescentando informações detalhadas sobre os prazos, agentes e recursos utilizados por cada atividade definida no processo. Esta etapa produz um processo instanciado a partir do modelo abstrato;
- Simulação do Processo: visa a verificação e validação dos processos definidos antes da execução, permitindo antever problemas de alocação e estimar a sua duração. Como resultado da fase de validação, pode ser necessário retornar a fases anteriores para aperfeiçoamento do modelo;
- Execução do modelo de processo: utiliza o processo instanciado e o executa através de ferramentas para guiar e assistir a realização do processo no mundo real. Informações sobre o andamento do processo (feedback) são coletadas e analisadas durante a execução. Nesta etapa, a máquina de processos coordena a interação com gerentes e desenvolvedores e obtém *feedback* sobre o andamento do processo;

- Avaliação do Processo: provê informação quantitativa e qualitativa que descreve o desempenho de todo o processo em execução. Nesta etapa novos requisitos podem ser gerados a partir do registro das ocorrências da execução. A avaliação pode ocorrer em paralelo com a execução do modelo de processo e as informações adquiridas podem ser utilizadas nas futuras ocorrências da atividade de análise de requisitos.

O meta-processo conta com a participação de agentes humanos (operam na fase de execução do processo), um projetista de processo (responsável por descrever o processo a ser executado) e um gerente do processo (acompanha a execução e a avaliação do processo) (LIMA REIS, 2002).

Em geral, os PSEEs estabelecem seu meta-processo através da provisão de tecnologia e paradigmas adotados para modelagem e execução de processos. A modelagem, execução e reutilização de processos são detalhadas a seguir.

4.5 Modelagem de Processo de Software

O processo de software envolve atividades complexas desempenhadas por pessoas (agentes⁶) com as mais diversas capacidades, sendo controladas por gerentes de desenvolvimento (REIS, 2002).

Um modelo de processo de software é uma descrição formal do processo de software. Vários tipos de dados são integrados em um modelo de processo para indicar quem, quando, onde, como e porque os passos são realizados (LONCHAMP, 1993). Para representar um modelo de processo de software é freqüentemente utilizada uma PML, a qual deve oferecer recursos para descrever e manipular os passos do processo.

A literatura da área distingue três tipos de modelos (DERNIAME; KABA; WASTELL, 1999; FEILER; HUMPHREY, 1993):

- Modelos Abstratos (ou *templates*), que fornecem moldes de solução para um problema comum, em um nível de detalhe que idealmente não está associado a uma organização específica. Um processo abstrato é um modelo de alto nível que é projetado para regular a funcionalidade e interações entre cargos de desenvolvedores, gerentes, usuários e ferramentas em um PSEE (WANG; KING, 2000);

- Modelos Executáveis⁷ (ou instanciados) são modelos prontos para execução, podendo ser submetidos à execução por uma máquina de processo. O modelo instanciado é um modelo de processo com características específicas do projeto em questão no contexto da organização de desenvolvimento de software envolvida. São exemplos de informações necessárias para instanciação de processos: a alocação de recursos, a definição dos desenvolvedores e a descrição de prazos das atividades (LIMA REIS, 2003).

- Modelos em Execução ou Executados registram o histórico da execução de um processo, incluindo os eventos e desvios (modificações realizadas no modelo associado).

⁶ Agentes, no contexto de processo de software, se refere aos participantes do processo: gerentes, desenvolvedores, etc. É importante destacar que, nesse contexto, não há nenhuma relação com os agentes definidos na área de Inteligência Artificial.

⁷ Do original em inglês *enactable* (RIBÓ; FRANCH, 2002).

Assim, a modelagem de processos de software é freqüentemente associada a sua execução, descrita na seção a seguir.

4.5.1 Execução de modelos de processo

Para automatizar a gerência do desenvolvimento, os modelos de processo de software devem ser executados. Na fase de execução, devem ser levadas em consideração as questões de coordenação de múltiplos usuários e a interação desses com as ferramentas disponíveis. Para isso, é necessário que se obtenha um modelo de processo executável, ou seja, um modelo que descreva o processo com tal nível de detalhe que permita a sua execução por uma máquina. Portanto, a execução de modelos de processo de software deve levar em consideração as questões de coordenação de múltiplos usuários, assim como a interação entre as ferramentas e os agentes envolvidos no processo (REIS, 2002).

A modelagem de processos de software não consiste tão somente em escrever programas que automatizem completamente o processo de desenvolvimento de software; tampouco descreve tudo o que as pessoas do processo devem fazer (REIS, 1999). Os programas de processo são escritos para definir possíveis padrões de comportamento entre pessoas e ferramentas automatizadas (TULLY, 1989). Por conseqüência, um PSEE deve permitir que as pessoas envolvidas no processo recebam orientação automatizada e assistência na realização de suas atividades, sem interferência no processo criativo (NGUYEN; WANG; CONRADI, 1997).

A execução de processos de software ocorre quando um modelo de processo está pronto para execução (está instanciado) e leva em consideração diversos fatores como a coordenação dos desenvolvedores, interação entre as ferramentas e desenvolvedores, garantia de que o processo está sendo executado conforme modelado, dentre outras questões (LONCHAMP, 1993; DERNIAME; KABA; WASTELL, 1999; FEILER; HUMPHREY, 1993). Com o apoio de um ambiente de desenvolvimento de software é possível obter resultados relevantes acerca do andamento da execução além de melhorar a gerência do desenvolvimento. Portanto, apesar de não ser totalmente automatizada, a fase de execução depende de um mecanismo automatizado que compreenda o processo modelado e oriente os desenvolvedores (agentes) no decorrer do seu trabalho, assim como seja responsável por executar automaticamente algumas tarefas repetitivas.

A arquitetura de um PSEE usualmente define como componente central a Máquina de Processo (do inglês *process engine*) que auxilia na coordenação das atividades realizadas por pessoas e por ferramentas automatizadas, sendo responsável pela interpretação/execução dos modelos de processos descritos com PMLs. Segundo Lima Reis (1998), uma máquina de processos é responsável por: ativar automaticamente atividades sem intervenção humana através de uma integração com as ferramentas do ambiente; apoiar o envolvimento cooperativo dos desenvolvedores; monitorar o andamento do processo e registrar o histórico da sua execução. A máquina de processo também deve garantir a execução das atividades na seqüência definida no modelo; a repetição de atividades; a informação de feedback sobre o andamento do processo; a gerência das informações sobre ele; a coleta automática de métricas; a mudança do processo durante sua execução; a interação com as ferramentas do ambiente e a gerência de alocação de recursos (LIMA REIS, 1998).

O Mecanismo de Execução de processos ou Máquina de Execução (Process Engine) interpreta o modelo de processo instanciado de acordo com a semântica da linguagem

de modelagem (HUFF, 1996), gerenciando as informações do ambiente e orientando os desenvolvedores de acordo com esse modelo. Um dos objetivos principais desse mecanismo é manter o estado da execução do processo consistente com o estado real da realização das tarefas. Além disso, monitoração e coleta de métricas sobre o processo são atividades relacionadas com esse mecanismo.

4.5.2 Reutilização de processos de software

O termo reutilização de processos de software define uma ampla área de estudo e aplicação prática relacionada aos diferentes aspectos envolvidos com a reutilização do conhecimento adquirido na condução de projetos anteriores. Estes estudos envolvem a definição de formalismos e ferramentas para facilitar a construção de modelos de processos a partir de processos e componentes já existentes. A evolução dessa área foi decisivamente influenciada pelos avanços nos estudos da reutilização de produtos de software e da tecnologia de processo de software (REIS, 2002).

A reutilização de processos de software pode trazer economia, tanto no custo quanto no prazo de entrega na produção do software em série, quando comparado com produtos independentemente produzidos. Atualmente, muitos processos similares são executados por diferentes organizações de desenvolvimento de software, surgindo a oportunidade de reutilizar a experiência adquirida durante o planejamento e condução de projetos anteriores, a fim de determinar melhores estratégias gerenciais em projetos futuros (REIS, 2002).

Para permitir a criação de uma base de conhecimento que possa ser reutilizada em projetos futuros, várias informações precisam ser registradas. Além das especificações de requisitos, projetos, programas, documentos, casos de teste e outros tipos de artefatos comumente produzidos, o conhecimento adquirido sobre o domínio da aplicação a partir das decisões gerenciais precisa ser armazenado (REIS, 2002).

A tarefa de descrever elementos reutilizáveis para processos de software não é equivalente ao problema de reutilização de produtos de software. Um modelo de processo de software envolve elementos relacionados com aspectos sociais, organizacionais, tecnológicos e ambientais.

A área de reutilização de processos de software recebeu muito mais atenção de pesquisadores e da indústria na segunda metade da década de 1990. Mecanismos vêm sendo aperfeiçoados e experimentados para o armazenamento e a definição de elementos reutilizáveis de processos. Essa evolução vem sendo motivada principalmente pela necessidade de novas abordagens no sentido de entender os processos correntes em uma organização, assim como no sentido de promover novas estratégias gerenciais (ELLMER et al., 1996). Alguns trabalhos apresentam propostas para reutilização de processos de software (REIS, 2002; SHENG; YUN, 2003; GANNOD et al., 2004).

4.6 Tecnologia de Processo de Software no Projeto PROSOFT

O projeto PROSOFT (NUNES, 1992; NUNES, 1994), desenvolvido no PPGC-UFRGS e na Universidade de Stuttgart (Baden-Württemberg, Alemanha) sob orientação do Prof. Dr. Daltro José Nunes, teve suas atividades iniciadas há mais de dez anos e tem como foco a construção de um ambiente integrado de desenvolvimento de software. Em

linhas gerais, o PROSOFT objetiva fornecer ferramentas que auxiliem na utilização de métodos de especificação formal em todas as fases do ciclo de vida de software. Esse ambiente vem servindo de laboratório para a experimentação de tecnologias que apóiam o desenvolvimento de software com alta qualidade e produtividade.

O PROSOFT, enquanto ambiente para construção de software, visa auxiliar o desenvolvimento de software desde as etapas iniciais - concepção e especificação de requisitos do produto - até a implementação do sistema, seguindo rigorosos padrões de qualidade.

A utilização de métodos formais é enfatizada na construção das ferramentas do ambiente a partir de um paradigma próprio, criado pelo próprio projeto, o que permite realizar verificações automáticas e provas matemáticas acerca da qualidade do produto em construção. Sob este paradigma, ferramentas CASE (Computer Aided Software Engineering, ou Engenharia de Software Apoiada por Computador) foram desenvolvidas para apoiar a utilização dos métodos formais e semi-formais para a construção de software, sendo então integradas ao ambiente.

Desse modo, uma recente linha de pesquisa do projeto PROSOFT envolve estudantes de pós-graduação e pesquisadores das instituições envolvidas na definição de ferramentas para apoiar a modelagem (REIS, 2001), execução (LIMA REIS, 1998), simulação (SILVA, 2001), e visualização (SOUSA, 2003) de processos cooperativos (REIS, 1998; REIS; LIMA REIS; NUNES, 1998) de desenvolvimento de software, constituindo uma estrutura integrada para automação de processos de software que apresenta o meta-modelo a ser utilizado por todas as ferramentas do ambiente. Esta estrutura é denominada APSEE (REIS, 2001; LIMA REIS, 2001).

Os mais recentes avanços agregados ao APSEE-PROSOFT foram:

- Um meta-modelo como contribuição para a tecnologia de processo de software, especificamente com o aumento do nível de automação fornecido na reutilização de modelos. Em Reis (2002) é apresentado um conjunto de ferramentas que apóiam especificamente a modelagem de processos reutilizáveis.
- Sousa (2003) apresenta um mecanismo que facilita a visualização do modelo de processos durante sua execução, permitindo assim uma representação adequada e operações que auxiliem na gerência e monitoração do processo, oferecendo uma interação eficaz do gerente com a máquina de processo.
- Integração de vários mecanismos de gerência de processos em um metamodelo comum, para auxiliar os usuários de PSEEs durante a gerência de processos. Este metamodelo foi definido através de uma abordagem formal que permite um melhor entendimento, análise e comparação com outros modelos (LIMA REIS, 2003).

A arquitetura atual do APSEE é descrita no próximo capítulo.

4.7 Considerações sobre o Capítulo

Este capítulo apresentou os principais conceitos envolvidos na tecnologia de processo de software, área que serviu como base para a proposta apresentada neste trabalho.

Dentre os conceitos discutidos, vale ressaltar que a definição de ciclo de vida de processos de software inspirou esse trabalho e os conceitos de reutilização de processos de software estão fortemente presentes na proposta da etapa de reutilização de processos de curso, presente no modelo.

No próximo capítulo é apresentado o ambiente APSEE, plataforma sobre a qual o modelo proposto foi concretizado, como será descrito no capítulo 7.

5 O AMBIENTE APSEE

O ambiente APSEE pode ser definido como um meta-modelo unificado para automação da gerência de processos de software (SOUSA, 2003). Este modelo serviu como base para o modelo de processo de curso proposto neste trabalho.

O APSEE é uma infra-estrutura de software que evoluiu a partir de um mecanismo de gerência de processos de software proposto em Lima (1998). Segundo Reis (2002), o modelo do Gerenciador de Processos, tinha como principal objetivo definir um meta-modelo para a execução automatizada de processos de software.

A seção 5.1 apresenta algumas classes do APSEE, especificadas segundo o paradigma do PROSOFT, e a seção 5.2 descreve a arquitetura do ambiente.

5.1 Classes PROSOFT

O formalismo utilizado no desenvolvimento do meta-modelo APSEE baseou-se na construção de ferramentas através da estratégia data-driven (NUNES, 1992). Portanto a especificação do ambiente começa pela construção das classes que compõem as ferramentas do ambiente (chamadas de ATOs – Ambientes de Tratamento de Objetos). A notação utilizada para construção das classes é descrita em (REIS, 1998).

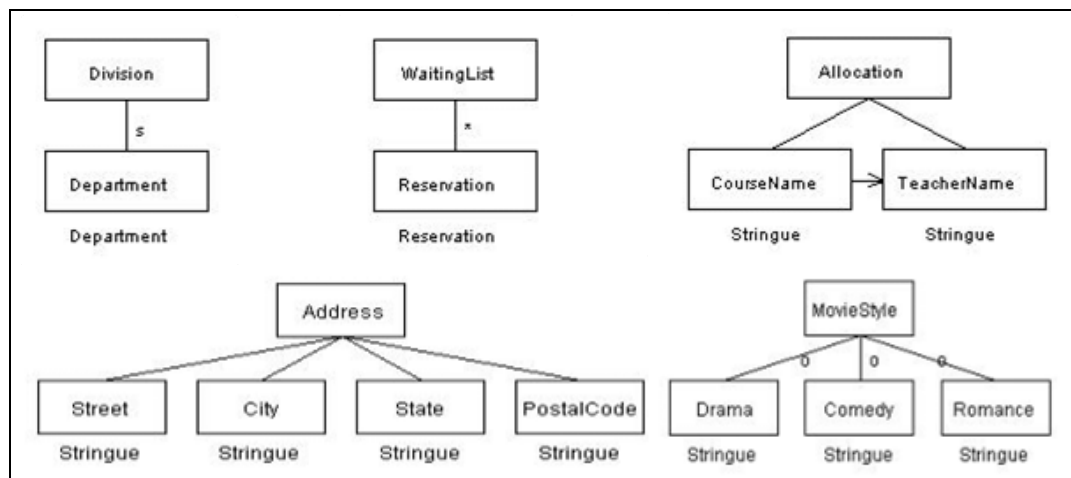


Figura 5.1: Exemplos dos construtores de classes no PROSOFT (LIMA REIS, 2003).

A Figura 5.1 mostra exemplos de classes utilizando os construtores da notação PROSOFT (LIMA REIS, 2003):

- conjunto (set - s);
- lista (*);
- mapeamento (\rightarrow);
- composição (Record)
- alternativa (seleção - o).

As operações disponibilizadas pelas classes são especificadas formalmente no paradigma Prosoft-Algébrico (NUNES, 1994). Por ser um método algébrico, o PROSOFT-algébrico especifica tipos abstratos de dados e operações sobre estes tipos. O ambiente PROSOFT proporciona que especificações destes tipos abstratos de dados e suas respectivas operações sejam implementadas e integradas ao ambiente, passando a fazer parte do mesmo (RANGEL, 2003).

O meta-modelo APSEE incorpora as tecnologias necessárias para atender cada fase do ciclo de vida e está especificado formalmente (mais de 60 classes) através do paradigma do ambiente PROSOFT (LIMA REIS, 2003).

Nesta seção são apresentadas algumas das principais classes do APSEE.

5.1.1 Classe APSEE

A classe APSEE é a classe principal do meta-modelo comum e é apresentada na figura 5.2, através da composição de vários elementos envolvidos na gerência de processos de software e descritos na seção 1.3. Os componentes da classe APSEE foram desenvolvidos por Reis, 2002; e Lima Reis, 2003.

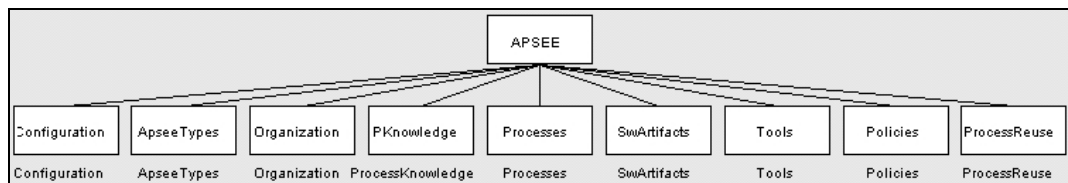


Figura 5.2: Classe APSEE

Alguns dos componentes principais da classe APSEE são APSEE-Types e Processes, cujas classes são apresentadas nas próximas seções.

5.1.2 Classe APSEE-Types

Os principais componentes da arquitetura APSEE são tipados. Um APSEE-Type é uma hierarquia de tipos para um componente do APSEE. Este componente será descrito com mais detalhes na seção 1.2.1.

A Figura 5.3 mostra a composição das classes APSEE, APSEE-Types e Types. Pode-se observar que a construção das classes de tipos através de mapeamento permite a inclusão de novas hierarquias no APSEE-Types e novos tipos, no Types.

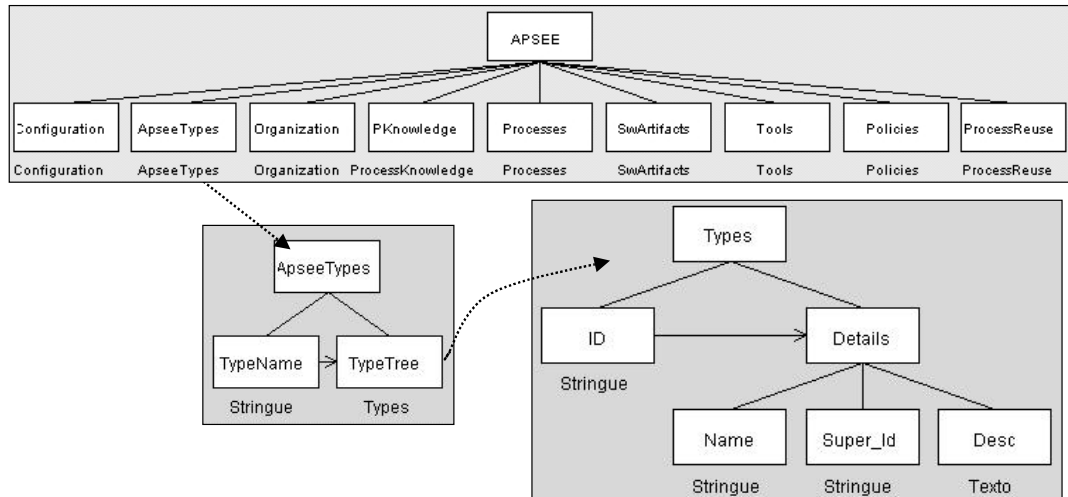


Figura 5.3: Classes APSEE-Types e Types no APSEE (LIMA REIS, 2003)

Um exemplo de hierarquia de tipos de recursos no APSEE é mostrado na Figura 5.4. Os computadores portáteis da organização estariam associados ao tipo *portable computer*, que é subtipo de *computer* e *exclusive*.

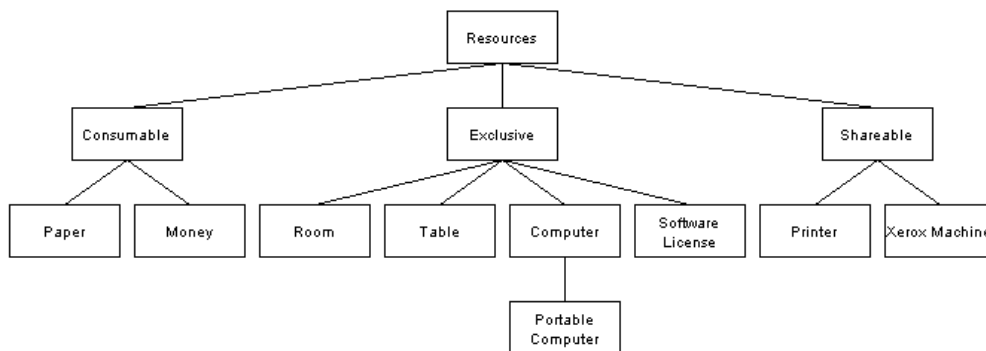


Figura 5.4: Exemplo de hierarquia de tipos de recursos (LIMA REIS, 2003).

5.1.3 Classe Processes

O componente *Processes* contém todos os processos de software descritos em diferentes estados de modelagem e de execução. A classe *Processes* é detalhada na Figura 5.5. Dessa classe foi detalhada a classe *ProcessModel* que descreve o processo, no quadro inferior da mesma figura.

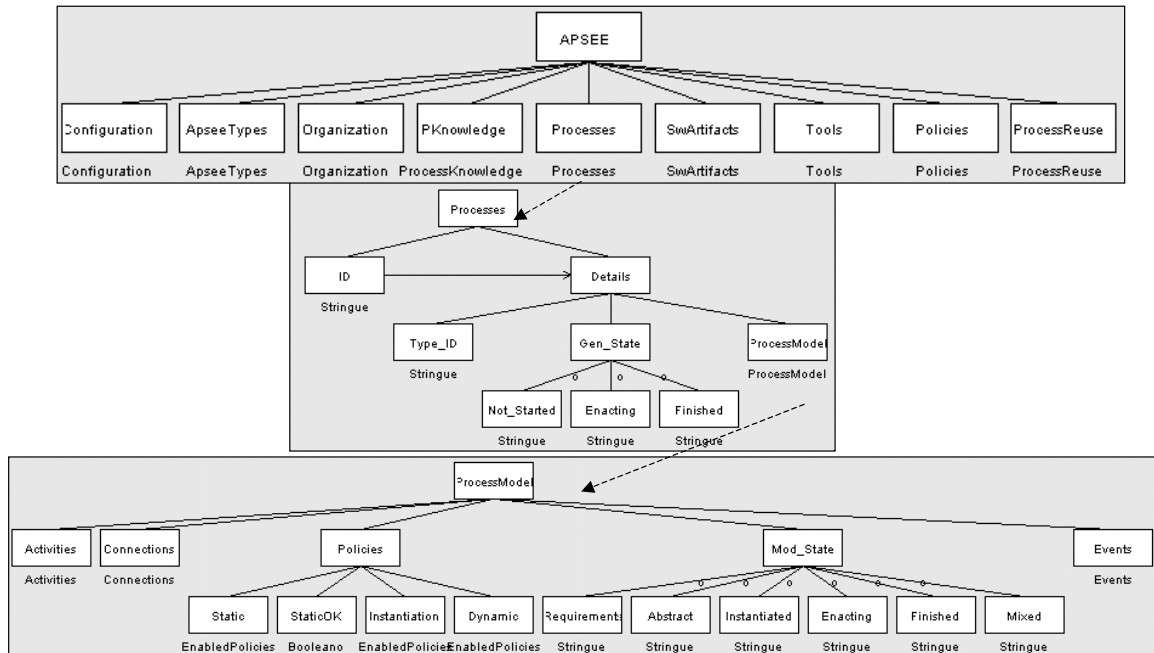


Figura 5.5: Classes APSEE, *Processes* e *ProcessModel* (LIMA REIS, 2003).

5.2 Arquitetura do APSEE

Nos últimos anos, a arquitetura do APSEE evoluiu, foram introduzidos novos recursos para apoiar de forma mais adequada as características evolucionárias e dinâmicas do processo de software (REIS, 2002). A Figura 5.6 apresenta os principais componentes do sistema APSEE, organizados em três camadas principais: meta-modelo, mecanismos de gerência de processos e nível de interação com o usuário. Cada uma destas camadas é descrita nas próximas seções.

5.2.1 Meta-modelo APSEE

O meta-modelo unificado (camada inferior) foi proposto para facilitar a integração de diferentes serviços de gerência de processos de software (que estão na camada do meio). As diferentes fases de evolução de um processo no ambiente APSEE são capturadas pelo meta-modelo unificado. É importante observar que os componentes centrais do modelo são a linguagem de modelagem e o mecanismo de execução, pois interagem com todos os outros componentes (a interação está representada através das setas).

O meta-modelo APSEE apresenta alguns dos tipos de dados mais importantes utilizados pelo ambiente (REIS, 2002; LIMA REIS, 2003), descritos nas seções a seguir.

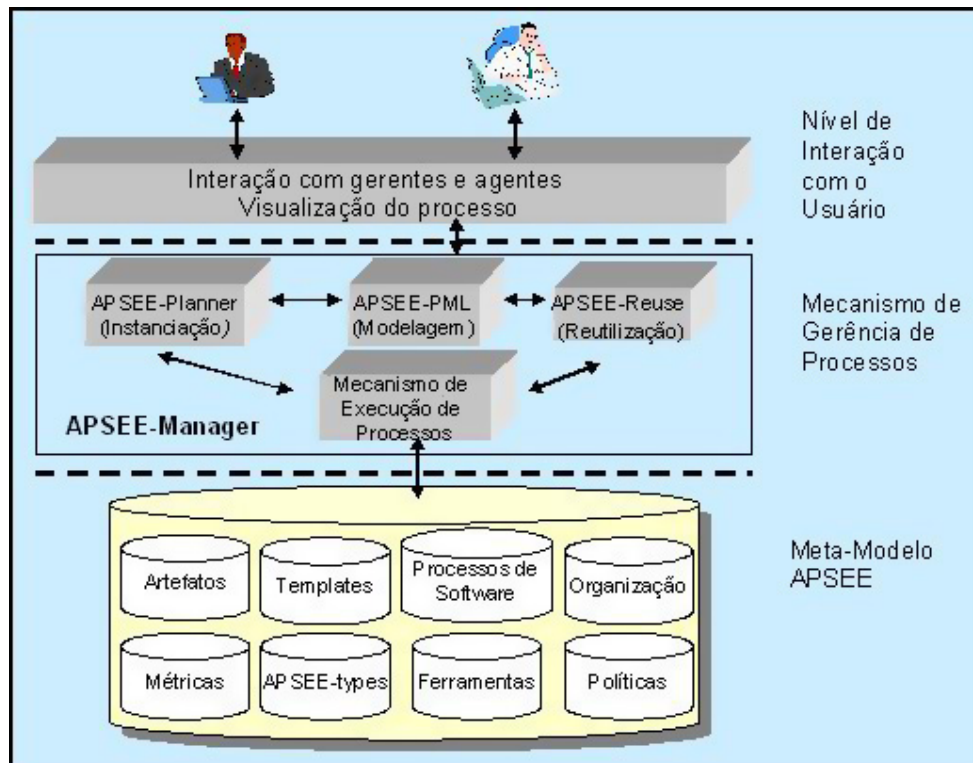


Figura 5.6: Visão geral do modelo APSEE (REIS, 2002; LIMA REIS, 2003)

5.2.1.1 Processos de Software

Este componente descreve modelos de processos de software em seus diferentes estados (modelados em nível abstrato ou instanciados, e em diferentes estados de execução);

O pacote Processos de Software foi dividido em quatro componentes (LIMA REIS, 2003):

- Modelos de processos: contém o modelo para descrever processos de software em diferentes estados de modelagem e de execução. Um processo da classe Process possui um identificador, um tipo na hierarquia de tipos de atividades, um estado e um modelo de processo. A distinção entre processo e modelo de processo (ProcessModel) se faz necessária para determinar a situação geral do nodo raiz da rede de atividades, pois modelos de processo são definidos de forma recursiva (são compostos de atividades que podem ser decompostas em novos modelos de processo). O modelo de processo é composto basicamente de atividades e conexões entre atividades;
- Atividades de um processo: Uma atividade pode ser simples ou decomposta. Se a atividade for decomposta, então é definida por um novo modelo de processo e seu conteúdo é representado pela classe ProcessModel mostrada na seção anterior. Caso contrário, trata-se de uma atividade simples na decomposição do modelo de processo. As características comuns a todas as atividades são: a habilitação de políticas, a indicação de que as políticas estáticas estão satisfeitas, o registro das modificações e as versões da atividade;

- Atividades simples: uma atividade simples pode ser normal ou automática. Atividades automáticas não consomem recursos nem tempo e são realizadas através de chamadas a operações de ferramentas integradas no ambiente. Já as atividades normais necessitam de recursos, agentes e envolvem artefatos de entrada e saída;
- Conexões entre atividades: servem para interligar atividades representando o fluxo de controle e de dados do processo e podem ser simples, múltiplas ou de artefatos. As conexões simples podem ser de seqüência ou de feedback, enquanto que as conexões múltiplas são branch e join combinados com operadores lógicos. As conexões simples e múltiplas também possuem tipo de dependências que influenciam diretamente na execução das atividades e podem representar diferentes dependências encontradas em processos reais. As conexões de artefato não influenciam no fluxo de controle do processo, servindo para descrever o fluxo de informações entre atividades, mas podendo ser ligadas a conexões múltiplas.

5.2.1.2 *Templates*

O tipo Template do APSEE apóia a descrição de processos abstratos e genéricos que podem ser reutilizados em diferentes contextos. Um processo abstrato é descrito como um conjunto parcialmente ordenado de atividades orientadas ao desenvolvimento de software. Este tipo é essencial ao componente APSEE-Reuse que será descrito na próxima seção.

5.2.1.3 *Hierarquia de tipos (APSEE-Types)*

Os principais componentes da arquitetura APSEE são classificados através de tipos pré-definidos. O APSEE-Types contém as hierarquias de tipos para os componentes do APSEE. São propostas as hierarquias dos tipos principais, fornecidas juntamente com o modelo, mas o usuário do ambiente poderá criar novas hierarquias de tipos e instâncias para hierarquias existentes de acordo com as necessidades da organização ou do processo. As hierarquias de tipos principais são: recursos, cargos, grupos, habilidades, métricas, atividades, conexões, políticas, artefatos e ferramentas.

O APSEE-Types exerce um papel importante na descrição do modelo, visto que a maioria dos componentes do meta-modelo APSEE estão associados a um Tipo (Type) que deve existir na hierarquia correspondente. As hierarquias de tipos relacionadas aos componentes do APSEE permitem que sejam descritos processos abstratos que podem ser refinados conforme necessário para execução ou usados para reutilização (LIMA REIS, 2003).

5.2.1.4 *Organização*

Engloba o modelo de recursos utilizados pelas atividades, o modelo de pessoas da organização (agentes), suas habilidades, afinidades, cargos e grupos de trabalho;

Este componente contém os aspectos relacionados à estrutura organizacional envolvida. As informações sobre a organização foram subdivididas, distinguindo informações sobre:

- Pessoas: As informações sobre as pessoas da organização visam apoiar os componentes de gerência de processos auxiliando na escolha de

desenvolvedores para tarefas específicas. Este tipo provê informações sobre os agentes, que incluem quaisquer pessoas envolvidas com o processo de software (desenvolvedores, gerentes e usuários, por exemplo), os cargos existentes, as habilidades reconhecidas (que fornecem um conhecimento importante para a sua alocação em atividades de um processo), os grupos formados pelos agentes e as agendas que são o mecanismo principal de comunicação entre o mecanismo de execução de processos do APSEE e os agentes. No APSEE um agente pode participar de vários processos. Assim, cada agenda representa a visão do agente sobre o processo em execução, ou seja, atividades alocadas a ele e os documentos que pode manipular (LIMA REIS, 2003);

- Recursos: O modelo de recursos do APSEE consiste de tipos de recursos, instâncias de recursos e relacionamentos entre os mesmos. Além de prover um mecanismo para descrever e organizar os recursos disponíveis na organização, o modelo de recursos pode ser usado para controlar dinamicamente o acesso concorrente aos recursos, escalonar sua utilização, analisar sua disponibilidade na organização e verificar consistência de seu uso. Os recursos, no APSEE, podem ser exclusivos (recursos que não podem ser alocados para várias atividades simultaneamente), compartilháveis (recursos que podem ser utilizados por várias atividades simultaneamente) ou consumíveis (recursos que não podem ser utilizados novamente);
- Políticas: são destacados os projetos da organização e são indicadas as políticas adotadas pela mesma.

