

Leonardo Nascimento dos Santos

*Desenvolvimento de um Multi-Organizador Flexível
de Espaços Virtuais*

Manaus
Março de 2009

Leonardo Nascimento dos Santos

***Desenvolvimento de um Multi-Organizador Flexível
de Espaços Virtuais***

Dissertação de mestrado apresentada ao
Curso de Mestrado em Informática da
Universidade Federal do Amazonas,
como requisito para obtenção do título
de Mestre em Informática

Orientador:
Professor Dr. Alberto Nogueira de Castro Júnior

UNIVERSIDADE FEDERAL DO AMAZONAS

Manaus
Março de 2009

Dissertação de Mestrado sob o título “*Desenvolvimento de um Multi-Organizador Flexível de Espaços Virtuais*”, defendida por Leonardo Nascimento dos Santos e aprovada em 31 de março de 2009, em Manaus, Estado do Amazonas, pela banca examinadora constituída por:

Prof. Dr. Alberto Nogueira de Castro Júnior
Universidade Federal do Amazonas
Orientador

Prof. Dr. Crediné Silva de Menezes
Universidade Federal do Espírito Santo
Co-orientador

Prof. Dr. José Francisco de Magalhães Netto
Universidade Federal do Amazonas

Prof. Dr. Davidson Cury
Universidade Federal do Espírito Santo

A Deus, o meu Senhor.

Agradecimentos

Antes de tudo agradeço a Deus, porque ele é quem me dá força para adquirir riquezas, e que me amou antes mesmo de eu ter fé Nele, sempre me abençoando e realizando os desejos do meu coração, preparando meu caminho para sua obra.

Queria agradecer aos meus pais que sempre me deram as condições ideais para que eu me dedicasse aos estudos, além de terem me dado o exemplo de caráter a ser seguido. Obrigado dona Suely e seu Lúcio!

Obrigado a meus irmãos por sempre acreditarem em mim. Ao meu irmão Leandro por ter me influenciado grandemente a fazer um curso de computação, vendo-o fazer seus primeiros códigos em Pascal, e a minha irmã Letícia, que está agora iniciando o seu curso de graduação em computação.

Queria agradecer a minha futura esposa Adrienne, que conheci na metade do meu curso de graduação e que já me viu na batalha e agonia do curso de graduação e agora no curso de mestrado. Esta que não me deixa esmorecer, sempre querendo que eu dê o meu melhor.

Aos amigos... Foram tantas pessoas que eu encontrei nessa jornada até hoje. Desde aqueles amigos de escola, depois os do curso técnico no ensino médio, e aqueles da faculdade. Mas amigos são aqueles que nos conhecem verdadeiramente e ainda sim gostam de nós. Tenho uma lista não muito grande deles: Fábio, Harry, Edley, Renê, Andrey e Eli. “O óleo e o perfume alegram o coração; assim é o doce conselho do homem para o seu amigo” (Pv 27.9). Mas a lista é muito maior daqueles que caminham e caminharam comigo, que tenho medo de esquecer algum na minha fraqueza de memória, mas não me interpretem mal.

Queria agradecer ao meu orientador, professor Alberto, que sempre acreditou em mim, fazendo-me crescer sempre no momento da dificuldade. E ao professor Crediné pela genial idéia que permeia este trabalho.

Agradeço também à FAPEAM pela bolsa de estudos, que possibilitou uma dedicação integral a este curso.

Obrigado a todos!

O que foi será, e o que se fez, se fará novamente; não há nada novo debaixo do sol. Será que existe alguma coisa da qual se possa dizer: Vê! Isto é novo? Não! Já existiu em épocas anteriores à nossa.

Eclesiastes 1:9-10.

Resumo

O trabalho e a aprendizagem construídos individual e coletivamente tomaram novo e determinante impulso a partir da disseminação das ferramentas de *software* que atualmente compõem os ambientes virtuais baseados na Web. Após um período de reconhecimento e apropriação desses recursos, o trabalho, o ensino, a aprendizagem e as relações sociais como um todo passam a apresentar demandas que os ambientes virtuais ainda não conseguem atender, em parte devido à conformação desses ambientes a certos aspectos funcionais mantidos principalmente por tradição. Uma estratégia possível para tratar esse problema é a concepção e desenvolvimento de ambientes flexíveis para apoiar a realização de atividades cooperativas, propiciando aos usuários uma melhor sintonia entre ferramentas de apoio e os objetivos e os perfis dos participantes de uma determinada atividade. O trabalho aqui descrito faz parte de um esforço multi-institucional de pesquisa no tema, onde, a partir dos elementos conceituais que definem uma abordagem de inovadora simplicidade à concepção e desenvolvimento de ambientes virtuais, é relatada a organização, modelagem e implementação de uma instância desse tipo de ambiente. O protótipo resultante, atualmente em plena utilização na UFAM, foi construído utilizando o Moodle como plataforma de desenvolvimento e avaliação.

Palavras-chave: Ambientes Virtuais Colaborativos; Flexibilidade em Software; CSCW/CSCL; Desenvolvimento de Software; Moodle.

Abstract

Work and learning built individually and in groups took a new and definitive impulse from popularization of software tools that currently compose web-based virtual environments. After a time of reckoning and appropriation of these resources, work, teaching, learning and social relations started to demand that virtual environments cannot yet answer, partially due to need of conformation by some of these environments to certain functional aspects maintained mainly by tradition. A possible strategy to deal with this problem is to devise and to develop flexible environments to support cooperative activities, given users a better balance between supporting tools and goals and profiles of participants in a specific activity. This work is part of a multi-institutional effort on the area where, from conceptual elements defining an innovative and simple approach to development of virtual environments, it is reported the organization, modeling and implementation of an instance of this kind of environment. The resulting prototype, currently in use at UFAM, was built using Moodle as development and evaluation platform.

Keywords: Collaborative Virtual Environments; Software Flexibility; CSCW/CSCL; Software Development; Moodle.

Sumário

Lista de Figuras	xi
Lista de Quadros.....	xiii
Lista de Siglas	xiv
1 Introdução	1
1.1 Objetivos	3
1.2 Metodologia	3
1.3 Organização do Trabalho	4
2 Ambientes Virtuais e a Organização dos Espaços de Trabalho.....	6
2.1 Trabalho apoiado pelo computador.....	6
2.1.1 Colaboração/Cooperação.....	8
2.1.2 Groupware, CSCW e CSCL	9
2.2 Ambientes Virtuais – Um Breve <i>Survey</i>	12
2.3 Ambientes de Trabalho e Conhecimento	15
3 Uma nova proposta para Organização dos Espaços Virtuais	17
3.1 Em busca de uma concepção inovadora.....	17
3.2 MOrFEU – Modelagem conceitual.....	19
3.3 Requisitos Não-Funcionais	22
4 Implementando o MOrFEU-UFAM	24
4.1 Utilizando o Moodle como Plataforma de Desenvolvimento	24
4.2 Reusando o Código do Moodle.....	26
4.3 Detalhes e Características da Implementação de um Módulo	26

4.4	Funcionalidades do Protótipo.....	30
4.5	Superclasses de Veículos de Comunicação.....	37
4.5.1	Superclasse Listagem.....	38
4.5.2	Superclasse Codificação	41
4.6	O Protótipo em Funcionamento	43
5	Discussão	45
5.1	Validação Conceitual	45
5.1.1	Cenário A: Projeto de Aprendizagem.....	45
5.1.2	Cenário B: Investigação em Grupo.....	48
5.1.3	Cenário C: Controvérsia Acadêmica	49
5.1.4	Cenário D: Jigsaw	51
5.1.5	Síntese dos Cenários	53
5.2	Avaliação do Protótipo.....	53
5.2.1	Funcionamento do Módulo do Moodle	53
5.2.2	Funcionalidades Comuns a Outros Ambientes.....	55
5.2.3	A Flexibilidade no Protótipo	56
5.2.4	Próximos Passos	58
6	Conclusão	59
6.1	Contribuições do Trabalho	59
6.2	Trabalhos Futuros.....	60
	Referências Bibliográficas	64
	APÊNDICE I: Diagrama de Navegação.....	69
	APÊNDICE II: Esquemas das Configurações dos Veículos.....	73

APÊNCIDE III: Configurações das Classes de Veículos de Comunicação: Fórum, Blog, Chat, Código Haskell e Código Prolog	77
--	-----------

Lista de Figuras

Figura 1. Modelo de colaboração 3C de Fuks. Fonte (Fuks <i>et al</i> , 2002).	9
Figura 2. Taxonomia espaço-tempo de Ellis para <i>groupware</i> . Adaptado de (Ellis <i>et al</i> , 1991).	10
Figura 3. Classificação de <i>Groupware</i> de acordo com aspectos tecnológicos. Adaptado de (Ávila, 1999).....	10
Figura 4. Artefato de Conhecimento em um Ambiente de Trabalho com Conhecimento. Adaptado de (Anttila, 2006).	16
Figura 5. Esquema Conceitual de Dados do MOrFEU.	19
Figura 6. Esquema de Dados do protótipo do MOrFEU-UFAM.	30
Figura 7. Esquema do Banco de Dados do MOrFEU-UFAM.....	33
Figura 8. Diagrama de Navegação do protótipo construído do MOrFEU-UFAM.....	34
Figura 9. Esquema de Dados de um Veículo Listagem.....	39
Figura 10. Esquema de Dados de um Veículo da Classe Blog.	39
Figura 11. Diagrama de Navegação da Superclasse de Veículo Listagem. Em azul estão as páginas que já existiam e foram reutilizadas.	40
Figura 12. Esquema de Dados de um Veículo Codificação.	43
Figura 13. Diagrama de Navegação da Superclasse de Veículo Codificação. Em azul estão as páginas que já existiam e foram reutilizadas.	43
Figura 14. Algumas páginas do protótipo funcionando: Meus Artigos (a); Meus Grupos (b); Meus Veículos (c); A Criação de um Novo Veículo (d); Editando as Configurações de um	

Veículo Blog (e); Visualizando o Veículo Blog (f); Vendo as Versões de um Artigo (g); Vendo as Diferenças entre duas Versões de um Artigo (h).....	44
Figura 15. Processo de desenvolvimento de Projetos de Aprendizagem. Fonte: (Monteiro, 2006).....	46
Figura 16. Modelagem do Veículo de Comunicação para Projetos de Aprendizagem. Uma classe especializada que pode ser instanciada para suportar a construção de Projetos de Aprendizagem específicos.....	47
Figura 17. Modelagem do Veículo de Comunicação para o Jigsaw. Sub-veículos de cada etapa compõe o veículo principal, que por sua vez são compostos por outros sub-veículos. Os veículos Exploração, Relato e Integração podem ter mais de uma instância, uma para cada grupo.....	52
Figura 18. Um Fórum para a Controvérsia Acadêmica implementado no protótipo do MOrFEU-UFAM. Em destaque nas elipses azuis, os <i>links</i> para responder aparecem apenas para os três tópicos principais, em destaque com os retângulos vermelhos.....	58
Figura 19. Uma página no Diagrama de Navegação.....	70
Figura 20. <i>Links</i> no diagrama de navegação.....	71
Figura 21. Inclusão no Diagrama de Navegação.....	71
Figura 22. Redirecionamento no Diagrama de Navegação.....	72
Figura 23. <i>Links</i> de sítios no Diagrama de Navegação.....	72

Lista de Quadros

Quadro 1. Quantidade de vezes que os ambientes foram avaliados ou comparados na literatura entre os anos de 1997 até 2004. Dados colhidos de (Itmazi, 2005). As células em branco indicam que o ambiente não foi levado em consideração na contagem.	14
Quadro 2. Configurações em XML de Veículos Listagem. Em azul e itálico, as descrições, e em vermelho e tachado, o que não foi implementado.	41
Quadro 3. Os seis estágios da Investigação em Grupo e um esboço de suas principais atividades. Fonte: (Silva <i>et al</i> , 2002).....	49
Quadro 4. Estágios do método de aprendizagem cooperativa Controvérsia Acadêmica. Fonte: (Mendonça <i>et al</i> , 2002).....	50
Quadro 5. Estágios do método de aprendizagem cooperativa Jigsaw. Fonte: (Pereira <i>et al</i> , 2002).....	52
Quadro 6. Esquema das configurações de veículos de comunicação da superclasse Listagem, feito em XMLSchema.	75
Quadro 7. Esquema das configurações de veículos de comunicação da superclasse Codificação, feito em XMLSchema	76
Quadro 8. Configuração das classes de veículos <i>Fórum</i> e <i>Blog</i>	77
Quadro 9. Configuração da classe de veículos <i>Chat</i>	78
Quadro 10. Configuração das classes de veículos <i>Código Haskell</i> e <i>Código Prolog</i>	78

Lista de Siglas

AVA	<i>Ambiente Virtual de Aprendizagem</i>
CASE	<i>Computer-Aided Software Engineering</i>
CMS	<i>Content Management System</i>
CMS	<i>Course Management System</i>
CSCL	<i>Computer-supported Cooperative Learning</i>
CSCW	<i>Computer-supported Cooperative Work</i>
HTML	<i>HyperText Modeling Language</i>
LCMS	<i>Learning Content Management System</i>
LMS	<i>Learning Management System</i>
MOrFEU	<i>Multi-Organizador Flexível de Espaços virtUais</i>
PHP	<i>Hypertext Preprocessor</i>
RSS	<i>Really Simple Syndication</i>
SGBD	<i>Sistema Gerenciador de Banco de Dados</i>
UFAM	<i>Universidade Federal do Amazonas</i>
UML	<i>Unified Modeling Language</i>
UPI	<i>Unidade de Produção Intelectual</i>
WYSIWYG	<i>What You See Is What You Get</i>
XML	<i>eXtensible Markup Language</i>

1 Introdução

O surgimento da Internet e, mais tarde, da Web provocou uma verdadeira revolução na organização e desenvolvimento das atividades humanas, notadamente aquelas realizadas de forma colaborativa (Berners-Lee, 2000). Inicialmente partiu-se das conquistas tecnológicas para oferecer ferramentas que pudessem ser usadas no apoio a atividades intelectuais, com ênfase nas formas de trabalho de grupos com maior visibilidade. Hoje, busca-se identificar e apoiar as formas de trabalho diferenciadas de cada comunidade.

Em um primeiro instante as atividades foram transportadas para o espaço virtual sob o condicionamento das ferramentas de comunicação, onde o compartilhamento das produções também ocorria através de tais ferramentas. Naquela fase, a produção coletiva foi socializada através dos programas de transferência de arquivos. Para cada atividade os usuários precisavam utilizar diferentes ferramentas.

O momento seguinte foi marcado pelo surgimento de ambientes voltados para a organização das produções coletivas, facilitando o compartilhamento e propiciando a integração de pessoas, documentos e interações. Assim surgiram os Wikis e os Blogs, onde os artefatos são diretamente integrados às ferramentas de gerência de revisão e de controle de versões. Entretanto, há ainda muitas situações trazidas por esse novo paradigma, que precisam ser exploradas e adequadamente incorporadas aos ambientes virtuais.

Diferentes grupos realizam suas atividades usando diferentes formas de organização e essas formas diferenciadas podem ser determinantes para a qualidade de suas produções. As formas de organização podem variar de atividade para atividade e podem ser criadas e modificadas sob medida para um dado empreendimento.

Black *et al* (2007) afirma que ambientes virtuais existentes no mercado são muito parecidos e padronizados, com apenas micro-detahes diferenciando-os. E ainda, que os Ambientes Virtuais de Aprendizagem (AVAs) são essencialmente produtos padronizados desenvolvidos para suporte de atividades não padronizadas, com diferentes áreas de conteúdo, filosofias e estilos de ensino. Portanto, observa-se um movimento de padronização dos ambientes, mas o mesmo não está acontecendo com as atividades realizadas dentro desses ambientes, pelo contrário, as atividades estão cada vez mais se diversificando.

Mesmo em atividades específicas, frequentemente vê-se que os ambientes disponíveis no mercado não são adequados. Tomando a área de ensino de computação, pode-se citar exemplos como o relatado em (Almeida Neto, 2007), que propôs um ambiente para autoria de programas, e em (Rößling *et al*, 2008), que discute as diversas formas de utilizar as ferramentas específicas existentes para ensino e aprendizagem de computação dentro de ambientes virtuais, mas deixa claro que isso não é possível ainda. Num contexto mais amplo de ensino e aprendizagem, o exemplo de (Mendonça *et al*, 2003) discute como o método de aprendizagem colaborativa “Controvérsia Acadêmica” pode fazer uso das ferramentas de um ambiente virtual, verificando que os ambientes tradicionais não suportam o tipo de fórum necessário para tal. Situação similar é relatada por (Menezes *et al*, 2008), que indica a inadequação de ambientes virtuais tradicionais para a arquitetura pedagógica chamada “Projeto de Aprendizagem” (Fagundes *et al*, 1999).

Em casos como os dos exemplos citados, a modificação de um ambiente existente mostrou-se inviável por diferentes fatores, provocando o desenvolvimento de várias ferramentas e ambientes virtuais específicos, o que evidencia tratar-se de um problema recorrente.

Chegamos então a um momento onde a produção de ambientes precisa deixar de ser centrada na tecnologia e deve voltar-se para as particularidades de grupos e tarefas. Uma

estratégia possível é a concepção e desenvolvimento de ambientes flexíveis para apoiar a realização de atividades cooperativas, propiciando aos usuários uma melhor sintonia entre ferramentas de apoio e os objetivos e o perfis dos participantes de uma determinada atividade. O trabalho aqui descrito faz parte de um esforço multi-institucional de pesquisa no tema onde, a partir dos elementos conceituais que definem uma nova abordagem à concepção e desenvolvimento de ambientes virtuais, relata a organização, modelagem e implementação de uma instância desse tipo de ambiente, atualmente em plena utilização na UFAM. Tal ambiente virtual recebe o nome de **MOrFEU**, um acrônimo para *Multi-Organizador Flexível de Espaços virtuais*.

1.1 Objetivos

O **objetivo geral** do trabalho foi a concepção de um ambiente virtual segundo um paradigma diferenciado para a organização da produção intelectual de seus usuários.

Foram **objetivos específicos** deste trabalho:

- A partir das limitações já registradas na literatura ou observadas em situações reais de uso, proceder a **elicitação de requisitos** para ambientes a serem desenvolvidos segundo o paradigma proposto;
- Desenvolver, segundo o paradigma discutido, uma **modelagem conceitual** para o MOrFEU;
- Construir uma **versão inicial** para o MOrFEU, utilizando o **Moodle** (Dougiamas&Taylor, 2003) **como plataforma de desenvolvimento**;
- **Avaliar** a adequação conceitual da ferramenta desenvolvida;

1.2 Metodologia

O trabalho foi desenvolvido através de um conjunto de procedimentos e estratégias próprias à área conforme segue:

- a) Levantamento Bibliográfico – na etapa inicial da investigação, além da revisão realizada em trabalhos da literatura relacionada, buscando contextualizar adequadamente a proposta, duas outras atividades merecem destaque:
1. Análise de ambientes virtuais – onde, de acordo com a disponibilidade de acesso aos ambientes, foram observadas suas ferramentas constituintes e metáforas de utilização.
 2. Elicitação de experiências no uso e desenvolvimento de ambientes virtuais – a partir de relatos e cenários de situações reais de uso, foram definidas estratégias para o desenvolvimento do projeto, especialmente aquelas relacionadas ao desenvolvimento de *software*.
- b) Registro de aspectos conceituais do MOrFEU – a partir de relatos e decisões de projeto definidas por um grupo de pesquisa interinstitucional, alguns pressupostos conceituais do MOrFEU foram listados.
- c) Exploração do Moodle – para definição do esquema de implementação a ser utilizado no projeto.
- d) Modelagem do ambiente – a partir dos requisitos definidos pela experiência (item a.1) e pressupostos conceituais (item b).
- e) Implementação – prototipagem do ambiente, aplicando as estratégias definidas no item (c).
- f) Testes para validação conceitual.

1.3 Organização do Trabalho

No Capítulo 2 busca-se situar a proposta através de um levantamento sobre como atividades relacionadas ao trabalho e aprendizagem, especialmente as que envolvem colaboração, são desenvolvidas fazendo-se uso de ambientes virtuais. A seguir, no Capítulo 3,

discute-se como é possível tornar esses espaços virtuais mais acessíveis através de uma maneira inovadora, onde se apresenta o MOrFEU, através de alguns elementos do seu arcabouço conceitual. No Capítulo 4 são descritos os detalhes de implementação do protótipo do MOrFEU na UFAM e suas funcionalidades. No Capítulo 5 são discutidos alguns testes para validação conceitual do ambiente implementado, seguido das conclusões, no Capítulo 6.

2 Ambientes Virtuais e a Organização dos Espaços de Trabalho

Desde seu surgimento, o computador tem sido utilizado como ferramenta de trabalho para automação de processos, passando também, em certo momento, a ter papel determinante na comunicação entre indivíduos e até na organização do trabalho. Mais recentemente, ambientes virtuais acessíveis através da Web têm sido desenvolvidos para apoiar a interação entre membros de uma comunidade e as construções coletivas. Neste capítulo fazemos uma breve discussão sobre esse panorama, tendo sempre como foco de interesse os ambientes para apoio ao trabalho ou à aprendizagem realizados de forma coletiva.

2.1 Trabalho apoiado pelo computador

Segundo Galbraith (1977 apud Lyytinen & Ngwenyama, 1992, p.2), uma visão tradicional de trabalho que embasava (e ainda embasa) muito da pesquisa em computação e sistemas de informação é fundada numa teoria de trabalho mecanicista, que sugere que o trabalho pode ser dividido em uma seqüência de tarefas caracterizadas somente em termos de sua incerteza e da necessidade de informação para a tomada de “decisões”.

Em um artigo discutindo questões relacionadas à reengenharia do trabalho e das organizações num contexto onde o computador e várias tecnologias relacionadas já se faziam presentes, Michael Hammer (1990) sustenta que pesados investimentos em Tecnologia da Informação frequentemente produzem resultados desanimadores devido ao fato de que as organizações tendem a usar a tecnologia para mecanizar antigas formas de trabalho, deixando os processos existentes intactos, usando computadores simplesmente para torná-los mais velozes. Segundo Hammer, é comum que o trabalho seja organizado como uma seqüência de tarefas separadas e que sejam empregados complexos mecanismos para monitorar seu

progresso. Apesar de que os sistemas para impor controle e disciplina sobre aqueles que de fato realizam o trabalho terem sido criados no período imediatamente posterior à 2ª. Guerra, tal arranjo remonta à Revolução Industrial, onde devido à escassez de pessoal com habilidades para o trabalho gerencial, uma forte hierarquia era necessária e os sistemas de controle conduziam informação para o topo da hierarquia para os poucos que supostamente saberiam o que fazer com ela.

Se por um lado Hammer ilustrava que bancos de dados compartilhados e redes de computadores possibilitavam diferentes tipos de informação a uma pessoa e que sistemas especialistas poderiam ajudar pessoas com experiência limitada a tomar decisões bem fundamentadas, caracterizando um cenário de confronto a noções centenárias sobre o trabalho, a importância das ferramentas de comunicação num contexto de apoio à decisão havia sido ressaltada na década anterior (Turoff&Hiltz, 1982), inclusive trazendo uma preocupação com a possibilidade de que uma excessiva ou inadequada estruturação dessa comunicação viesse a prejudicar a interação entre os membros de um grupo, uma vez que “Muitas empresas que passaram a usar computadores em sua comunicação pensam neles [apenas] como um correio eletrônico que provê uma simulação automatizada de memorandos internos, servindo apenas para estabelecer canais de comunicação mais rápidos e baratos. Essas empresas não estão preocupadas que possam existir melhores modos de comunicar ou, especificamente, se o computador possibilitaria novas formas de comunicação”.

A integração de grupos de trabalho geograficamente dispersos foi apenas uma das possibilidades trazidas pelas ferramentas de comunicação, ajudando a tornar mais explícita a natureza multidimensional e a riqueza social na articulação dos processos de trabalho, características antes negligenciadas no desenvolvimento de aplicações da Tecnologia da Informação – situação que mudaria significativamente a partir de meados da década de oitenta (Lyytinen & Ngwenyama, 1992).

2.1.1 Colaboração/Cooperação

Existem pelo menos três pontos de vista com relação aos conceitos de colaboração e cooperação. Um diz respeito que colaboração tem o mesmo sentido de cooperação; um segundo ponto de vista diz que colaboração é mais geral que cooperação; e o terceiro, que cooperação é mais geral que colaboração.

Dois autores não fazem distinção entre os dois conceitos. Ferreira (1986, p. 38 apud Maçada&Tijiboy, 1998) define colaboração como “trabalho em comum com uma ou mais pessoas; cooperação; auxílio; contribuição”. E ainda, Kaye (1991, p. 20 apud Maçada&Tijiboy, 1998) acredita que:

[...] colaborar (co-labore) significa trabalhar junto, que implica no conceito de objetivos compartilhados e uma intenção explícita de somar algo - criar alguma coisa nova ou diferente através da colaboração, se contrapondo a uma simples troca de informação ou passar instruções.

No entanto, para Maçada&Tijiboy (1998), que compartilha as idéias de Piaget (1973 apud Maçada&Tijiboy, 1998) e Vygotsky (1987 apud Maçada&Tijiboy, 1998), o conceito de cooperação é mais complexo, pois pressupõe a interação e a colaboração, de forma que perceberam que:

[...] a diferença fundamental entre ambos conceitos reside no fato de que para haver colaboração um indivíduo deve interagir com o outro, existindo ajuda - mútua ou unilateral. Para existir cooperação deve haver, interação, colaboração, mas também objetivos comuns, atividades e ações conjuntas e coordenadas.

O terceiro ponto de vista é o de Fuks *et al* (2002) que, baseado em (Ellis *et al*, 1991), acredita que o conceito de colaboração é mais geral e abrange os conceitos 3C de comunicação, coordenação e cooperação. A Figura 1 apresenta de modo esquemático os conceitos 3C as relações entre si.

Neste trabalho, adota-se a visão de Maçada&Tijiboy (1998): sendo a cooperação um conceito mais complexo e estando inclusos os conceitos de colaboração e interação, sendo necessários para se haver cooperação.