5.2.1.5 Artefatos

Uma das principais tarefas dos ambientes de desenvolvimento de software é armazenar, estruturar e controlar o desenvolvimento de informação, produtos de software e os objetos gerados durante o desenvolvimento de software. Esses objetos são, por natureza, mais complexos que os objetos tratados por sistemas de banco de dados tradicionais, pois são mais estruturados e requerem operações complexas (GODART apud LIMA REIS, 2003).

Um ambiente de desenvolvimento de software deve prover um módulo que seja responsável por controlar o acesso e a evolução de objetos compartilhados (LIMA REIS, 2003). Como o APSEE foi integrado ao ambiente PROSOFT, o qual já possuía um gerenciador de objetos, chamado PROSOFT Cooperativo (REIS, 1998), na arquitetura do APSSE este gerenciador foi integrado através do componente Artefatos.

Esse componente trata objetos de qualquer tipo no ambiente, guardando seu estado, gerenciando o seu acesso e suas versões. Descreve os itens de dados manipulados, criados e utilizados durante o desenvolvimento de software. Exemplos de artefatos são documentos e código.

5.2.1.6 Conhecimento sobre o processo (*ProcessKnowledge*)

As pesquisas nas áreas de gerência de processos e projetos, métricas de processo e produto e qualidade de software indicam que o conhecimento útil a ser armazenado sobre a realização de processos de desenvolvimento de software, além da própria descrição do processo executado, pode ser modelado como métricas (ROCHA et al., 2001; PRESSMAN, 2001; JONES, 1996; FERNANDES, 1995; FLORAC, 1999; FRANCA, 1998; HUMPHREY, 1989; 1995; 1997).

O componente de Conhecimento sobre o Processo do APSEE funciona como uma base de conhecimento sobre todos os componentes do meta-modelo. O conhecimento sobre processo é representado através de métricas resultantes da execução dos processos definidos. Esse componente pode ser consultado nas fases de modelagem, instanciação, simulação e execução, além de permitir estudos empíricos acerca do impacto de algumas métricas em aspectos importantes, como por exemplo, na qualidade do software produzido (LIMA REIS, 2003).

É importante observar que o conhecimento armazenado no componente Artefatos é o conhecimento factual (ABEL apud LIMA REIS, 2003) sobre os componentes do modelo, enquanto que o conhecimento heurístico que pode ser definido pelo usuário é representado por políticas.

Uma definição de métrica, no APSEE, possui (LIMA REIS, 2003):

- Tipo: métrica de processo (indica uma característica de um processo, fragmento de processo ou atividade), métrica de produto (indica uma característica de um artefato produzido ou consumido por atividades) ou métrica individual (indica características de pessoas e recursos envolvidos no processo).
- Unidades de medida da métrica;
- Intervalo de validade da métrica;
- Forma de obtenção: método utilizado para medir, como por exemplo, “cálculo automático”, “contagem manual” ou “chamada de operação de uma ferramenta”, dentre outros.

5.2.1.7 Ferramentas

Este componente apresenta informações sobre as ferramentas disponíveis para o desenvolvimento de software. No APSEE as ferramentas podem ser *ATOs* (ferramentas do ambiente PROSOFT) ou ferramentas externas ao PROSOFT.

5.2.1.8 Políticas

Segundo Feiler apud Lima Reis (2003), políticas de processos de software consistem em "princípios que conduzem o desenvolvimento e/ou a execução de processos de software".

No modelo APSEE, uma política pode ser informalmente descrita como um conjunto de propriedades que atuam na formação e execução de modelos de processos de software, representando um conhecimento gerencial genérico e reutilizável para diferentes contextos.

Uma política contém critérios da organização definidos pelo usuário e pode estar habilitada em uma atividade, em um processo, ou na organização, permitindo a sua reutilização em diferentes contextos. No APSEE foram definidos três tipos principais de políticas com diferentes finalidades:

- Políticas Estáticas: definem regras sintáticas para a formação dos modelos. Podem ser úteis para definir formalmente boas práticas de gerenciamento de projetos que podem ser reutilizadas em diferentes processos. Essas políticas foram definidas em (REIS; LIMA; NUNES, 2002; REIS 2002);

- Políticas de Instanciação: Definem critérios para a alocação de recursos e pessoas utilizadas por uma organização em contextos ou processos específicos, podendo ser baseadas no histórico da execução de processos anteriores;
- Políticas Dinâmicas: Baseiam-se nos eventos que podem ocorrer durante a execução de processos e realizam ações em resposta aos eventos de acordo com estratégias definidas pelo usuário.

5.2.2 Mecanismo de gerência de processos

A camada mecanismo para gerência de processos é composta por um conjunto de serviços que constituem o componente denominado *APSEE-Manager* (LIMA REIS, 2003).

Este componente da abordagem APSEE foi proposto com o objetivo de aumentar o nível de automação e flexibilidade na gerência de processos de software e é composto por quatro componentes que serão descritos nas próximas seções: *APSEE-PML*, *APSEE-Planner*, Mecanismo de Execução de Processos e *APSEE-Reuse*.

5.2.2.1 APSEE-PML

É o formalismo de modelagem do APSEE que permite representação gráfica de processos criados a partir do meta-modelo descrito na seção anterior.

Essa linguagem de modelagem de processos é baseada em redes de atividades que podem ser decompostas. Neste formalismo, um modelo de processo pode ser construído a partir de símbolos gráficos conectados e o detalhamento do relacionamento com os outros componentes do modelo é feito através de formulários específicos que apóiam essa tarefa. A gramática da linguagem e as dependências e restrições de consistência para a construção de processos foram definidas formalmente em Lima (2003).

A APSEE-PML além de auxiliar na descrição de processos e *templates*, também é útil para habilitar políticas em modelos de processos de software (LIMA REIS, 2003).

5.2.2.2 APSEE-Planner

A instanciação de processos de software é a transformação de um modelo de processo em processo executável (DERNIAME; KABA; WASTELL, 1999) através do planejamento do cronograma e alocação de desenvolvedores e recursos para as tarefas, dentre outros ajustes.

O componente APSEE-Planner é responsável por fornecer assistência automática para instanciação de recursos e agentes durante a execução de processos de software através de políticas de instanciação definidas pelo usuário (descrito em Lima Reis et al (2002)).

5.2.2.3 Mecanismo de execução de processos

Durante a execução de processos de software as atividades modeladas e instanciadas são realizadas tanto pelos desenvolvedores quanto automaticamente. Esta fase envolve questões importantes acerca de planejamento, controle, monitoração, garantia de conformidade com o processo modelado, treinamento, segurança e recuperação do processo (FEILER; HUMPHREY, 1993).

O mecanismo de execução de processos do APSEE exerce um papel central na organização dos mecanismos existentes, coordenando as atividades do processo em execução através da interpretação da linguagem de modelagem de processos (REIS, 2002). Esse mecanismo tem como principais características (LIMA REIS, 2003):

- Permitir execução de processos incompletos e modificação dinâmica;
- Possibilitar a escolha dentre várias alternativas para o fluxo de execução dependendo da situação corrente;
- Permitir a repetição de atividades, verificando condições da conexão de feedback;
- Fornecer a instanciação automática para o processo ou seus fragmentos.
- Fornecer um registro histórico dos processos;
- Fornecer a semântica da linguagem APSEE-PML. Essa semântica é definida formalmente no paradigma do PROSOFT (algébrico) combinado com gramática de grafos (LIMA REIS et al., 2002);
- Estar integrado com o mecanismo de gerência de objetos fornecido pelo componente Artefatos;
- Permitir execução de processos colaborativos;
- Permitir visualização da execução através do formalismo gráfico e executável para modelagem de processos. O mesmo formalismo é usado para modelar processos e acompanhar sua execução;
- Para facilitar a interação com os usuários e obter feedback sobre a realização do processo são fornecidas agendas de tarefas para cada agente envolvido no processo.

A execução trabalha com transformações em vários níveis, atendendo às solicitações de vários usuários, ao mesmo tempo, e continuamente verificando o estado corrente do processo. Deve-se destacar que para uma compreensão geral da execução é muito importante considerar a semântica das conexões entre atividades (LIMA REIS, 2003).

5.2.2.4 APSEE-Reuse

APSEE-Reuse é o componente de apoio à reutilização de processos responsável pela criação, recuperação e adaptação de *templates* reutilizáveis (REIS; LIMA REIS; NUNES, 2001; REIS, 2002);

O APSEE-Reuse está centrado em apoiar a modelagem de processos, sendo decisivamente influenciado pela necessidade de fornecer apoio à separação de detalhes na modelagem de processos. Assim, o modelo combina uma série de elementos sintáticos que auxiliam na descrição de processos reutilizáveis.

O APSEE-Reuse foi projetado para apoiar a construção de processos genéricos a partir da composição de elementos independentes definidos em diferentes dimensões (REIS, 2002).

5.2.3 Interação com o Usuário

A camada de interação é responsável pela coordenação da interação entre o usuário e o sistema através da tradução das ações do usuário pela ativação de uma determinada funcionalidade do sistema, e também, permitindo que os resultados destas ações possam ser observados (SOUSA, 2003).

A camada de interação do APSEE provê mecanismos de interação especializados para os diferentes usuários do ambiente. Por exemplo, agendas para os agentes e facilidades de visualização de processos para projetista/gerentes, conforme proposto por Sousa (SOUSA, 2003).

É através desta camada que o usuário se comunica com o sistema, avaliando o estado de um determinado processo, percebendo a evolução dinâmica das entidades que compõem todo o processo e intervindo quando necessário.

A interação com o usuário é fornecida através de uma interface gráfica, sendo a mesma fornecida pelo ambiente PROSOFT-Java, integrando serviços fornecidos pela linguagem de programação do ambiente. A Figura 5.7⁸ apresenta um exemplo de visão para a execução de um processo.

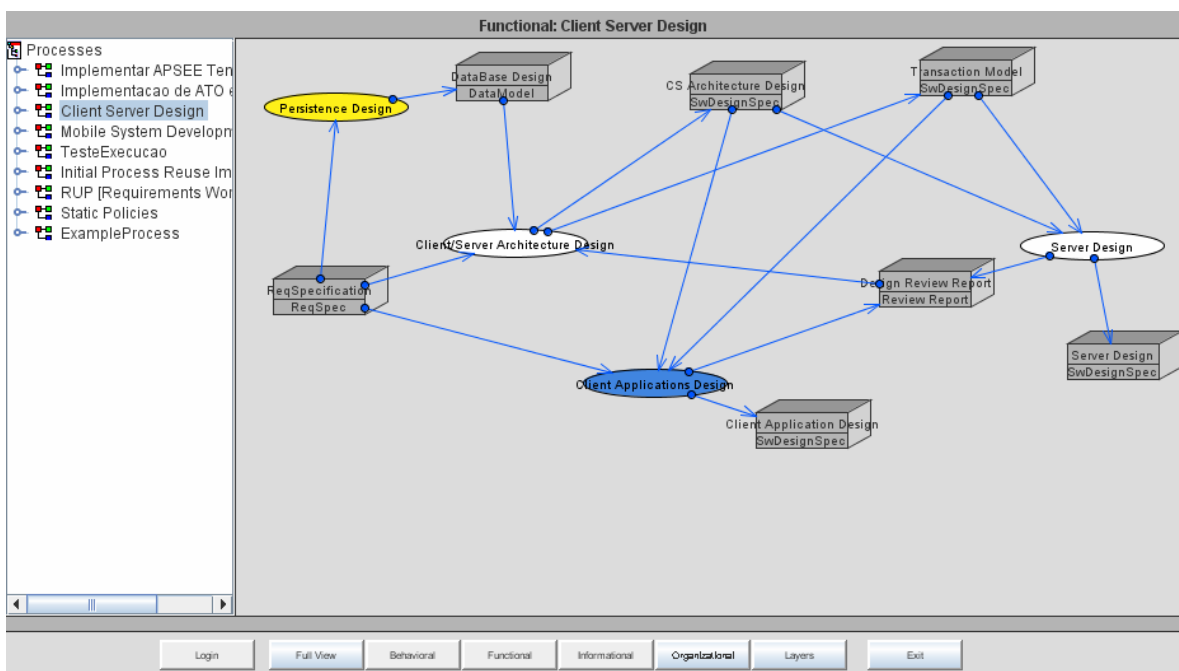


Figura 5.7: Visão funcional de processos de software no ambiente APSEE

Esta camada da arquitetura do modelo APSEE tem como principal componente o APSEE-Monitor que visa apoiar o papel do gerente de processos durante a monitoração do processo de desenvolvimento através de um mecanismo de apoio à visualização de processos de software(SOUSA, 2003).

O principal objetivo do APSEE-Monitor é apresentar ao gerente uma forma adequada de representação do fluxo de controle, dados, o estado atual e histórico do processo, além de fornecer independência do formalismo de modelagem de processos, e, principalmente, oferecer ao gerente, diferentes visões do processo durante a monitoração de sua execução (Figura 5.8).

⁸ As figuras deste capítulo e do capítulo 7 que ilustram o ambiente PROSOFT possuem alguns problemas de legibilidade. Isto deve-se a limitações na ferramenta de visualização do ambiente, impossibilitando a captura com uma melhor resolução.

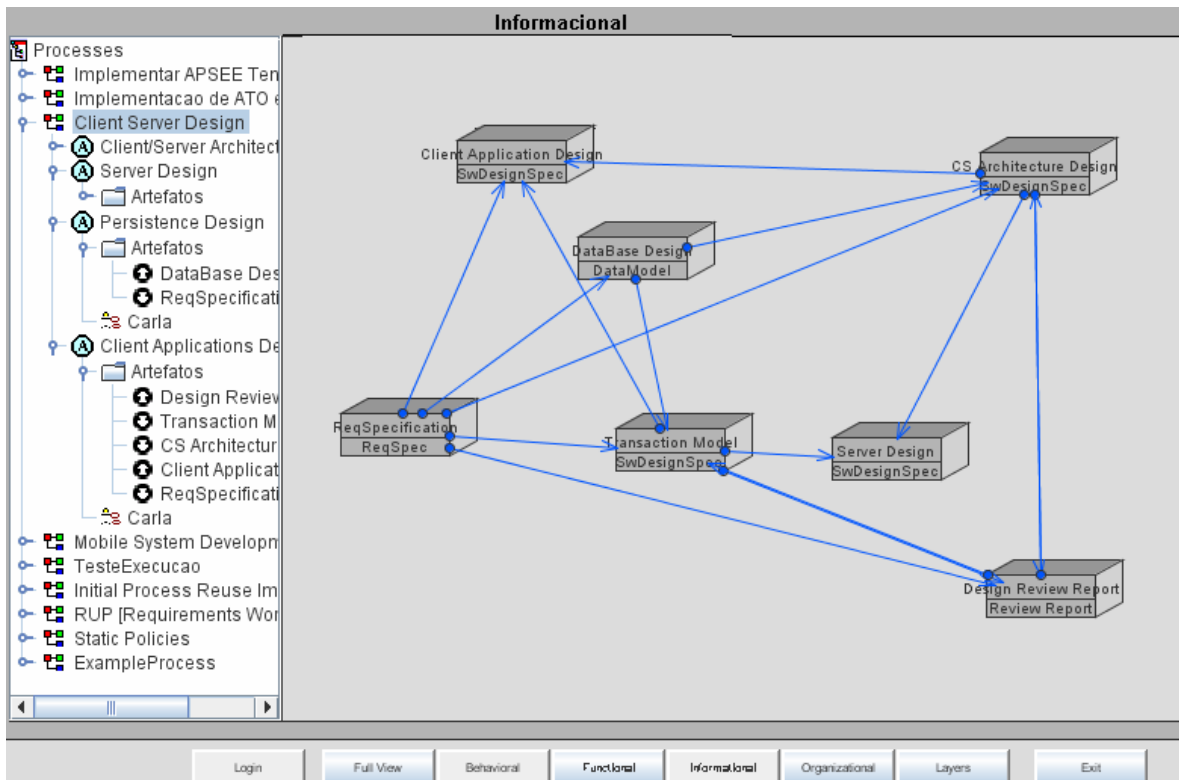


Figura 5.8: APSEE-Monitor: visão informacional

5.3 Implementação do Modelo APSEE em PROSOFT-Java

O modelo APSEE, apresentado em Reis (2002) e Lima Reis (2003), foi implementado no ambiente PROSOFT-Java. O objetivo inicial dessa implementação foi permitir uma experimentação dos componentes do modelo proposto.

As seções a seguir descrevem o ambiente PROSOFT-Java, o ambiente APSEE implementado, o editor de processos de software, o mecanismo de execução de processos e o editor e interpretador de políticas de instanciação (REIS, 2002; LIMA REIS, 2003).

5.3.1 Ambiente PROSOFT-Java

Esta seção fornece uma descrição geral do ambiente PROSOFT-Java de forma análoga ao encontrado em outros trabalhos recentes do mesmo grupo, que envolvem o mesmo ambiente (REIS, 2002; LIMA REIS, 2003; SOUSA, 2003).

PROSOFT-Java é a denominação da mais recente implementação do paradigma PROSOFT⁹, na forma de um ambiente homogêneo e integrado de desenvolvimento de

⁹ A primeira versão monousuária do PROSOFT foi desenvolvida em Solaris-Pascal, como descrito por Nunes (1992). Posteriormente, o PROSOFT-Distribuído foi desenvolvido em Pascal e C, apresentado em Granville (1996) e Schlebbe (1994). Finalmente, conforme Schlebbe (1997) foi selecionado Java como nova linguagem hospedeira para o ambiente.

software escrito na linguagem Java (SCHLEBBE, 1997). O desenvolvimento atual do PROSOFT-Java é resultado do esforço cooperativo de estudantes e pesquisadores do PPGC-UFRGS e da Fakultät Informatik da Universität Stuttgart (Alemanha), sob orientação do Prof. Dr. Daltro José Nunes.

O principal objetivo do ambiente é estabelecer uma infra-estrutura que apóie o desenvolvimento de software de alta complexidade através da integração de ferramentas escritas em um paradigma próprio. Atualmente, o PROSOFT-Java é um ambiente portátil, distribuído¹⁰ e cooperativo¹¹ que integra ferramentas CASE para auxiliar diferentes fases do processo de software. Uma descrição completa sobre o desenvolvimento de ATOs-Java foi disponibilizada em Schlebbe (1997) e Schlebbe (2002).

5.3.2 Ambiente APSEE

A implementação atual do sistema APSEE é composta por mais de uma centena de ATOs-Java, como resultado do trabalho de diferentes membros do grupo de pesquisa. Os ATOs estão relacionados entre si como definido pela arquitetura apresentada na seção 5.2, no qual o gerenciador de processos (APSEE-Manager) interconecta os diferentes serviços para definição, visualização e execução de processos.

O APSEE se vale dos serviços de apoio ao trabalho distribuído e cooperativo fornecidos pelo PROSOFT-Java permitindo, por exemplo, que o gerenciador de processos execute em um servidor, enquanto que instalações remotas do PROSOFT podem executar as agendas de tarefas (LIMA REIS, 2003).

5.3.3 Editor de processos de software

Um editor para processos de software (Figura 5.9) foi desenvolvido em PROSOFT-Java e integrado ao sistema APSEE. O editor implementa os tipos de dados, funções e regras de execução descritas em Lima Reis (2003).

¹⁰ A distribuição entre ATOs é fornecida através do mecanismo de comunicação denominado ICS-Distribuído que está implementado sobre o protocolo Java-RMI (SUN, 2003).

¹¹ A funcionalidade para trabalho cooperativo está disponível a partir da extensão denominada PROSOFTCooperativo, descrita por Reis (1998).

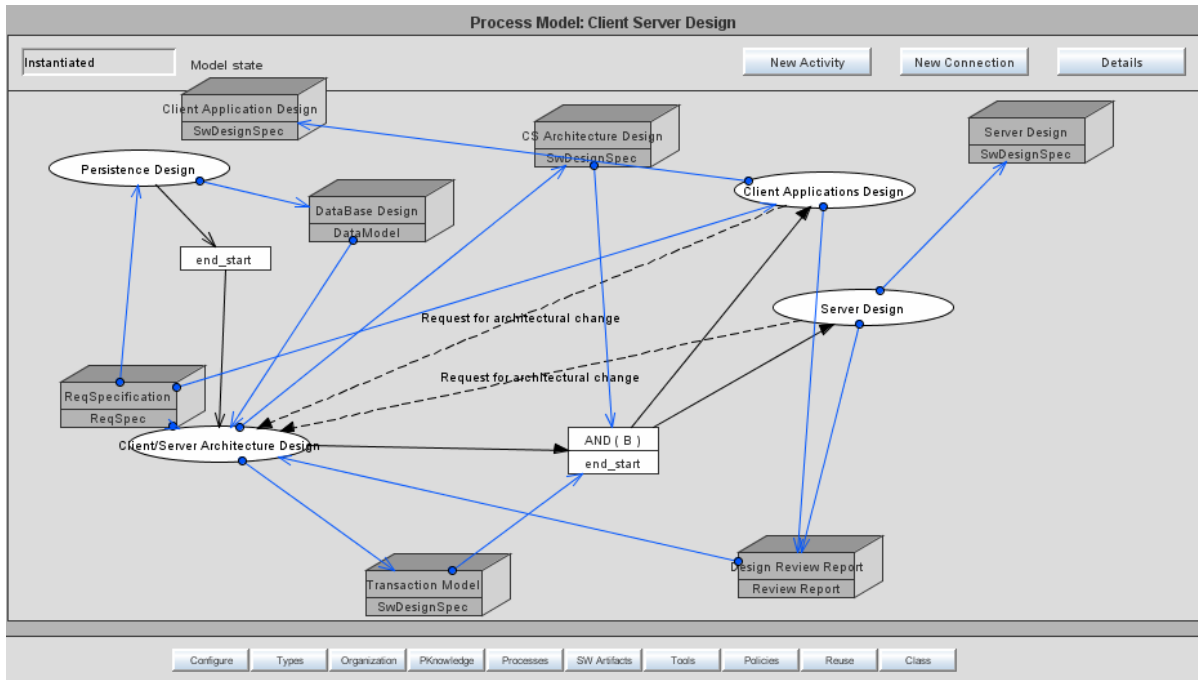


Figura 5.9: Editor de processos do APSEE

Alguns atributos dos ATOs especificados para o meta-modelo APSEE armazenam informações essencialmente textuais, e por isso a implementação está associada a formulários a serem manipulados pelo usuário. A Figura 5.10 apresenta um exemplo de formulário, exibido para descrever os detalhes de uma atividade.

Figura 5.10: Formulário para definição de detalhes de uma atividade.

Nos detalhes de uma atividade normal é possível definir através de formulários quais os agentes ou grupos serão envolvidos, assim como recursos necessários. A Figura 5.11 apresenta o formulário de definição de pessoas para uma atividade.

The screenshot shows a software interface titled "Required People: Client Server Design.Client Applications Design". It contains several panels for configuring roles and abilities. The "Required Roles" panel lists "1: Chief Analyst". The "Needed Abilities" panel is currently empty. The "Selected Agents for this Role/Abilities" panel shows "Paulo" selected in a dropdown menu, with a "Set" button below it. The "Role IDs" panel shows "Chief Analyst" in a dropdown menu, with "Add", "Delete", and "Clean" buttons below. The "Ability IDs" panel shows "Algebraic Specification" in a dropdown menu, also with "Add", "Delete", and "Clean" buttons. A "Minimal Degree" input field is located between the "Needed Abilities" and "Ability IDs" panels. The "Required Group Types" panel lists "Groups" and "Quality Group". The "Selected Group Instance" panel is empty. A "Show Suggestion" button is present in both the "Selected Agents" and "Selected Group Instance" panels. At the bottom of the window, there is a navigation bar with buttons for "Configure", "Types", "Organization", "PKnowledge", "Processes", "SW Artifacts", "Tools", "Policies", "Reuse", and "Class".

Figura 5.11: Formulário para definição de agentes em uma atividade

5.3.4 Mecanismo de execução de processos do APSEE

Assim como o editor para processos, o mecanismo de execução também foi desenvolvido em PROSOFT-Java e integrado ao sistema APSEE. A Figura 5.12 apresenta a tela onde o usuário pode selecionar um processo para execução.

Após o início da execução o gerente pode acompanhar o andamento do processo usando a mesma ferramenta de edição de processos. A partir desse momento o editor possui todas as suas funcionalidades originais, fornecendo ainda informações sobre a dinâmica de execução do processo. Uma das maneiras de informar o gerente sobre o andamento da execução é através da mudança de cor das atividades de acordo com o seu estado.

Do ponto de vista do desenvolvedor, o mecanismo de execução fornece informações através da agenda de atividades (Figura 5.13). Nessa agenda, o usuário pode visualizar o estado da atividade, pode interagir solicitando mudanças de estado e pode visualizar os detalhes da atividade.

The 'Processes' window displays a list of processes on the left, including 'Client Server Design', 'ExampleProcess', 'Implementacao de ATO e', 'Implementar APSEE Temp', 'Initial Process Reuse Imp', 'Mobile System Developm', 'RUP [Requirements Work', 'Static Policies', and 'TesteExecucao'. The central form shows the details for 'Client Server Design':

- ID:** Client Server Design
- Type ID:** Act Type 1
- General State:** Not_Started
- Static Policies Status:** Not verified

Buttons on the right include: Process Model, Execute Process, Textual Description, Check Static Policies, Show Process Events, and Reset Process. At the bottom left are 'Add/Update', 'Delete', and 'Clean' buttons. The bottom navigation bar includes: Configure, Types, Organization, PKnowledge, Processes, SW Artifacts, Tools, Policies, Reuse, and Class.

Figura 5.12: Formulário para seleção de processos para execução.

The 'TaskAgenda APSEE-Manager' window features a menu bar with File, View, Operation, Cooperation, Customization, and Help. The 'Process' dropdown is set to 'Client Server Design'. The 'Task' section includes buttons for Start, Finish, Pause, Delegate, Filter, and Sort by (ID). The task list is as follows:

ID	State	Priority	Begin Date	End Date
Client Applications Desi...	Active	0	2002-03-02	2002-03-02
Client/Server Architectur...	Waiting	0		
Persistence Design	Ready	0		

Below the table are fields for 'Delegated From', 'Delegated To', and 'Working Hours' (0.0). The 'Task Log' section shows a list of events:

- [2002-03-02 10:34:12] - Added
- [2002-03-02 10:34:32] - Finished
- [2002-03-02 10:34:35] - Paused
- [2002-03-02 10:34:48] - Finished
- [2002-03-02 10:34:53] - Paused

Figura 5.13: Agenda do desenvolvedor.

5.3.5 Editor e interpretador de políticas de instanciação

Foram implementados no ambiente APSEE um editor e um interpretador de políticas de instanciação (LIMA REIS, 2003).

O editor de políticas de instanciação foi implementado para que o usuário crie políticas através de um formulário apropriado, que contém todas as operações básicas da linguagem de políticas. No componente *Policies* do ambiente APSEE, é possível visualizar e manipular as políticas de instanciação existentes através da verificação dos seus detalhes, como mostra a Figura 5.14.

Figura 5.14: Editor de políticas do APSEE.

Quando uma atividade habilita uma política de instanciação, o usuário pode chamar o interpretador de políticas através do formulário de edição da atividade. Além disso, caso a instanciação automática esteja habilitada, são executadas regras que chamam o interpretador automaticamente durante a execução.

5.4 Considerações sobre o Capítulo

Este capítulo apresentou a arquitetura do modelo APSEE e a implementação do ambiente resultante desse modelo. A apresentação do APSEE é fundamental, pois o modelo para Processo de Curso, proposto neste trabalho, foi implementado através desta plataforma.

No próximo capítulo, o modelo para Processo de Curso, proposto nesta tese, será apresentado.

6 UM MODELO PARA PROCESSO DE CURSO

Nesta tese, é proposto um modelo de processo de curso a distância desenvolvido através da analogia entre processo de curso e processo de software. Esse modelo está focado no apoio à gerência de cursos e busca minimizar as dificuldades encontradas pelos professores através da utilização de experiências anteriores como base para criação de novos cursos.

Para definição do modelo, foram definidos os requisitos e estudados vários trabalhos que abordam a questão da gerência de cursos a distância. Os requisitos definidos e os estudos realizados são descritos nas próximas seções.

6.1 Requisitos do Modelo

Através da análise de trabalhos que abordam a gerência de cursos a distância (TAVARES et al., 2001; TAVARES et al.; 2001; BESSAGNET et al., 2002; SANTORO; BORGES; SANTOS, 2002; CAMOLESI; NETO, 2002) foram definidos os requisitos para ambientes de apoio à gerência de cursos a distância que nortearam o desenvolvimento do modelo (Tabela 6.1).

Tabela 6.1 – Requisitos para Gerência de Cursos a Distância

R1	Oferecer um formalismo para modelagem dos cursos
R2	Fornecer independência do formalismo de modelagem em relação aos recursos utilizados
R3	Permitir instanciação individualizada, de acordo com o perfil do aluno
R4	Permitir monitoração da execução do curso
R5	Fornecer um mecanismo de avaliação flexível
R6	Permitir reutilização de modelos de cursos
R7	Permitir modificação dinâmica do processo durante a execução

Os requisitos listados se referem a uma característica que um ambiente de apoio à gerência de cursos deve oferecer. Cada um deles é explicado a seguir:

- R1: Um ambiente para gerência de cursos deve oferecer algum formalismo que permita modelar os cursos. Este formalismo deve possibilitar a definição de

conexão entre atividades, definição de tipos de recursos e produções relacionados a cada atividade.

- R2: O formalismo para modelagem dos cursos deve ser independente dos conteúdos, ferramentas de comunicação e demais recursos utilizados nas atividades. A seleção dos materiais e demais recursos deve ser prevista para a etapa de instanciação. Desta forma, um modelo de curso possui independência de conteúdo, podendo ser utilizado em diferentes contextos.
- R3: Um sistema para EAD deve ser flexível e adaptável às características do professor e do aluno. Com relação ao aluno, diferentes perfis de aprendizagem devem ser considerados na etapa de instanciação. Um mesmo curso pode ter implementações diferentes, dependendo do perfil dos aprendizes.
- R4: A monitoração da execução do curso é essencial para um bom andamento do mesmo. Através da visualização do andamento do curso, o professor/tutor¹² pode identificar atrasos nas atividades, abandono do curso por parte de algum aluno, participações aquém do esperado em atividades propostas, etc. Se identificar alguma dessas características, o professor/tutor pode, por exemplo, intervir através de mensagens de estímulo para os alunos participantes do curso. A monitoração da execução fornece ao professor um *feedback* que procura substituir o contato visual de uma aula presencial.
- R5: Um ambiente para gerência de cursos a distância deve permitir que um curso seja avaliado durante e após sua execução. Esse mecanismo de avaliação deve ser flexível e permitir a configuração de métricas e combinação das mesmas, de acordo com os objetivos dos professores e da instituição onde o curso é oferecido. É necessário um mecanismo de avaliação eficiente para que apenas cursos de qualidade possam ser reutilizados, conforme previsto pelo próximo requisito.
- R6: A reutilização de modelos de cursos já executados é uma característica muito importante para o apoio às atividades docentes em EAD e busca um aumento na produtividade dos professores. Se o ambiente atende ao requisito 2, ou seja, possui um mecanismo de modelagem independente de conteúdo, um modelo de curso já executado pode ser reutilizado em outro curso, com o mesmo conteúdo, ou não. Um mecanismo de busca deve recuperar cursos com características semelhantes ao que se deseja modelar e a reutilização deve prever que o professor possa adaptar o modelo recuperado, de acordo com seus objetivos. É importante ressaltar que a etapa de avaliação também é determinante para o sucesso da reutilização dos modelos de curso, pois apenas cursos de qualidade devem ser reutilizados.
- R7: Um modelo de curso está sujeito a alterações. Portanto, é importante que o ambiente forneça meios para alteração dinâmica, ou seja, permita que o professor altere o modelo de um curso em execução. Em uma modificação devem ser levados em consideração aspectos como reutilização de modelos e reinstanciação de elementos do curso. Esta flexibilidade deve estar voltada a fornecer *feedback* ao professor acerca das conseqüências das alterações.