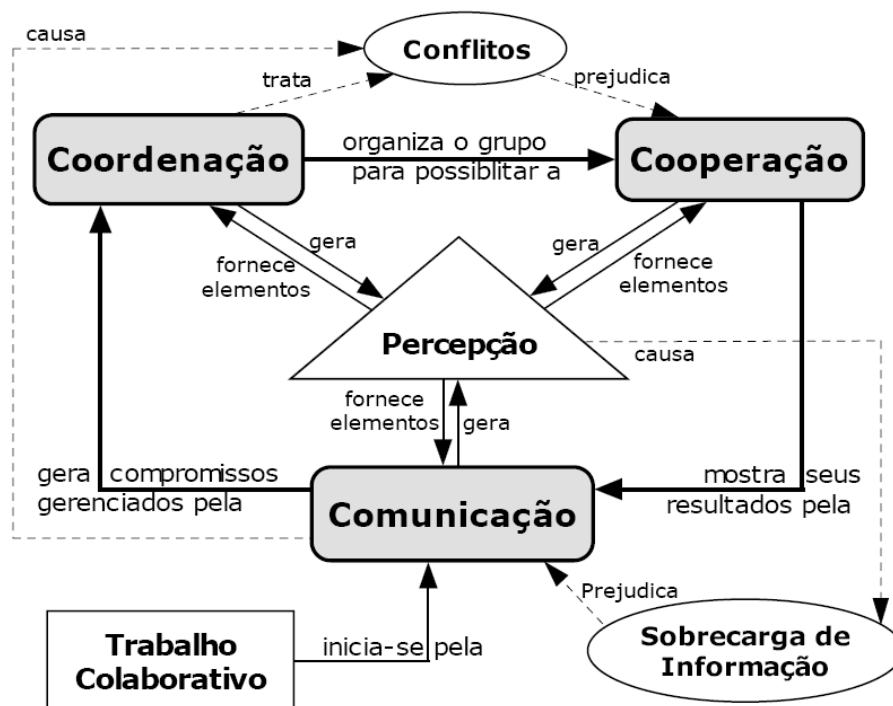


Figura 1. Modelo de colaboração 3C de Fuks. Fonte (Fuks *et al*, 2002).

2.1.2 Groupware, CSCW e CSCL

Os sistemas *groupware* são aplicações que dão suporte a grupos. (Ellis *et al*, 1991) define os *groupware* como sistemas de computação que apóiam grupos de pessoas que tomam parte em uma tarefa (ou meta) em comum e que provêm uma interface para um ambiente compartilhado. Na Figura 2 é apresentada a taxonomia espaço-tempo de Ellis *et al* (1991). Ainda existe outras definições para o termo, que segundo (Johnson-Lenz&Johnson-Lenz, 1981 apud Penichet *et al*, 2007) a definição original é: um processo intencional de grupos mais o *software* para dar suporte a eles.

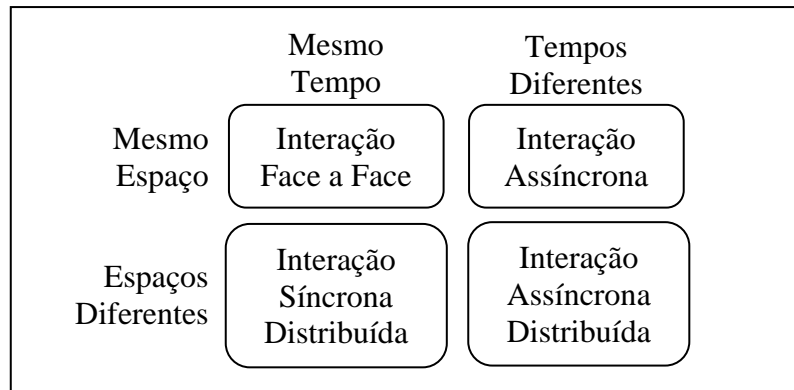


Figura 2. Taxonomia espaço-tempo de Ellis para *groupware*. Adaptado de (Ellis *et al*, 1991).

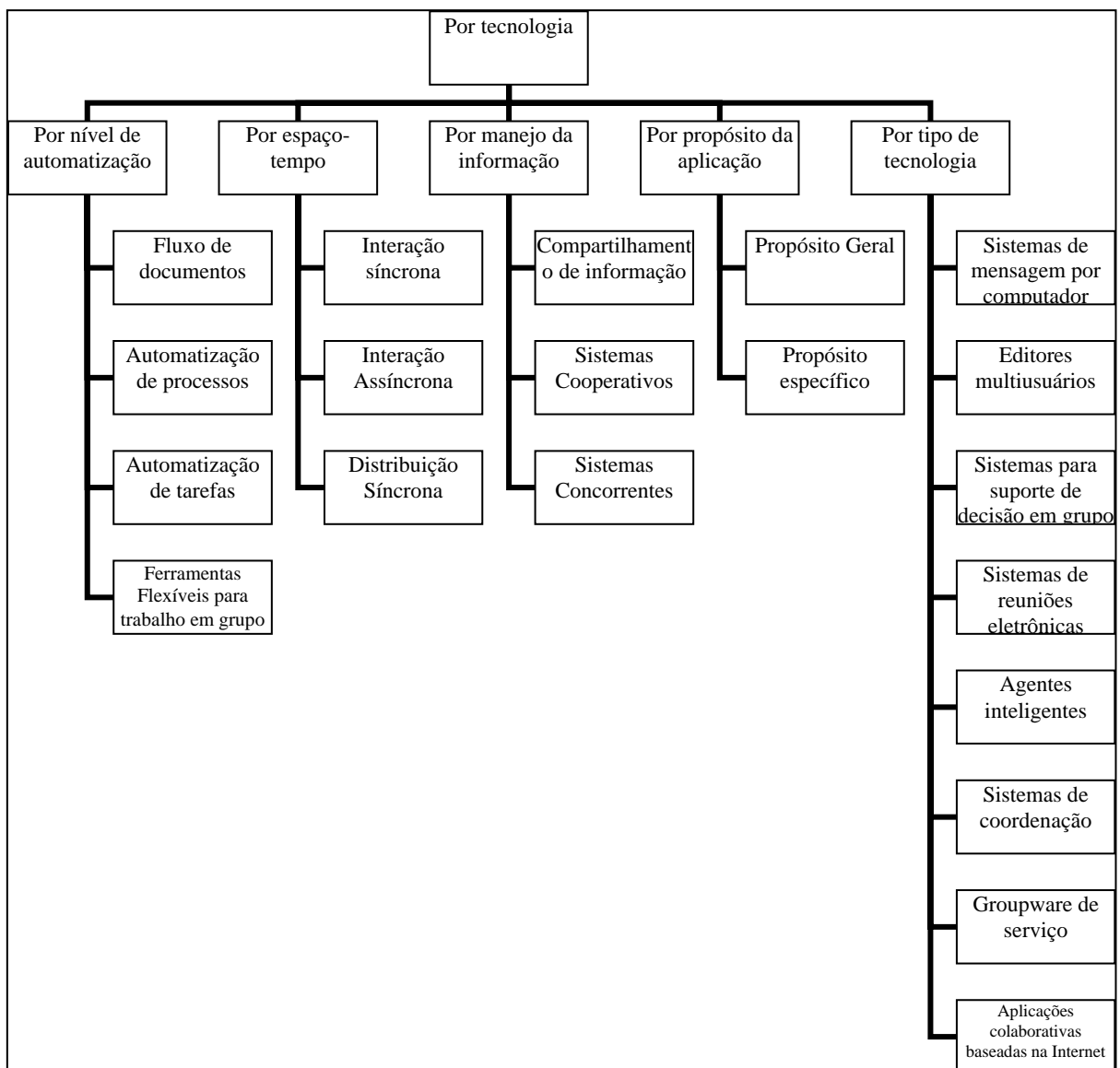


Figura 3. Classificação de *Groupware* de acordo com aspectos tecnológicos. Adaptado de (Ávila, 1999).

Ávila (1999) classifica os *groupware* em diversas categorias e em diversos ângulos, que primeiramente divide em “tipo de tecnologia” e “propósito do grupo”. A Figura 3 mostra um quadro de classificação das aplicações *groupware* de acordo com a tecnologia.

Por sua vez, CSCW (*Computer-supported Cooperative Work*) é o estudo de como as pessoas utilizam a tecnologia, com relação a *hardware* e *software*, para trabalhar juntas em tempo e espaço compartilhados, segundo (Rama&Bishop, 2006). Há ainda outros autores com outras definições sobre o que a área de CSCW estuda, tal como em (Schmidt&Bannon, 1992), que afirma que CSCW é uma área de pesquisa destinada à exploração e descoberta dos requisitos de suporte à organização do trabalho cooperativo. Portanto, a diferença entre CSCW e *groupware* é que CSCW descreve a pesquisa e *groupware* descreve a tecnologia (Grudin, 1994 apud Schmidt&Bannon, 1992).

Observa-se que os Ambientes Virtuais de Aprendizagem (AVAs) são *groupwares* que compartilham a classificação de Ávila (1999). Eles são estudados pela área de pesquisa chamada de CSCL (*Computer-supported Cooperative Learning*), uma derivação do termo CSCW. No entanto, para esses há ainda outras nomenclaturas e classificações, que foram exploradas por (Watson&Watson, 2007), que define LMS, CMS e LCMS. Assim, um *Learning Management System* (LMS) é um *framework* que cuida de todos os aspectos do processo de aprendizagem (Watson&Watson, 2007), não se limitando apenas ao ambiente, mas também cuida das atividades e processos realizados nesse ambiente. No entanto, LMS é um termo muito utilizado na literatura para se referir a AVAs em geral. E um *Course Management System* (CMS) é um ambiente que provê o instrutor com um conjunto de ferramentas e um *framework* que permite a criação relativamente fácil de conteúdos de cursos *online* e o subsequente ensino e gerência do curso incluindo as interações dos estudantes do curso (EDUCAUSE Evolving Technologies Committee, 2003, p.1 apud Watson&Watson, 2007). Por fim, um *Learning Content Management System* (LCMS) é um sistema usado para

criar, armazenar, montar e entregar conteúdo personalizado de *e-Learning* na forma de objetos de aprendizagem (Oakes, 2002, p. 73 *apud* Watson&Watson, 2007). Aqui, entende-se por *e-Learning* (aprendizagem eletrônica), como a aprendizagem realizada através de tecnologias digitais.

2.2 Ambientes Virtuais – Um Breve *Survey*

Existem vários ambientes virtuais baseados na Web. E os AVAs são os mais abundantes nesse meio. Itmazi (2005) afirma que o mercado de AVAs no ano de 2005 possuía pelo menos 200 produtos. Em (Cátedra Unesco de Educación a Distancia, 2009) há uma lista com 116 nomes de AVAs e *links* para suas páginas oficiais. (EduTools, 2009a) e (EduTools, 2009b) trazem outras listas de AVAs (muitos dos presentes na lista anterior), diferenciando por versões, o primeiro com AVAs mais antigos e o segundo com mais recentes, e dando acesso a comentários feitos sobre algumas categorias de características, possibilitando uma comparação entre os ambientes.

Muitos desses produtos surgiram ao longo dos anos e se desenvolveram, tornando-se populares e ganhando sempre novas versões. No entanto, outros são apenas citados, mas não houve um real desenvolvimento destes ambientes. Dois históricos sobre o desenvolvimento de AVAs baseados na web podem ser encontrados em (Wikipedia, 2009) e (Moodle.org, 2009a). Baseado nestas duas listas e no surgimento de outros AVAs que se tem conhecimento, principalmente no cenário nacional, elaborou-se a seguinte lista, indicando o ano da aparição da primeira versão do dado ambiente. Nesta lista foram omitidos vários ambientes, deixando apenas aqueles com maior popularidade:

- 1995: BSCW¹, TopClass²;
- 1997: WebCT³, Blackboard⁴;
- 1999: LON-CAPA⁵;
- 2000: Claroline⁶, ILIAS⁷, AmCorA (Menezes *et al*, 2000), AulaNet (Fuks, 2000), e-Prinfo (Ministério da Educação, 2000);
- 2002: Moodle⁸ (Dougiamas&Taylor, 2003), dotLRN⁹, ATutor¹⁰, Fle3¹¹;
- 2003: TelEduc (Rocha, 2003);
- 2004: Sakai¹²;
- 2005: AMADIS (Monteiro *et al*, 2005);
- 2006: COFALE (Chieu, 2006), ACAI (Lima, 2006);
- 2007: TIDI-Ae (Beder *et al*, 2007), Amadeus¹³;

Itmazi (2005) apresenta um estudo sobre 58 trabalhos de comparações de AVAs para *e-Learning*, encontrados em sua biblioteca e na Internet até princípios de 2005, visando à escolha de um para a implementação do seu protótipo. Se um ambiente faz parte de um trabalho de comparação com outros ambientes, quer dizer que ele é popular e está sendo utilizado pela comunidade ou possui características interessantes levadas em consideração na comparação. Assim, elaborou-se o Quadro 1 para mostrar a evolução de popularidade em

¹ <http://www.bscw.de/>

² <http://www.wbtsystems.com/>

³ <http://www.webct.com/>

⁴ <http://www.blackboard.com/>

⁵ <http://www.lon-capa.org/>

⁶ <http://www.claroline.net/>

⁷ <http://www.ilias.de/>

⁸ <http://moodle.org/>

⁹ <http://dotlrn.org/>

¹⁰ <http://www.atutor.ca/>

¹¹ <http://fle3.uiah.fi/>

¹² <http://sakaiproject.org/>

¹³ <http://amadeus.cin.ufpe.br/>

trabalhos de comparação de AVAs. Neste quadro só foram apresentados os AVAs com um bom número de comparações, pois havia muitos AVAs que foram comparados apenas pouquíssimas vezes.

Ambientes Virtuais	Ano						
	1997-1998	1999	2000	2001	2002	2003	2004
	4 trab.	3 trab.	6 trab.	7 trab.	10 trab.	18 trab.	8 trab.
WebCT	4	3	3	7	10	10	4
Blackboard	2	2	5	7	6	11	4
TopClass	3	2	2	1	0	2	0
ILIAS						5	3
ATutor						5	3
dotLRN						4	2
Moodle						8	5
Claroline						4	1
Fle3						3	0
LON-CAPA						3	0

Quadro 1. Quantidade de vezes que os ambientes foram avaliados ou comparados na literatura entre os anos de 1997 até 2004. Dados colhidos de (Itmazi, 2005). As células em branco indicam que o ambiente não foi levado em consideração na contagem.

Observa-se no Quadro 1 a popularidade do WebCT e do Blackboard, estando sempre entre os mais citados em trabalhos de comparação de AVAs, que, por serem *software* comerciais, tiveram um bom investimento nos seus desenvolvimentos. Mas observa-se nos últimos anos analisados por (Itmazi, 2005) uma crescente recorrência para o Moodle. O sítio (Google Directory, 2009) traz um lista de produtos para criação de cursos na Web, ordenados por sua popularidade em *links* de páginas Web, e em primeiro lugar encontra-se o Moodle, seguido pelo WebCT e o Blackboard. O sítio (Moodle.org, 2009b) traz estatísticas a respeito da utilização do Moodle, que já conta com quase de trinta milhões de usuários cadastrados em cerca de cinquenta mil sítios registrados em 208 países.

Black *et al* (2007) discute o que pode ser feito para aumentar as chances de escolha de um AVA, demonstrando quais características devem ser levadas em consideração, como compatibilidade, vantagem relativa, capacidade de ser testado, complexidade e satisfação do usuário. No entanto, ele ressalta essas características por concluir que os AVAs existentes no mercado são muito parecidos e padronizados, com apenas micro-detahes diferenciando-os. E

ainda que os LMSs são essencialmente produtos padronizados desenvolvidos para suporte de atividades não padronizadas, com diferentes áreas de conteúdo, filosofias e estilos de ensino. Portanto, observa-se este movimento de padronização dos ambientes, mas não é isto o que está acontecendo com as atividades realizadas dentro desses ambientes, pelo contrário, as atividades estão cada vez mais se diversificando. Assim, o que se espera de um ambiente virtual de uso geral é que ele seja adequado para todos esses tipos de atividades, sendo flexível o suficiente para tal.

2.3 Ambientes de Trabalho e Conhecimento

O trabalho e a aprendizagem colaborativos necessitam do compartilhamento de estruturas conceituais para a incorporação de um conceito já existente ou a criação de um novo conhecimento. Anttila (2006) destaca os requisitos básicos para um ambiente de trabalho colaborativo com conhecimento intenso (“*collaborative knowledge-intensive work*”), seguindo os modelos da Web 2.0 (também chamada de Web Social) (O’Reilly, 2005), utilizando algumas de suas ferramentas: *agregadores, fóruns, wikis, blogs e arquivos*. Mas o interessante é que o conhecimento é tratado como uma unidade, um *artefato de conhecimento* (ver Figura 4). Isto se reflete da necessidade de unificação das unidades que compõem o conhecimento. E demonstra que as ferramentas de interação e comunicação compõem uma parte importante no desenvolvimento deste conhecimento, e seus produtos não podem ser vistos separadamente.

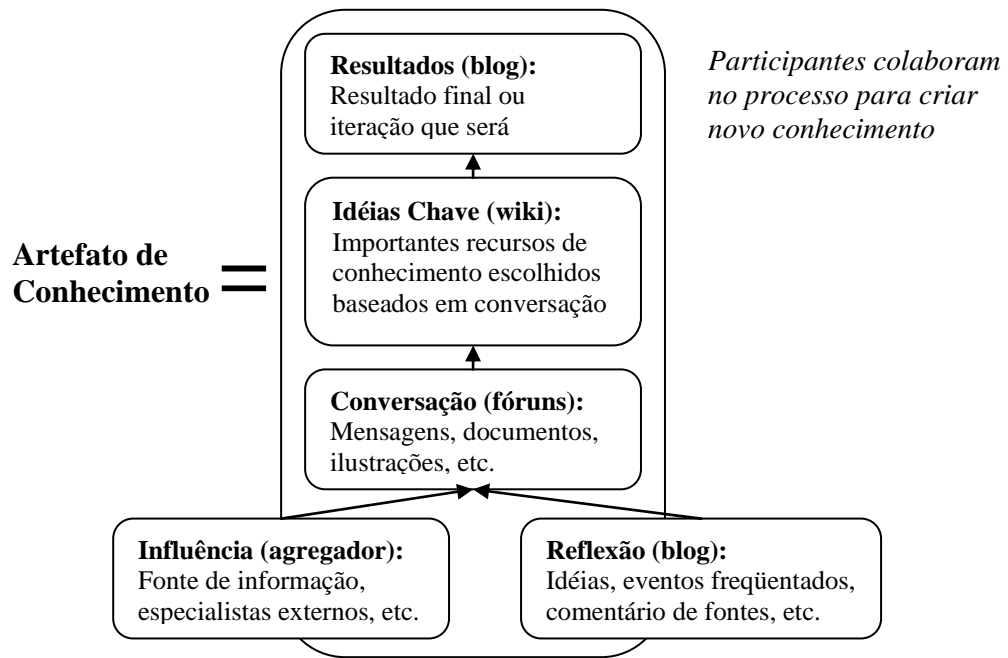


Figura 4. Artefato de Conhecimento em um Ambiente de Trabalho com Conhecimento. Adaptado de (Anttila, 2006).

3 Uma nova proposta para Organização dos Espaços Virtuais

Para se obter uma real inovação no uso dos ambientes virtuais é preciso pensá-los de forma diferenciada. Muitas das ferramentas de comunicação disponíveis nesses ambientes foram construídas de acordo com uma concepção definida historicamente. Ou seja, o surgimento de novas ferramentas na Web frequentemente está associado a ferramentas já existentes, de forma que um ambiente virtual se torna um agregador dessas ferramentas, nem sempre adequando-se aos requisitos para a realização de uma dada atividade. Este capítulo descreve a busca por uma abordagem mais flexível, apresentando um modelo conceitual para tal paradigma.

3.1 Em busca de uma concepção inovadora

Atividades relacionadas com o trabalho, os negócios, o lazer, a filantropia, a aprendizagem e outras, podem ocorrer em ambientes virtuais. Embora visem diferentes objetivos, todas elas requerem facilidades para autoria cooperativa, aquisição e socialização de conhecimento.

A Internet está povoada por ferramentas de comunicação e interação que são implementadas pela maioria dos sistemas de ambientes virtuais usados por diferentes grupos de desenvolvimento de *software*, sejam eles CMS (*Content Management System*) ou LMS (*Learning Management System*), além de sistemas de relacionamentos, entre outros.

Das ferramentas usuais podemos citar: fóruns, e-mails, chats, murais, wikis, blogs, FAQs, organizadores de fotos, organizadores de músicas, *webfolios*, etc. Em geral, cada ambiente permite a configuração de uso para um elenco restrito de ferramentas, de estrutura

predefinida, com pequenas facilidades de configurações. A hipótese que seguimos é que as ferramentas hoje disponíveis ainda seguem os modelos surgidos historicamente e baseados nas possibilidades disponíveis na Web quando foram criadas. Essas ferramentas foram construídas buscando modelar as necessidades de grupos e atividades estereotipadas, e em situações reais de uso, especialmente onde estratégias pedagógicas mais recentes são aplicadas (Nevado *et al*, 2007), têm se mostrado cada vez mais inadequadas.

É sabido que quando uma nova idéia surge para ferramentas dessa natureza, ela, em geral, é implementada em ferramentas específicas, independente dos ambientes virtuais integrados, e para que possamos usá-las nesses ambientes é necessário aguardarmos a implementação por uma equipe de programação. Por outro lado, à medida que uma ferramenta dessas ganha popularidade, seus idealizadores tendem a potencializá-la como sistema independente, incorporando facilidades presentes nos ambientes integrados mais populares. Podemos citar como exemplo o caso dos sistemas para produção de blogs, uma ferramenta bastante conhecida hoje, muito usada por profissionais ou amadores de diferentes ramos. À medida que o uso do blog foi se popularizando, foram sendo incorporados outros recursos tais como fóruns, marcadores e enquetes. Por outro lado, os ambientes integrados buscam a incorporação dessas novas ferramentas, como podemos observar em ambientes virtuais de aprendizagem como o Moodle, ou ambientes gerenciadores de conteúdo como o Drupal¹⁴, que incorporaram blogs. Isso é semelhante ao que ocorre com ferramentas tais como wikis, fóruns, RSS e outros.

Nesse contexto, participamos de um esforço de investigação para identificação e modelagem de elementos básicos das atividades intelectuais individuais e coletivas realizadas em ambientes virtuais, que permitam a construção de ferramentas flexíveis que possam ser facilmente integradas aos perfis dos participantes e às características de atividades específicas.

¹⁴ <http://drupal.org/>

Como catalisador para essa investigação, em (Menezes *et al*, 2008) foi proposta a concepção e implementação de um ambiente virtual flexível para organização das interações de pessoas, de suas bibliotecas digitais (multimidiáticas) e de suas produções individuais e coletivas, inclusive as resultantes de interações, de maneira tal que resgate os aspectos singulares de cada indivíduo, promovendo características adaptativas ao ambiente. Tais produções serão posteriormente, objeto de gestão do conhecimento e deverão receber suporte inteligente de agentes de *software* dedicados. Esse mundo virtual foi denominado MOrFEU, um acrônimo para *Multi-Organizador Flexível de Espaços virtUais*.

3.2 MOrFEU – Modelagem conceitual

A concepção primordial do sistema é o suporte à autoria, à publicação e à socialização das produções intelectuais. Ao invés de colocar o foco no uso de ferramentas, dentro de um determinado contexto, teremos como base a manifestação dos sujeitos através do que chamaremos de Unidade de Produção Intelectual (UPI). A Figura 5 apresenta os elementos básicos da estrutura conceitual inicial, proposta para o ambiente.

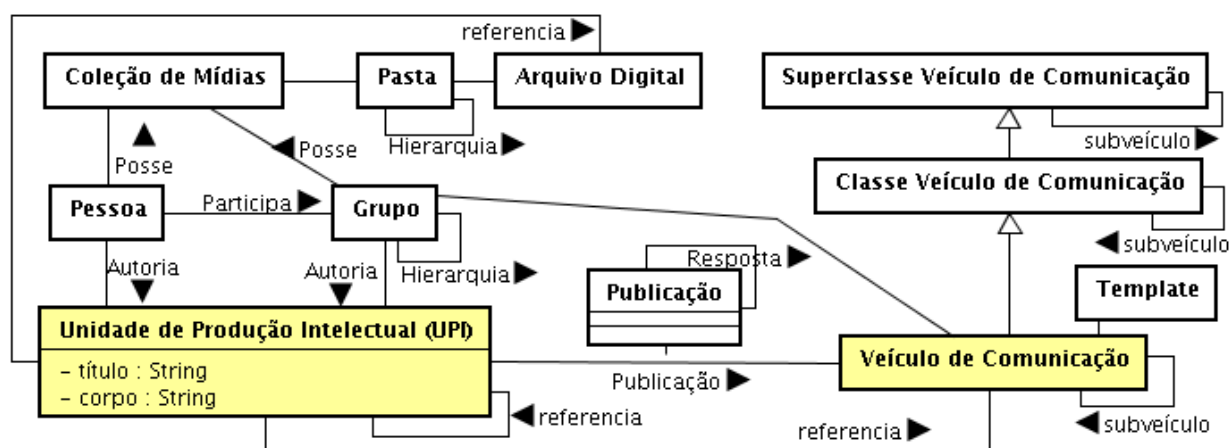


Figura 5. Esquema Conceitual de Dados do MOrFEU.

Uma UPI tem a forma de uma mensagem que se deseja registrar. Ela tem autor(es), um título e um corpo. Os autores podem ser tanto pessoas quanto grupos. A revisão de uma UPI produz sempre uma nova versão da mesma, mantendo intacta a versão anterior. Isto permite a gerência de versões. O suporte à publicação de UPIs é o que se denominou de

Veículo de Comunicação. Assim, as UPIs são sempre socializadas e publicadas através dos Veículos de Comunicação, que são responsáveis também pela interação entre as pessoas.

Cada usuário do ambiente pode escrever inúmeras UPIs, as quais podem ser ou não socializadas. As UPIs não socializadas ficam automaticamente na coleção de UPIs privadas, que é um veículo de comunicação básico do ambiente. As produções podem ser organizadas de diferentes maneiras pelo autor, sendo a ordenação padrão por ordem cronológica (o mais recente está no topo da pilha). Neste sentido, a organização básica lembra a caixa de produções de um escritor, na qual as páginas produzidas vão sendo empilhadas. As UPIs podem ser socializadas através de um ou mais veículos de comunicação. Para socialização, o veículo de comunicação básico é a coletânea de UPIs socializadas, por autor. Cada autor tem o seu veículo individual de socialização.

As UPIs podem fazer referências a outros elementos do ambiente. Estas referências podem ser a arquivos digitais presentes na coleção de mídias, a outras UPIs ou a veículos de comunicação. Estas referências devem ser apresentadas como *links* e deve-se ter uma listagem destas referências em cada UPI.

O Ambiente deve prover suporte à criação e edição de Veículos de Comunicação. Existe uma organização de Veículos de Comunicação, em forma de Superclasse, Especialização e Instância. Por exemplo, um jornal é uma classe geral de veículo (ou uma Superclasse), o jornal a Folha de São Paulo é uma especialização da classe Jornal (ou uma Classe) e um exemplar da Folha, de 20 de janeiro de 2008, é uma instância (ou um Veículo).

Todo e qualquer suporte à interação é modelado através do conceito de Veículo de Comunicação. Na concepção do MOrFEU, os chats, os fóruns, o correio eletrônico, os murais e outros, são modelados como veículos de comunicação. Mensagens enviadas a um fórum, a um correio eletrônico, a um chat, etc., são UPIs que também estarão disponíveis na coletânea de publicações individuais, acessíveis diretamente por este veículo. Um dado veículo possui

periodicidade e sua publicação pode estar condicionada ou não à análise de revisores. O formato de exibição de UPIs e de veículos será definido por *templates* (inicialmente através de CSS) com idéias baseadas no ambiente MAMBO¹⁵ e congêneres.