¹² O acompanhamento do curso pode ser realizado por tutores ou pelo próprio professor.

Esses requisitos identificados foram empregados na análise dos trabalhos descritos na próxima seção.

6.2 Trabalhos Relacionados

Na literatura são apresentadas diversas abordagens para estruturação e representação de cursos a distância (JUNG, 2001; PETERS, 2003; KRAMER et al., 1999; HORTON, 2000). Vários destes trabalhos foram analisados detalhadamente e serviram como inspiração ao modelo proposto (MACHADO, J., 2000; RAPCHAN et al.; SIZILIO, 2000; KUNDE; SOUTO; OLIVEIRA; 2001; SILVA et al., 2003; BROADBENT, 2002; TAVARES et al.; 2001; BESSAGNET et al., 2002; SANTORO; BORGES; SANTOS, 2002; CAMOLESI; NETO, 2002). Os trabalhos mais significativos para a formulação desse modelo e a análise comparativa desses trabalhos são descritos nas próximas seções.

6.2.1 Hyper-Automaton (MACHADO, J., 2000; MACHADO, C., 2002)

Machado, J. (2000) apresenta um ambiente semi-automatizado para o suporte ao gerenciamento de hipertextos para o Ensino a Distância através da aplicação de conceitos da teoria de autômatos e teoria das categorias, reunindo os benefícios de ambas.

O trabalho apresenta uma alternativa de criação e modelagem de sistemas para o apoio à criação de cursos na *Web*, visando fornecer maior flexibilidade e facilidades a professores e alunos. A solução é suportada por uma arquitetura cliente-servidor modular, chamada de Hyper-Automaton, baseada em um modelo de organização de hiperdocumentos definido sobre o formalismo de Autômatos Finitos com Saída (Máquina de Mealy e Máquina de Moore).

O modelo é inspirado por pesquisas na área de hiperdocumentos, com especial enfoque no desenvolvimento de sistemas de hipertexto onde a base de hiperdocumentos é projetada de forma independente da estrutura de controle da aplicação hipermídia, e suporta algumas facilidades como a composição de estruturas hierárquicas, especificação de vários conjuntos de links sobre um mesmo corpo de hiperdocumentos e objetos separados da estrutura de navegação (MACHADO, J., 2000).

Nessa abordagem, cada curso consiste em um conjunto de hiperdocumentos independentes e é definido através de um autômato com saída (Figura 6.1). A função de transição (representada na figura pelas setas) funciona como ligação lógica entre os hiperdocumentos e a função de saída (indicada nos estados do autômato) compõe as páginas. O resultado final é a estrutura básica de páginas e links de hipertexto em um site na *Web*.

A interface do ambiente de um navegador *Web* fornece uma interpretação tangível para a estrutura de autômatos em hiperdocumentos. Os símbolos do alfabeto de saída são anotados com unidades de informação (páginas HTML) e, neste caso, o resultado das funções de saída (Máquina de Moore) ou de transição (Máquina de Mealy) é a apresentação de um hiperdocumento na janela do navegador, de acordo com os fragmentos indicados na palavra de saída.

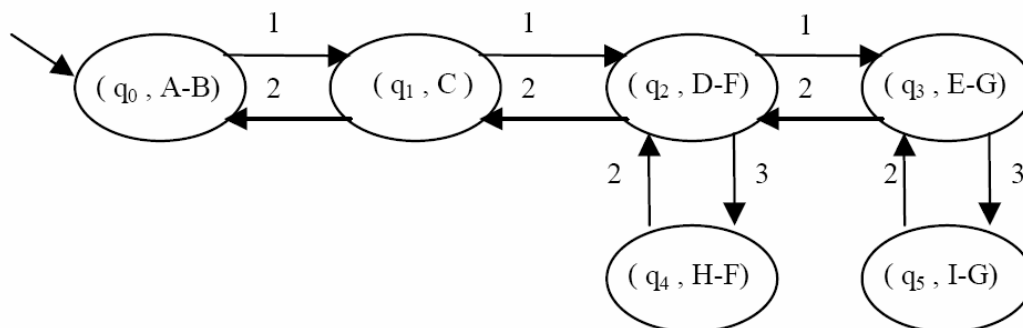


Figura 6.1: Curso modelado como Máquina de Moore (MACHADO, J., 2000)

O modelo leva a um alto grau de modularização do material instrucional, permitindo reuso de páginas em diversos cursos, com eliminação da redundância, elaboração de seqüências de aprendizagem com enfoques específicos (permite estudo individualizado) e um esquema de composição de cursos que permite a construção de novos cursos sobre cursos já existentes.

O modelo do Hyper-Automaton foi estendido por Machado, C. (2002). O XHA (Extensible Hyper- Automaton) é a versão revisada e atualizada Hyper-Automaton. Alguns dos avanços dessa versão são: flexibilização e personalização do layout de cursos, incremento de recursos didáticos e facilidade de manutenção.

6.2.2 EduQNet (RAPCHAN et al., 2002)

Rapchan et al. (2002) propõem um modelo de qualidade de processo para a elaboração, operação, gerência e manutenção de cursos a distância que utilizem a Internet como elemento mediador: EduQNet (Educação com Qualidade Mediada pela Internet).

O EduQNet foi definido a partir de um mapeamento dos indicadores de qualidade de EAD. Os autores consideraram, para este mapeamento, o guia de qualidade da HRDC (Human Resources Development Canadá) (BARKER, 1999), os indicadores de qualidade do MEC/SEED (MEC/SEED, 2002) e as referências de qualidade do IHEP (Institute for Higher Education Policy) (PHIPPS; MERISOTIS, 2000).

Além disso, o trabalho de Massy (2001) que sugere o uso experimental do CMM (*Capability Maturity Model*) para avaliar a penetração do trabalho de qualidade educacional nas instituições de ensino a distância foi a inspiração para estender este mapeamento entre modelos de qualidade de EAD para modelos de qualidade de processos de desenvolvimento de software. No EduQNet foi utilizado como referência o modelo NBR ISO/IEC 12207, para o mapeamento dos modelos de qualidade de EAD.

O modelo EduQNet está organizado em 14 atividades principais. Cada atividade indica uma série de sub-atividades relacionadas com o seu cumprimento:

1. Aquisição de produtos e serviços de terceiros.
2. Admissão de alunos
3. Desenvolvimento do curso

4. Operação do curso
5. Manutenção do curso
6. Documentação
7. Gerência de configuração
9. Revisão conjunta, verificação e validação e auditoria
10. Processo de resolução de problemas
11. Processo de gerência
12. Processo de infra-estrutura
13. Processo de melhoria
14. Processo de treinamento

Este trabalho utiliza um modelo de qualidade já definido, o NBR ISO/IEC 12207. A organização e arquitetura do modelo foram totalmente aproveitadas.

O trabalho apresenta apenas a definição conceitual do EduQNet. Não há descrição de nenhuma implementação desse modelo, apenas uma sugestão para projetos futuros.

6.2.3 Modelos baseados em técnicas de workflow

Sizilio (2000) aponta as técnicas de modelagem de *workflow* (WF) como uma tecnologia capaz de modelar a criação de cursos de EAD e implementar os conceitos relacionados à interação entre os processos, proporcionando um tratamento adequado aos aspectos dinâmicos que se apresentaram, às restrições temporais para sua execução e à troca de controle entre tarefas.

A Figura 6.2 apresenta o WF de autoria de cursos a distância. Ele utiliza a notação de WF apresentada no trabalho estudado e representa quatro supertarefas: cadastrar curso, definir critérios de análise de desempenho do aluno, implementar módulo e liberar curso para execução (SIZILIO, 2000).

Para verificar a exatidão do WF gerado foi simulada sua aplicação para gerar um curso de línguas pela Internet. Foram realizadas, também, algumas experiências isoladas de utilização do WF de autoria na modelagem de cursos on-line da modalidade salas de aulas virtuais.

O WF de autoria engloba as tarefas relacionadas à criação do curso (concepção, planejamento, definição e implementação do mesmo) (SIZILIO; EDELWEISS, 2001). Sugere-se que, na execução das tarefas, estejam envolvidos os seguintes agentes: autor do curso (responsável pelo projeto e por convocar, se necessário, os demais agentes) e profissionais das áreas de computação, pedagogia, comunicação e psicologia. Na utilização do WF de autoria foi possível constatar que para um docente utilizá-lo torna-se necessário algum treinamento.

O trabalho de Sizilio (2000) destacou as técnicas de modelagem de workflow como uma tecnologia capaz de modelar a criação de cursos de EAD e implementar os conceitos relacionados à interação entre os processos, proporcionando um tratamento adequado aos aspectos dinâmicos que se apresentaram, às restrições temporais para sua execução e à troca de controle entre tarefas.

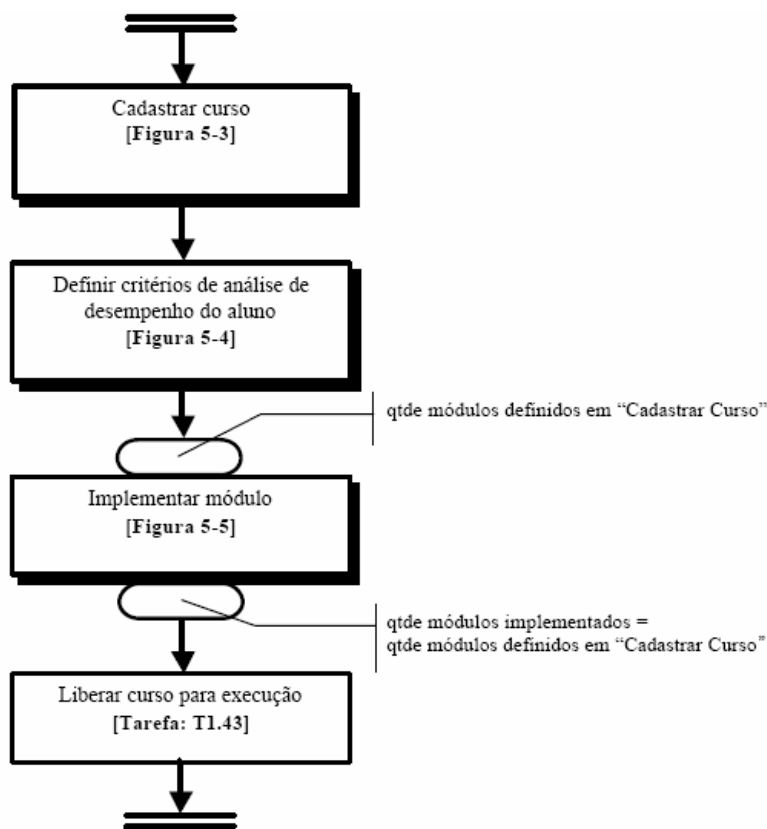


Figura 6.2: Autoria de um curso em EAD (SIZILIO, 2000)

Em outro trabalho (KUNDE; SOUTO; OLIVEIRA; 2001), a evolução dinâmica de cursos modelados através de técnicas de WF é prevista. São caracterizadas as mudanças dinâmicas que podem ocorrer no esquema conceitual de um curso via Web. O conteúdo do curso é representado conceitualmente e executado como um workflow. É apresentada uma análise de possíveis modificações no plano do curso e suas repercussões nas atividades dos alunos inscritos.

Neste trabalho, um estudo de caso foi elaborado para demonstrar as mudanças dinâmicas que podem ocorrer em um modelo de curso a distância e a propagação destas mudanças nas instâncias em execução. Cada aluno é representado por uma instância descrita pelo modelo e terá um andamento próprio. Assim, foi considerada a evolução dinâmica, pois há uma entrada contínua de alunos e, a cada alteração do curso, diferentes instâncias estarão em execução, em estágios de andamento diferentes. No gerenciamento das mudanças todas as instâncias em execução são consideradas.

Nesta proposta, os alunos podem iniciar as aulas a qualquer momento, não há prazos definidos para a inscrição nos cursos. Além disso, existe um modelo padrão de curso, composto pelas etapas de inscrição, questionário, avaliação, nivelamento, módulo de estudo, cálculo da média e certificado. Essas etapas são, por sua vez, decompostas em outras mais detalhadas.

Os autores destacam que para um sistema de workflow possuir flexibilidade para representar e gerenciar um curso a distância, com entrada contínua de alunos, é necessário que possua meios para permitir que as alterações no esquema possam ser propagadas para as instâncias de forma transparente e rápida. Porém, esta propagação não é trivial, pois pode causar erros ou inconsistências na execução. Muitos fatores devem ser levados em conta: estágio de execução de cada instância, dependência de dados entre as etapas do processo, formas de sincronização, etc.

A abordagem proposta pelos autores (KUNDE; SOUTO; OLIVEIRA; 2001) alia os benefícios da modelagem de workflow para cursos a distância com o suporte a mudanças dinâmicas.

6.2.4 Arquitetura baseada em agentes (SILVA et al., 2003)

Este trabalho apresenta uma arquitetura visando a construção de um espaço integrado, personalizado e multiparadigmático. A arquitetura proposta baseia-se em agentes, componentes e framework e foi inspirada pelo modelo APSEE (LIMA REIS, 2003).

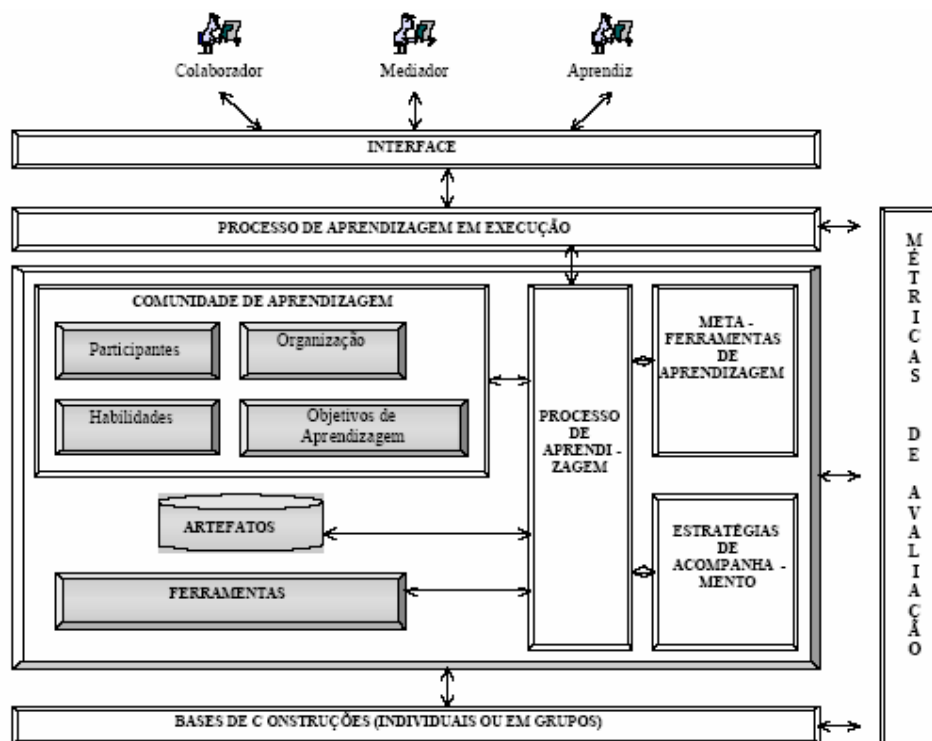


Figura 6.3: Arquitetura para ambientes de aprendizagem (SILVA et al., 2003)

Os componentes dessa arquitetura (Figura 6.3) foram definidos para fornecer os elementos necessários para a execução de um processo de aprendizagem considerando a reutilização de ferramentas, o compartilhamento do conhecimento, a integração entre os conhecimentos produzidos em diferentes comunidades, as atividades cooperativas, a interface adaptativa, o modelo do aprendiz (habilidades) e o modelo de estratégias de

acompanhamento, além da coordenação e avaliação necessária na união desses elementos.

Os componentes do modelo são:

- Processo de aprendizagem: define, para uma ou mais comunidades de aprendizagem, um conjunto de tarefas a serem realizadas de forma a alcançar os objetivos educacionais;
- Comunidade de aprendizagem: possui informações sobre as diversas comunidades de aprendizagem, como por exemplo, informações sobre os participantes e suas habilidades;
- Artefatos: um artefato é qualquer construção em uso por um ou vários aprendizes no decorrer do processo de aprendizagem;
- Ferramentas: estão relacionadas aos diversos recursos tecnológicos com fins pedagógicos que estão sendo utilizados pelas comunidades e individualmente pelos aprendizes, na produção dos artefatos;
- Meta-ferramentas de aprendizagem: contém as especificações necessárias para a geração do conjunto de ferramentas selecionadas (por mediadores e aprendizes) para serem utilizadas pelas comunidades virtuais de aprendizagem;
- Estratégias de acompanhamento: formas de avaliar e acompanhar as atividades desenvolvidas pelos aprendizes;
- Base de construções: contem os artefatos produzidos por aprendizes em atividades individuais ou em grupo;
- Processo de aprendizagem em execução: refere-se ao componente responsável pela execução dos processos de aprendizagem definidos.
- Interface: mecanismo de interação com aprendizes, colaboradores e mediadores.
- Métricas de avaliação: registra o conhecimento acerca dos resultados relacionados aos processos de aprendizagem (em execução e já executados) e a execução de estratégias de acompanhamento. O registro desse conhecimento torna possível a avaliação das diferentes formas de estruturação das etapas do processo de aprendizagem com o objetivo de provocar reflexões a respeito das abordagens utilizadas em uma ou mais comunidades.

A partir da arquitetura definida, o trabalho apresenta um cenário de uso, considerando os papéis de mediador, aprendiz e colaborador, no Ambiente AmAm (Ambiente de Aprendizagem Multiparadigmático). Todavia, nenhum resultado é descrito.

6.2.5 Um Modelo Baseado em Projetos (SANTORO; BORGES; SANTOS, 2002)

Neste trabalho é apresentado um modelo de cooperação para aprendizagem baseado em projetos. Tal modelo contempla a definição de processos e apóia o desenvolvimento de ambientes no contexto da aprendizagem cooperativa apoiada por computadores (CSCL – *Computer-Supported Collaborative Learning*).

O modelo de cooperação está centrado nos conceitos de objetivo e processo cooperativo. Esses dois elementos determinam as características que o ambiente CSCL deve ter. O objetivo está relacionado a quatro aspectos: conhecimentos prévios, teorias de aprendizagem, formas de cooperação e fatores culturais. Já o processo engloba seis aspectos: atividades, avaliação, memória (armazenamento de dados), percepção, papéis e coordenação.

Nesta abordagem, existe o conceito de reuso de processos, É possível reutilizar, refinar e trabalhar a partir de processos já executados. Assim, processos já definidos podem servir como aprendizado para os grupos.

Também é descrito o protótipo de uma infra-estrutura, desenvolvida com base nesse modelo, composta de um ambiente cooperativo e um framework para auxílio à implementação de ferramentas educacionais cooperativas.

O ambiente instanciado recebeu o nome de COPLE (Cooperative Project Learning Environment) e é composto por um editor de processos e um editor de textos cooperativo.

6.2.6 Abordagem ISD (BROADBENT, 2002)

Broadbent (2002) apresenta a abordagem baseada em design instrucional, a ISD (*Instrucional Systems Design*) para criação de programas de EAD. Esta abordagem propõe um ciclo de oito etapas: definição do escopo do projeto, análise, projeto, desenvolvimento, teste piloto, entrega, avaliação, manutenção. Entre cada etapa existem momentos de decisão, para checagem do processo e reelaboração de estratégias (Figura 6.4).

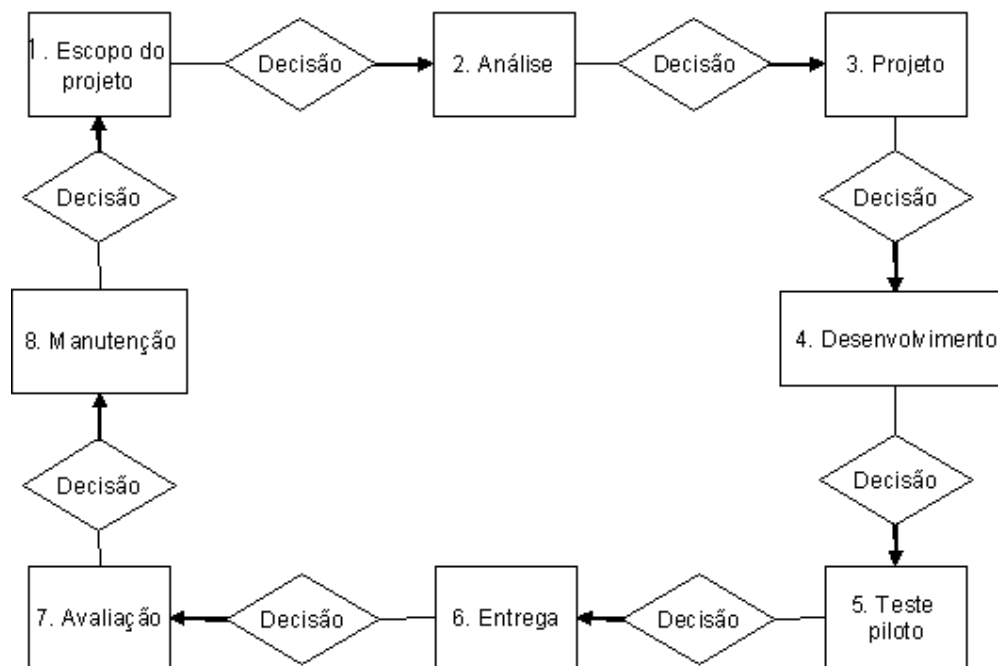


Figura 6.4: Abordagem ISD (BROADBENT, 2002)

Broadbent (2002) apresenta os prós e contras da abordagem ISD e destaca que, para alguns autores, a ISD é considerada “morta” porque possui uma estrutura linear, seqüencial, engessada. Por outro lado, afirma que para designers experientes a estrutura não é tão linear quanto parece.

Apesar de ser uma abordagem que tem cada vez menos adeptos, alguns trabalhos atuais estão centrados na ISD (BRITO; NÓBREGA; OLIVEIRA, 2005).

6.2.7 Análise Comparativa

Para analisar os principais trabalhos pesquisados, foram considerados os requisitos descritos na seção 6.1.

A Tabela 6.2 apresenta um quadro comparativo que analisa as abordagens estudadas, a partir dos requisitos e indica se o trabalho estudado apresentou descrição de implementação realizada.

Tabela 6.2 – Análise dos Trabalhos Relacionados

Abordagem	Hyper-Automaton	EduQNet	Baseada em Workflow	Baseada em Agentes	Baseada em Projetos	ISD
R1	Sim	Não	Sim	Sim	Sim	Não
R2	Não	N.A.	Não	Não	Sim	N.A.
R3	Sim	Não	Não	Não	Não	Não
R4	Sim	N.A.	Sim	Sim	Sim	N.A.
R5	Não	N.A.	Não	Sim	Sim	N.A.
R6	Sim	Não	Não	Sim	Sim	Não
R7	Não	Não	Sim	Não	Não	Não
Implement.	Sim	Não	Sim	Não	Sim	N.A.

Como as abordagens estudadas estão focadas em diferentes etapas da gerência de cursos, alguns requisitos não se aplicam a alguns dos trabalhos estudados. A indicação de N.A. significa que o requisito em questão não se aplica àquela abordagem.

6.3 Modelo Proposto

A partir dos estudos realizados com os trabalhos relatados nas seções anteriores e outros (JUNG, 2001; ANDRADE, 2003; PETERS, 2003; KRAMER et al., 1999; HORTON, 2000; RUHE, 1999), optou-se por definir um modelo para “Processo de Curso”. Este modelo baseia-se em uma analogia entre o curso a distância e o processo de desenvolvimento de software. Enquanto no primeiro é realizado um curso, no segundo é realizado um processo que gera a liberação de um produto de software.

O termo “Processo de Curso” é proposto aqui para o contexto de cursos a distância. Em alguns trabalhos relacionados (SIZILIO, 2000; RAPCHAN, 2002) o curso a distância já era visto como um processo, mas a metáfora de processo de software para Processo de Curso é proposta neste trabalho. Considera-se esta metáfora uma contribuição importante desta tese, no contexto de modelagem de cursos em EAD.

Para definição do modelo foram considerados os requisitos listados na seção 6.1 e os conceitos de Engenharia de Software, principalmente da área de tecnologia de processo de software, descritos no capítulo 4. É importante destacar que o modelo proposto não

está focado na questão da melhoria de processo, como outros trabalhos da área (HANNEY; WATSON, 2005), mas pode ser estendido, no futuro, para incorporar esta característica.

Desta forma, o processo de curso corresponde a um conjunto de atividades relacionadas que são desempenhadas por professores, alunos e tutores do curso, desde a concepção até a completa execução do mesmo.

A figura 6.5, inspirada no meta-modelo de processo de software (DERNIAME; KABA; WASTELL, 1999; LIMA REIS, 1998), ilustra o proposto para processo de curso.

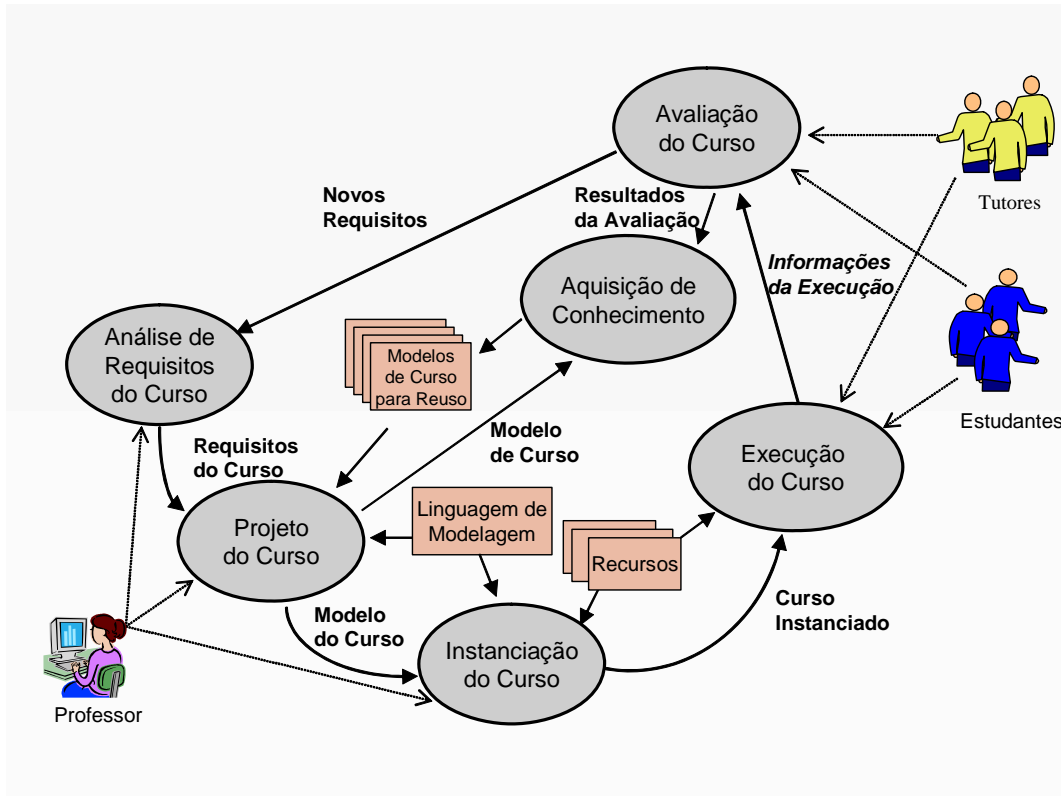


Figura 6.5: Modelo de Processo de Curso

Nas próximas seções os componentes do modelo são apresentados com detalhes. Como o modelo baseia-se na analogia com processo de software, em cada seção é descrito um mapeamento dos componentes do modelo de processo de curso para o modelo do processo de desenvolvimento de software.

Também é importante ressaltar que este é um modelo abstrato, ficando características específicas de algumas etapas restritas à implementação desse modelo.

6.3.1 Atividades

O processo de software pode ser compreendido como o conjunto de todas as atividades necessárias para transformar os requisitos do usuário em software (HUMPHREY, 1989; OSTERWEIL, 1987). Já o processo de curso compreende todas

as atividades necessárias para criar, executar e avaliar um curso que transforme os requisitos em aprendizagem.

O processo de curso a distância engloba seis atividades: análise de requisitos, projeto do curso, instanciação, execução do curso, avaliação e aquisição de conhecimento. Cada uma delas é descrita a seguir.

6.3.1.1 *Análise de Requisitos*

Nesta etapa são analisados os requisitos para criação e execução do curso e também aqueles resultantes da avaliação de cursos anteriores.

Apesar da maioria dos autores incluir a análise de requisitos dos cursos na etapa de projeto, optou-se por colocá-la em uma etapa independente devido a complexidade de sua implementação. No modelo de processo de software, esta etapa também está destacada devido a sua importância.

Descrever detalhadamente os requisitos do curso pode determinar o sucesso da etapa de projeto. Vários requisitos podem ser considerados na criação de um curso a distância. Como exemplo desses requisitos, podem ser destacados: tipo de público alvo (adultos, adolescentes, idosos, etc), área do conteúdo (exatas, humanas, etc), tempo disponível para realização do curso, número de alunos, número de tutores disponíveis, entre outros.

A representação dos requisitos depende da implementação do modelo. O uso de *templates* próprios é recomendado, mas a representação seguirá o padrão do ambiente de implementação utilizado em cada caso.

6.3.1.2 *Projeto do Curso*

A estrutura do curso a distância considerada na elaboração do modelo foi apresentada na seção 3.2 e pode ser sintetizada como uma seqüência de atividades relacionadas. Este seqüenciamento é definido pelo professor na etapa de projeto do curso. O professor pode definir, por exemplo, uma relação de dependência entre duas atividades (só quando uma termina a outra pode iniciar).

Nesta etapa, o professor define o modelo de um curso, sem escolher recursos ou identificar participantes. Estas ações serão realizadas na etapa de instanciação do modelo.

Como discutido anteriormente, a etapa de projeto é essencial para o sucesso de um curso. Para realizar projetos de uma forma mais eficiente, além dos requisitos identificados na etapa anterior, os docentes necessitam de uma linguagem para modelar o curso. Assim, um modelo de curso pode ser construído através dessa linguagem.

Torres e Dodero (2005) caracterizam as linguagens do tipo LPCEL (*Learning Processes Composition and Execution Languages* – Linguagens para Composição e Execução de Processos de Aprendizagem) como linguagens apropriadas para modelar e executar processos complexos de aprendizagem que integram diversas abordagens pedagógicas e baseiam-se na cooperação entre instrutores e aprendizes.

As linguagens do tipo LPCEL suportam adaptabilidade, flexibilidade e reusabilidade e possuem as seguintes características (TORRES; DODERO, 2005):

- Diversidade pedagógica
- Descrição do fluxo de aprendizagem

- Composição dinâmica e flexível
- Descentralização dos recursos de aprendizagem
- Separação do processo de aprendizagem e dos recursos
- Descrição das atividades e sua avaliação

A seleção da linguagem de modelagem, também restrita à implementação deste modelo, é muito importante para o sucesso da mesma. A escolha de uma linguagem adequada é imprescindível para que o professor possa modelar cursos com eficiência. Diversos trabalhos apresentam propostas para linguagens de modelagem de cursos (LAFORCADE, 2005; GAO; ZHANG; HAWRYSZKIEWYCZ, 2005; KARAMPUPERIS; SAMPSON, 2005; SIZILIO; EDELWEISS, 2001) e este é um tema de pesquisa bem explorado na atualidade.

Outro ponto importante da etapa de projeto do curso, no modelo proposto nesta tese, é a possibilidade dos professores reutilizarem ou não outros cursos. A reutilização de modelos de cursos, já realizados e avaliados positivamente, pode auxiliar o professor que deseja desenvolver um novo curso e não está familiarizado com as etapas de criação. Esta característica baseia-se no conceito de fábrica de experiências, proposto por Basili (1994), e é um dos pontos principais do modelo proposto.

A fábrica de experiências analisa e sintetiza todos os tipos de experiência, incluindo lições aprendidas, dados de projetos, relatórios e, finalmente, explicitam estas experiências através de repositórios (RUHE, 1999). Assim, os dados de cursos anteriores são analisados para que estas experiências sirvam de subsídio para novos cursos.

Se o professor optar por recuperar um modelo anterior de curso, um mecanismo de busca é utilizado e os modelos de cursos que atendem aos requisitos definidos são apresentados. O mecanismo de busca não é definido nesta proposta, ficando para a implementação a definição da estrutura de armazenamento e recuperação dos modelos.

6.3.1.3 *Instanciação*

Nesta etapa é feita a criação do curso propriamente dita. O modelo de curso é instanciado através da definição do cronograma, dos participantes (alunos e tutores) que interagirão com o curso, do detalhamento das atividades e dos recursos a serem alocados.

Tudo o que foi planejado, na etapa de projeto é colocado em prática. Pressupõe-se a existência de materiais didáticos e ferramentas computacionais para apoiar o curso. Na etapa de instanciação estes materiais e recursos são apenas selecionados.

É importante destacar que a produção dos materiais é uma tarefa externa ao modelo proposto, realizada por professores conteudistas. Algumas abordagens consideram a criação e edição do material como parte integrante do processo (SIZILIO; EDELWEISS, 2001; HASEGAWA; OCHIMIZU, 2005; TORRES; DODERO; ZARRAONANDIA, 2005), mas nesse modelo, pressupõe-se apenas a utilização de materiais pré-elaborados.

Esta etapa de instanciação poderia, em uma implementação do modelo, consultar automaticamente um repositório de objetos de aprendizagem e selecionar materiais didáticos. Atualmente, as pesquisas têm avançado muito na área de objetos de

aprendizagem (MIRANDA, 2004; GOMES, 2004; HASEGAWA; OCHIMIZU, 2005; TSUKAHARA; SEKI; OKAMOTO, 2005; CHEN et al., 2005).

Além disso, é na etapa de instanciação que o curso pode ter uma relativa personalização, dependendo do perfil de cada aluno. O modelo de curso é o mesmo (atividades propostas, seqüência entre elas, etc), mas na instanciação podem ser definidos recursos (materiais didáticos, ferramentas computacionais) diferentes para apoiar cada atividade, permitindo a execução diferenciada para cada aluno.

A etapa de projeto do processo de curso é análoga a de projeto do processo de software. Assim como modelos de cursos podem ser reutilizados, modelos de processos de software também servem de *feedback* para novos processos. Além disso, as linguagens para modelagem de cursos tem características sintáticas semelhantes às linguagens para modelagem de processos de software.

No modelo de processo de software um processo instanciado está pronto para execução. Da mesma forma, no processo de curso um curso instanciado é um curso que já pode ser executado.

6.3.1.4 Execução do Curso

Esta etapa do curso é realizada pelos alunos, sob supervisão dos tutores. Como será descrito na seção 6.3.3, os tutores podem ser agentes com este papel específico ou professores atuando nesta função.

Uma vez definida a estratégia de ensino, a execução de um curso corresponde à execução de um conjunto de atividades componentes do curso. Cada atividade é executada por um ou vários alunos (atividade cooperativa), necessita de recursos, possui cronograma previsto e cronograma real, possui um conjunto de atividades das quais ela depende, recursos que aloca e a descrição da tarefa a ser feita.

É importante ressaltar que os tutores podem ser os próprios professores que modelaram o curso ou outros professores, ou ainda monitores habilitados a realizar o acompanhamento do curso. Nesta fase é muito importante a existência de mecanismos para acompanhar e avaliar remotamente o desempenho do aluno no curso e coletar informações que serão utilizadas na etapa de avaliação.

A execução dos cursos ocorre da mesma forma que a execução de um processo de software. Um mecanismo de execução de processo de software que defina como serão executadas as atividades que compõem o processo pode ser utilizado para executar um processo de curso. Isto, se as linguagens de modelagem utilizadas para o processo de software e para o curso possuírem as mesmas características.

6.3.1.5 Avaliação

A etapa de avaliação é essencial para a qualificação de qualquer programa de EAD. Essa etapa visa analisar se o curso atingiu seus objetivos. Na avaliação é verificado se tudo o que foi planejado foi implementado. Além disso, é necessário avaliar se o objetivo pretendido (a aprendizagem dos alunos) foi atingido.

A avaliação de cursos, assim como acontece com a avaliação de processos de software, pode considerar dados coletados durante a execução do curso e outros, obtidos através de questionários respondidos por alunos e professores.

A avaliação do curso é realizada, principalmente, através da análise de sua execução. Diversos trabalhos propõem mecanismos que podem ser utilizados na avaliação dos alunos durante o curso (KASHIHARA; HASEGAWA, 2005; ARAÚJO; LUCENA FILHO, 2005; CINELLI et al., 2005; HATZILYGEROUDIS et al., 2005; ALVES; ERRICO; MESQUITA, 2002; SILVA; FEIJÓ, 2002; OTSUKA; ROCHA, 2002; CUNHA; FERNANDES, 2002), permitindo a obtenção de dados úteis também para a avaliação do curso.

A avaliação pode ocorrer em paralelo ou após a execução do modelo de processo. As conclusões obtidas na etapa de avaliação serão requisitos muito úteis ao projeto de novos cursos e as informações adquiridas são utilizadas nas atividades de análise de requisitos e aquisição de conhecimento. Esta característica também é análoga à avaliação de processos de software.

Uma definição abrangente de critérios para avaliação de cursos é muito importante. Vários trabalhos propõem metodologias para avaliação de cursos a distância (SOONG et al., 2001; JONSON, 2005; BELANGER; JORDAN, 2000; MOURA, 2002; AEDO; DIAZ, 2000; WESSON, 2002; BONAMY; CHARLIER; SAUNDERS, 2001), mas esta tese não aborda esta questão. Um estudo mais detalhado para definição de uma proposta para avaliação dos curso é uma sugestão para continuar a pesquisa realizada nesta tese. Este estudo é apresentado no Apêndice A.

6.3.1.6 Aquisição de Conhecimento

As pesquisas nas áreas de gerência de processos indicam que o conhecimento útil a ser armazenado sobre a realização de processos de desenvolvimento de software, além da própria descrição do processo executado, pode ser modelado através de métricas (ROCHA et al., 2001; PRESSMAN, 2001; JONES, 1996; FERNANDES, 1995; FLORAC, 1999; FRANCA, 1998; HUMPHREY, 1989; 1995; 1997).

A partir deste conceito, foi proposta a etapa de aquisição de conhecimento deste modelo. Essa etapa é responsável por armazenar na base de conhecimento os modelos de cursos já executados. Só serão retidos aqueles modelos de cursos considerados de qualidade, ou seja, que obtiveram boa avaliação.

A definição dos modelos que serão armazenados é feita através da análise do resultado das métricas resultantes da execução dos processos definidos e dos instrumentos de avaliação aplicados.

É importante observar que será armazenado apenas o modelo do processo e não detalhes de sua instanciação.

6.3.2 Artefatos

Os componentes denominados artefatos correspondem aos itens de dados manipulados, criados e utilizados durante o processo de curso. São eles:

- Modelo de curso: o modelo de curso, definido através da linguagem de modelagem, corresponde a representação das atividades que compõe o curso, suas dependências, e recursos a serem alocados. Em uma implementação do modelo, o mecanismo de execução será responsável por gerenciar o andamento do curso, baseado na definição e instanciação do modelo;
- Linguagem de modelagem: um modelo de curso é construído através da linguagem de modelagem. Como os docentes, de um modo geral, não possuem

conhecimentos técnicos, é interessante que a linguagem possua uma sintaxe simples, de preferência gráfica;

- Recursos: representam o material didático e todos os recursos tecnológicos de apoio à EAD (chat, fórum, exercícios automatizados, etc). Na instanciação de um curso são selecionados os recursos que apoiarão cada atividade. Dependendo do contexto de implementação do modelo, estes recursos são diferentes.

Como o modelo proposto prevê uma modelagem independente de conteúdo, a definição dos materiais e personalização do curso é feita através da escolha dos recursos a serem utilizados na execução.

6.3.3 Participantes

O processo de curso conta com a participação de atores (agentes humanos) que realizam funções diferentes. Para facilitar o entendimento das funções de cada agente, a Figura 6.6 apresenta o diagrama de Casos de Uso para gerência de cursos a distância.

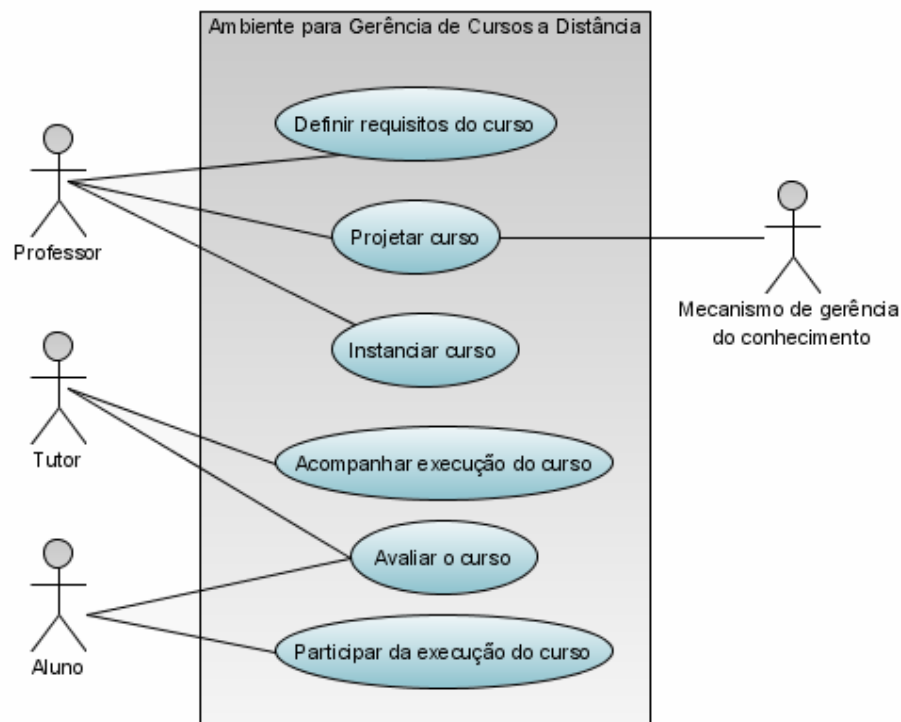


Figura 6.6: Diagrama de Casos de Uso

O diagrama apresenta uma visão geral da interação dos atores, no contexto da gerência de cursos a distância, definido no modelo de Processo de Curso.

No processo de curso estão definidos 3 atores primários (professor, tutor e aluno) e um secundário que apóia a realização do caso de uso Projetar Curso. A seguir cada um dos casos de uso indicados no diagrama é descrito:

- Definir requisitos do curso: esta função corresponde à definição dos requisitos que servirão de referência para o projeto do curso. O professor deve definir,

através de uma notação adequada (definida na implementação do modelo), os requisitos que devem ser atendidos na etapa de projeto do curso;

- **Projetar curso:** este caso de uso representa a função de modelagem do curso, desempenhada pelo professor. Ao projetar o curso, o professor descreve o modelo do curso, definindo um conjunto de atividades a serem realizadas, a estratégia de execução das tarefas, etc. Nesta função, o professor é apoiado pelo mecanismo de gerência de conhecimento. Esse mecanismo é responsável pela retenção de modelos de cursos de qualidade (com avaliações positivas) e recuperação de modelos de curso que possam ser reutilizados pelo professor, durante o projeto do curso;
- **Instanciar curso:** o caso de uso que se refere a instanciação do curso representa a tarefa na qual o professor define os métodos e técnicas que serão utilizados em cada tarefa prevista no projeto do curso e as ferramentas que irão automatizar a aplicação desses métodos e técnicas;
- **Acompanhar execução do curso:** este caso de uso ilustra a função que deve ser realizada pelo tutor para garantir o bom andamento do curso. O tutor precisa estar atento, acompanhando os aprendizes no curso, permitindo a identificação de atrasos, participação abaixo da média nas atividades por parte de alguns alunos, etc;
- **Avaliar o curso:** os tutores e alunos atuam da etapa de avaliação, opinando sobre o andamento do curso. Além disso, os alunos participam das atividades previstas para os cursos, gerando dados que serão utilizados na etapa de avaliação;
- **Participar da execução do curso:** o aluno, durante a execução do curso, realiza as atividades previstas pelo professor na etapa de projeto do curso. Este caso de uso ilustra esta participação dos alunos em atividades de conteúdo, interações ou trabalhos individuais ou em grupo. O aluno participa ativamente da construção do seu próprio conhecimento, podendo assumir os papéis de leitor, autor e colaborador, durante as atividades.

É importante ressaltar que o diagrama apresenta as funções do professor e do tutor separadamente, mas o docente pode assumir as funções de tutoria também, dependendo da situação de aplicação do modelo. Existe também a possibilidade de vários tutores atuarem em um mesmo curso, de acordo com o número de alunos.

6.4 Vantagens e Limitações do Modelo

Depois da definição do modelo e análise dos diversos trabalhos relacionados, alguns pontos positivos do modelo precisam ser destacados:

- **Analogia entre Processo de Curso e Processo de Software:** a utilização da metáfora para processo de software possibilita a utilização dos conceitos e ferramentas desenvolvidos para essa área. Três dos trabalhos citados na seção 6.2 (SIZILIO, 2000; RAPCHAN, 2002; SANTORO; BORGES; SANTOS, 2002) já apresentavam uma visão do processo de um curso a distância, mas não utilizando esta metáfora. Além disso, o estudo da área de Engenharia de Software para aplicação em Informática na Educação tem recebido atenção dos pesquisadores da área (GIRAFFA; MARCZAK; PRIKLADNICKI, 2005), demonstrando ser um domínio de aplicação que merece ser explorado;

- Flexibilidade e Dinamicidade do Modelo: apesar de dividir o processo de curso em etapas como em outras abordagens (BROADBENT, 2002; RAPCHAN, 2002) propõe uma visão mais flexível, representando-o como um ciclo que se auto-alimenta em alguns pontos, como na recuperação de novos requisitos e modelos de curso. Esta característica dinâmica do Processo de Curso merece ser destacada;
- Reutilização de Modelos de Curso: o reuso do modelo de cursos de qualidade já executados na criação de novos cursos pode melhorar muito a produtividade do professor. A utilização do conceito da fábrica de experiências na fundamentação do modelo oferece este recurso. O Hyper-Automaton (MACHADO, J., 2000) permite a utilização de uma mesma página em cursos diferentes, mas nenhuma das abordagens estudadas apresenta um repositório de modelos de processo para reutilização;
- Notação Utilizada: a utilização de uma notação abstrata na definição do modelo facilita sua instanciação e implementação. Alguns dos modelos apresentados (SILVA et al., 2003; SIZILIO et al.) utilizam-se de notações muito específicas, dificultando a implementação da estrutura proposta;
- Concretização do modelo: a implementação do protótipo PRO-EAD, descrito no próximo capítulo, permitiu a avaliação do modelo e comprovou sua viabilidade.

Uma limitação do modelo que pode ser citada é a impossibilidade de elaboração de seqüências de aprendizado diferenciado, com enfoques específicos para oferecer estudo individualizado. Só é possível realizar personalização do conteúdo, mas o modelo do curso permanece o mesmo.

Outra limitação refere-se a uma possível necessidade de alterações no modelo do curso durante a execução do processo. Esta possibilidade não está prevista no modelo.

6.5 Considerações sobre o Capítulo

Este capítulo apresentou o modelo para Processo de Curso, a principal contribuição deste trabalho. Também foram descritos alguns trabalhos que embasaram esta proposta e as limitações e vantagens do modelo em relação a esses trabalhos.

No próximo capítulo a metodologia de avaliação é descrita. Também é apresentado o PRO-EAD, um protótipo implementado no ambiente APSEE, que permitiu a realização de uma avaliação do modelo e demonstrou sua viabilidade de concretização.

7 AVALIAÇÃO DO MODELO

Como descrito no capítulo 2, a metodologia de avaliação empregada neste trabalho foi composta pela implementação de um protótipo, baseado no modelo proposto, e pela realização de um estudo de casos com professores com experiência em EAD e conhecimentos da área de Engenharia de Software.

Neste capítulo, o protótipo será apresentado brevemente e depois, a avaliação e os dados resultantes dessa avaliação serão descritos.

7.1 O Protótipo PRO-EAD

O protótipo implementado como parte da avaliação do modelo de Processo de Curso foi denominado PRO-EAD¹³ e foi implementado no ambiente APSEE-PROSOFT (descrito com detalhes no capítulo 5).

O PRO-EAD foi implementado com o auxílio do Sr. Heribert Schlebbe (pesquisador da Universidade de Stuttgart, associado do projeto PROSOFT)¹⁴. Este capítulo apresenta alguns pontos do desenvolvimento considerados relevantes para esta tese, uma vez que o protótipo não é o foco da mesma. Maiores detalhes de implementação podem ser encontrados em Schlebbe (2005).

Os objetivos específicos do desenvolvimento do protótipo foram:

- Fornecer uma infra-estrutura computacional que permitisse a experimentação prática dos conceitos propostos pelo modelo de processo de curso;
- Demonstrar a viabilidade de implementação, em um ambiente de desenvolvimento de software centrado no processo, do modelo proposto.

Como a definição de métricas de avaliação e políticas de instanciação extrapolam o contexto desta tese, as etapas de avaliação e aquisição do conhecimento, previstas no modelo, não foram implementadas. É importante ressaltar, porém, que uma vez definidas métricas de avaliação e políticas de reutilização para os cursos, o ambiente APSEE já possui toda a estrutura para implementação das etapas de avaliação e aquisição do conhecimento.

¹³ Disponível em <http://www.inf.ufrgs.br/prosoft>

¹⁴ Como o ambiente APSEE foi todo implementado no idioma inglês, devido ao projeto de cooperação com a Alemanha, este idioma foi também empregado na implementação do PRO-EAD.

7.1.1 Classes do PRO-EAD

O PRO-EAD foi implementado em PROSOFT-Java que é a denominação da atual implementação do paradigma PROSOFT, como já explicado anteriormente. O seu núcleo consiste de uma coleção de classes e interfaces Java que cooperam entre si para fornecer a funcionalidade requerida pelo paradigma PROSOFT.

Como descrito na seção 5.1, a especificação no ambiente PROSOFT começa pela construção das classes que compõem as ferramentas do ambiente (chamadas de ATOs – Ambientes de Tratamento de Objetos).

A Figura 7.1 mostra a classe que compõe o Ato PRO-EAD, definida para implementação do modelo. Esta classe representa uma extensão da classe APSEE, através da inclusão da classe ProEAD (indicada na figura).

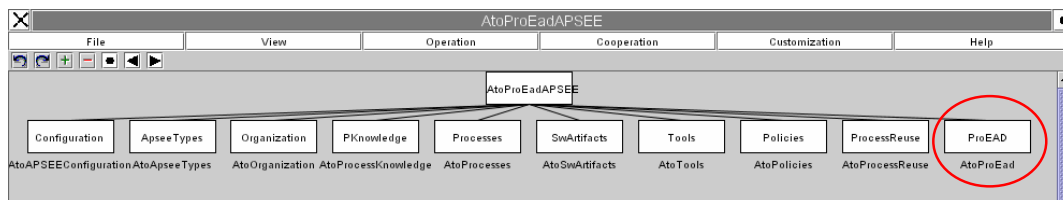


Figura 7.1: Classe ProEadAPSEE

A classe ProEad (Figura 7.2), por sua vez, é composta por um mapeamento de grupos para processos de EAD. A identificação do grupo é mapeada para uma classe Processes, do tipo AtoProEadProcesses. Este mapeamento pode permitir a instanciação de um mesmo processo de curso para grupos diferentes. Com isto, a personalização de curso pode ser alcançada.

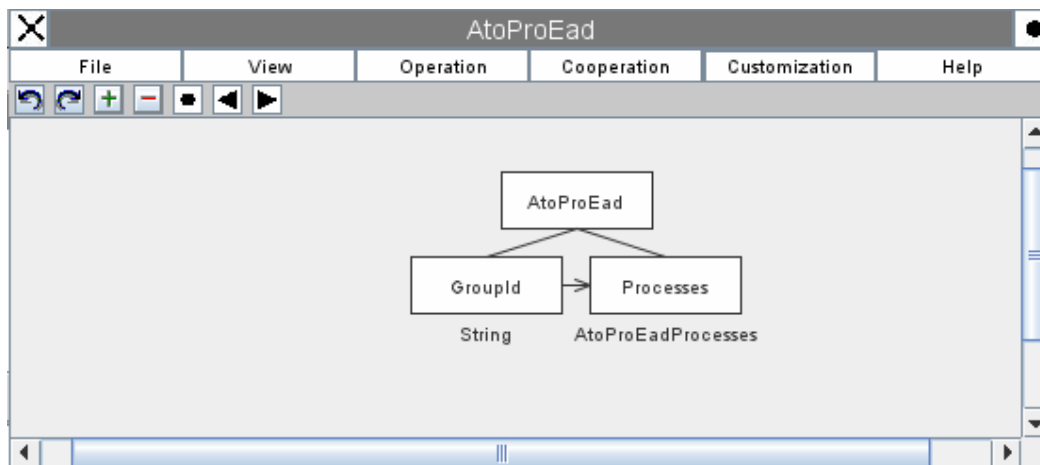


Figura 7.2: Classe ProEad

A Figura 7.3 apresenta a classe que representa o tipo *ProEadProcesses*. Essa classe representa as características principais do PRO-EAD. A composição da classe inclui o processo definido para o curso (*Group Process*) e um mapeamento dos alunos para instâncias do processo definido (*StudentProcess*). Em ambos os casos, o tipo utilizado é o processo (*Process*) definido no modelo APSEE.

O mapeamento do curso para cada um dos alunos permite que, durante a execução, cada um deles possa estar em pontos diferentes das atividades propostas.

Além disso, o tipo *ProEadProcesses* inclui o objeto *Cooperation* que permite definir que atividades do curso serão cooperativas, ou seja, realizadas em conjunto pelo grupo.

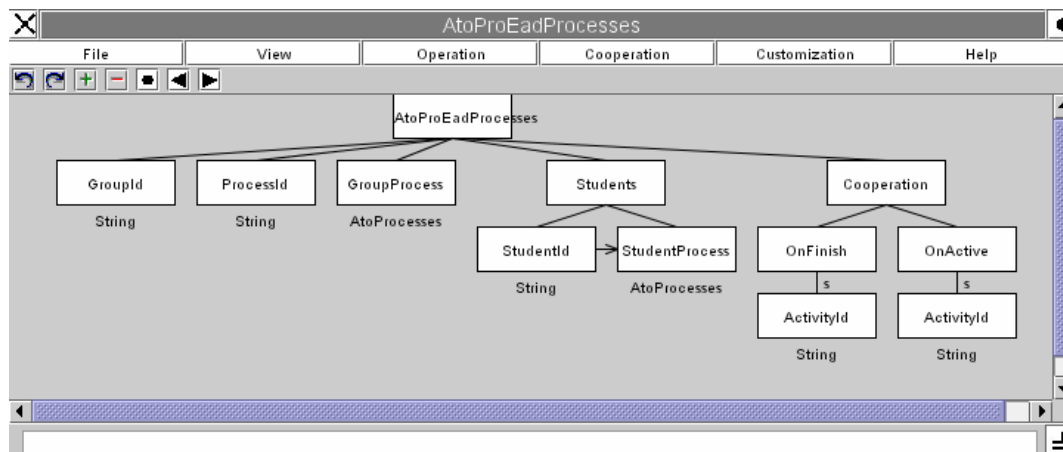


Figura 7.3: Classe ProEadProcesses

As demais classes que compõem o PRO-EAD foram herdadas do APSEE e suas definições podem ser encontradas em Lima Reis(2003).

7.1.2 Arquitetura do PRO-EAD

A arquitetura do PRO-EAD foi herdada do APSEE e adaptada para o novo contexto. A figura 7.4 apresenta esta arquitetura, explicada a seguir.

7.1.2.1 Meta-modelo

O Meta-modelo do APSEE pode ser reutilizado no contexto do PRO-EAD. Como foi demonstrado no capítulo 6, os componentes do modelo são análogos para Processo de Curso e Processo de Software.

Para adequá-los melhor à área, alguns repositórios do APSEE, foram renomeados para integrar o modelo do PRO-EAD (indicados em vermelho na Figura 7.4)

- Artefatos para Produções;
- Processos de Software para Processos de Curso ;
- Ferramentas para Recursos;
- Organização para Instituição.

Como definido na seção anterior, os processos no PRO-EAD possuem uma característica que os diferencia dos processos do APSEE. Os processos que serão armazenados no repositório do PRO-EAD terão a indicação de quais atividades foram definidas como cooperativas.

Além disso, na implementação do PRO-EAD, dois repositórios não foram considerados: Métricas e Políticas. Estes componentes contêm dados totalmente relacionados ao processo de software e não poderiam ser reaproveitados para processo de curso. Como a definição de métricas para avaliação de software e políticas para reutilização de modelos de processo não se enquadra no escopo desta tese, estes repositórios foram desconsiderados na implementação do PRO-EAD.

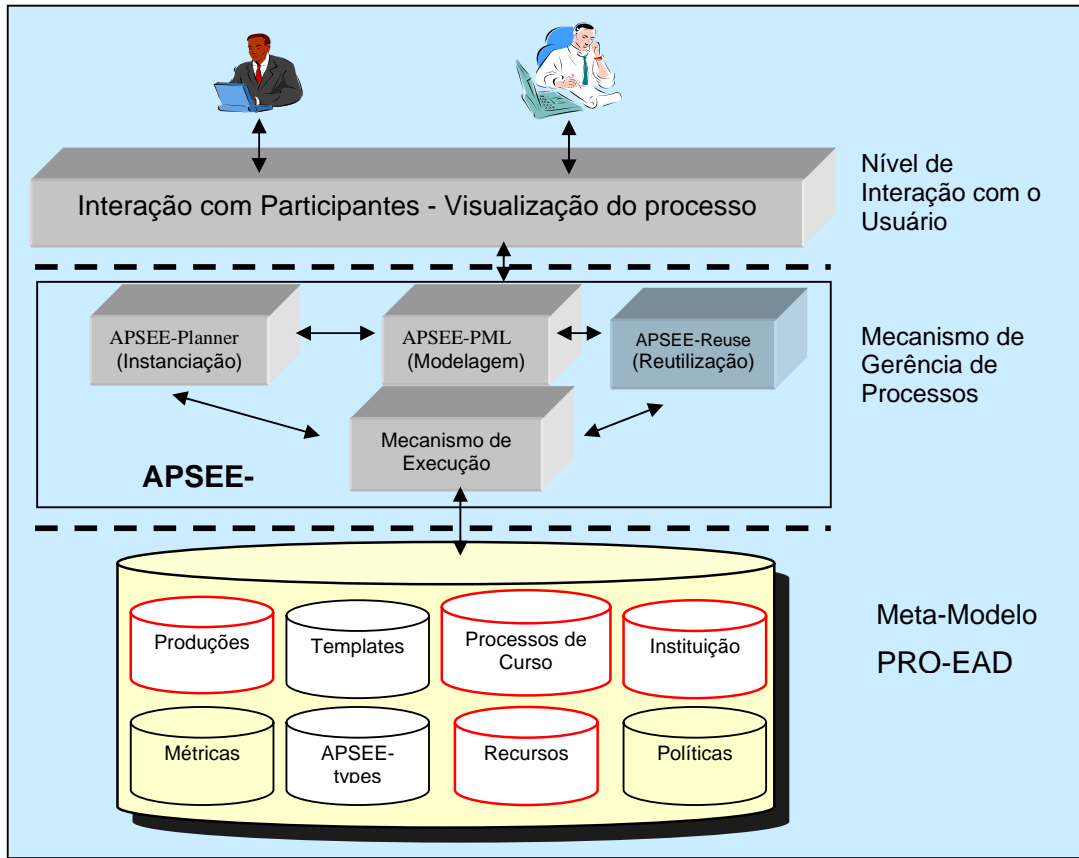


Figura 7.4: Arquitetura do PRO-EAD

O anexo A apresenta o diagrama de pacotes UML (BOOCH, 1998) do meta-modelo APSEE, ilustrando os componentes que foram utilizados no PRO-EAD. Alguns componentes do meta-modelo são refinados em novos pacotes. O anexo A apresenta também os diagramas das classes *SoftwareProcesses* e *ProcessModels*, muito importantes para a implementação do PRO-EAD. Maiores detalhes sobre os componentes e classes que compõem o APSEE podem ser encontrados em Lima Reis (2003).

7.1.2.2 Mecanismo de Gerência de Processos

Na camada do mecanismo de gerência de processo não foi necessária nenhuma alteração. A linguagem para modelagem de processos (APSEE-PML) é utilizada na modelagem dos cursos, instanciados através do APSEE-Planner. O Mecanismo de Execução de Processos também se adequou totalmente à execução dos processos de curso.

A modelagem de cursos requer formalismos de alto nível para descrever suas diversas características. Estes formalismos devem ser convenientes não somente para representação, mas também para execução. Com base nisso, o PRO-EAD utiliza-se da linguagem de modelagem do APSEE (denominada *APSEE-PML*) que permite representação gráfica de processos de software e, conseqüentemente, de cursos.

A linguagem de modelagem de processos de software do APSEE é baseada em redes de atividades que podem ser decompostas. Neste formalismo, um modelo de processo

pode ser construído a partir de símbolos gráficos conectados e o detalhamento do relacionamento com os outros componentes do modelo é feito através de formulários específicos que apóiam essa tarefa.

Os símbolos visuais do alfabeto da APSEE-PML são mostrados na Figura 7.5. A gramática dessa linguagem é definida formalmente em Lima Reis (2003).

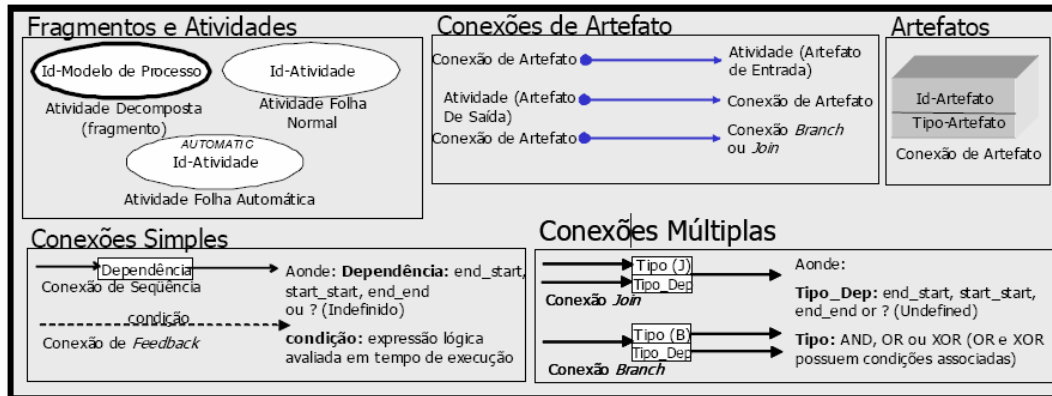


Figura 7.5: Símbolos da Linguagem APSEE-PML (Extraído de Lima Reis (2003))

Um exemplo de modelo de processo, definido através da APSEE-PML, contendo três atividades normais e duas conexões simples, é apresentado na Figura 7.6.