Os veículos possuem configurações que moldam suas características, e estas configurações são passíveis de modificações a qualquer momento. Cada superclasse de veículo é responsável por definir seu conjunto de configurações possíveis, que serão repassadas aos seus “herdeiros”, suas especializações e instâncias. Inicialmente, podem existir configurações de permissões de visualização e navegação, de permissões de publicação, de permissões de modificação de configurações, de modos de apresentação (que pode estar agregado ao modelo de *templates*).

Um veículo pode fazer o papel de agregador, de forma a centralizar o conhecimento, tendo função semelhante ao “artefato de conhecimento” de (Anttila, 2006), ilustrado na Figura 4. Isto é feito através da relação de “sub-veículo” entre veículos. Como as superclasses de veículos modelam todos os veículos “herdeiros”, esta deve definir as características e configurações de quais tipos e como os veículos serão agregados nos veículos herdeiros. Já o papel da classe de veículos é servir de modelo inicial para a agregação dos veículos, já que as características de um veículo podem ser modificadas a qualquer momento, seguindo as regras impostas pela sua superclasse. Desta forma, um veículo pode possuir sub-veículos diferentes dos definidos pela sua classe, como um modelo.

Assim, as superclasses de veículos têm como papel definir os limites dos veículos herdeiros, assim como seu funcionamento e navegação. As classes farão papel de modelo para os veículos herdeiros, um molde inicial para os veículos instanciados nesta classe. E os veículos devem ter a capacidade de se modificar a qualquer instante, seguindo sempre as definições de sua superclasse.

¹⁵ <http://mambo-foundation.org/>

O sistema pode dar suporte também às coleções de mídias, onde podem ser organizados arquivos digitais em diferentes formatos e que podem ser referenciados e utilizados pelas UPIs.

Os usuários do ambiente podem se organizar em grupos. Os grupos, por sua vez, são organizados hierarquicamente, com cada grupo com o seu grupo-pai, seguindo a idéia de Menezes *et al* (2000). Os níveis de visibilidade das produções e das mídias são: “privado”, “grupos selecionados”, “meus grupos”, “usuários cadastrados” e “todos os usuários da Web”.

3.3 Requisitos Não-Funcionais

Uma vez que a Seção 3.2 apresentou algumas funcionalidades da proposta, enumeramos a seguir alguns requisitos não-funcionais já identificados.

- a) **Usabilidade:** pode ser definida como a capacidade de uma ferramenta ser usada facilmente e eficientemente por humanos (Shackel, 1991, p.24 apud Hornbæk, 2006), ou a efetividade, a eficiência e a satisfação com que usuários específicos podem alcançar metas em ambientes particulares (ISO, 1998, p.2 apud Hornbæk, 2006). As possibilidades de organização e visualização das UPIs de cada indivíduo e dos grupos a que pertence necessitam contar com recursos de manipulação adequados à simplicidade e “naturalidade” da proposta;
- b) **Aceitação e Adoção:** é necessário que o MOrFEU tenha uma estratégia de desenvolvimento com diferentes etapas de avaliação formativa que permita-lhe ter uma boa aceitação e adoção como ambiente virtual de uso geral na Web. Uma discussão mais aprofundada sobre o que pode ser feito para que se aumentem as possibilidades de isso acontecer pode ser encontrado em (Black *et al*, 2007);
- c) **Suportabilidade:** este requisito se refere às estratégias de desenvolvimento, suporte e manutenção do ambiente, iniciando-se por um bom conjunto de recursos facilitador

(bibliotecas de componentes, tecnologias, etc) por parte dos desenvolvedores, incluindo adaptabilidade a diferentes plataformas e personalização.

4 Implementando o MOrFEU-UFAM

A construção de ambientes virtuais baseados na Web para apoiar o trabalho ou a aprendizagem colaborativa é evidentemente uma tarefa árdua. E para construir um protótipo que agregue as funcionalidades que se deseja, é necessário ter muitas das funcionalidades básicas de um sistema Web. Mesmo considerando-se a construção de um AVA com técnicas de Engenharia de *Software* baseadas em componentes, onde são agregadas todas as vantagens do método, a exemplo do que ocorreu no desenvolvimento do TIDIA-Ae (Beder *et al*, 2007), o processo pode se tornar muito complexo.

Se por um lado o desenvolvimento de todas as funcionalidades de um ambiente virtual, que envolve a construção dos serviços mais simples até a composição de intrincados módulos, proporciona autonomia ao desenvolvedor, por outro lado a manutenção e integração do grande número de serviços requer uma enorme quantidade de tempo e esforço. De fato, a participação do autor no desenvolvimento de novas funcionalidades para um ambiente virtual, o AmCorA (Menezes *et al*, 2000), confirmou que a maior parte do tempo e esforço foi aplicado em administrar serviços básicos do sistema como o de correio eletrônico, de banco de dados, do servidor Web, de cadastro e atualização de usuários, etc.

4.1 Utilizando o Moodle como Plataforma de Desenvolvimento

Os relatos na literatura relacionada e a experiência pessoal no desenvolvimento de ambientes virtuais baseados na Web motivaram a busca por uma abordagem orientada ao reuso que, além da economia de tempo e esforço, incorporasse confiabilidade no que diz respeito ao funcionamento dos serviços básicos (grandes demandantes para o desenvolvedor). Após um levantamento em diferentes plataformas que pudessem viabilizar nossa estratégia,

adotamos o Moodle, principalmente por ser um projeto de *software* consolidado, desenvolvido em código livre, com uma grande e ativa comunidade de desenvolvedores. E a associação da grande comunidade de desenvolvedores, a grande comunidade de usuários (quase trinta milhões cadastrados (Moodle.org, 2009b)) e as várias versões lançadas trouxe um grande grau de confiabilidade para que se fizesse uso do Moodle como ponto inicial de reuso de código.

A partir de uma experiência anterior na modificação e extensão das funcionalidades do Moodle (Santos *et al*, 2007), definiu-se as possíveis estratégias de desenvolvimento, onde o maior desafio seria reusar a maior quantidade possível de código sem que isto alterasse as características pretendidas para o MOrFEU, ou seja, que não herdasse características indesejáveis do Moodle, como a estrutura de cursos, grupos e ferramentas de comunicação.

Essas últimas características citadas são indesejáveis pelo fato de não se adequarem a arquitetura proposta do MOrFEU. Porque o Moodle é centrado em cursos, que são os espaços compartilhados onde as atividades são organizadas e disponibilizadas, sendo que as outras unidades dependem da definição destes. Como os grupos, que são sub-organizações das pessoas de um curso, e a partir da versão 1.9 o Moodle ganhou a funcionalidade de *groupings*, que são agrupamentos de grupos. Por sua vez, as ferramentas de comunicação e interação (como *Fórum*, *Wiki*, *Chat*, *Entrega de Atividades*, *Calendário*) são altamente dependentes da estrutura de *Cursos+Grupos+Groupings* definida, ou seja, elas só existem dentro desta organização do Moodle. Por exemplo, um fórum pode ser definido para todos os grupos de um *grouping* definido dentro de um curso, de forma que as pessoas de cada grupo desse *grouping* possua seu espaço reservado nesse fórum.

Assim, um reuso dessas partes do código do Moodle implicaria em replicar toda essa estrutura organizacional, o que acarretaria na descaracterização do conceito do MOrFEU em

relação ao seu protótipo. No entanto, ainda sim, desejou-se reutilizar máximo possível de códigos do Moodle.

4.2 Reusando o Código do Moodle

O Moodle é formado por um núcleo principal constituído pelo gerenciamento de usuários, de cursos, de grupos e do próprio sistema. E possui uma série de bibliotecas com os mais diversos procedimentos e funções para apoiar toda a construção do sistema. E os componentes que podem ser separados e adicionados são de dois tipos, blocos (*blocks*) e módulos (*mods*).

Alterar o núcleo do sistema não é uma boa opção, pois este está sempre sendo em desenvolvido e atualizado, devido ao processo de evolução natural do Moodle. Os blocos, por sua vez, são elementos da página de cursos no Moodle, porém o MORFEU se baseia em uma estrutura diferenciada, o que implica que desenvolver um bloco não seria adequado, tampouco.

Outra opção seria utilizar as bibliotecas e criar todo o sistema, no entanto, construir um **módulo** traria mais vantagens. Além disso, ainda seria possível aproveitar uma parte do núcleo do sistema, o gerenciamento de usuários e o gerenciamento do sistema. Isto não interferiria no processo de desenvolvimento do núcleo do Moodle, podendo este módulo ser instalado em versões posteriores do Moodle.

4.3 Detalhes e Características da Implementação de um Módulo

Quando se criou o módulo para o protótipo do MORFEU-UFAM, o seu código-fonte ficou no diretório “morfeu” dentro do diretório “mod” do Moodle. Esse diretório contém alguns elementos principais de interação com o sistema do Moodle:

- O arquivo “**version.php**”, contendo: o número da versão deste módulo; a versão do Moodle compatível com este módulo; e a frequência com que o *script* “cron.php”¹⁶ irá ser executado para este módulo.
- O arquivo “**mod.html**”, que deve conter um formulário de adição de novas instâncias deste módulos a um curso. Mas no caso do MORFEU, ele não tem instâncias nos cursos, pois não tem ligação alguma com os cursos do Moodle. Portanto, este arquivo contém apenas uma mensagem dizendo que o MORFEU já está instalado.
- O diretório “**db**”, que contém dois elementos principais: um *script* “mysql.sql”, com instruções para o SGBD que são executadas no momento da instalação do módulo (para MORFEU foi definido apenas as instruções para o MySQL¹⁷, mas o Moodle ainda dá suporte para PostgreSQL¹⁸, Oracle¹⁹ e MS SQLServer²⁰); e o *script* “upgrade.php” que define procedimentos que devem ser adotados quando há uma modificação na versão do módulo. E para instalar ou atualizar um módulo basta acessar num navegador Web o endereço “http://domínio.../moodle/admin/index.php”.
- O diretório “**lang**”, que possui arquivos com as *strings* em várias línguas para o suporte multilíngüe do Moodle. Inicialmente, o MORFEU só dá suporte para o inglês, que é o padrão, e o português brasileiro.
- O arquivo “**lib.php**”, que é a biblioteca de funções do módulo. Neste *script* está contido o procedimento “morfeu_cron”, que é executado quando chamado pelo *script* “cron.php”.

¹⁶ O *script* “cron.php” é executado periodicamente executando tarefas do sistema e de módulos instalados, como, por exemplo, *backups*, envio de *e-mails* e processamento de estatísticas.

¹⁷ <http://www.mysql.com/>

¹⁸ <http://www.postgresql.org/>

¹⁹ <http://www.oracle.com/>

²⁰ <http://www.microsoft.com/sqlserver/>

Deste modo, o MOrFEU agrega características importantes que incrementam sua robustez, como: uma instalação e atualização simples, o que facilita sua evolução; funciona com vários tipos de SGBDs; funciona em várias línguas; e pode realizar tarefas de autogerenciamento, como *backups*.

Além disso, o MOrFEU faz uso de funcionalidades e bibliotecas do núcleo do Moodle:

- Incluindo o *script* “**config.php**” encontrado no diretório raiz do Moodle (utilizando a função do PHP “*require_once*”, por exemplo), automaticamente são incluídas as bibliotecas-padrão do Moodle, a variável “\$CFG” com as definições de configuração do sistema, a sessão é iniciada e o banco de dados é conectado.
- Possui funções de abstração de passagem de **parâmetros de entrada** nas páginas, podendo estes serem parâmetros obrigatórios (“*required_param*”) ou parâmetros opcionais (“*optional_param*”), que possuem um valor padrão se não estiver definido. E não importa o método de envio destes parâmetros, se “GET” ou “POST”.
- Em uma página pode ser obrigatório o *login* do **usuário**. Desta maneira, pode-se aproveitar todo o gerenciamento de usuários do Moodle: *login*, *logout*, registro de novos usuários, perfil do usuário (dados e informações fornecidas pelo próprio usuário), etc. E a variável de sessão “\$USER” contém os dados do usuário *logado* que está acessando aquela página.
- O Moodle possui um sistema de temas de **apresentação**. Para utilizar estes temas, basta incluir um cabeçalho no início da página e toda a página fica com o tema que foi configurado para o Moodle. Há a possibilidade, também, de um rodapé da página.
- Nesse cabeçalho que pode ser acrescentado, há um lugar para os **links de navegação**, que são definidos a cada página.
- Há um sistema de *log*, de **registro de informações** de utilização do sistema, e para utilizá-lo basta fazer a chamada do procedimento “*add_to_log*”.

- Quando se está trabalhando com formulário HTML que possui um campo de texto “textarea” é possível fazer uso de um **editor de HTML** do tipo WYSIWYG. Desta maneira, o usuário pode editar textos com formatação, abstraindo a marcação HTML atribuída a este texto. Ou seja, o usuário pode dar cor ao texto, mudar o tipo e o tamanho da fonte, desenhar tabelas, inserir figuras, etc.
- No Moodle, associado a cada curso há um gerenciador de arquivos e diretórios. Como se pode ver na Figura 5, é desejável esse tipo de gerenciamento de arquivos, com a biblioteca de mídias. Logo, tentou-se aproveitar mais esta parte do Moodle. Para tal, o MOrFEU ficou associado a um curso, definido no arquivo de configurações, para que se pudesse utilizar os arquivos deste curso. Outro problema era que somente os tutores do curso podem navegar, inserir ou alterar os arquivos, então foi preciso trocar o papel de novos membros do curso para tutores. Assim, todos os usuários do MOrFEU poderiam gerenciar os arquivos. Porém, ainda resta um pequeno problema que não afeta tanto o protótipo, mas deve ser corrigido em versões posteriores: todos manipulam um mesmo diretório de arquivos.
- Como foi dito, quando se faz a inclusão do arquivo “config.php”, é feita também a conexão com o **banco de dados**. E há uma camada de abstração do banco de dados que permite fazer consultas e alterações no banco, sem levar em conta qual SGBD se está usando.