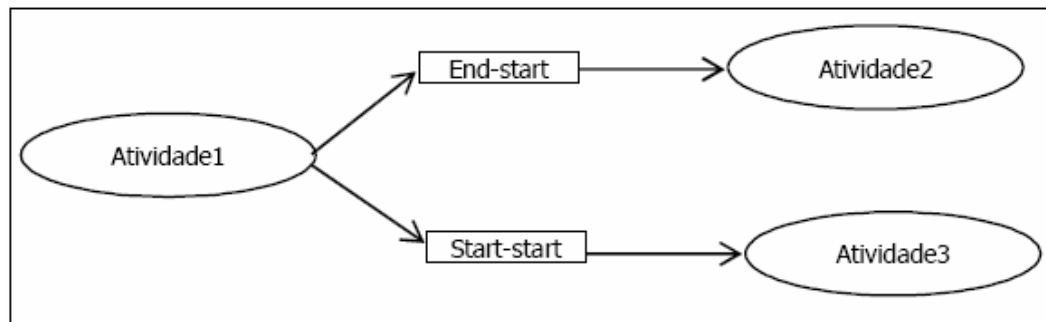


Figura 7.6 - Exemplo de processo na APSEE-PML com conexões simples de seqüência (LIMA REIS, 2003).

A APSEE-PML possui as características recomendadas para linguagens do tipo LPCEL, descritas na seção 6.3.1.2, permitindo assim uma representação do curso.

Como o modelo de curso, representado através da APSEE-PML, corresponde a representação das atividades que compõe o curso, suas dependências, e recursos, o fluxo de execução é indicado pelas conexões entre as atividades. O mecanismo de execução fica responsável em gerenciar o andamento do curso.

O Mecanismo de Execução de Processos do APSEE, formalmente definido através de gramática de grafos (LIMA REIS, 2003), interpreta o modelo de processo instanciado e gerencia as informações do ambiente de acordo com esse modelo. Como já foi demonstrada a analogia entre a modelagem de um processo de software e um curso, o mecanismo de execução é adequado para executar um curso modelado através da APSEE-PML.

O único componente da camada intermediária que não foi incorporado pela implementação do PRO-EAD foi o *APSEE-Reuse*. Como este componente combina as métricas e políticas para descrição de processos reutilizáveis e a definição dessas métricas e políticas não se constitui objetivo desta tese, o APSEE-Reuse é previsto na arquitetura do PRO-EAD mas não foi acessado na implementação.

7.1.2.3 *Nível de Interação com o Usuário*

A camada de interação com o usuário pôde ser integralmente utilizada na definição do PRO-EAD. Ela permite que o usuário (professor, tutor ou aluno) acompanhe o curso através da visualização da execução. Como o modelo de curso e o modelo de processo de software possuem as mesmas características, os mecanismos de visualização desenvolvidos para o APSEE (SOUSA, 2003) podem ser aplicados.

Na próxima seção o funcionamento do protótipo PRO-EAD será ilustrado através de um exemplo que engloba as etapas de projeto, instanciação e execução do processo de curso.

7.1.3 Funcionamento

A figura 7.7 apresenta a tela inicial do PRO-EAD. Quando o protótipo é acessado o usuário tem diversas opções, herdadas do ambiente APSEE.



Figura 7.7: Tela inicial do PRO-EAD

7.1.3.1 *Definição dos Requisitos*

A primeira etapa do modelo proposto corresponde à análise de requisitos do curso que será realizada pelo professor. Esta etapa é realizada no PRO-EAD através da definição das características do curso (Figura 7.8). O tipo do processo é definido e uma descrição textual pode ser incluída.

Este formulário já existia no APSEE e foi reutilizado no PRO-EAD.

Figura 7.8: Definição dos processos de curso no PRO-EAD

7.1.3.2 Projeto do Curso

Após a definição dos requisitos, vem a etapa de modelagem do curso, que corresponde ao projeto do curso, no modelo proposto. Esta etapa é realizada através da APSEE-PML (Figura 7.9) e permite a definição do modelo de curso, formado pelas atividades que compõe o curso, suas dependências e recursos. O fluxo de execução é indicado pelas conexões entre as atividades.

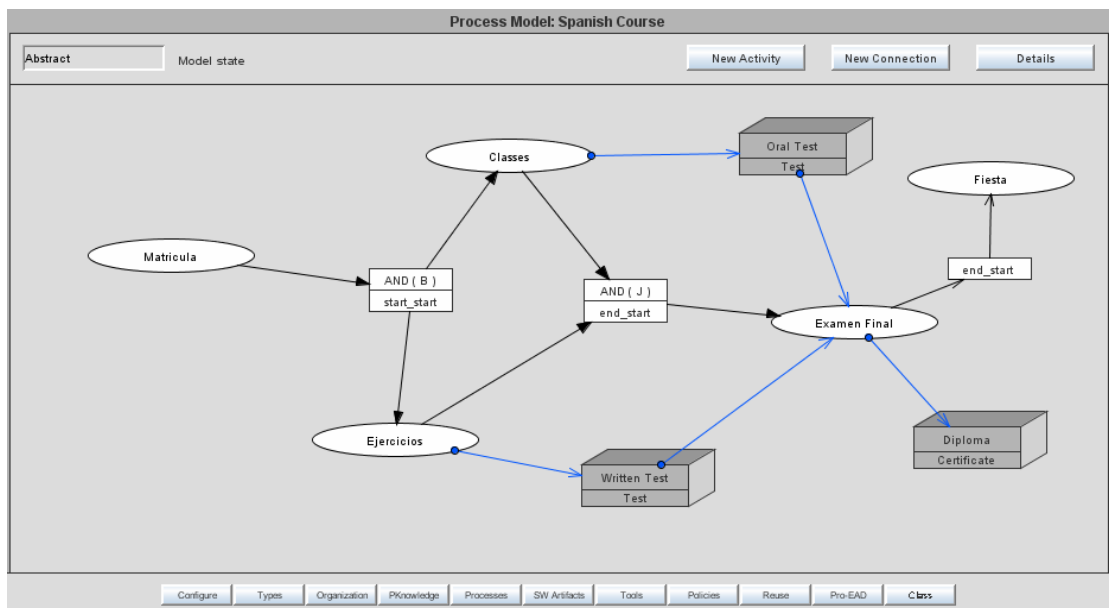


Figura 7.9: Modelagem dos processos de curso no PRO-EAD

O professor pode definir, através da linguagem gráfica, fluxos de aprendizagem, condições para avanço no curso e outras características providas pela APSEE-PML.

7.1.3.3 Instanciação

Depois da modelado, o curso está pronto para ser instanciado. Na implementação do PRO-EAD, um curso é instanciado para um grupo de alunos (Figura 7.10), Esse grupo já deve ter sido previamente cadastrado no ambiente, através do formulário de definição de agentes, herdado do APSEE (Figura 7.11). Como definido anteriormente, no PRO-EAD existem três tipos de agentes: estudantes, professores e tutores.

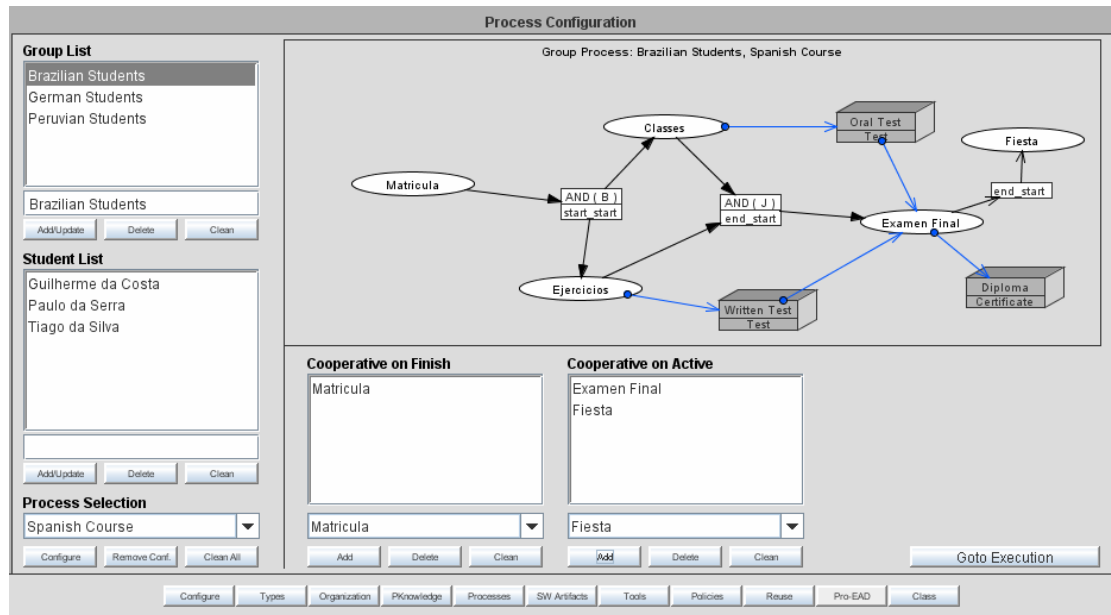


Figura 7.10: Instanciação de processos de curso no PRO-EAD

Figura 7.11: Formulário para definição de Agentes

É na etapa de instanciação, também, que são definidas que operações cooperativas serão cooperativas. Na implementação foram definidos dois tipos de atividades cooperativas: *Cooperative on Finish* (cooperativa na finalização) e *Cooperative on Active* (cooperativa na ativação).

A definição de uma atividade como cooperativa na ativação significa que esta atividade só entrará em execução quando todos os alunos do grupo estiverem aptos a iniciá-la. Já a cooperativa no final só libera as atividades posteriores a ela quando todos os alunos a tiverem finalizado.

Os formulários e estruturação da tela de visualização foram desenvolvidos nesta tese, assim como a implementação da instanciação das atividades cooperativas. A implementação dessas atividades era essencial para que as características de um curso a distância pudessem ser representadas.

O mecanismo de visualização do modelo definido na etapa de projeto já existia no APSEE e foi apenas combinado com os demais recursos.

7.1.3.4 Execução

Após a etapa de instanciação, o curso pode entrar em execução.

Durante a execução, o professor ou tutor pode visualizar a estrutura global de um curso e os estados das atividades que compõe o curso (Figuras 7.12 e 7.13). Os estados das atividades são representados por cores na representação gráfica e textualmente na janela inferior.

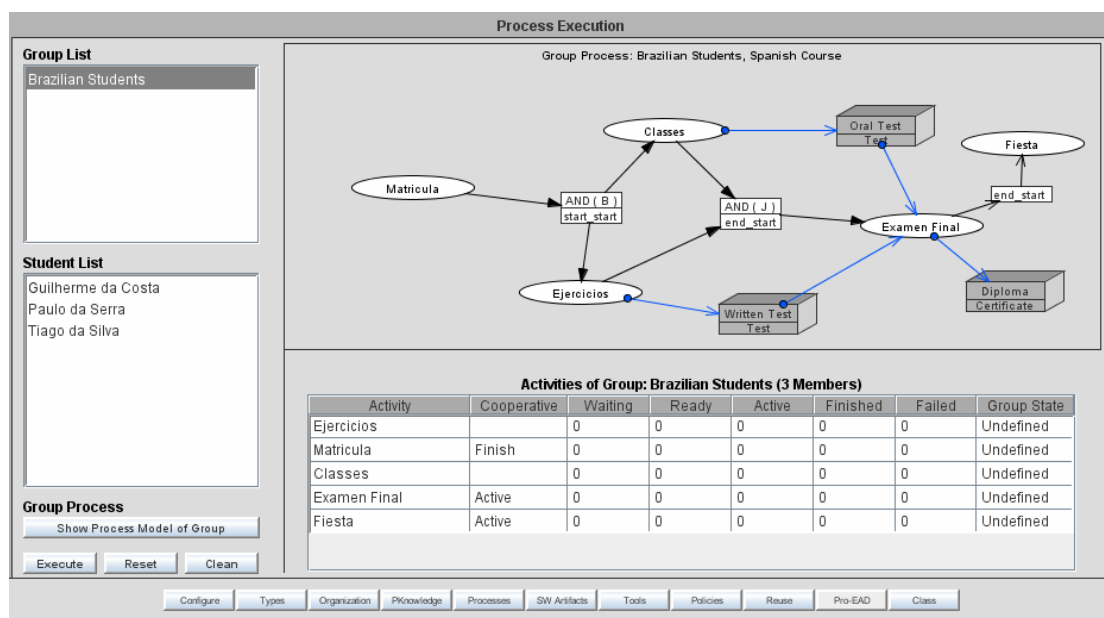


Figura 7.12: Visão global de um processo de curso não iniciado

A Figura 7.12 mostra o processo ainda não iniciado e na Figura 7.13 um processo em execução é exibido (com as cores indicando o estado das atividades¹⁵). A cor azul

¹⁵ No caso de cópias em preto e branco, o leitor poderá distinguir esta diferenciação nos tons de cinza da figura.

escuro indica uma atividade em execução e o amarelo representa que a atividade está pronta para ser iniciada.

Na visão global da execução de um curso (Figura 7.13) o professor pode visualizar no quadro inferior direito, em cada atividades, quantos alunos estão executando esta atividade, quantos estão esperando para executá-la, etc.

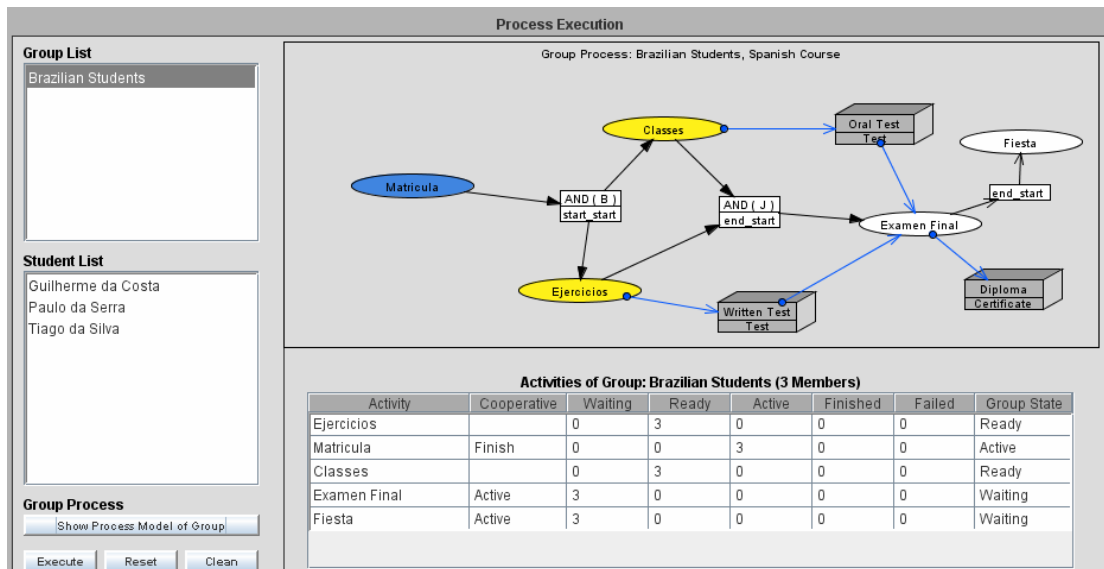


Figura 7.13: Visão global de um processo de curso em execução

O professor pode visualizar o processo de curso de cada aluno, com as cores indicando o estado das atividades (Figura 7.14). Esta visualização permite ao professor identificar, por exemplo, alunos com atraso na realização das atividades.

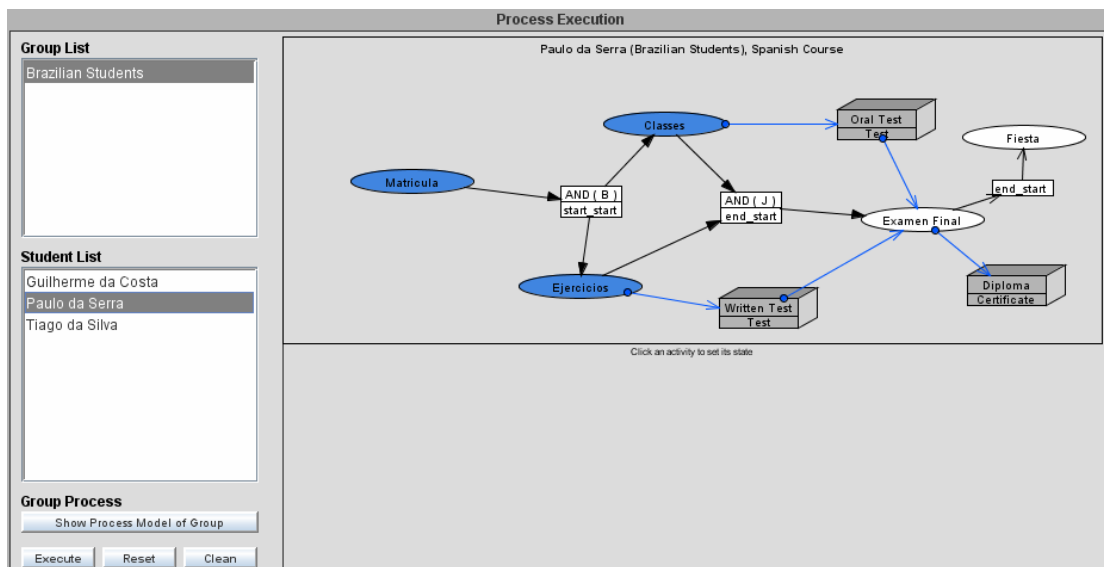


Figura 7.14: Visão do aluno de um processo em execução

Cada um dos alunos pode visualizar o curso e receber informações sobre as atividades que o compõem, selecionando cada uma delas (Figuras 7.15 e 7.16).

Durante a execução, o aluno deve selecionar a atividade que quer realizar e, se ela estiver disponível para execução, ativá-la através do botão *Active* (grifado na Figura 7.15). O aluno pode também pausar uma atividade através do botão *Pause* ou finalizá-la no botão *Finish*.

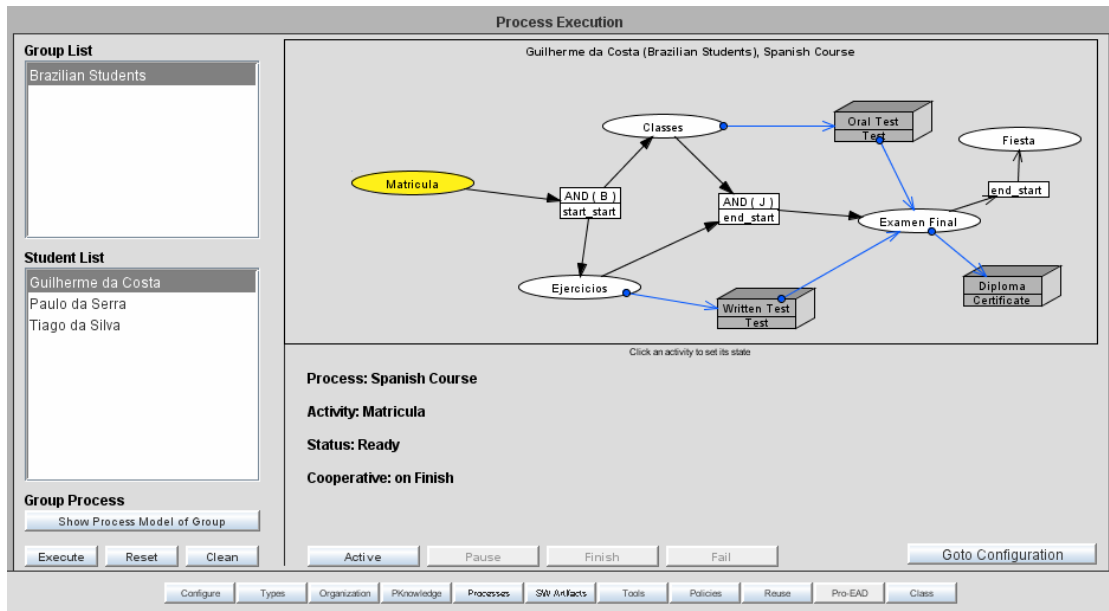


Figura 7.15: Visualização do curso pelo aluno, com atividade pronta para execução

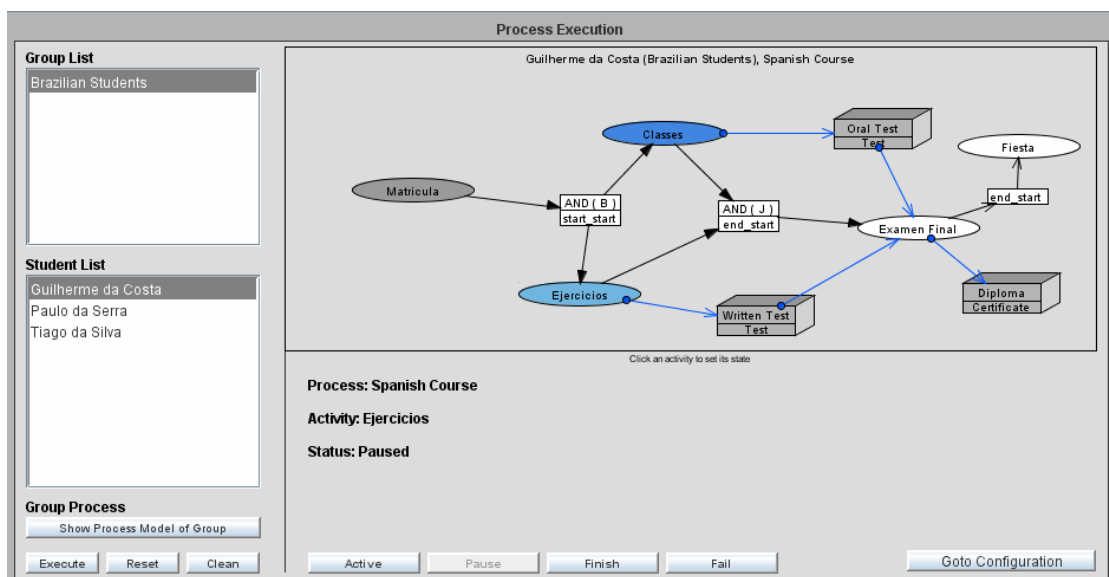


Figura 7.16: Visão do curso pelo aluno, com uma atividade ativada e outra em pausa

Finalmente, dois aspectos referentes à implementação do PRO-EAD precisam ser ressaltados:

- Atividades definidas como cooperativas devem ser executadas por todos os participantes do grupo para o qual o curso foi instanciado.
- Considera-se, na execução que o aluno realiza as atividades através de recursos externos ao ambiente PRO-EAD. Como, por exemplo, acesso a páginas de conteúdo, chats e fóruns de discussão. No ambiente são indicados apenas o início, término e possíveis pausas na execução das atividades.

7.2 Metodologia de Avaliação

A metodologia do estudo de caso prevê partir de uma necessidade especificada, selecionar uma amostra do universo em questão, elaborar um instrumento de coleta de dados, aplicar o instrumento e realizar uma análise dos dados (JUNG, 2004).

7.2.1 Seleção da Amostra

A população identificada para realização do estudo de caso representa a amostra da pesquisa. A amostra é uma parcela selecionada do universo ou população, um subconjunto do universo de pesquisa (REA; PARKER, 2002).

A maior vantagem da técnica de pesquisa por amostragem é a capacidade de generalizar a respeito de uma população inteira, extraindo-se inferências com base em dados obtidos de uma pequena parcela da população (REA; PARKER, 2002). Mas, para isso, o pesquisador deve estar relativamente certo de que a população selecionada possui o conhecimento e informações necessários à satisfação dos objetivos da pesquisa (SANTOS, 2000).

Por essas razões, a amostra desta pesquisa é não-probabilística e intencional, pois foram selecionadas pessoas com características específicas. Os sujeitos selecionados são professores extraídos de uma população considerada útil para a pesquisa.

A amostra contou com um universo de quatro professores que receberam os arquivos de instalação do protótipo e um questionário. A seleção desses professores foi realizada segundo os seguintes critérios:

- Experiência prévia em cursos a distância;
- Conhecimentos da área de Engenharia de Software.

Esses critérios de seleção da amostra foram considerados essenciais, pois a hipótese a ser testada pelo estudo de caso relaciona a área de EAD com a Engenharia de Software.

O objetivo deste instrumento é levantar a opinião dos professores que utilizaram o PRO-EAD, tanto sobre os aspectos referentes ao modelo quanto aos que se relacionam ao protótipo.

7.2.2 Questionário

Questionário é um instrumento de coleta de dados, constituído por um série ordenada de perguntas, que deve ser respondido por escrito e sem a presença de entrevistador. A elaboração de um questionário requer a observância de regras precisas, a fim de aumentar sua eficácia e validade (LAKATOS; MARCONI, 2001). O

questionário deve ser de fácil compreensão e consistente, além de representar uma análise apropriada e significativa dos dados (REA; PARKER, 2002).

Um bom questionário deve apresentar tanto questões abertas, quanto fechadas. Enquanto as questões fechadas apresentam opções de resposta e solicitam a escolha de uma delas, as perguntas abertas não possuem categorias pré-fixadas de respostas e dão ao entrevistado ampla liberdade para respondê-las (REA; PARKER, 2002).

A preocupação básica ao elaborar as perguntas deve ser, além da validade, a finalidade e a relação das questões com o objetivo de pesquisa (CERVO; BERVIAN, 2002).

Para realizar a avaliação da hipótese desta tese foi elaborado, como instrumento de pesquisa, um questionário apresentado no Anexo B. Esse questionário é formado por questões abertas e fechadas e foi elaborado pela autora e por uma especialista em pesquisa experimental¹⁶ para atender aos objetivos esperados para a etapa de avaliação.

O questionário foi elaborado para permitir a avaliação das principais características do modelo e do protótipo.

7.2.3 Aplicação do instrumento

Os professores participantes do estudo de caso modelaram cursos já ministrados por eles anteriormente, utilizando o protótipo implementado. Depois da modelagem, instanciação e execução do curso, responderam ao questionário (Anexo B) que foi enviado por correio eletrônico.

Através da resposta a esses instrumentos de avaliação e dos cursos modelados pelos professores, foi possível avaliar as contribuições deste trabalho. Os cursos modelados estão incluídos no Anexo C deste trabalho.

Buscou-se, com a aplicação deste questionário, avaliar os seguintes aspectos:

- Quanto ao modelo:
 - Analogia entre Processo de Curso e Processo de Software;
 - Completude (atividades previstas no modelo);
 - Adequação ao contexto;
 - Possibilidade de reutilização de modelos de cursos;
 - Limitações.
- Quanto ao protótipo:
 - Recursos disponíveis;
 - Linguagem de modelagem;
 - Mecanismos de instanciação e visualização da execução;
 - Facilidade de Uso;
 - Limitações.

¹⁶ O acesso a esta profissional foi possibilitado através de consulta ao Núcleo de Assessoria Estatística da UFRGS

Todos os professores participantes da avaliação enviaram o questionário preenchido. Este índice de 100% de respostas ao instrumento de avaliação deve-se ao pequeno tamanho da amostra.

7.2.4 Análise dos Dados

Os professores modelaram cursos já realizados anteriormente, utilizando o protótipo implementado. Antes e depois da modelagem, responderam a questionários (ANEXO I).

Através da resposta a esses instrumentos de avaliação e dos cursos, modelados pelos professores especialistas, foi possível avaliar as contribuições deste trabalho.

Vale ressaltar que a avaliação realizada é qualitativa já que a amostra é pouco significativa, mas justificada pela necessidade de participação de professores especialistas nas duas áreas (Engenharia de Software e EAD).

Para realizar uma validação do protótipo seria interessante submeter a ferramenta completa a uma amostra mais significativa, ou seja, um grupo bem maior de avaliadores.

A Tabela 7.1 apresenta um resumo das respostas dadas às questões fechadas do questionário desenvolvido.

Tabela 7.1 – Análise dos Questionários de Avaliação

Questão	Resultados Obtidos
1 - Você acredita que a analogia entre processo de curso e processo de software agrega vantagens à gerência de cursos a distância?	Sim – 100%
2 - Ainda sobre a analogia proposta, você identifica algum tipo de obstáculo que interfira na gerência de cursos?	Sim – 50% Não – 50%
3 - Você considera importante a possibilidade de reutilização de cursos a distância?	Sim – 100%
4 - Como você classificaria o PRO-EAD com relação à modelagem e gerência de cursos a distância?	Adequado – 100%
5 - O PRO-EAD oferece recursos para gerência de cursos não encontrados em outros ambientes de apoio à EAD não orientados a processo?	Sim – 100%
6 - Classifique a linguagem de modelagem de cursos oferecida pelo ambiente PRO-EAD:	Adequada – 100%
7 - Classifique o mecanismo de instanciação de cursos do ambiente PRO-EAD:	Adequado – 100%
8 - Classifique o mecanismo de visualização da execução de cursos disponibilizado pelo ambiente:	Adequado – 75% Parcialmente – 25%
9 - Você considera que o ambiente PRO-EAD é de fácil utilização?	Sim – 25% Parcialmente – 75%
13 - O ambiente PRO-EAD correspondeu às suas expectativas?	Sim – 100%

Os resultados obtidos demonstram, principalmente, a adequação da proposta deste trabalho ao contexto (Educação a Distância), pois todos os entrevistados declararam que

a analogia proposta nesta tese agrega vantagens à gerência dos cursos. Dentre as justificativas apresentadas pelos docentes para esta questão, vale destacar as seguintes:

- “Do ponto de vista da organização das atividades, do estabelecimento de um processo com etapas bem definidas, dos papéis em cada uma nas atividades, da visão de artefatos como resultado das atividades realizadas, considero isto muito importante para a questão do planejamento de um curso, principalmente pela organização e padronização obtida. Esta analogia permite agregar muitas contribuições da área de gerência de processos, uma vez que todo processo pode ser planejado, executado, controlado e avaliado. Além disto, a possibilidade de reuso é uma importante contribuição, pois um bom processo de curso, reutilizável, pode diminuir de forma significativa o esforço de planejamento de novos cursos”.
- “Diversos fatores justificam as vantagens da analogia. Entre elas: melhor visualização do fluxo de execução do curso; maior controle sobre sua execução; facilita o planejamento do curso; pode viabilizar a avaliação da qualidade e conseqüentemente a melhoria continua do curso; e permite a monitoração da execução do curso”.

Pode-se perceber, nas respostas, que os especialistas vislumbraram uma provável melhoria nos cursos, decorrente da incorporação das contribuições da área de processos de software proporcionada pela analogia. Entretanto, dois especialistas identificaram obstáculos, decorrentes da mesma:

- “Os conceitos relativos aos processos de software não são fortemente conhecidos por outros profissionais (p.ex: profissionais da área de educação), que muitas vezes não tem a visão sistêmica e organizada em termos de processo como a desta proposta. Apesar disto, considero a abordagem de fácil aprendizado, o que facilitaria sua utilização por pessoas de outras áreas que não de Engenharia de Software”.
- “A analogia pressupõe que seja conhecido pelo menos um dos modelos de processo proposto. No caso, considera-se que o educador conheça o modelo de processo de software e possa adaptá-lo ao processo de curso. Acredito que este seja um dos principais obstáculos.”

O obstáculo apontado pelos docentes foi justificado pela própria resposta do primeiro. Apesar da maioria dos docentes não possuir conhecimentos da área de processos de software, este não é um pré-requisito para a utilização da abordagem proposta.

A terceira pergunta do questionário abordou a questão da reutilização de cursos. Neste ponto, as respostas foram unânimes. Todos os docentes consideraram importante a possibilidade de reutilizar cursos anteriores. Algumas das justificativas:

- “A possibilidade de reutilização de cursos pode auxiliar no controle da qualidade dos cursos oferecidos, pois permite o planejamento de um novo curso via o histórico de cursos anteriores. Além disso, a reutilização pode viabilizar o processo de melhoria continua dos cursos oferecidos”.
- “Sem dúvida. O planejamento de um curso é sempre uma tarefa complexa, e exige um grande embasamento de experiências anteriores para que possamos definir um plano bem adequado ao contexto do curso em questão. Aproveitar

experiências anteriores, permitir a adaptação de processos existentes e bem sucedidos é uma grande vantagem, pois irá qualificar de forma significativa o planejamento realizado”.

- “Extremamente importante, pois o processo de modelagem de um curso demanda um certo tempo do professor. O reuso trás uma grande vantagem, pois pode-se realizar essa tarefa uma vez e depois reusar partes do curso para outros cursos ou alunos. Atualmente o reuso está sendo a alternativa encontrada para os professores não precisarem realizar toda a tarefa de modelagem a cada novo curso”.

Nas justificativas dadas à questão número 3, mais uma vez a idéia de melhoria na qualidade dos cursos está atrelada a uma característica do modelo. Este é um ponto muito importante para justificar a abordagem escolhida para a realização desta tese.

As questões de número 4 à 8 estão relacionadas aos recursos do protótipo PRO-EAD. Na maioria das respostas, os mecanismos oferecidos pelo ambiente foram considerados adequados:

- “Considero o protótipo adequado no sentido de que todas as etapas de um curso podem ser previstas, planejadas, executadas, monitoradas e avaliadas. Além disto, a abordagem de utilização de requisitos para um curso me parece adequada, no sentido de que cada curso tem suas peculiaridades e a definição de requisitos auxiliam nesta reflexão. Somente o ato de analisar e elicitar os requisitos do curso, já o torna mais qualificado, uma vez que o curso planejado será focado em objetivos bem definidos e planejados (isto, muitas vezes nem é feito de forma organizada em iniciativas manuais)”.
- “Fiquei satisfeito com a facilidade de representar as etapas do curso. A flexibilidade que a ferramenta apresenta permite modelar os processos de praticamente qualquer curso. A gerência do curso terá como consequência uma ferramenta de apoio a manutenção do curso”.
- “A linguagem é adequada, pois possibilita a definição apropriada das atividades, suas conexões, dependências, etc., sendo de fácil utilização e interpretação”.
- “A linguagem apresenta os elementos sintáticos e semânticos necessários para a modelagem do curso. Além disso, apresenta um conjunto bastante rico de possibilidades para a conexão entre as atividades”.
- “Considero o mecanismo de instanciação adequado, principalmente em função da simplicidade e coerência permitindo uma atividade bastante intuitiva”.
- “O mecanismo permite a monitoração da execução do curso de forma adequada. O estado de cada atividade é apresentado, possibilitando a percepção do início, atraso, e conclusão. A estratégia de visualização permite o acompanhamento da execução do curso, identificando as atividades realizadas e por realizar por cada grupo ou aluno”.
- “O mecanismo permite a definição dos artefatos específicos a serem produzidos/utilizados de forma bastante flexível. Além disto, o ambiente permite a definição dos grupos e agentes e sua associação de forma simples às atividades do processo”.

Apenas um docente considerou o mecanismo de visualização da execução parcialmente adequado. Sua justificativa foi:

- “É adequado no sentido de que a visualização é bastante fácil de ser utilizada. Apesar disto a interface poderia ser melhorada, principalmente nos casos onde são modelados processos médios/grandes, onde a visualização é prejudicada”.

O ponto que determinou sua resposta foi a questão da interface da linguagem de modelagem de processos. A linguagem, assim como a interface, são oferecidas pelo ambiente APSEE e utilizadas na implementação do PRO-EAD. Logo, esta limitação será reportada à equipe de implementação do grupo PROSOFT.

A questão de número 5 indaga se o protótipo implementado nesta tese oferece recursos não encontrados em ambientes não orientados a processo. Para esta questão, todos os docentes listaram recursos que se destacam no PRO-EAD, como os listados abaixo:

- “Os ambientes mais conhecidos são mais focados em conteúdo do que propriamente no planejamento e controle das atividades e seus resultados. Além disto, a possibilidade de reuso, mensuração e avaliação do processo são características que não são comumente encontradas nos demais ambientes”.
- “A representação do curso através de redes de atividades. A definição explícita das dependências entre as atividades (fluxo de execução). Controle individual e em grupo da execução das atividades do curso”.

Realmente, acredita-se que a representação da estrutura do curso através de uma rede de atividades possa facilitar, ao professor, as tarefas envolvidas na gerência de cursos a distância.

Já a questão 9, foi a que identificou o maior índice de docentes apenas parcialmente satisfeitos. A maioria dos docentes considerou que o PRO-EAD possui uma facilidade parcial de utilização:

- “O sequenciamento das ações na ferramenta é adequado e bastante completo, porém a usabilidade da ferramenta é um pouco complexa para usuários mais leigos em informática”.
- “A ferramenta pode ser melhorada em relação a facilidade de uso. Acredito que para usuários não especialistas em computação tenham bastante dificuldade no uso da ferramenta”.

A interface do PRO-EAD, herdada do APSEE, realmente possui limitações, principalmente no que se refere a aspectos de usabilidade. Estes resultados também serão repassados à equipe de implementação do grupo PROSOFT.

As questões abertas (10, 11 e 12) indagam sobre limitações e possíveis sugestões para melhoria do ambiente. As limitações apontadas foram em relação à modelagem dos cursos:

- “Algumas disciplinas que ministram ocorrem simultaneamente com outros cursos. Em determinado momento do curso, os professores devem ter concluído determinados módulos para que eu possa continuar ministrando a minha disciplina. Tive dificuldades em representar isto através do PRO-EAD. Acredito possa existir na visão do professor subdivisões, por exemplo: a visão do

professor e visão do professor coordenador onde podem ser visualizadas as relações entre as disciplinas”.

- “As principais limitações foram em relação a ferramenta, principalmente em sobre a usabilidade na criação das atividades do processo. Quanto à sistemática proposta e o entendimento da mesma nenhum problema”.

Com relação à primeira limitação, o modelo pressupõe a gerência de cursos isolados, mas este problema poderia ser resolvido com a possibilidade de decomposição de atividades, prevista no modelo APSEE. Cada um dos cursos que se relacionam seria uma atividade, decomposta em sub-atividades. A descrição da modelagem e gerência de atividades decompostas pode ser encontrada em Lima Reis (2003).

Na questão 12, quando indagados acerca de sugestões para melhoria do ambiente, os docentes citaram:

- “Melhorar alguns recursos de interface e algumas facilidades no software”.
- “Acredito que a evolução normal deste ambiente seria sua migração para um ambiente web”.
- “Melhoria da usabilidade na modelagem das atividades (está um pouco complicado para leigos). Possibilidade de vinculação do conteúdo das aulas aos artefatos indicados no processo, suporte a definição e registro dos requisitos do curso. Outra melhoria interessante seria a portabilidade futura para ambiente *Web*.”

Finalmente, a questão 13 é referente à satisfação dos docentes com o protótipo. Todos os docentes declaram que o PRO-EAD correspondeu totalmente às suas expectativas. Algumas das justificativas foram:

- “É possível modelar um curso, com todas as suas atividades e pré-requisitos. As funcionalidades oferecidas de instanciação de curso e visualização do status de cada aluno são pontos fortes da ferramenta.”
- “O ambiente se apresentou como uma ferramenta de fácil manipulação permitindo representar o ambiente do curso. Também atendeu as necessidades gerenciais relacionadas principalmente no que se refere ao planejamento do curso.”

A avaliação realizada permitiu constatar a validade da analogia proposta e a adequação do modelo ao contexto da Educação a Distância.

Além disso, a satisfação relatada pelos docentes com as ferramentas do PRO-EAD, implementado no ambiente APSEE, comprova a viabilidade de implementação do Modelo de Processo de Curso em um ambiente de desenvolvimento de software orientado a processos.

Estas duas conclusões obtidas através da avaliação, comprovam a hipótese formulada nesta tese e apresentada no capítulo 2.

7.3 Considerações sobre o Capítulo

Este capítulo descreveu a metodologia de avaliação do modelo de Processo de Curso, proposto no capítulo 6.

Primeiramente, foi apresentada uma visão geral do PRO-EAD, protótipo implementado no ambiente APSEE, que permitiu a realização da avaliação e demonstrou a viabilidade de concretização do modelo de Processo de Curso. É importante ressaltar que, apesar do ambiente APSEE ter sido escolhido, o modelo pode ser concretizado em qualquer outro ambiente que possua apoio para gerência de processos de software, já que esta foi a metáfora utilizada na definição do modelo.

Após a apresentação do protótipo, o estudo de caso realizado com os professores selecionados foi descrito, assim como a análise realizada com os dados obtidos.

No próximo capítulo, as conclusões obtidas com a pesquisa realizada são apresentadas através da descrição das contribuições da tese. Além disso, são propostos temas para dar seguimento a este trabalho.

8 CONCLUSÃO

Esta tese situa-se na área de Engenharia de Software aplicada à Educação a Distância e propôs uma solução para a seguinte questão de pesquisa:

Que elementos um modelo, embasado pela Engenharia de Software, precisa conter para representar as atividades envolvidas na gerência de cursos a distância?

Esta solução baseou-se na seguinte hipótese:

Um modelo que descreve as atividades de gerência de cursos a distância deve oferecer elementos análogos aos que compõem o modelo do ciclo de vida de um processo de software, e pode ser implementado em um ambiente de desenvolvimento de software orientado a processos;

Foi definido um modelo para Processo de Curso a Distância baseado em conceitos da área de Tecnologia de Processo de Software. Para entendimento do modelo, o texto apresentou uma revisão bibliográfica de conceitos de Educação a Distância e Tecnologia de Processo de Software, além da descrição do modelo.

Um das metas almejadas na definição do modelo é auxiliar o professor que não possui um conhecimento prévio de desenvolvimento de cursos a distância, através da recuperação de experiências passadas de criação de cursos.

Para demonstrar a viabilidade de concretização do modelo, foi implementado um protótipo, utilizando o ambiente APSEE-PROSOFT, que também foi descrito no decorrer do trabalho.

Nas seções a seguir são descritas as contribuições desta tese, os resultados alcançados e são discutidas as perspectivas para dar continuidade a este trabalho.

8.1 Contribuições

A principal contribuição deste trabalho está centrada na definição de um modelo para processo de curso, através de uma metáfora com o processo de desenvolvimento de software.

Foram estudados diversos trabalhos que propõem representações para cursos a distância (BROADBENT, 2002; MACHADO, J., 2000; SIZILIO, 2000; KRAMER et al., 1999; RAPCHAN et al., 2002; HORTON, 2000; SILVA et al., 2003), mas nenhum deles utiliza-se desta metáfora para representá-los.

Outro diferencial do modelo proposto é o foco na reutilização de modelos de curso, baseado no conceito da fábrica de experiências proposto por Basili (1994). Alguns trabalhos citam a utilização deste conceito na educação (RUHE, 1999), mas não apresentam nenhuma proposta de modelo para viabilizar isto.

Conforme apresentado na seção 6.4, os principais pontos positivos do Modelo de Processo de Curso são:

- Analogia entre Processo de Curso e Processo de Software, possibilitando a utilização dos conceitos e ferramentas desenvolvidos para a área de Tecnologia de Processo de Software;
- Flexibilidade e dinamicidade do modelo, representado como um ciclo que se auto-alimenta;
- Reutilização de modelos de curso, permitindo a melhoria da produtividade do professor;
- Utilização de uma notação abstrata na definição do modelo, facilitando sua instanciação e implementação;

Além da definição do modelo, outras contribuições secundárias do trabalho precisam ser destacadas:

- Concretização do modelo: a implementação do protótipo PRO-EAD permitiu a avaliação do modelo e comprovou sua viabilidade de implementação.
- Definição de um conjunto de critérios para avaliação de cursos em EAD (DAHMER; VICCARI; NUNES, 2005a; DAHMER; VICCARI; NUNES, 2005b). Estes critérios (Apêndice A) buscam identificar os aspectos que poderiam ser considerados na avaliação de cursos. Eles foram propostos como um passo inicial para a uma metodologia de avaliação que possa ser empregada na implementação do modelo de processo de curso ou em outras abordagens;

8.2 Análise dos Resultados

Esta seção faz uma análise do modelo proposto com base na avaliação apresentada no capítulo 7, enfatizando os requisitos listados no capítulo 6.

Com a definição do modelo a Tabela 6.2 foi estendida através do acréscimo de mais uma coluna indicando a abordagem proposta nesta tese (Tabela 8.1).

Tabela 8.1 – Quadro Comparativo com os Trabalhos Relacionados

Abordagem Requisito	Hyper- Automaton	EduQNet	Baseada em Workflow	Baseada em Agentes	Baseada em Projetos	ISD	Modelo de Processo de Curso
R1	Sim	Não	Sim	Sim	Sim	Não	Sim
R2	Não	N.A.	Não	Não	Sim	N.A.	Sim
R3	Sim	Não	Não	Não	Não	Não	Parcial
R4	Sim	N.A.	Sim	Sim	Sim	N.A.	Sim
R5	Não	N.A.	Não	Sim	Sim	N.A.	Sim
R6	Sim	Não	Não	Sim	Sim	Não	Sim
R7	Não	Não	Sim	Não	Não	Não	Não
Implem.	Sim	Não	Sim	Não	Sim	N.A.	Sim

Como indicado na tabela 8.1, o modelo apresentado nesta tese atende a maioria dos requisitos listados na seção 6.1 e foi implementado, comprovando sua viabilidade e permitindo a realização da avaliação descrita no capítulo 7 desta tese.

As limitações relacionadas aos requisitos R3 e R7 são discutidas na próxima seção.

8.3 Questões em aberto e perspectivas para o futuro

O trabalho aqui apresentado representa um primeiro passo no sentido de fornecer uma solução para o problema da gerência de cursos em Educação a Distância. Muitas questões ainda estão em aberto, definindo pontos que devem ser explorados em outros trabalhos com o mesmo tema. A seguir são apresentadas algumas sugestões para futuras pesquisas:

- Uma limitação do modelo que pode ser citada é a impossibilidade de elaboração de seqüências de aprendizado diferenciado, com enfoques específicos para oferecer estudo individualizado. Só é possível realizar personalização do conteúdo, mas o modelo do curso permanece o mesmo. Pesquisas podem ser realizadas para estender o modelo, permitindo estruturas de cursos diferenciadas;
- Outra limitação refere-se a uma possível necessidade de alterações no modelo do curso durante a execução do processo. Esta possibilidade não está prevista no modelo e pode ser explorada por estudos que investiguem o impacto dessas alterações durante a execução, definindo mecanismos permitam a realização das mesmas.
- Para permitir o reuso de processos de curso é necessário definir métricas que serão utilizadas na etapa de avaliação e políticas adequadas a reutilização de modelos de curso. Estas tarefas são extremamente complexas, por isso, decidiu-se propor um conjunto de métricas para avaliação de cursos a distância, como uma contribuição adicional deste trabalho. Estes critérios são descritos no Apêndice A e servem como sugestão para continuidade deste trabalho.
- Uma etapa fundamental do modelo é a análise de requisitos do curso. Esses requisitos podem desempenhar um papel importante para o reuso de curso. Considerar uma ligação entre a especialização dos requisitos e as possibilidades de reuso pode colaborar para a definição de *templates* cada vez mais completos, permitindo uma maior objetividade na recuperação dos modelos de cursos para reutilização.
- Uma importante atividade a ser realizada é a condução de uma avaliação empírica que determine através de um estudo de caso, com uma amostra mais consistente e não especializada, os benefícios reais da utilização da ferramenta durante a gerência de cursos.

Espera-se que essas perspectivas de continuidade para o trabalho possam estimular outros pesquisadores à avançar ainda mais na busca de modelos e ferramentas adequados à gerência de cursos a distância.

REFERÊNCIAS

ABEL, M. **Introdução aos Sistemas Especialistas**. Porto Alegre: Instituto de Informática – UFRGS, 1994.

AEDO, I.; DIAZ, P. Evaluation Criteria for Hypermedia Educational Systems. In: INTERNATIONAL SYMPOSIUM ON COMPUTERS IN EDUCATION, 2., 2000. **Proceedings...** Dordrecht: Kluwer Academic Publishers, 2000. p. 45-60.

ALMEIDA, A. J. N. **Ensino a distância Suportado pela Internet**, 1998. Disponível em: <<http://lsm.dei.uc.pt/Aulas/SF/PaginasAlunos/Alfredo/ENSINO.HTM>> . Acesso em: jul. 2002.

ALMEIDA, C. B. F. Comunicação Interativa em EAD. **Revista Tecnológica Educacional**, Rio de Janeiro, v.25, p. 39-41, nov./dez. 1997.

ALONSO, K. M. A Educação a Distância no Brasil: a Busca de Identidade. In: PRETI, Oreste (Org.). **Educação a distância: inícios e indícios de um percurso**. Cuiabá: NEAD/IE - UFMT, 1996.

ALVES, J. R. M. **A educação a distância no Brasil: síntese histórica e perspectivas**. Rio de Janeiro: Instituto de Pesquisas Avançadas em Educação, 1994.

ALVES, L.; NOVA, C. **Educação a Distância: Uma Nova Concepção de Aprendizado e Interatividade**. São Paulo: Futura, 2003.

ALVES, R. M.; ERRICO, L.; MESQUITA, R. C. Um Modelo Informacional para Avaliações de Alunos no Ensino a Distância via Web. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 13., 2002, São Leopoldo. **Anais...** São Leopoldo: UNISINOS: SBC, 2002. p. 464-473

AMBRIOLA, V.; CONRADI, R.; FUGGETTA, A. Assessing process-centered software engineering environments. **ACM Transactions on Software Engineering and Methodology**, New York, v.6, n.3, p. 283-328, 1997. Disponível em: <<http://doi.acm.org/10.1145/258077.258080>>. Acesso em: jan. 2003.

ANDRADE, A. A. M. Qualidade em Projetos de Educação à Distância. **Revista Tecnológica Educacional**, Rio de Janeiro, v.25, p. 32-34, nov./dez. 1997.

ANDRADE, A. F. de; VICCARI, R. M. Construindo um Ambiente de Aprendizagem a Distância Inspirado na Concepção Sociointeracionista de Vygotsky. In: SILVA, M. (Org.). **Educação Online: Teorias, Práticas, Legislação, Formação Corporativa**. São Paulo: Loyola, 2003. p. 255-272.

- ANDRADE, M.M. **Introdução à Metodologia Científica**. São Paulo: Atlas, 2003.
- ARAÚJO, L. H. L.; LUCENA FILHO, G. J. Comunidades Virtuais de Aprendizagem: Novas Dinâmicas de Aprendizagem exigem novas formas de avaliação. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 16., 2005, Juiz de Fora. **Anais...** Porto Alegre: SBC, 2005. p. 340-350.
- ARBAOUI, S.; DERNIAME, J.; OQUENDO, F.; VERJUS, H. A comparative review of Process-Centered Software Engineering Environments. **Annals of Software Engineering**, The Netherlands, v.14, p. 311-340, 2002.
- ARETIO, L. G. **La Educación a Distancia y la UNED**. Madrid: UNED, 1998.
- ARETIO, L. G. **La Educación a Distancia: De la teoría a la práctica**. Barcelona: Ariel Educación, 2001.
- ASSISTANT-GDPA. **G. D. P. A.** Disponível em: <<http://www.informatik.unibremen.de/uniform/gdpa/>>. Acesso em: dez. 2003.
- AZEVÊDO, W. **Muito além do Jardim de Infância: O desafio do preparo de alunos e professores on-line**. 1999. Disponível em: <<http://www.stprj.br/abed/99.html>>. Acesso em: jan.2003.
- BANDINELLI, S. et al.: An Environment for Software Process Analysis, Design and Enactment. In: FINKELSTEIN, A. et al. (Ed.). **Software Process Modelling and Technology**. Tauton: Research Studies Press, 1994. p.223-248.
- BARKER, K. **Quality guidelines for technology assisted distance education**. 1999. Disponível em: <<http://www.furured.com>>. Acesso em: jul.2002.
- BASILI, V. R.; CALDIEIRA, G.; ROMBACH, H. D. The Goal Question Metric Approach. In: **Encyclopedia of Software Engineering**. New York: J.Willey & Sons, 1994.
- BASILI, V. R.; CALDIEIRA, G.; ROMBACH, H. D. The Experience Factory. In: **Encyclopedia of Software Engineering**. New York: J.Willey & Sons, 1994.
- BELANGER, F.; JORDAN, D. H. **Evaluation and Implementation of Distance Learning: Technologies, tools and techniques**. London: Idea Group Publishing, 2000.
- BELLONI, M. L. **Educação a Distância**. 2. ed. São Paulo: Autores Associados, 2001.
- BENIGNO, V.; TRENTIN, G. The Evaluation of Online Courses. **Journal of Computer Assisted Learning**, [S.l.], v. 16, p. 259-270, Feb. 2000.
- BERCHT, M. **Avaliação Pedagógica como Fator para a Construção de Estratégias de Ensino em Ambientes de Ensino e Aprendizagem Computadorizados**. 1997. (Exame de Qualificação) – Instituto de Informática, UFRGS, Porto Alegre.
- BERGE, Z. L. (Ed.) **Sustaining Distance Training: integrating learning technologies into the fabric of the enterprise**. San Francisco: Jossey-Bass, 2001.
- BESSAGNET, M. N.; MARQUESUZAÀ, C.; NODENOT, T., SALLABERRY, C.; LAFORCADE, P. Information Systems and Educational Engineering. In: IFIP WORLD

COMPUTER CONGRESS, 17., 2002, Montréal. **Tele-Learning: the challenge for the third millennium.** Boston: Kluwer Academic, 2002. p. 269-276.

BOOCH, G. et al. **The Unified Modeling Language User Guide.** [S.l.]: Addison-Wesley, 1998.

BORONI, C. M.; GOOSEY, F. W. et al. Trying it All Together: Creating Self-Contained, Animated, Interactive, Web-Based Resources for Computer Science Education. **SIGCSE Bulletin**, New Orleans, v.31, n.1, p.7-11, Mar. 1999.

BRITO, M. C. A.; NÓBREGA, G. M.; OLIVEIRA, K. M. Capturando Experiência Docente para Guiar o Processo de Design Instrucional Colaborativo e Contínuo. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 16., 2005, Juiz de Fora. **Anais...** Porto Alegre: SBC, 2005. p. 147-157.

BROADBENT, B. **ABCs of E-learning: Reaping the Benefits and Avoiding the Pitfalls.** San Francisco: ASTD, 2002.

CAMOLESI, A. R.; NETO, J. J. Modelo AMBER-Adp de um Ambiente para Gerenciamento de Ensino a Distância. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 13., 2002, São Leopoldo. **Anais...** São Leopoldo: UNISINOS: SBC, 2002. p. 401-409.

CASTRO, A. P. **Uma Ferramenta para Auxiliar o Professor no Ensino a Distância.** 2002. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

CERVO, A. L.; BERVIAN, P. A. **Metodologia Científica.** 5. ed. São Paulo: Prentice Hall, 2002.

CHAVES, M. C. S. **O perfil do novo educador frente à informatização no processo de ensino aprendizagem.** Disponível em: <<http://sites.uol.com.br/cdchaves/perfileduca.htm>>. Acesso em: out. 2003.

CHEN, M. C. et al. On the Development and Implementation of a Sequencing Engine for IMS Learning Design Specification. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, 5., 2005, Kaohsiung. **Proceedings...** Los Alamitos: IEEE Computers Society, 2005. p.636-640.

CINELLI, G. B. et al. Acompanhamento de Sessões de Fórum Através da Ferramenta Faucon. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 16., 2005, Juiz de Fora. **Anais...** Porto Alegre: SBC, 2005. p. 801-810.

CONRADI, R. EPOS: Object-Oriented Cooperative Process Modelling. In: FINKELSTEIN, A. et al. (Ed.). **Software Process Modelling and Technology.** Somerset: John Wiley, 1994. p. 33-70.

CUNHA, F. B. R. P.; KIENBAUM, G. S.; OLIVEIRA, C. A. **O Aprendizado a Distância Através de um Ambiente Computacional Inteligente para Educação na WEB.** Taubaté: Universidade de Taubaté (UNITAU), 2000.

CUNHA, M. J. S.; FERNANDES, C. T. AC3As-Web: Ambiente Cooperativo de Apoio à Avaliação de Aprendizagem Significativa na Web. In: SIMPÓSIO BRASILEIRO DE

INFORMÁTICA NA EDUCAÇÃO, SBIE, 13., 2002, São Leopoldo. **Anais...** São Leopoldo: UNISINOS: SBC, 2002. p. 146-156.

DAHMER, A.; ASSIS, G. A.; LEÃO, B. SADE - Sistema de Apoio a Denúncia - O Raciocínio Baseado em Casos Aplicado a Área Jurídica. In: SIMPÓSIO BRASILEIRO DE INTELIGÊNCIA ARTIFICIAL, SBIA, 13., 1996, Curitiba. **Anais...** Curitiba: SBC, 1996. p. 23-28.

DAHMER, A.; FERNANDES, U.; ASSIS, G. A.; LEÃO, B. Um modelo Híbrido de Sistemas Especialistas para a Área Jurídica. In: CONGRESSO ARGENTINO DE CIÊNCIAS DE LA COMPUTACION, CACIC, 2., 1996, San Luis. **Proceedings...** San Luis: Universidad Nacional de San Luis, 1996. p. 349-359.

DAHMER, A.; FROZZA, R.; GASPARY, L. P. Uma Arquitetura para Acompanhamento Pedagógico em Ensino a Distância. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 11., 2000, Maceió. **Anais...** Maceió: SBC, 2000. p. 290-296.

DAHMER, A.; FROZZA, R.; GASPARY, L. P.; RIZZATO, E.; SANTOS, B. S. Ensino de Graduação em Computação a Distância: Relato de um Experimento. In: CONGRESO IBEROAMERICANO DE EDUCACIÓN SUPERIOR EN COMPUTACIÓN, CIESC, 8., 2000, Cidade do México. **Proceedings...** Cidade do México: [s.n.], 2000.

DAHMER, A.; GASPARY, L. P.; FROZZA, R.; SANTOS, B. S.; RIZZATO, E. Uma Experiência de Ensino de Redes de Computadores via Internet. In: CONGRESSO NACIONAL DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO - WORKSHOP DE EDUCAÇÃO EM INFORMÁTICA, WEI, 10., 2000, Curitiba. **Anais...** Curitiba: Champagnat, 2000.

DAHMER, A.; GASPARY, L. P.; FROZZA, R. Ambiente Integrado de Apoio ao Ensino a Distância: Gerenciamento de Aulas, Tutores Inteligentes e Avaliação Remota. In: TALLER INTERNACIONAL DE SOFTWARE EDUCATIVO, 1., 1999, Santiago. **Proceedings...** Santiago: [s.n.], 1999.

DAHMER, A.; GASPARY, L. P.; FROZZA, R. Um Ambiente para Desenvolvimento de Ensino a Distância. In: WORKSHOP INTERNACIONAL SOBRE EDUCAÇÃO VIRTUAL, WISE, 1999, Fortaleza. **Anais...** Fortaleza: [s.n.], 1999.

DAHMER, A.; GASPARY, L. P.; FROZZA, R.; KIST, T. Disponibilização de um Ambiente Integrado para Gerenciamento e Acompanhamento de Aulas a Distância. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 12., 2001, Vitória. **Anais...** Vitória: SBC, 2001. p. 60-67.

DAHMER, A.; GASPARY, L. P.; FROZZA, R.; SANTOS, B. S.; RIZZATO, E. Teaching Undergraduate Students through the Web: Report of an Experiment. In: IFIP WORLD CONFERENCE ON COMPUTERS IN EDUCATION, 7., 2001, Copenhagen. **Book of Abstracts...** Copenhagen: UNI-C, 2001. p.130.

DAHMER, A.; VICCARI, R. M.; NUNES, D. J. A Proposal of Criteria for Evaluation of Distance Education Courses. In: INTERNATIONAL CONFERENCE ON WEB-

- BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 481-485.
- DAHMER, A.; VICCARI, R. M.; NUNES, D. J. Critérios para Avaliação de Cursos em Educação a Distância. In: GLOBAL CONGRESS ON ENGINEERING AND TECHNOLOGY EDUCATION, GCETE, 2005, Santos. **Proceedings...** Santos: [s.n.] , 2005.
- DEMO, P. **Avaliação sob o Olhar Propedêutico**. Campinas: Papirus, 1996.
- DEMO, P. **Universidade, Aprendizagem e Avaliação**. Porto Alegre: Mediação, 2004.
- DEPRESBITERIS, L. Avaliação da Aprendizagem: revendo conceitos e posições. In: SOUZA, C. P. de (Org). **Avaliação do rendimento escolar**. Campinas: Papirus, 1995. p.67-70.
- DERNIAME, J. C.; KABA, B. A.; WASTELL, D. (Ed.). **Software Process: Principles, Methodology, and Technology**. Berlin: Springer-Verlag, 1999. p.165-200. (Lecture Notes in Computer Science, 1500).
- DOWSON, M.; FERNSTRÖM, C. Towards Requirements for Enactment Mechanisms. In: EUROPEAN WORKSHOP ON SOFTWARE PROCESS TECHNOLOGY, EWSPT, 3., 1994. **Proceedings...** Berlin: Springer-Verlag, 1994.
- DOWSON, M.; NEJMEH, B.; RIDDLE, W. Fundamental Software Process Concepts. In: EUROPEAN WORKSHOP ON SOFTWARE PROCESS MODELLING, 1., 1991, Milan, Italy. **Proceedings...** [S.l.]: AICA Press, 1991.
- ELIASQUEVICI, M. K.; FONSECA, N. A. **Educação a Distância: Orientações para o início de um percurso**. Belém: EDUFPA, 2004.
- ELLMER, E. et al. Process Model Reuse to Promote Organizational Learning in Software Development. ANNUAL INTERNATIONAL COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE, COMPSAC, 20., 1996. **Proceedings...** Los Alamitos: IEEE Computer Society Press, 1996. p.21-26.
- FACHIN, O. **Fundamentos de Metodologia Científica**. 3. ed. São Paulo: Saraiva, 2001.
- FEILER, P.; HUMPHREY, W. Software Process Development and Enactment: Concepts and Definitions. In: INTERNATIONAL CONFERENCE ON THE SOFTWARE PROCESS, ICSP, 2., 1993. **Proceedings...** Berlin: IEEE Computer Society Press, 1993.
- FERNANDES, A. A. **Gerência de Software através de métricas: Garantindo a qualidade do projeto, processo e produto**. São Paulo: Atlas, 1995.
- FERREIRA, L. F. **A evolução dos ambientes de aprendizagem construtivistas**. UFRGS. 1998. Disponível em: <<http://penta.ufrgs.br/~luis/Ativ1/AmbApC.html>>. Acesso em: nov. 2003.
- FLORAC, W. A.; CARLETON, A. D. **Measuring the Software Process: Statistical Process Control for Software Process Improvement**. [S.l.]: Addison Wesley, 1999.

- FRANCA, L. P. A.; STAA, A.; LUCENA, C. J. P. de. Medição de Software para Pequenas Empresas: Uma solução baseada na Web. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE, SBES, 12., 1998. **Anais...** Maringá: SBC, 1998.
- FREIMUT, B. et al. **State of art in empirical studies**. Berlin: Fraunhofer IESE, 2002. (Technical Report Visek/007/E).
- FROZZA, R. **Critérios para Avaliação de Coordenação Multiagente**. 2004. 136 f. Tese (Doutorado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- FUGGETTA, A. Software process: a roadmap. In: THE FUTURE OF SOFTWARE ENGINEERING, 2000, Limerick - Ireland. **Proceedings**. . . New York: ACM Press, 2000. p.25.34. Disponível em: <<http://doi.acm.org/10.1145/336512.336521>>. Acesso em: maio 2002.
- FUKS, H.; CUNHA, L. M.; GEROSA, M. A.; LUCENA, C. J. P. Participação e Avaliação no Ambiente Virtual AulaNet da PUC-Rio. In: SILVA, M. (Org.). **Educação Online: Teorias, Práticas, Legislação, Formação Corporativa**. São Paulo: Loyola, 2003. p. 231-254.
- GAGNÉ, R. **The Conditions of Learning**. 4th ed. New York: Holt Rinehart and Winston, 1999.
- GANNOD, G. C. et al. Using Simulation to Facilitate the Study of Software Product Line Evolution. In: INTERNATIONAL WORKSHOP OF THE PRINCIPLES OF SOFTWARE EVOLUTION, IWPSE, 7., 2004. **Proceedings...** Washington: IEEE Computer Society, 2004.
- GAO, S.; ZHANG, Z.; HAWRYSZKIEWYCZ, I. Supporting Adaptive Learning in Hypertext Environment: A High Level Timed Petri Net Based Approach. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, 5., 2005, Kaohsiung. **Proceedings...** Los Alamitos: IEEE Computers Society, 2005. p.735-739.
- GARG, P. K.; JAZAYERI, M. **Process-Centered Software Engineering Environments**. New York: IEEE Computer Society Press, 1996.
- GIMENES, I. M. S. **ExPSEE: Um Ambiente Experimental de Engenharia de Software Orientado a Processos**. Universidade Estadual de Maringá, Depto. de Informática, Maringá, 2000. Relatório de Projeto.
- GIMENES, I. M. S. **Uma Introdução ao Processo de Engenharia de Software: ambientes e formalismos**. Caxambu, MG: SBC, 1997. Trabalho apresentado na 13. Jornada de Atualização em Informática, 1994, Caxambu.
- GIRAFFA, L.; MARCZAK, S.; PRIKLADNICKI, R. PDS-E: Em direção a um processo para desenvolvimento de Software Educacional. In: CONGRESSO DA SBC - WORKSHOP SOBRE INFORMÁTICA NA ESCOLA, WIE, 25., 2005, São Leopoldo. **Anais...** São Leopoldo: SBC, 2005. p. 2833-2841.
- GODART, C.; CHAROY, F. **Databases for Software Engineering**. [S.l.]: Prentice Hall International. 1994.