Assim, foram utilizados da arquitetura do Moodle apenas a infra-estrutura, o gerenciamento de usuário e o gerenciamento de arquivos de um curso. No entanto, este último deve ser retirado em versões posteriores, já que ele não atenderá todos os requisitos para esta funcionalidade no MOrFEU.

4.4 Funcionalidades do Protótipo

Depois de descritas as vantagens e detalhes de como o MORFEU foi desenvolvido como um módulo do Moodle, passamos a descrever como e quais idéias iniciais foram implementadas no protótipo. Aqui, descreve-se a evolução e pequenas alterações que foram realizadas naquele modelo inicial (ver Figura 5), representando apenas o que foi implementado na versão corrente do protótipo.

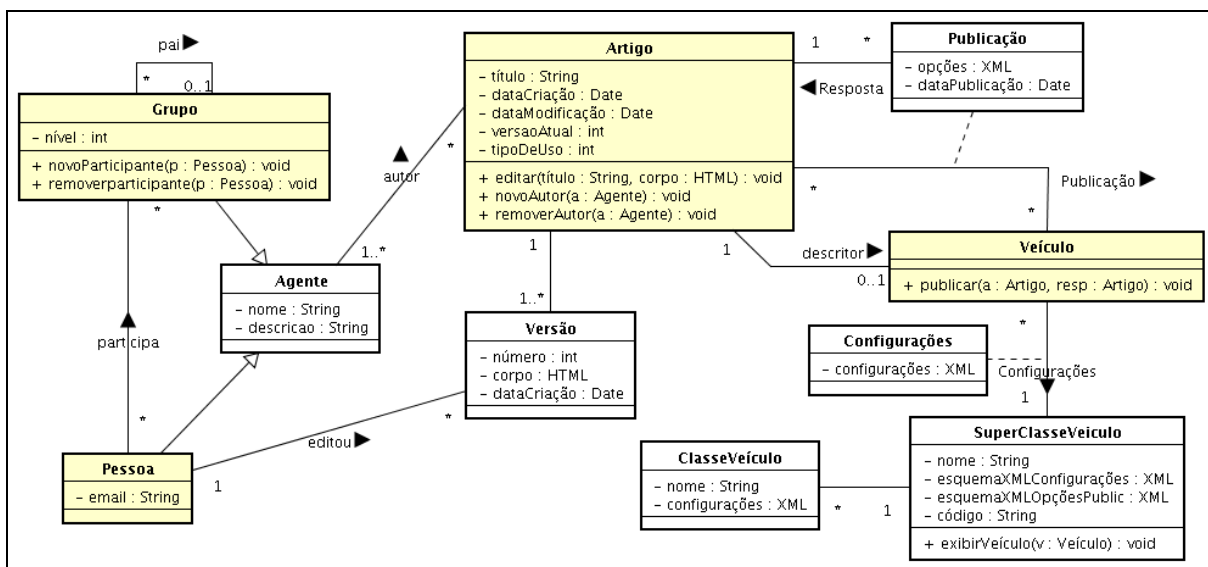


Figura 6. Esquema de Dados do protótipo do MORFEU-UFAM.

A Figura 6 representa o Diagrama de Dados em Diagrama de Classes UML. Comparando com a modelagem inicial (Figura 5), há algumas pequenas mudanças, mas as unidades principais de dados continuam: Pessoas, Grupos, UPIs e Veículos de Comunicação.

Aqui, as UPIs são chamadas Artigos. Isto se refere ao início da construção do protótipo, quando este item possuía o nome de Artigo. No entanto, mais tarde, adotou-se o nome de Unidade de Produção Intelectual (UPI), por ser uma denominação mais adequada. Porém, como o protótipo já estava em grande parte implementado, manteve-se este nome. Assim, o item com nome Artigo possui a mesma semântica do item anteriormente denominado UPI. Isso será retificado em versões posteriores.

As unidades Pessoa e Grupo, nesta modelagem, participam em uma generalização, ambas são subclasses de Agente. Isto porque um grupo deve ser tratado como uma unidade

dentro de um ambiente virtual, agindo de forma particular, assim como uma pessoa (Stahl, 2005). Esta generalização ainda pode agregar outro tipo de agente, um agente artificial, ou agente de *software*, que pode atuar em conjunto com as pessoas e grupos, auxiliando-os e trabalhando na gestão de conhecimento. No entanto, neste protótipo, não foi implementado nenhum agente deste tipo, ressaltando que isto foi apenas preparado para uma futura intervenção nesta direção.

Assim, um autor de um Artigo pode ser um Grupo ou uma Pessoa. O Artigo é um texto com marcações HTML para representar as formatações. Agregado a um artigo estão suas versões, que é sempre editada por uma Pessoa, gerando uma nova versão. Os arquivos digitais não estão representados nesta modelagem, pois foi utilizado o gerenciamento de arquivos do Moodle, e este é muito limitado, como já foi dito na seção 4.3 *Detalhes e Características da Implementação*.

Todo Veículo precisa de uma descrição, que pode ser feita por um Artigo. Desta maneira foi definido um tipo de artigo especial, o Artigo Descritor, que representa um Veículo. Os autores deste artigo serão, conseqüentemente, os autores do Veículo. Como todo Veículo é formado por artigos que são publicados em resposta a outros artigos, este Artigo Descritor é o primeiro artigo a ser publicado e é uma exceção a esta regra. Assim, é evidente que os artigos publicados em um veículo compõem uma árvore, com o Artigo Descritor como sua raiz.

Para os Veículos e as Classes de Veículos, foi adotado um esquema diferente do apresentado no esquema conceitual, mas pretende-se obter o mesmo resultado. Cada veículo está associado a uma Superclasse de Veículo. Esta contém o esquema de configurações de um veículo desta classe e o esquema das opções que um artigo publicado num veículo deste tipo pode ter. Essas configurações são todas feitas em XML.

Uma Classe de Veículo é um tipo de veículo pré-definido. Mas propositadamente não está associada diretamente a um Veículo. Isto é para manter a flexibilidade. Tomar-se-á um exemplo para melhor entendimento deste ponto. Como será visto a seguir, está definida a Superclasse de Veículos Listagem, que modela ferramentas como Fórum, Blog e Chat. Suponha agora um veículo que foi inicialmente criado como um Fórum, então ele é parte da Classe de Veículos Fórum, que também está definida. Então, decide-se tornar este veículo em um Chat que é organizado de forma parecida com um Fórum, hierarquicamente. A partir daí, este veículo deixa de pertencer à Classe de Veículos Fórum, passando a fazer parte de uma outra Classe de Veículos, que ainda não está definida. As configurações feitas em XML permitem este tipo de flexibilidade, e ainda mais, permitem uma flexibilidade de mudança de Super Classes de Veículos, mas mantendo a estrutura dos artigos publicados neste veículo.

Na Figura 7, está a tradução do esquema de dados em diagrama de classes para um esquema de banco de dados relacional, que é tipo de banco de dados que o Moodle trabalha. Este esquema foi feito com o auxílio do DBDesigner²¹ e feito especificamente para MySQL.

A maioria das traduções é feita de forma direta, assim como descrito na literatura. Mas alguns detalhes são importante notar:

- A tabela “mdl_user” representa a classe Pessoa, e já é existente no Moodle. Nela não foram descritos todos seus campos, apenas os mais importantes.
- A relação “morfeu_agent” não é real, na implementação, ela é uma *view* que agrega todas as informações de pessoas e de grupos. O campo *name* é o nome de grupo ou o de usuário, e idem para *description*. O campo *fullname* é o nome completo do usuário (*firstname+lastname*) ou o caminho completo da hierarquia do grupo, todos os grupos desde seu grupo raiz até ele, separados por barra. Há ainda o campo *agenttype* que descreve qual o tipo de agente a que se refere, 1 se pessoa e 2 se grupo.

²¹ <http://www.fabforce.net/dbdesigner4/>

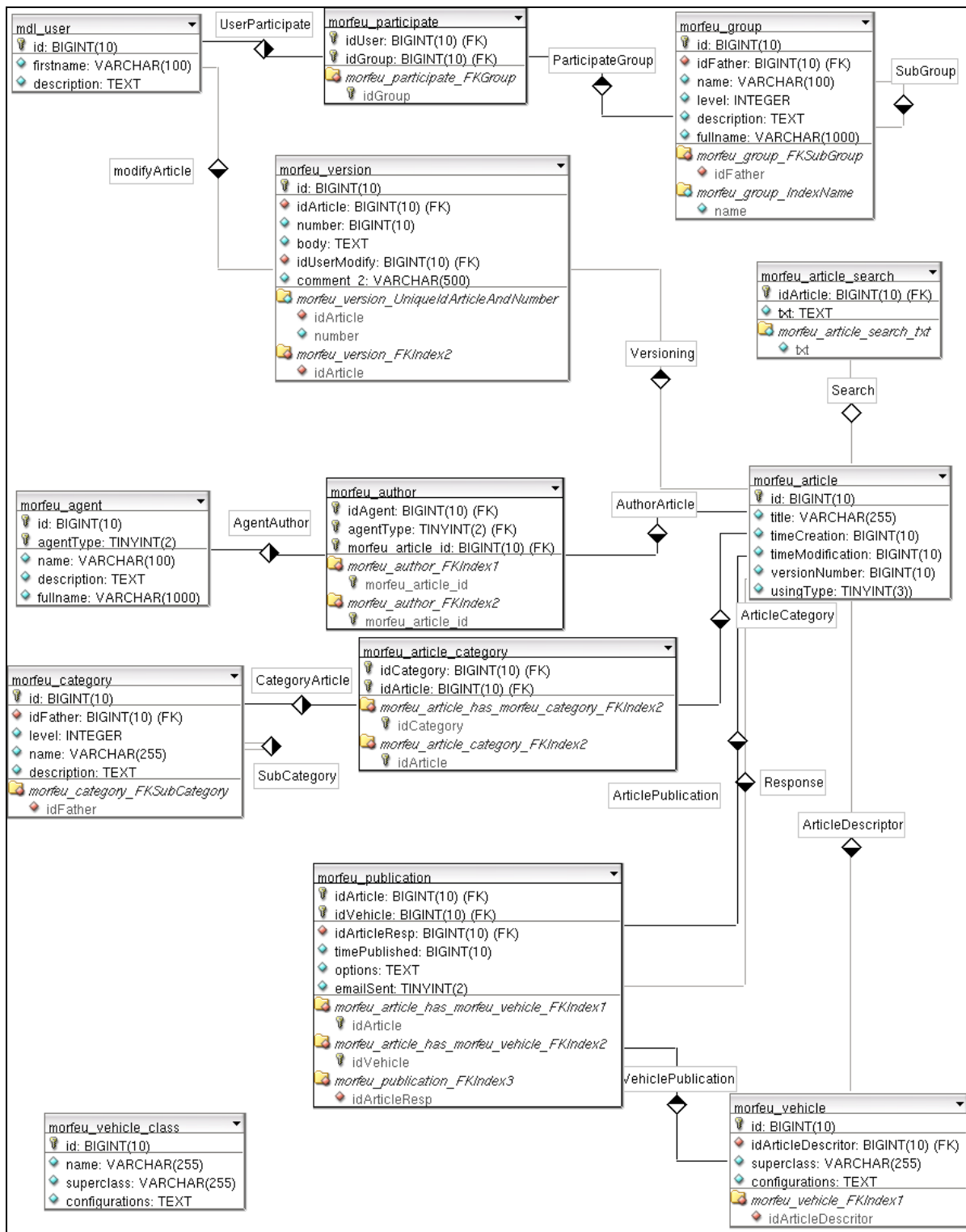


Figura 7. Esquema do Banco de Dados do MORFEU-UFAM.

- Há uma tabela separada de “morfeu_article”, a tabela “morfeu_article_search”. Isto acontece porque todas as tabelas, com exceção desta, são do tipo *InnoDB*, que permite integridade referencial no MySQL, e esta é do tipo *MyISAM*. Este último tipo de tabela permite consultas do tipo *full text*, ou seja, consultas por palavras-chaves, como

numa máquina de busca. A tabela “morfeu_article_search” guarda, então, o texto puro, sem as marcações HTML do artigo, permitindo assim que seja feita uma pesquisa de artigos.

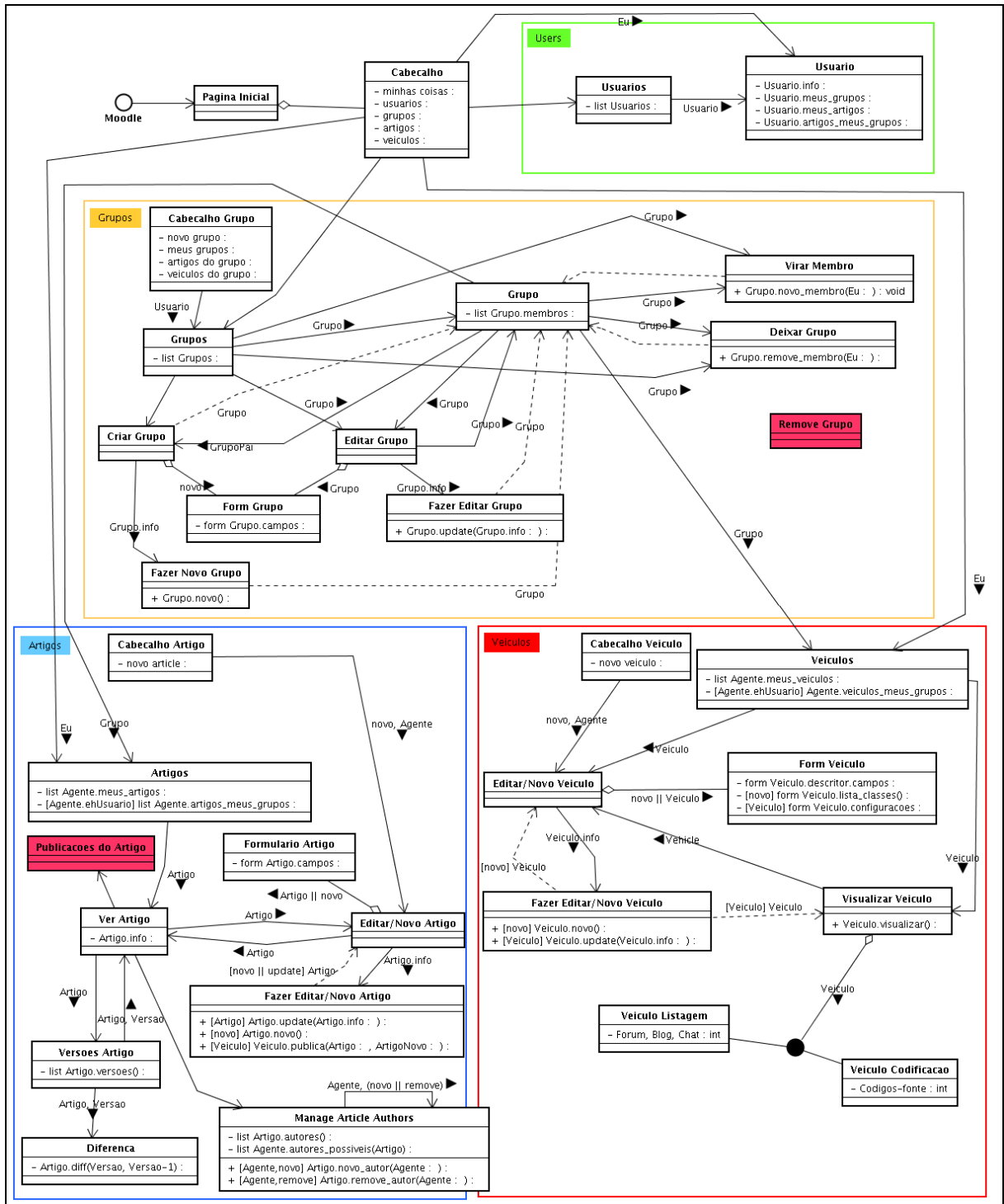


Figura 8. Diagrama de Navegação do protótipo construído do MORFEU-UFAM.

Mas a questão mais importante desta tradução recaia sobre os Veículos. Uma Superclasse de Veículo é a unidade responsável pelo funcionamento do veículo. Ela necessita definir como um veículo será exibido, quais as regras de publicação de artigos, quem é responsável por essa publicação, etc. Assim sendo, as Superclasses de Veículos foram implementadas como *scripts* PHP. Cada Superclasse é representada por um arquivo com o seu nome. Por exemplo, a superclasse Listagem possui um arquivo “listing.php” (escrito em inglês para manter o padrão da implementação). Este *script* é uma biblioteca de funções, e precisa implementar uma função que vai ser chamada no momento de se visualizar o veículo. Deve implementar também as outras “páginas” de sua navegação. Todas as “páginas” serão referenciadas a uma só, mas deveram ser passados parâmetros para determinar qual a página que se está querendo acessar.

Na Figura 8, tem-se diagrama de navegação do protótipo construído do MOrFEU. Para melhor entendimento, ele está dividido em seções: Usuários, Grupos, Artigos e Veículos. Não foi utilizada uma notação usual para diagramas de navegação de páginas dinâmicas na Web, como WebML (Ceri *et al*, 2000) ou UWE (Hennicker&Koch, 2001). Isto se deve ao fato de que esses modelos citados não atendem a uma necessidade de representação desse diagrama, quando desenvolvendo o protótipo. Desejava-se uma representação em mais baixo nível, representando quais as informações de formulários e/ou de *links* que trafegam de uma página para outra e qual o processamento é realizado em cada página. O WebML e UWE são modelos de mais alto nível, e possuem até ferramentas CASE para geração automática de código, mas não seria possível ser feito devido ao fato de grande parte do processamento das informações no protótipo do MOrFEU fica a cargo do *script* de criação da página, em oposição ao que ocorre no WebML e UWE, que modelam o processamento das informações como sendo feito em sua maior parte pelo Sistema Gerenciador de Banco de Dados (SGBD).

Neste diagrama de navegação, as caixas representam as páginas, divididas em três partes: a primeira parte contém o nome da página; a segunda parte descreve o conteúdo da página; e a terceira parte descreve o processamento realizado por essa página. As setas representam a passagem de uma página para outra, podendo ser através de *links* ou formulários, não há diferença no diagrama. Acompanhadas dos *links*, estão as informações que são passadas de uma página para outra. As setas pontilhadas são redirecionamentos instantâneos de página. As linhas com losango na ponta representam inclusões de páginas, a página do lado do losango inclui a outra página, em PHP são funções como “include” e “require”. Alguns caracteres têm funções especiais nesse diagrama: os colchetes representam condições, ou seja, o *link* só aparece ou uma operação é realizada, se a condição for satisfeita; o *pipe* | indica uma disjunção, um “ou”; e a vírgula indica uma conjunção, um “e”. Por fim, um círculo cheio pintado de preto indica um “ou exclusivo”. Maiores detalhes podem ser encontradas no *Anexo A: Diagrama de Navegação*.

No diagrama da Figura 8 ainda existe uma observação especial sobre o Cabeçalho, que ele está presente em todas as páginas do MORFEU. Os Cabeçalhos de Grupo, Artigo e Veículo estão presentes nas respectivas páginas de suas seções.

As páginas da seção Usuário são cópias das páginas das páginas de usuários do Moodle. Mas a página de visualizar informações do usuário foi acrescida com informações sobre a produção de Artigos do usuário e os Grupos que este participa.

Os Grupos são organizados de forma hierárquica. No entanto, não existe apenas um grupo “raiz”, mas podem existir vários deles. No momento que um usuário vira membro de um subgrupo, ele também vira membro de todos os grupos “ancestrais” deste. Neste protótipo, não foram desenvolvidas funcionalidades avançadas de gerenciamento de participantes do grupo, como adicionar ou remover vários participantes ao mesmo tempo, nem há papéis de coordenação envolvidos no grupo, todos têm a mesma autonomia. O que há

é uma listagem de participantes daquele grupo. Cada pessoa é responsável por participar ou deixar de participar de qualquer grupo.

Os Artigos podem ser de pessoas ou de grupos. Existem funcionalidades de criar um novo Artigo, editar um Artigo, gerenciar autores e gerenciar versões, com uma ferramenta de verificar diferenças. Como o SGBD dá suporte, há possibilidade de se fazer uma busca por palavras-chave de artigos. Se não houvesse este suporte, seria preciso implementar um sistema de busca para se obter esta funcionalidade.

Os Veículos são tratados de forma diferenciada. Existem funcionalidades como criar um novo veículo, editar informações e configurações, exibir uma listagem de veículos disponíveis, que podem ser de pessoa ou de grupo. Mas o principal acontece quando se “visualiza” um veículo, sendo toda navegação e regras definidas pela superclasse de veículo. Isto vai ser será abordado mais detalhadamente na próxima seção.

4.5 Superclasses de Veículos de Comunicação

Nesta seção, serão detalhadas as duas Superclasses de Veículos desenvolvidas. Essas superclasses são *scripts* PHP dentro do diretório “morfeu/vehicles/superclass”. Cada *script* desse tipo é uma biblioteca de funções, com a obrigação de implementar apenas uma função, a função principal de visualizar o veículo, que é chamada na página “Visualizar Veículo” (ver Diagrama de Navegação na Figura 8). Esta página passa alguns parâmetros para esta função, mas esta função pode tomar qualquer parâmetro de entrada da página acessando variáveis de ambiente do PHP, como “\$_POST” e “\$_GET”.

A página de uma Superclasse é apenas uma, a página “Visualizar Veículo”, mas podem ser enviados parâmetros que fazem a diferenciação lógica de páginas. Ou seja, quando certo parâmetro é enviado para a página “Visualizar Veículo”, quer dizer que o destino é uma certa página daquela Superclasse. Veremos, a seguir, que a Superclasse Codificação funciona desta maneira.

As configurações de cada veículo se encontram em formato XML. E cada Superclasse manipula esta configuração como lhe convier. Atualmente, não está sendo verificado se o estilo do XML utilizado para estas configurações está de acordo como especificado pela sua Superclasse.

4.5.1 Superclasse Listagem

Esta foi a primeira Superclasse criada, com objetivo de generalizar ferramentas bem semelhantes em sua estrutura e que existem em muitos ambientes virtuais e na Web em geral. São elas o Fórum, o Blog e o Chat. São ferramentas de comunicação, indispensáveis para trabalho ou aprendizagem colaborativos baseados em ambientes virtuais na Web.

A estrutura desses três, se modelados como Artigos do MOrFEU é bem semelhante e simples: Artigos que respondem a Artigos, formando uma árvore de Artigos. E é claro, o artigo raiz desta árvore é o Artigo Descritor do Veículo. A Figura 9 representa esta estrutura. Não há limites para esta estrutura, mas as ferramentas que ela representa necessitam de alguns limites:

- Um **Fórum** comum, levando em consideração apenas uma discussão, atua sem limites no número de níveis da árvore ou do número de artigos por nível. Ou seja, é a mesma estrutura descrita na Figura 9.
- Mas um **Blog** é um pouco diferente. Seu autor do Blog publica suas mensagens e estas mensagens são alvo de comentários das outras pessoas. Assim sendo, sua estrutura possui número limitado de níveis de resposta, como demonstrado na Figura 10. O Blog ainda tem outra característica diferente do Fórum, as mensagens mais recentes aparecem no topo da página.
- Um **Chat**, no entanto, possui uma estrutura mais solta, não dependendo da estrutura de ligações de resposta. Ele depende da data e da hora em que foram publicados os Artigos. Assim, ele pode possuir dois tipos de estrutura: todos os artigos respondendo

ao artigo descritor; ou cada artigo respondendo ao último artigo publicado. Mas qualquer outra estrutura ainda pode ser reorganizada pelo momento em que foram publicados os artigos.

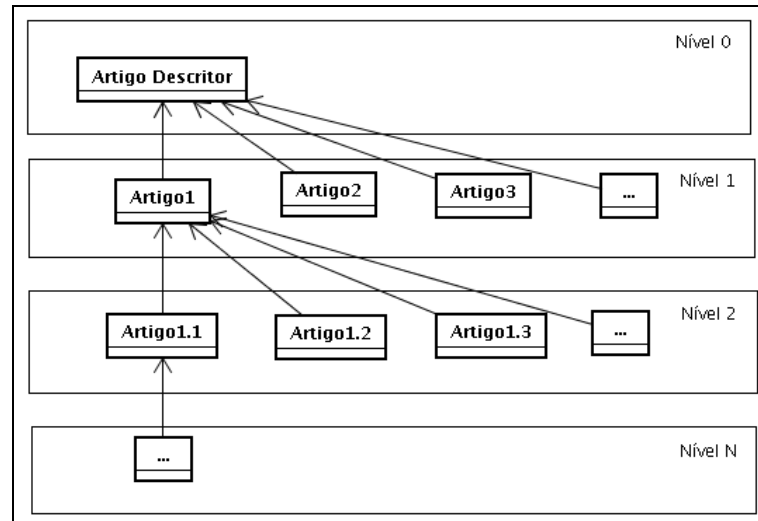


Figura 9. Esquema de Dados de um Veículo Listagem.

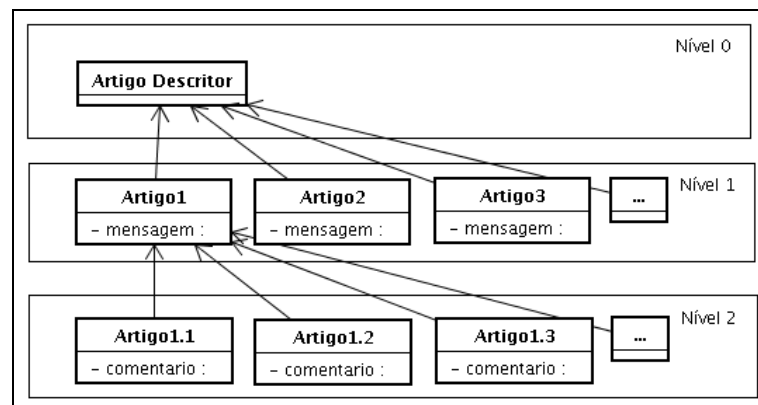


Figura 10. Esquema de Dados de um Veículo da Classe Blog.

Observa-se que esses tipos de ferramentas baseiam-se apenas na resposta direta de outros artigos já publicados. Logo, ter-se-á um diagrama de navegação simples, apresentado na Figura 11.