GODOY, A. S. Introdução à Pesquisa Qualitativa e suas Possibilidades. **Revista de Administração de Empresas**, São Paulo, v. 35, n. 2, p. 57-63, mar./abr., 1995.

GOMES, E. R. Objetos inteligentes de aprendizagem: uma abordagem baseada em agentes para objetos de aprendizagem. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 15., 2004, Manaus. **Anais...** Manaus: SBC, 2004. p. 408-417.

GRANVILLE, L.Z.; SCHLEBBE, H. **Distributed PROSOFT: Management of Tools and Memory**. Porto Alegre: UFRGS e Uni-Stuttgart, 1996. Relatório Técnico.

GRINSPUN, M. P. S. Z. (Org.). **Educação Tecnológica: Desafios e Perspectivas**. São Paulo: Cortez, 1999.

GRUHN, V. Process-Centered Software Engineering Environments: A brief history and future Challenges. **Annals of Software Engineering**, The Netherlands, v. 14, p. 363-382, 2002.

HACK, J. R. **A avaliação em educação a distância (EAD) e cursos em modalidade alternativa**. Disponível em: <<http://www.unoescjba.rct-sc.br/~hack/textos/texto6.htm>>. Acesso em: jun. 2000.

HADJI, C. **Avaliação Desmitificada**. Porto Alegre: ARTMED, 2001.

HANNEY, N.; WATSON, P. Ensuring Quality in the design, development and implementation of computer based learning packages – a best practice framework. In: INTERNATIONAL CONFERENCE ON WEB-BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 475-480.

HASEGAWA, S.; OCHIMIZU, K. A Learning Management System based on the Life Cycle Management Model of e-Learning Courseware. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, 5., 2005, Kaohsiung. **Proceedings...** Los Alamitos: IEEE Computers Society, 2005. p.35-37.

HATZILYGEROUDIS, I. et al. Using expert systems technology for student evaluation in a web based educational system. In: INTERNATIONAL CONFERENCE ON WEB-BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 534-538.

HEIDE, A.; STILBORNE, L. **Guia do Professor para a Internet: completo e fácil**. Porto Alegre: ARTMED, 2000,

HEIMANN, P.; KRAPP, C. A.; WESTFECHTEL, B. An Environment for Managing Software Development Processes. In: CONFERENCE ON SOFTWARE ENGINEERING ENVIRONMENTS, 8., 1997. **Proceedings...** New York: IEEE Computer Society, 1997. p. 101-109. Disponível em: <<http://www-i3.informatik.rwthachen.de/research/dynamite/>>. Acesso em: maio 2002.

HOLMBERG, B. **Educación a distancia: situación y perspectivas**. Buenos Aires: Editorial Kapelusz, 1981.

HORTON, S. **Web Teaching Guide: a practical approach to creating course web sites**. New Haven: Yale University, 2000.

- HORTON, W. **Designing Web-based Training**. New York: Wiley, 2000.
- HORTON, W. **Evaluating E-Learning**. Alexandria: ASTD, 2001.
- HUFF, K. Software Process Modeling. In: FUGGETTA, A.; WOLF, A. (Eds.). **Software Process**. [S.l.]: John Wiley & Sons, 1996.
- HUMPHREY, W. S. **A Discipline for Software Engineering**. [S.l.]: Addison Wesley, 1995.
- HUMPHREY, W. S. **Introduction to the Personal Software Process**. [S.l.]: Addison Wesley, 1997.
- HUMPHREY, W. S. **Managing the software process**. [S.l.]: Addison-Wesley, 1989.
- JONES, C. **Applied Software Measurement: Assuring Productivity and Quality**. 2nd ed. [S.l.]: McGraw Hill, 1996.
- JONSON, E. Quality Issues in Web-based Education: is a six week on-line course quality professional development? In: INTERNATIONAL CONFERENCE ON WEB-BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 448-451.
- JUNG, C. F. **Metodologia para Pesquisa e Desenvolvimento: Aplicada a Novas Tecnologias, Produtos e Processos**. Rio de Janeiro: Axcel, 2004.
- JUNG, J. L. **Concepção e Implementação de um Agente Semiótico como Parte de um Modelo Social de Aprendizagem a Distância**. 2001. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- KAISER, G.; FELLER, P. H.; POPOVICH, S. S. Intelligent Assistance for Software Development and Maintenance. **IEEE Software**, Los Alamitos, v. 5, n. 3, May 1988.
- KARAMPIPERIS, P.; SAMPSON, D. Towards Next Generation Activity-based Educational Systems. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, 5., 2005, Kaohsiung. **Proceedings...** Los Alamitos: IEEE Computers Society, 2005. p.868-872.
- KASHIHARA, A.; HASEGAWA, S. A model of meta-learning for web-based navigational learning. In: INTERNATIONAL CONFERENCE ON WEB-BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 249-254.
- KRAMER, E. A. W. C. et al. **Educação a Distância: da teoria à prática**. Porto alegre: Alternativa, 1999.
- KUNDE, G.F.; SOUTO, M.A.; OLIVEIRA, J.P. Evolução Dinâmica de um Curso a Distância Modelado por Workflow. **Revista Brasileira de Informática na Educação**, Florianópolis, v. 9, p. 35-50, set. 2001.
- LAFORCADE, P. Towards a UML-based Educational Modeling Language. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING

- TECHNOLOGIES, 5., 2005, Kaohsiung. **Proceedings...** Los Alamitos: IEEE Computers Society, 2005. p.855-859.
- LAKATOS, E. M.; MARCONI, M. A. **Fundamentos de Metodologia Científica**. 4. ed. São Paulo: Atlas, 2001.
- LANDIM, C. M. M. P. F. **Educação a distância: algumas considerações**. Rio de Janeiro: [s.n.], 1997.
- LEITE, L. S.; SILVA, C. M. T. **A educação a distância capacitando professores: em busca de novos espaços para a aprendizagem**. 1998. Disponível em: <<http://www.intelecto.net/ead/ligia-cris.htm>>. Acesso em: maio 2002.
- LIMA REIS, C. A. A Abordagem APSEE para Modelagem e Gerência de Recursos em Ambientes de Processos de Software. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE, 15., 2001. **Anais...** Rio de Janeiro: IME/SBC, 2001.
- LIMA REIS, C. A. **Um Gerenciador de Processos de Software para o Ambiente PROSOFT**. 1998. 197 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.
- LIMA REIS, C. A. **Uma Abordagem Flexível para Execução de Processos de Software Evolutivos**. 2003. 267 f. Tese (Doutorado em Ciência da Computação). Instituto de Informática, UFRGS, Porto Alegre.
- LIMA REIS, C. A.; REIS, R.Q.; SCHLEBBE, H.; NUNES, D.J. A Policybased Resource Instantiation Mechanism to Automate Software Process Management. In: WORKSHOP ON SOFTWARE ENGINEERING DECISION SUPPORT, SEDECS; SOFTWARE ENGINEERING & KNOWLEDGE ENGINEERING, SEKE, 14., 2002, Ischia. **Proceedings...** Ischia: ACM Press, 2002.
- LIMA REIS, C. A. **Ambientes de Desenvolvimento de Software e seus Mecanismos de Execução de Processos de Software**. 2000. (Exame de Qualificação) - Instituto de Informática, UFRGS, Porto Alegre.
- LIMA REIS, C. A.; REIS, R. Q.; ABREU, M. M.; SCHLEBBE, H.; NUNES, D. J. Using Graph Transformation as the Semantical Model for Software Process Execution in the APSEE Environment. In: INTERNATIONAL CONFERENCE ON GRAPH TRANSFORMATION, 1., 2002, Barcelona. **Proceedings...** Barcelona: Springer Verlag, 2002.
- LIMA REIS, C. A.; REIS, R. Q.; SCHLEBBE, H.; NUNES, D. J. Resource Instantiation Policies in Software Process Environments. In: ANNUAL INTERNATIONAL COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE, COMPSAC, 26., 2002, Oxford. **Proceedings...** Los Alamitos: IEEE Computer Society Press, 2002.
- LOBO NETO, F. J. S. **Educação a Distância: Regulamentação, Condições de Êxito e Perspectivas**. 1998. Disponível em: <<http://www.intelecto.net/ead/lobo1.htm>>. Acesso em: maio 2004.
- LONCHAMP, J. A Structured Conceptual and Terminological Framework for Software Process Engineering. In: INTERNATIONAL CONFERENCE ON THE SOFTWARE

PROCESS: CONTINUOUS SOFTWARE PROCESS IMPROVEMENT, ICSP, 2., 1993, Berlin, Germany. **Proceedings...** New York: IEEE Computer Society, 1993. p. 41-53.

MACHADO, C. C. **XHA: eXtensible Hyper-Automaton**. 2002. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

MACHADO, J. H. A. P. **Hyper-Automaton: hipertextos e cursos na web usando autômatos finitos com saída**. 2000. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

MACHADO, L. S. **Mineração do Uso da Web na Educação a Distância: propostas para a condução de um processo a partir de um estudo de caso**. 2003. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

MACIEL, C.; COSTA, R.; SILVA, R.L. Critérios de Avaliação para Portfólios Eletrônicos. In: INTERNATIONAL CONFERENCE ON ENGINEERING AND TECHNOLOGY EDUCATION, 7., 2002, Santos. **Proceedings...** Santos: SENAC, 2002.

MAIA, C. **Guia Brasileiro de Educação a Distância**. São Paulo: Esfera, 2001.

MASSI, W.F. Applying business quality principles to academic audit: Hong Kong's second-round TLQPR. In: BIENNIAL CONFERENCE OF THE INTERNATIONAL NETWORK FOR QUALITY ASSURANCE AGENCIES IN HIGHER EDUCATION, 6., Bangalore. **Proceedings...** Bangalore: [s.n.], 2001.

MATTELART, A. **Comunicação no Mundo: História das Idéias e das Estratégias**. Petrópolis: Vozes, 1994.

MCCORMACK, C.; JONES, D. **Building a Web Based Education System**. [S.l.]: John Wiley, 1998.

MCISAAC, M.; RALSTON, K. **Third generation distance learning**. Educational Media and Computer Program at Arizona State University. Disponível em: <<http://seamonkey.ed.asu.edu/~mcsiaac/disted/week1/2focuslt.html>>. Acesso em: ago. 1997.

MEC/SEED. **Referenciais de Qualidade de EaD para Cursos de Graduação a Distância**. 2002. Disponível em: <<http://portal.mec.gov.br/seed/>>. Acesso em: abr. 2004.

MIRANDA, R. M. **GROA: Um Gerenciador de Repositórios de Objetos de Aprendizagem**. 2004. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

MOORE, M.; KEARSLEY, G. **Distance Education: A Systems View**. Belmont: Wadsworth Publishing Company, 1996.

MORAN, J. M.; MASETTO, M. T.; BEHRENS, M. A. **Novas Tecnologias e Mediação Pedagógica**. Campinas: Papirus, 2000.

MOURA, S. L. Indicadores de Qualidade de Cursos Virtuais. In: CONGRESSO DA SBC - WORKSHOP SOBRE INFORMÁTICA NA ESCOLA, WIE, 8., 2005, Florianópolis. **Anais...** Florianópolis: SBC, 2002. p. 511-518.

NGUYEN, M. N.; WANG, A. I.; CONRADI, R. Total Software Process Model Evolution in EPOS. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, 19., 1997. **Proceedings...** New York: ACM Press, 1997. p. 390-399. Disponível em: <http://www.idi.ntnu.no/_epos/Papers/icse97.ps>. Acesso em: ago. 2002.

NISKIER, A. **Tecnologia educacional: uma visão política**. Petrópolis: Vozes, 1993.

NUNES, D. J. Estratégia Data-driven no Desenvolvimento de Software. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE, SBES, 6., 1992, Gramado. **Anais...** Porto Alegre: Instituto de Informática da UFRGS, 1992. v.6, p.81.95.

NUNES, D. J. **PROSOFT, Um ambiente de Desenvolvimento de Software baseado no Método algébrico**. Porto Alegre: PPGC/UFRGS, 1994. Relatório Técnico.

NUNES, I. B. **Noções de educação a distância**. 1992. Disponível em: <http://www.intelecto.net/ead_textos/ivonio1.html>. Acesso em: mar. 2002.

OSTERWEIL, L. Software Processes are Software Too. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING, ICSE, 1987, Monterey. **Proceedings...** Monterey: [s.n.], 1987. p. 2-13.

OTSUKA, J.L.; ROCHA, H.V. Avaliação Formativa em Ambientes de EaD. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 13., 2002, São Leopoldo. **Anais...** São Leopoldo: UNISINOS: SBC, 2002. p. 146-156.

PALLOFF, R. M.; PRATT, K. **Construindo Comunidades de Aprendizagem no Ciberespaço**. Porto Alegre: ARTMED, 2002.

PALLOFF, R. M.; PRATT, K. **O Aluno Virtual: Um Guia para Trabalhar com Estudantes On-line**. Porto Alegre: ARTMED, 2004.

PARRA FILHO, D.; SANTOS, J.A. **Metodologia Científica**. São Paulo: Futura, 2001.

PETERS, O. **A Educação a Distância em Transição**. São Leopoldo: Ed. UNISINOS, 2003.

PETERS, O. **Didática do Ensino a Distância**. São Leopoldo: Ed. UNISINOS, 2003.

PHIPPS, R.; MERISOTIS, J. **Quality on the line: benchmarks for success in Internet-based distance education**. Washington: IHEP, DC. Disponível em: <www.ihep.com/quality.pdf>. Acesso em: out. 2000.

PIMENTEL, N. O ensino a distância na formação de professores. **Revista Perspectiva**, Florianópolis, n. 24, 1995.

POHL, K. et al. - toward process-integrated modeling environments. **ACM Transactions on Software Engineering and Methodology (TOSEM)**, [S.l.], v.8, n.4,

p.343.410, 1999. Disponível em: <<http://doi.acm.org/10.1145/322993.322995>>. Acesso em: maio 2004.

PORTER, L. R. **Creating the Virtual Classroom**. New York: Wiley, 1997.

PRESSMAN, R. S. **Software Engineering: A Practitioner's Approach**. 5th ed. London: McGraw-Hill, 2001.

PRETI, O. **Educação a Distância: Inícios e Indícios de um Percorso**. Cuiabá: NEAD/IE - UFMT, 1996.

RANGEL, G. S. **ProTool: Uma Ferramenta de Prototipação de Software para o Ambiente PROSOFT**. 2003. 223 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

RAPCHAN, F. J. C.; CURY, D.; MENEZES, C.; FALBO, R. A. EduQNet: Um Modelo de Qualidade de Processo para Cursos a Distância Mediados pela Internet. In: SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE, SBQS, 1., 2002, Porto Alegre. **Anais...** Porto Alegre: II da UFRGS, 2002.

REA, L.M.; PARKER, R.A. **Metodologia de Pesquisa: do Planejamento à Execução**. São Paulo: Pioneira Thomson Learning, 2002.

REIS, R. Q. **APSEE-REUSE- Um Meta-Modelo para Apoiar a Reutilização de processos de Software**. 2002. 213 f. Tese (Doutorado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

REIS, R. Q.; LIMA REIS, C. A. NUNES, D. J. Automated Support for Software Process Reuse: Requirements and Early Experiences with the APSEE model. In: INTERNATIONAL WORKSHOP ON GROUPWARE, 7., 2001, Darmstadt. **Proceedings...** Los Alamitos: IEEE Computer Society Press, 2001.

REIS, R.; LIMA REIS, C.; NUNES, D. Gerenciamento do Processo de Desenvolvimento Cooperativo de Software no Ambiente PROSOFT. In: SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SOFTWARE, 12., 1998, Maringá. **Anais...** [S.l.]: SBC, 1998. p. 221-236.

REIS, R. Q. **Uma Proposta de Suporte ao Desenvolvimento Cooperativo de Software no Ambiente PROSOFT**. 2002. 177 f. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

REIS, R. Q.; LIMA REIS, C. A.; YAMIN, A.; AUGUSTIN, I.; NUNES, D. J.; GEYER, C. Towards a Software Process Model to Support the Design of Mobile Computing Applications. In: WORLD CONFERENCE ON INTEGRATED DESIGN & PROCESS TECHNOLOGY, IDPT, 6., 2002, Pasadena. **Proceedings...** Pasadena: Society for Design and Process Science, 2002.

REIS, R. Q.; NUNES, D. J. **Uma Avaliação dos Paradigmas de Linguagens de Processo de Software**. 1999. 49 f. Trabalho Individual (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

RIBÓ, J. M.; FRANCH, X. **Supporting Process Reuse in PROMENADE**. Technical Report, LSI - Universitat Politècnica de Catalunya, Espanha. Disponível em <<http://griho.udl.es/josepma/tesi/report.pdf>>. Acesso em: out. 2003.

ROCHA, A. R. C.; MALDONADO, J. C.; WEBER, K. C. **Qualidade de Software: Teoria e Prática**. São Paulo: Prentice Hall, 2001.

ROCHA, A. R.; TRAVASSOS, J. C. **Estação TABA: Ambiente de Desenvolvimento de Software**. 2005. Disponível em: <<http://ramses.cos.ufrj.br/tab/>>. Acesso em: set. 2005.

RODRIGUES, R. S. **Modelo de Avaliação para Cursos no Ensino a Distância: estrutura, aplicação e avaliação**. 1998. Dissertação (Mestrado em Engenharia de Produção) – Departamento de Engenharia de Produção, UFSC, Florianópolis.

RODRIGUEZ, A. M. et al. A Transformação do Ensino Através do Uso da Tecnologia na Educação. In: CONGRESSO NACIONAL DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO, 19., 1999, Rio de Janeiro. **Anais...** Rio de Janeiro: SBC, 1999.

ROSENBERG, M. J. **E-Learning**. São Paulo: Pearson Education do Brasil, 2002.

RUHE, G. Experience Factory-Based Professional Education and Training. In: CONFERENCE ON SOFTWARE ENGINEERING EDUCATION AND TRAINING, 12., 1999, New Orleans. **Proceedings...** New York IEEE Press, 1999. p. 62-72.

SANTORO, F. M.; BORGES, M. R. S.; SANTOS, N. Um Modelo de Cooperação para Aprendizagem Baseada em Projetos com Foco no Processo Cooperativo e Workflow. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 13., 2002, São Leopoldo. **Anais...** São Leopoldo: UNISINOS: SBC, 2002. p. 358-367.

SANTOS, A. R. **Metodologia Científica: a construção do conhecimento**. 3. ed. Rio de Janeiro: DP&A Ed., 2000.

SANTOS, C. T. ; FROZZA, R.; DAHMER, A.; GASPARY, L. P. Doris - Pedagogical Agent in Intelligent Tutoring Systems. In: INTERNATIONAL CONFERENCE ON INTELLIGENT TUTORING SYSTEMS, 6., 2002, Biarritz. **Proceedings...** Biarritz: Springer, 2002.

SANTOS, C. T.; FROZZA, R.; DAHMER, A.; GASPARY, L. P. DÓRIS - Um Agente de Acompanhamento Pedagógico em Sistemas Tutores Inteligentes. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 12., 2001, Vitória. **Anais...** Vitória: SBC, 2001. p. 437-444.

SANTOS, D. S. dos. **Ferramenta para Visualização Gráfica da Interação de Alunos em um Curso de Ensino a Distância na Web**. 2002. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Departamento de Informática, UNISC, Santa Cruz do Sul.

SANTOS, N.; MELLO, W.; SEGRE, L. **Sistemas de Autoria para Cursos à Distância Apoiados em Tecnologias da Internet: Diretrizes para Seleção**. Universidade Federal do Rio de Janeiro. Disponível em: <http://www.cos.ufrj.br/~neide/artigos/analise_sistemas.htm>. Acesso em: abr. 2003.

SCHLEBBE, H. **Distributed PROSOFT** : report on a working stay at the institute of computer science of the state university of Rio Grande do Sul (UFRGS) at Porto Alegre, Brazil from May 1 to June 15, 1994. Stuttgart: Universität Stuttgart - Fakultät Informatik, 1994.

SCHLEBBE, H. **Java-PROSOFT Manual**. Fakultät Informatik, Universität Stuttgart. 2002. Disponível em: <http://www.informatik.uni-stuttgart.de/ifi/bs/schlebbe/prosoft_doc/guide>. Acesso em: out. 2003.

SCHLEBBE, H.; SCHIMPF, S. **Reengineering of PROSOFT in Java**. Stuttgart: Universität Stuttgart - Fakultät Informatik, 1997. Technical Report.

SCHLEBBE, H. **PRO-EAD**: Report on a working stay at the institute of computer science of the state university of Rio Grande do Sul at Porto Alegre, from March 10 to April 10, 2005. Stuttgart: Universität Stuttgart - Fakultät Informatik, 2005.

SHENG, Z. Y.; YUN, H. Y. Architecture-based SW Process Model. **SIGSOFT Software Engineering Notes**, New York, v. 28, n. 2, 2003.

SILVA, A. S.; BRITO, S. R.; FAVERO, E. L.; DOMÍNGUEZ, A. H.; TAVARES, O. L.; FRANCÊS, C. R. L. Uma Arquitetura para Desenvolvimento de Ambientes Interativos de Aprendizagem Baseado em Agentes, Componentes e Framework. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 14., 2003, Rio de Janeiro. **Anais...** Rio de Janeiro: UFRJ, 2003.

SILVA, C. S. **Medidas e avaliação em educação**. Petrópolis: Vozes, 1992.

SILVA, F. **Um Modelo de Simulação de Processos de Software Baseado em Conhecimento para o Ambiente PROSOFT**. 2001. 124 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

SILVA, J. C. T.; FEIJÓ, B. Uma Máquina de Estados Finitos para Avaliação de Desempenho em um Grupo de Discussão On-line. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, SBIE, 13., 2002, São Leopoldo. **Anais...** São Leopoldo: UNISINOS: SBC, 2002. p. 420-427.

SILVA, M. (Org.). **Educação Online**: Teorias, Práticas, Legislação, Formação Corporativa. São Paulo: Loyola, 2003.

SILVA, M. L. (Org.) **Novas Tecnologias**: Educação e Sociedade na Era da Informação. Belo Horizonte: Autêntica, 2001.

SILVA, R. V.; SILVA, A. V. (Org.). **Educação, Aprendizagem e Tecnologia**: um paradigma para professores do século XXI. Lisboa: Edições Silabo, 2005.

SIZILIO, G. R. M. A. **Técnicas de Modelagem de Workflow Aplicadas à Autoria e Execução de Cursos à Distância**. 2000. 115 f. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

SIZILIO, G. R.; EDELWEISS, N. Modelo de Autoria de Cursos de Ensino a Distância. **Revista Brasileira de Informática na Educação**, Florianópolis, v. 8. p. 35-51, abr. 2001.

SOONG, M. H. et al. Critical Success Factors for On-line Course Resources. **Computers & Education**, Amsterdam, v. 36, 2001.

SOUSA, A. L. R. **APSEE-Monitor: Um Mecanismo de Apoio à Visualização de Modelos de Processos de Software**. 2003. 112 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

SOUSA, A. L. R.; LIMA REIS, C. A.; REIS, R. Q.; NUNES, D. J.; PIMENTA, M. S. Analizando a Interação de Gerentes e Desenvolvedores em ambientes de processos de software. In: WORKSHOP CHILENO DE ENGENHARIA DE SOFTWARE, 1., 2001, Punta Arenas. **Proceedings...** Punta Arenas: [s.n.], 2001.

SOUSA, A. L. R.; LIMA REIS, C. A.; REIS, R. Q.; NUNES, D. J. **Interação Humana Durante Execução de Processos de Software: Classificação e Exemplos**. Porto Alegre: PPGC-UFRGS, 2001. (RP-310).

STRUCHINER, M. et al. Elementos Fundamentais para o Desenvolvimento de Ambientes Construtivistas de Aprendizagem a Distância. **Tecnologia Educacional**, Rio de Janeiro, v.26, n.142, jul./ago./set. 1998.

SUN MICROSYSTEMS. **Java™ Remote Method Invocation (RMI)**. Disponível em: <<http://java.sun.com/products/jdk/rmi/>> Acesso em: mar. 2003.

TAJRA, S. F. **Comunidades Virtuais: um Fenômeno na Sociedade do Conhecimento**. São Paulo: Érica, 2002.

TAVARES, O.L. et al. Ambiente de Apoio à Mediação da Aprendizagem: Uma abordagem Orientada por Processos e Projetos. **Revista Brasileira de Informática na Educação**, Florianópolis, v. 9, p. 77-87, set. 2001.

TAYLOR, R. N.; BOLCER, G. A. Endeavors: A Process System Integration Infrastructure. In: INTERNATIONAL CONFERENCE ON SOFTWARE PROCESS, ICSP, 4., 1996, Brighton. **Proceedings...** [S.l.]: IEEE Press, 1996. Disponível em: <<http://www.ics.uci.edu/pub/endeavors/>>. Acesso em: mar.2004.

TESSAROLLO, M. R. M. **AutorWeb - Ambiente de Autoria de Cursos a Distância**. 2000. 97 f. Dissertação (Mestrado em Ciência da Computação) – Instituto de Computação, Unicamp, Campinas. Disponível em: <http://teleduc.nied.unicamp.br/teleduc/publicacoes/marcia_disser.pdf> Acesso em: jun. 2005.

TORRES, J.; DODERO, J. M. A Characterization of Composition and Execution Languages for Complex Learning Processes. In: INTERNATIONAL CONFERENCE ON WEB-BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 255-260.

TORRES, J.; DODERO, J. M.; ZARRAONANDIA, T. An Architectural Framework for Composition and Execution of Complex Learning Processes. In: IEEE INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, 5., 2005, Kaohsiung. **Proceedings...** Los Alamitos: IEEE Computers Society, 2005. p.143-147.

TRAVASSOS, G. H. **O Modelo de Integração de Ferramentas da Estação TABA**. 1994. Tese (Doutorado em Engenharia de Sistemas e Computação) – COPPE/UFRJ, Rio de Janeiro.

TSUKAHARA, W.; SEKI, K.; OKAMOTO, T. Development of an integrated E-learning system and implications from its practice. In: INTERNATIONAL CONFERENCE ON WEB-BASED EDUCATION, WBE, 4., 2005, Grindelwald. **Proceedings...** Calgary: ACTA Press, 2005. p. 125-130.

TULLY, C. Representing and Enacting the Software Process. **ACM SigSoft Software Engineering Notes**, New York, v. 14, n. 4, June 1989.

TURRA, C.M.G. et al. **Planejamento de ensino e avaliação**. Porto Alegre: Emma, 1982.

TYLER, L. E. **Testes e medidas**. Rio de Janeiro: Zahar, 1981.

VALIATI, E. R. A. **Elaboração e Avaliação de um Guia de Recomendações para Auxílio no Desenvolvimento de Interfaces com Usabilidade em Softwares Educacionais do Tipo Hipertexto/Hipermídia Informativo**. 2000. Dissertação (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

WANG, Y.; KING, G. **Software Engineering Processes: Principles and Applications**. [S.l.]: CRC Press, 2000.

WESSON, J. Usability evaluation of web-based learning: an essential ingredient for success. In: IFIP WORLD COMPUTER CONGRESS, 17., 2002, Montréal. **TeIE-Learning: the challenge for the third millennium**. Boston: Kluwer Academic, 2002. p. 357-363.

WILLIS, B. **Distance education at a Glance**. Engineering Outreach at the University of Idaho, 1996. Disponível em: <<http://www.uidaho.edu/evo/distglan.html>>. Acesso em: mar. 2005.

WOHLIN, C.; HÖST, M.; HENNINGSSON, K. **Empirical Research Methods in Software Engineering**. Berlin: Springer-Verlag, 2003, p. 7–23. (Lecture Notes in Computer Science, v. 2765).

YIN, R. K. Case Study Research Design and Methods. **Applied Social Research Methods Series**, [S.l.], v.5, 1994.

ZAMBALDE, A. L.; ALVES, R. M.; SANTOS, A. B. **Ensino a Distância: aspectos teóricos e práticos de implantação e uso de um ambiente virtual de aprendizado**. Lavras: UFLA, 2001.

ANEXO A DIAGRAMAS DE PACOTES UML

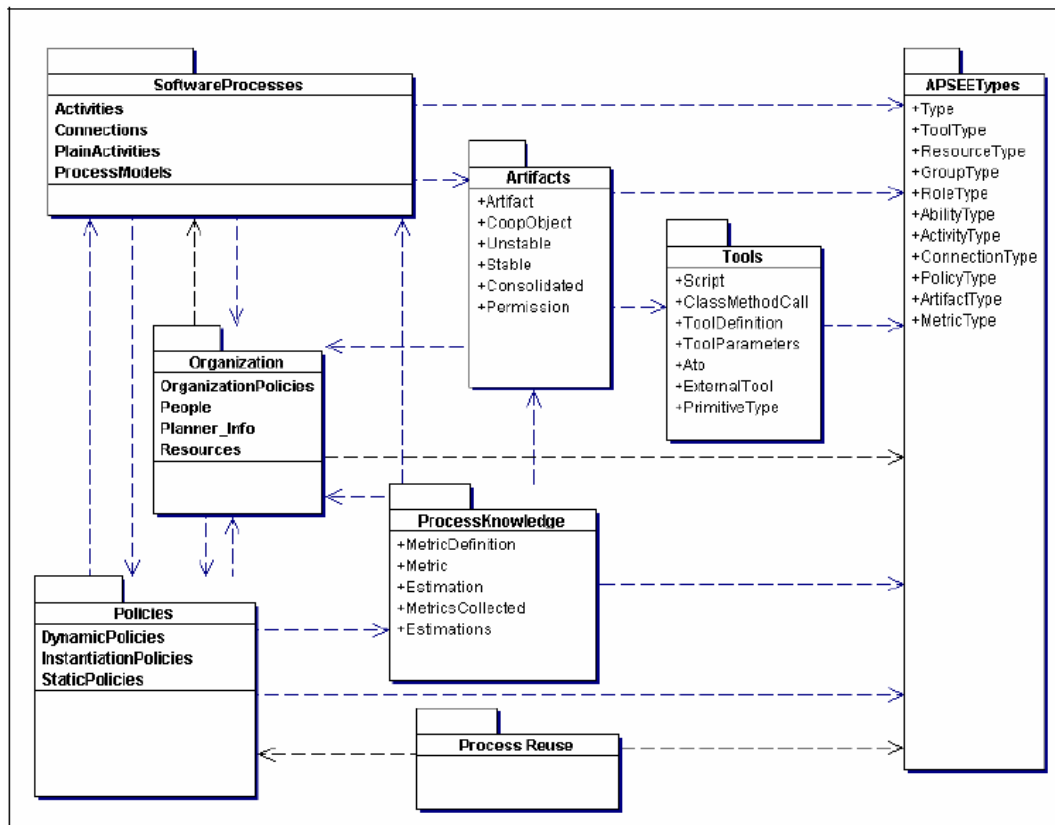


Figura A.1: Meta-modelo *APSEE* representado por um diagrama de pacotes UML

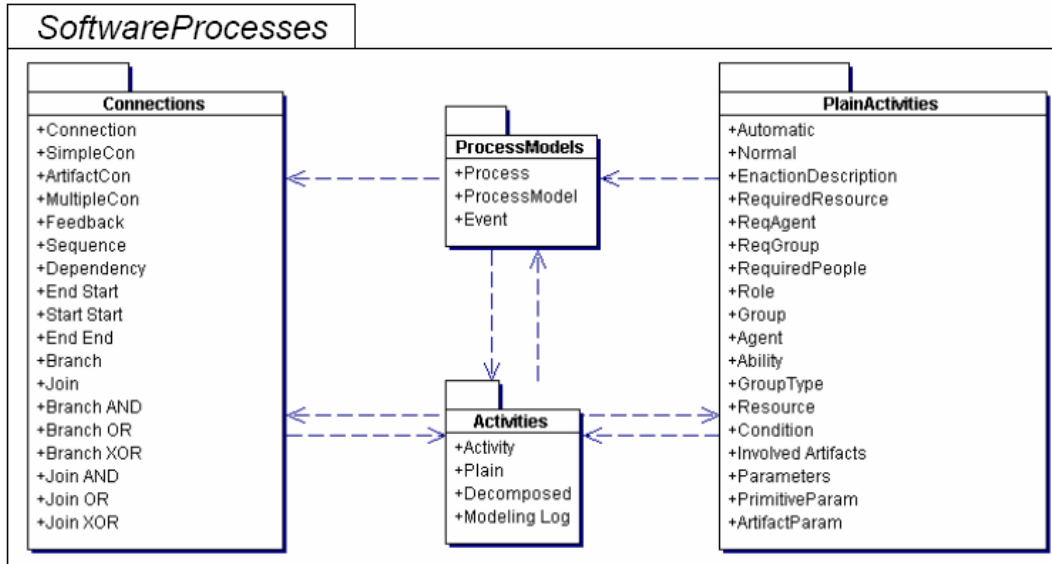


Figura A.2: Classe *SoftwareProcesses* representada por um diagrama de pacotes UML

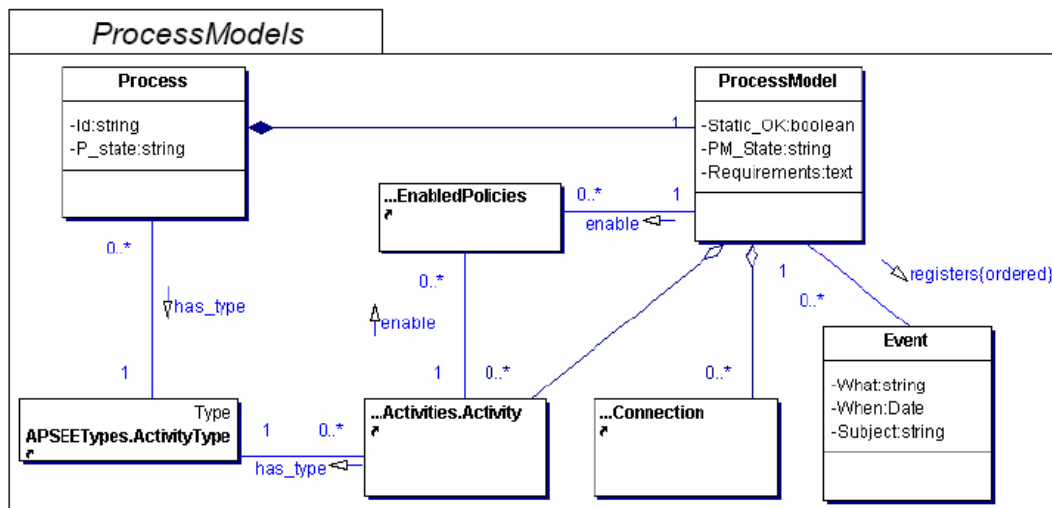


Figura A.3: Classe *ProcessModels* representada por um diagrama de pacotes UML

ANEXO B INSTRUMENTO DE AVALIAÇÃO UTILIZADO NO ESTUDO DE CASO

Dados de Identificação:

I. Sexo: Feminino () Masculino ()

II. Idade:

III. Área de atuação:

IV. Curso:

Questionário:

1. Você acredita que a analogia entre processo de curso e processo de software agrega vantagens à gerência de cursos a distância?

() sim

() não

Justifique:

2. Ainda sobre a analogia proposta, você identifica algum tipo de obstáculo que interfira na gerência de cursos?

() sim

() não

Justifique:

3. Você considera importante a possibilidade de reutilização de cursos a distância?

() sim

() não

() não tenho como prever

Justifique:

4. Como você classificaria o PRO-EAD com relação à modelagem e gerência de cursos a distância?

- adequado
- parcialmente adequado
- inadequado

Justifique:

5. O PRO-EAD oferece recursos para gerência de cursos não encontrados em outros ambientes de apoio à EAD não orientados a processo?

- sim
- não

Em caso afirmativo citar:

6. Classifique a linguagem de modelagem de cursos oferecida pelo ambiente PRO-EAD:

- adequada
- parcialmente adequada
- inadequada

Justifique:

7. Classifique o mecanismo de instanciação de cursos do ambiente PRO-EAD:

- adequado
- parcialmente adequado
- inadequado

Justifique:

8. Classifique o mecanismo de visualização da execução de cursos disponibilizado pelo ambiente PRO-EAD:

- adequado
- parcialmente adequado
- inadequado

Justifique:

9. Você considera que o ambiente PRO-EAD é de fácil utilização?

- sim
- parcialmente
- não

Justifique:

10. Quais foram as limitações que você encontrou na utilização do PRO-EAD, em relação à modelagem dos cursos?

11. Quais foram as limitações que você encontrou na utilização do PRO-EAD, em relação à execução dos cursos?

12. Quais sugestões você gostaria de fazer para melhoria do ambiente PRO-EAD?

13. O ambiente PRO-EAD correspondeu as suas expectativas?

- sim, totalmente
- parcialmente
- não

Justifique:

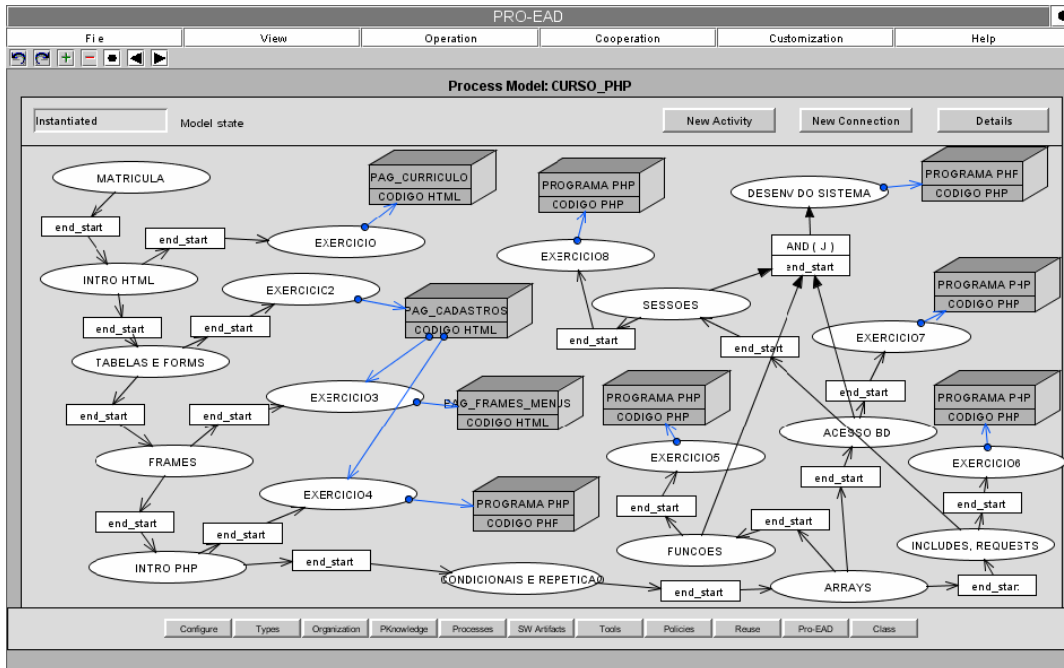


Figura C.2: Curso modelado pelo docente B

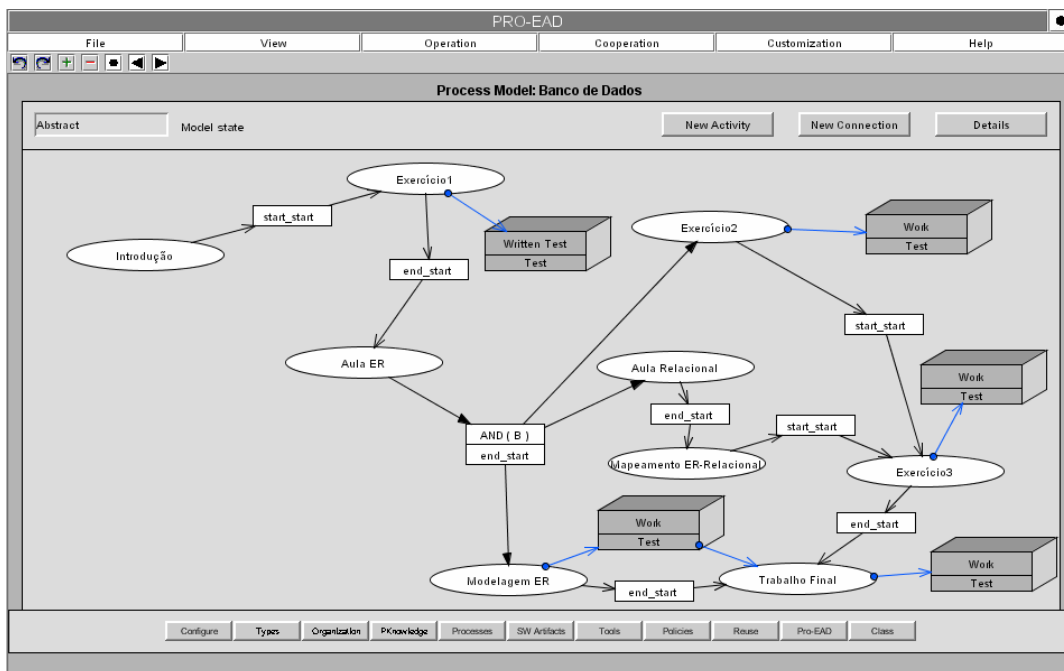


Figura C.3: Curso modelado pelo docente C

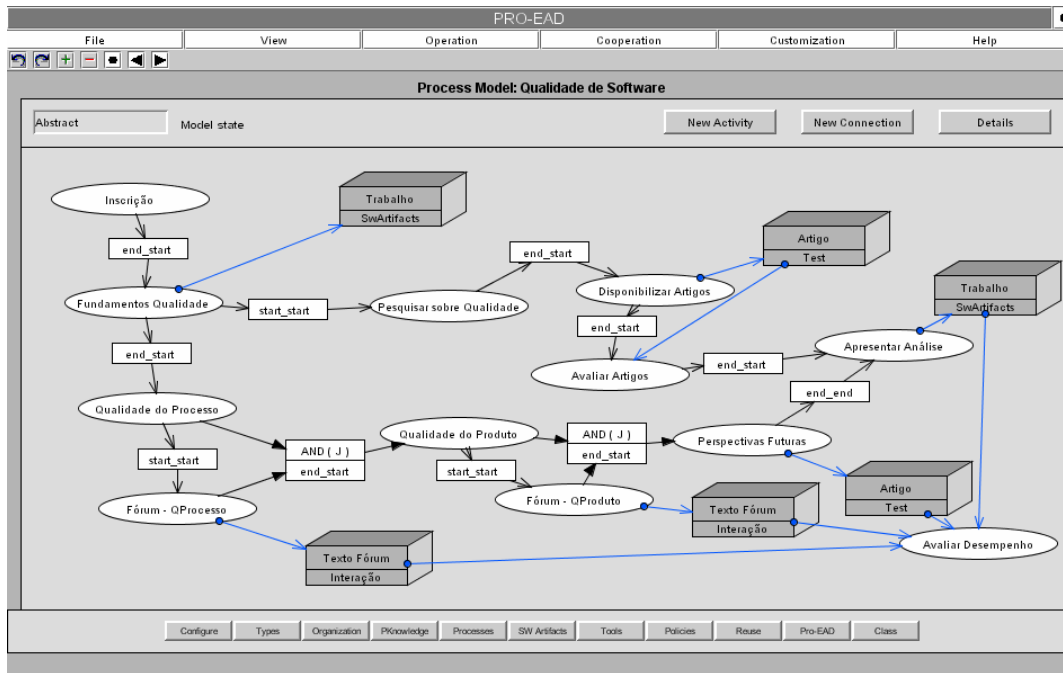


Figura C.4: Curso modelado pelo docente D

APÊNDICE A AVALIAÇÃO DE CURSOS A DISTÂNCIA

Toda avaliação é questionável (DEMO, 2004). A avaliação é um processo subjetivo e não existe uma única proposta a ser seguida. Portanto, é considerada como um desafio, pela dificuldade em saber o que é importante e válido avaliar, em como definir critérios para realizar a avaliação e em como mostrar os resultados (DAHMER, 2005). A proposta de avaliação apresentada neste trabalho provavelmente apresenta limitações.

No modelo proposto neste trabalho a avaliação tem um papel fundamental. Este apêndice discute a problemática da avaliação e apresenta uma abordagem para definição de critérios para avaliação de cursos a distância que pode servir como ponto de partida para um estudo mais completo.

Avaliação

Conforme Silva (1992), para realizar uma avaliação, primeiro devem ser traçados alguns objetivos a serem seguidos e depois escolhidos os métodos e os critérios.

Segundo Demo (1996), a avaliação precisa ser sistemática, ou seja, fazer parte diária do processo reconstrutivo do curso.

Turra (1982) reforça a necessidade de planejamento da avaliação. Ele afirma que para que a avaliação adquira a importância que realmente tem no processo ensino-aprendizagem, é necessário seguir alguns princípios básicos:

- Estabelecer com clareza o que vai ser avaliado. Se não se sabe o que avaliar não se pode avaliar de maneira eficiente;
- Selecionar técnicas adequadas para avaliar o que se pretende avaliar. Nem todas as técnicas e instrumentos são adequados aos mesmos fins;
- Utilizar, na avaliação, uma variedade de técnicas;
- Ter consciência das possibilidades e limitações das técnicas de avaliação.

De uma forma mais ampla, a avaliação é um conjunto de atividades, observações e coleta de informação que, se bem realizada, permite (DAHMER, 2005):

- Saber quanto foi alcançado e quanto falta para atingir os objetivos e metas, visando resolver um determinado problema ou responder a uma necessidade.
- Identificar qual o alcance (resultados positivos) e quais as limitações e obstáculos que impediram um maior avanço
- Decidir que modificações são necessárias para otimizar os resultados positivos.

As avaliações são realizadas com os seguintes objetivos (FROZZA, 2004):

- fornecer informações para auxiliar a tomada de decisão para a solução de um problema em particular. Por exemplo, um avaliador pode comparar operações e resultados de um programa em relação a um conjunto de padrões (critérios). Tomadores de decisão podem usar estas comparações para julgar o valor de um programa;
- ajudar a melhorar sistemas, por meio da utilização de técnicas mais adequadas em função das características do problema sendo solucionado;
- aprender sobre as razões de sucesso e de falhas;
- verificar se o objeto de avaliação atende de forma confiável, acessível, segura e no tempo as necessidades da situação em questão.

O objetivo previsto não é apenas alcançar um resultado positivo na solução do problema em questão, mas também saber se as ações propostas para atingir esta solução foram as mais corretas e eficazes.

Tipos de Avaliação

São citados, geralmente, dois tipos de avaliação (GODOY, 1995):

- avaliação qualitativa: é o processo que responde à questão "O quão bem nós fazemos?". O pesquisador não procura enumerar e/ou medir os eventos estudados, nem emprega instrumental estatístico para a análise dos dados, uma vez que é uma avaliação de cunho exploratório, descritivo e subjetivo. Baseia-se em critérios qualitativos, que identificam características adequadas a um problema específico;
- avaliação quantitativa: é o processo que responde à questão "O quanto nós fazemos?". O pesquisador preocupa-se com a medição objetiva e a quantificação dos resultados, uma vez que é uma avaliação que se baseia em critérios quantitativos, que mostram resultados numéricos para comparação.

Tanto na abordagem qualitativa quanto quantitativa, a pesquisa se caracteriza como um esforço cuidadoso para a descoberta de novas informações ou relações e para a verificação e ampliação do conhecimento existente (FROZZA, 2004).

A avaliação quantitativa é mais pragmática e é mais simples de ser realizada, e pode determinar alguns dados importantes sobre o curso. Por exemplo: Quantas interações foram realizadas no curso? Quantos módulos de conteúdo foram acessados? Quantos alunos participaram do curso?

A avaliação qualitativa é com certeza a mais difícil, mas produz resultados mais relevantes e completos. Os pesquisadores que seguem a abordagem qualitativa estão preocupados com o processo e não simplesmente com os resultados ou produto. O interesse está em verificar como determinado fenômeno se manifesta nas atividades, procedimentos e interações diárias. Como exemplo de pesquisas qualitativas, pode-se citar a análise da conversação entre sujeitos em processos de interação (GODOY, 1995).

Definição dos Critérios

Toda avaliação, seja quantitativa ou qualitativa, é elaborada a partir de critérios. Critérios são parâmetros, padrões, regras que servem como base para a avaliação (MACIEL; COSTA; SILVA, 2002). Por isso, para realizar uma avaliação eficiente, é necessária a definição de um conjunto de critérios adequados. Critério de avaliação é um princípio que se toma como referência para julgar alguma coisa (DEPRESBITERIS, 1995).

Este estudo propõe um conjunto de critérios para avaliação de cursos a distância. Para definição desses critérios foi utilizada a abordagem GQM, descrita na próxima seção.

Abordagem GQM (Goal Question Metric)

A abordagem GQM é um mecanismo utilizado na definição de metas mensuráveis, descrito na literatura de engenharia de software, no contexto de processo de software que requer critérios de medida para feedback e avaliação, assim como o processo de curso a distância apresentado neste trabalho.

O GQM, proposto por Basili (1994) baseia-se na definição de metas, questões e métricas e é usado para identificar e refinar metas, baseadas nas características do objeto a ser medido (produto, processo, recurso, curso), a fim de melhorar a sua qualidade.

Na fase de interpretação, as medidas são usadas para responder as questões e para concluir se a meta é ou não atingida. Assim, GQM usa uma abordagem *top-down* para definir as métricas e uma abordagem *bottom-up* para analisar e interpretar os dados das medidas.

A abordagem GQM possui três níveis (BASILI, 1994):

- nível conceitual (meta);
- nível operacional (questão);
- nível quantitativo (métrica).

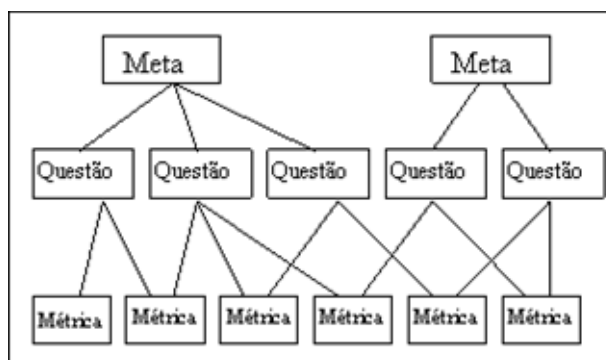


Figura A.1: Níveis do GQM (BASILI, 1994)

GQM é uma estrutura hierárquica (Figura A.1), iniciada por uma meta, que especifica o objetivo da medida, o objeto a ser medido e sob o ponto de vista de quem. A meta é refinada em questões e cada questão é então refinada em métricas. Uma mesma métrica pode ser usada para responder diferentes questões e pode ter diferentes valores em função dos diferentes pontos de vista.

Uma grande dificuldade no processo de avaliação é saber o quê avaliar. A abordagem GQM auxilia a decisão sobre quais critérios podem ser avaliados e como devem ser aplicados, a partir dos objetivos que se esperam com a avaliação.

Na próxima seção são apresentados os critérios definidos, a partir da abordagem GQM, para avaliação de cursos a distância.

Critérios para Avaliação de Cursos a Distância

Tabela A.1: Critérios para Avaliação de Cursos a Distância

Critério	Meta	Tipo das Métricas
Qualidade dos Resultados	Verificar se os resultados do curso foram satisfatórios	Objetivas e Subjetivas
Qualidade da Interação	Verificar se houve qualidade nas interações	Objetivas e Subjetivas
Estrutura do Curso	Verificar se a estrutura do curso foi adequada ao conteúdo	Objetivas e Subjetivas
Participação dos alunos	Verificar a frequência de participação dos alunos nas atividades propostas	Objetivas
Motivação dos Alunos	Verificar se os alunos permaneceram motivados	Subjetivas
Diversidade das Mídias	Verificar se houve variedade nas mídias utilizadas no curso	Objetivas
Qualidade do Conteúdo	Avaliar a qualidade do material didático preparado para o curso	Subjetivas
Adequação da Linguagem	Verificar se a linguagem utilizada foi adequada ao público-alvo	Subjetivas
Estratégias de Ensino	Analisar se a estratégia educacional utilizada foi adequada para o curso	Subjetivas
Nível de Aceitação	Verificar se curso correspondeu às expectativas dos alunos	Subjetivas
Atividades em Grupo	Analisar o número de atividades em grupo e a qualidade das participações	Objetivas e Subjetivas
Cronograma do Curso	Analisar se o tempo estimado para as atividades mostrou-se adequado	Subjetivas

Este estudo propõe um conjunto de critérios para avaliação de cursos através do estudo de outras propostas de avaliação (HORTON, 2001; RODRIGUES, 1998; PALLOFF; PRATT, 2002; VALIATI, 2000; BROADBENT, 2002) e da experiência da autora em EAD (DAHMER et al., 2000; DAHMER et al., 2001; JONSON, 2005; BELANGER; JORDAN, 2000; MOURA, 2002; AEDO; DIAZ, 2000; BONAMY; CHARLIER; SAUNDERS, 2001).

Os critérios de avaliação propostos levam em conta a organização do curso, metodologia adotada, os recursos utilizados na preparação do curso, os mecanismos de comunicação utilizados e os resultados obtidos.

Para determinar se as metas foram atingidas, alguns dos critérios propostos utilizam métricas que podem ser obtidas automaticamente através da monitoração dos cursos, outras necessitam da aplicação de um questionário com os professores ou participantes do curso, pois envolvem questões subjetivas. Assim, as métricas utilizadas são classificadas em dois tipos: objetivas e subjetivas.

Em cada um dos critérios as métricas são calculadas e combinadas. Os limites e pesos de cada métrica podem ser customizados, de acordo com os objetivos da instituição onde a avaliação será aplicada. Pois, segundo Hadji (2001), todo julgamento de avaliação é institucional e na avaliação não há legitimidade senão institucional.

A tabela A.1 apresenta um resumo dos critérios definidos através da abordagem GQM e as subseções a seguir mostram a descrição desses critérios. Em cada critério está indicada a meta a ser atingida, as questões definidas para que a meta seja atingida e as métricas que serão combinadas para responder a questão.

Critério 1: Qualidade do Resultado

Este critério relaciona-se à avaliação da qualidade do aprendizado atingido com o curso. É importante ressaltar que o critério utiliza-se das avaliações realizadas pelos alunos no decorrer do curso, desenvolvidas pelo professor. Este trabalho não entra no mérito de como avaliar alunos, assunto polêmico na EAD, assim como na educação presencial.

Meta: verificar se o modelo de curso utilizado levou a um resultado satisfatório na aprendizagem (resultados acima de um mínimo definido para cada avaliação do curso).

- *Questão 1.1: A aprendizagem desejada foi atingida?*

Métricas:

- Total de alunos com resultados satisfatórios;
- Taxa de resultados satisfatórios (T1);
- Número de avaliações com todos resultados satisfatórios;
- Número de avaliações com mais da metade dos resultados satisfatórios;
- Percentual de avaliações com resultados satisfatórios.

Obs: Por resultado satisfatório entende-se que o aluno atingiu os requisitos mínimos para aprovação. A definição dos requisitos mínimos fica sob responsabilidade do professor. Este trabalho sugere que esta métrica seja obtida através de objetivos, e não

apenas com uma nota absoluta. O professor define quais objetivos precisam ser atingidos com o curso (conteúdos a serem aprendidos) e verifica se estes conteúdos foram atingidos através do rendimento dos alunos no curso e nas avaliações realizadas. A subjetividade deste critério reside na definição da forma de avaliação dos alunos.

- Questão 1.2: *Qual é a taxa de resultados insatisfatórios?*

Métricas:

- Total de alunos com resultados insatisfatórios;
- Taxa de resultados insatisfatórios(T2).

As taxas T1 e T2, por exemplo, são calculadas pelo quociente entre o número de resultados satisfatórios (SR - por exemplo, alunos que atingiram um rendimento mínimo nas avaliações da aprendizagem) ou insatisfatórios (IR- alunos que não atingiram o rendimento esperado) e o número total de alunos do curso e o número de alunos que já realizaram o curso, conforme as fórmulas:

$$T1 = (SR / \text{total de avaliações de aprendizagem}) * 100$$

$$T2 = (IR / \text{total de avaliações de aprendizagem}) * 100$$

- Questão 1.3: *Houve acréscimo no conhecimento dos alunos sobre os temas abordados na disciplina?*

Métricas:

- Resultados das pré-avaliações;
- Resultados das avaliações realizadas ao final do curso;
- Variação dos resultados entre a pré e a pós-avaliação.

O aspecto avaliado nesta questão refere-se à avaliação do nível de informação dos participantes sobre o assunto abordado, tanto no início das atividades como durante o curso.

A realização de uma avaliação prévia é importante porque serve como indicador, quando repetida ao final do curso, do nível de aprendizado.

Critério 2: Qualidade da Interação

Este critério relaciona-se à avaliação da qualidade das participações dos alunos nas atividades de interação do curso.

Meta: Verificar se as participações dos alunos, nas atividades que exigiam interação, foram satisfatórias (não só em quantidade, mas também em qualidade).

- Questão 2.1: *A participação dos alunos foi efetiva?*

Métricas:

- Número de participações dos alunos nas ferramentas de comunicação;

- Média de participações nas ferramentas;
- Número de participações de cada aluno.

- *Questão 2.2: Houve qualidade nas participações?*

Métricas:

- Relação entre o número de participações relevantes e o número total de participações dos alunos nas ferramentas de comunicação;
- Média de participações relevantes nas ferramentas;
- Número de participações relevantes de cada aluno.

Obs: A definição de quais participações são relevantes é feita pelo professor e define a subjetividade desta questão.

Critério 3: Estrutura do Curso

Este critério relaciona-se à avaliação da estruturação do curso. Um curso mal estruturado (dividido) pode dificultar a aprendizagem dos alunos

Meta: Verificar se a estrutura do curso foi adequada ao conteúdo e se a divisão realizada foi equilibrada.

- *Questão 1.1: A divisão dos conteúdos foi bem realizada?*

Métricas:

- Média de tempo dos alunos nas atividades;
- Variação entre a média de tempo nas atividades mais longa e mais curta.

- *Questão 1.2: A estrutura foi aprovada por alunos e professores?*

Métricas:

- Taxa de aprovação da estrutura do curso (opinião dos alunos - instrumento de avaliação);
- Avaliação da estrutura pelo professor (instrumento de avaliação).

Critério 4: Participação dos Alunos

Este critério relaciona-se à análise da participação dos alunos nas atividades que compõem o curso.

Meta: Verificar a forma de participação dos alunos nas atividades propostas.

Métricas: Objetivas

- Questão 4.1: *Os alunos participaram regularmente?*

Métricas:

- Média de tempo dos alunos nas atividades;
- Variação entre a média de tempo nas atividades mais longa e mais curta.

- Questão 4.2: *Houve regularidade nos horários de acesso?*

Métricas:

- Horários de maior número de acessos;
- Variação entre a média de tempo nas atividades mais longa e mais curta.

- Questão 4.3: *O tempo das participações foi significativo?*

Métricas:

- Média de tempo dos alunos em cada participação;

Critério 5: Motivação dos Alunos

Este critério relaciona-se à motivação dos alunos. Um dos principais fatores de insucesso de cursos a distância é a falta de motivação dos alunos para realizarem as atividades.

Meta: Verificar se os alunos permaneceram motivados.

Métricas: Subjetivas

- Questão 5.1: *Os alunos mantiveram o ritmo de participação nas atividades?*

Métricas:

- Variação entre a média de tempo na atividades mais longa e mais curta.
- Evolução do tempo de participação nas atividades, no decorrer do curso.

- Questão 5.2: *Os alunos acreditaram estar motivados durante o curso?*

Métricas:

- Avaliação da motivação dos alunos (instrumento de avaliação);
- Avaliação da motivação dos alunos pelo professor (instrumento de avaliação).

Critério 6: Diversidade das Mídias

Este critério relaciona-se à diversidade das mídias utilizadas. Acredita-se que um curso que utiliza várias mídias tem uma probabilidade maior de prender a atenção dos alunos e, conseqüentemente, mantê-los motivados (interessados na aprendizagem).

Meta: Verificar se houve variedade nas mídias utilizadas no curso.

Métricas: Objetivas

- Questão 6.1: *Houve variação no uso das mídias?*

Métricas:

- Número de mídias diferentes utilizadas no curso;
- Distribuição das mídias pelas atividades do curso;
- Tempo de acesso a cada uma das mídias;

Critério 7: Qualidade do Conteúdo

Este critério relaciona-se à qualidade do material didático desenvolvido para os módulos de conteúdo pois, com materiais de baixa qualidade, a aprendizagem dos alunos fica comprometida.

Meta: Avaliar a qualidade do material didático preparado para o curso.

Métricas: Subjetivas

- Questão 7.1: *O material do curso era de boa qualidade?*

Métricas:

- Análise do material pelos alunos (instrumento de avaliação).
- Análise do material por especialistas da área (instrumento de avaliação).

Critério 8: Adequação da Linguagem

Este critério relaciona-se à linguagem utilizada no material didático e nas interações realizadas com os alunos.

Meta: Verificar se a linguagem utilizada foi adequada ao público-alvo.

Métricas: Subjetivas

- Questão 8.1: *A linguagem utilizada foi adequada ao público-alvo?*

Métricas:

- Análise da linguagem pelos alunos (instrumento de avaliação).
- Análise da linguagem por especialistas da área (instrumento de avaliação).

Critério 9: Estratégias de Ensino

Este critério relaciona-se às estratégias de ensino utilizadas durante o curso.

Meta: Analisar se a estratégia educacional utilizada foi adequada para o curso.

Métricas: Subjetivas

- Questão 9.1: *As estratégias de ensino utilizadas mostraram-se adequadas ao público alvo?*

Métricas:

- Análise das estratégias por especialistas da área (instrumento de avaliação).

Critério 10: Nível de Aceitação

Este critério relaciona-se à aceitação do curso por parte dos alunos.

Meta: Verificar se curso correspondeu às expectativas dos alunos.

Métricas: Subjetivas

- Questão 10.1: *O curso correspondeu à expectativa dos alunos?*

Métricas:

- Opinião dos alunos sobre o curso (instrumento de avaliação).

Critério 11: Atividades em Grupo

Este critério relaciona-se às atividades cooperativas realizadas pelos alunos durante o curso.

Meta: Analisar o número de atividades em grupo e a qualidade das participações.

Métricas: Objetivas e Subjetivas

- Questão 11.1: *As atividades realizadas em grupo foram adequadas ao curso?*

Métricas:

- Número de atividades em grupo;
- Avaliação, por alunos e professores, das atividades em grupo realizadas (instrumento de avaliação);
- Qualidade da participação dos alunos nas atividades em grupo.

Critério 12: Cronograma do Curso

Este critério relaciona-se ao cronograma do curso. Um curso com um cronograma mal estimado pode prejudicar o andamento do mesmo.

Meta: Analisar se o tempo estimado para as atividades mostrou-se adequado.

Métricas: Subjetivas

- Questão 12.1: *A divisão dos conteúdos foi bem realizada?*

Métricas:

- Média de tempo dos alunos nas atividades;
- Variação entre a média de tempo na atividades mais longa e mais curta.

Combinação dos Critérios

Em cada um dos critérios propostos, as métricas devem ser calculadas e combinadas. Os limites e pesos de cada métrica podem ser ajustados, de acordo com os objetivos da instituição onde a avaliação será realizada.

Considerações sobre os Critérios Propostos

Este apêndice apresentou uma proposta de um conjunto de critérios para avaliação de cursos a distância.

É importante ressaltar que existem vários trabalhos que propõem outros critérios e que esta é apenas uma proposta inicial, considerando a experiência da autora e o levantamento bibliográfico realizado. Essa proposta para avaliação de cursos em EAD precisa ser complementada e pode ser o passo inicial para futuros trabalhos, focados na avaliação dos cursos.