Mas a idéia principal era generalizar esses tipos de ferramentas, de forma a torná-las mais flexíveis, sendo este último o objetivo geral do MORFEU. Então a Superclasse Listagem não se resume a estas três ferramentas, ou a essas três Classes de Veículos. Assim, foram definidas configurações que possibilitassem esta flexibilidade. Entre elas estão configurações de quantidade de níveis de resposta, forma de ordenação dos artigos, permissões de

publicação e dinâmica da página. O Quadro 2 apresenta um esquema XML das configurações de veículos da superclasse Listagem. Este não está muito formal para efeitos de melhor compreensão. Para maiores detalhes, ver Quadro 6 no Apêndice II.

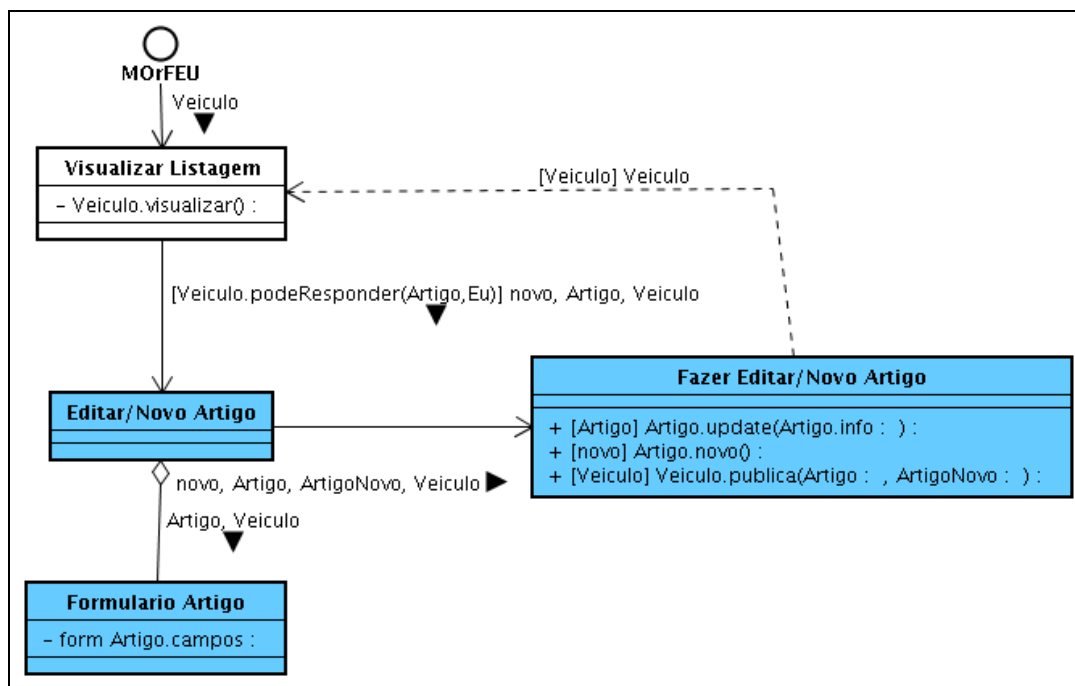


Figura 11. Diagrama de Navegação da Superclasse de Veículo Listagem. Em azul estão as páginas que já existiam e foram reutilizadas.

Assim, um chat pode virar um fórum, um fórum pode ser usado de forma síncrona (tendo a estrutura de organização das mensagens, mas para ser utilizado como um chat), e outras muitas possibilidades. Um exemplo disso é a definição de um Fórum para um método de aprendizagem colaborativa chamado Controvérsia Acadêmica (Mendonça *et al*, 2003). Naquele trabalho, há a necessidade de se definir um Fórum que possua, para cada tema a ser discutido, três tópicos, “Concordo”, “Discordo” e “Depende”. Para fazer isso no MORFEU, basta criar um Veículo da Classe Fórum com o tema a ser discutido. Cria-se então, no nível 1, três respostas, “Concordo”, “Discordo” e “Depende”. A partir deste momento, basta alterar as configurações para que somente haja respostas a partir do nível 2 da árvore de Artigos. Idealmente, este tipo de procedimento que se acabou de ser descrito poderia vir vinculado à Classe de Veículo, para que seja realizado no momento de criação do Veículo.

```

<configurations>
  <treeOrder>
    <criteria>{ tree | date }</criteria>
    tree: formato de árvore;
    date: ordenado pela data de publicação.
    <order>{ ASC | DESC }</order>
    ASC: do menor para o maior;
    DESC: do maior para o menor.
    <printType>{ indented | flat }</printType>
    indented: imprime de forma tabulada, formando a árvore;
    flat: todos são impressos no mesmo nível (mais a esquerda na página).
  </treeOrder>

  <publication>
    <newArticle>{ yes | no }</newArticle>
    O novo artigo que será publicado deve ser necessariamente novo?
    <editArticle>{ yes | no }</editArticle>
    O artigo pode ser editado?
    <removeArticle>{ yes | no }</removeArticle>
    O artigo pode ser removido? Ou seja, pode ser despublicado?
  </publication>

  <page>
    <allowView>{ author | all | none }</allowView>
    Quem pode visualizar o veículo?
    <header>{ yes | no }</header>
    A página do veículo tem o header do Moodle?
    <refresh>{ no | N }</refresh>
    Qual a taxa de atualização da página (em segundos)?
    <css>[STRING]</css>
    CSS específico para o veículo.
  </page>

  <permission>*
    <allowPublish>{ author | all | none }</allowPublish>
    Quem pode publicar?
    <treeLevel>{ all | N | fromN | untilN | fromNuntilM }</treeLevel>
    Em que nível esse alguém pode publicar?
  </permission>
</configurations>

```

Quadro 2. Configurações em XML de Veículos Listagem. Em azul e itálico, as descrições, e em vermelho e tachado, o que não foi implementado.

4.5.2 Superclasse Codificação

Esta Superclasse define ferramentas como AAEP (Almeida Neto, 2007), ou seja, interpretadores *on-line* de códigos-fontes. No caso do AAEP, ele era um ambiente virtual separado, tendo que lidar com todas as preocupações anteriormente citadas: usuários, banco de dados, etc. Ainda tinha que lidar com o controle de versões, onde utilizava um sistema separado, e com ferramentas de consulta de diferenças, quando utilizava o comando “diff” do Linux.

Tirando vantagem da arquitetura do MOrFEU de resposta de artigos num veículo e de que já existe o controle de versões, só restou desenvolver o interpretador, que foi feito reutilizando parte do código do AAEP. Dessa forma, foi desenvolvido um interpretador para a linguagem Haskell utilizando o Hugs²², mas há ainda possibilidades de se fazer interpretadores para outras linguagens. Dessa forma, foi feito um interpretador para a linguagem Prolog, utilizando-se GProlog²³, mas o funcionamento deste último não ficou muito bom.

Na Figura 12, é mostrado como a estrutura do MOrFEU é utilizada em um Veículo do tipo Codificação. No nível 0 está o Artigo Descritor com a descrição do problema a ser resolvido. No nível 1, ficam os códigos-fontes das respostas desenvolvidas. E no nível 2, ficam guardadas as tentativas de se executar o *script* desenvolvido, que o AAEP não guardava. Além disso, foi utilizado o Geshi²⁴ para colorização dos códigos-fontes, que dá suporte para colorização de códigos em diversas linguagens de programação.

O Diagrama de Navegação é mostrado a seguir na Figura 13. Têm-se basicamente cinco funcionalidades: criar uma nova solução, editar uma solução existente, executar uma solução e ver as versões e suas diferenças.

Nas configurações, são definidas a linguagem e o interpretador. Além disso, existem dois modos de visualização das soluções: no primeiro, somente são visualizadas as respostas do próprio usuário; e no outro modo, são visualizadas as respostas de todos os usuários. O Quadro 7, no Apêndice II, mostra o esquema de configurações.

Outro exemplo de como toda a arquitetura é flexível é este: inicialmente, é criado um veículo do tipo Codificação para resolução de um problema, de forma individual, com cada aluno fazendo seu código de forma individual. Após isto, este veículo é transformado num

²² <http://www.haskell.org/hugs/>

²³ <http://www.gprolog.org/>

²⁴ <http://qbnz.com/highlighter/>

veículo da Superclasse Listagem, para que haja uma discussão ou comentários dos códigos e execuções que haviam sido desenvolvidos.

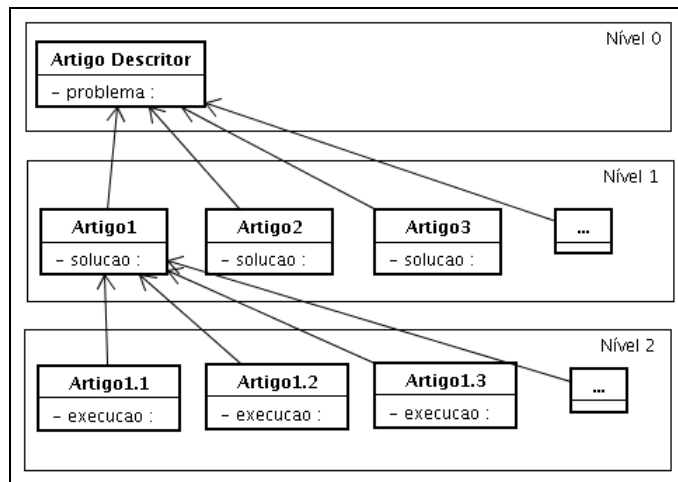


Figura 12. Esquema de Dados de um Veículo Codificação.

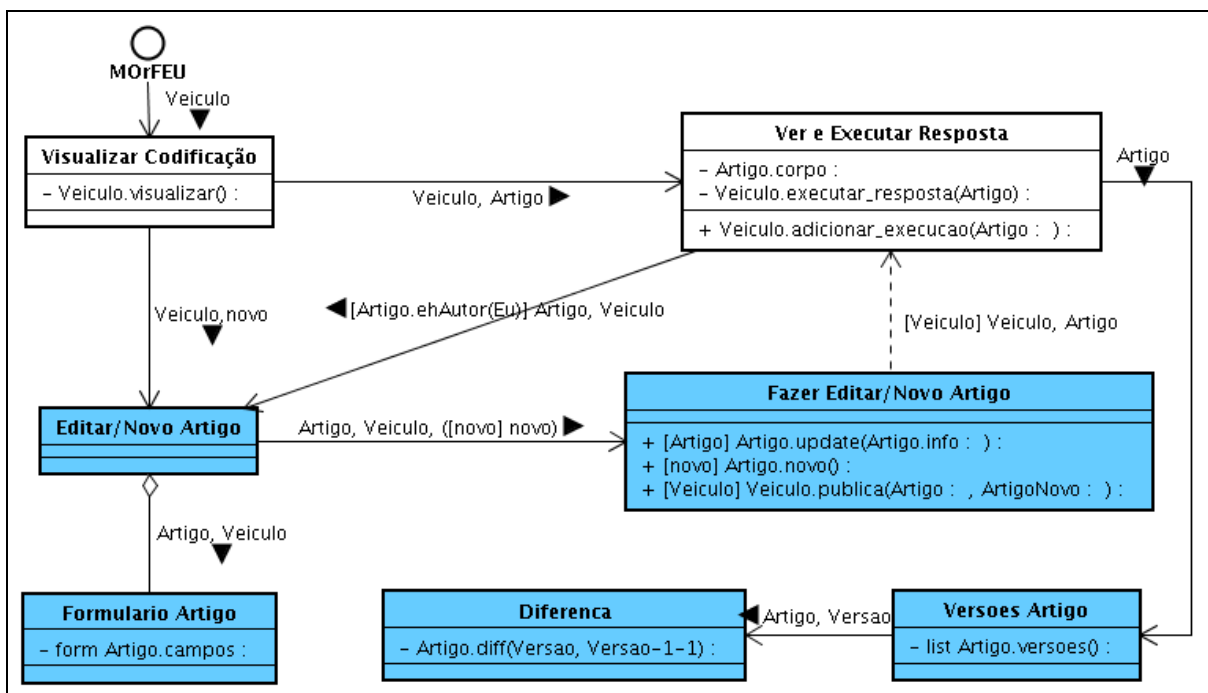


Figura 13. Diagrama de Navegação da Superclasse de Veículo Codificação. Em azul estão as páginas que já existiam e foram reutilizadas.

4.6 O Protótipo em Funcionamento

O protótipo está funcional no endereço <<http://colabweb.ufam.edu.br/morfeu>>, e é facilmente instalável em um Moodle de versão 1.9 ou superior. A seguir, na Figura 14, são mostradas algumas de suas páginas funcionando.



Figura 14. Algumas páginas do protótipo funcionando: Meus Artigos (a); Meus Grupos (b); Meus Veículos (c); A Criação de um Novo Veículo (d); Editando as Configurações de um Veículo Blog (e); Visualizando o Veículo Blog (f); Vendo as Versões de um Artigo (g); Vendo as Diferenças entre duas Versões de um Artigo (h).



5 *Discussão*

Neste capítulo, discutimos alguns elementos relacionados à validação conceitual da proposta e do protótipo construído. Uma vez que os pressupostos sobre os quais o MOrFEU foi concebido foram propostos a partir de uma longa experiência em situações reais de uso de AVAs, e seu arcabouço teórico ainda está sendo desenvolvido, buscamos obter subsídios para o processo de refinamento das idéias, caracterizando um processo de desenvolvimento em espiral.

5.1 **Validação Conceitual**

Nesta etapa, fazemos uma avaliação das capacidades do MOrFEU, apresentando cenários que demonstram a necessidade das características de flexibilidade propostas. No entanto, não se trata de cenários de uso específico, a maioria deles são métodos de ensino cooperativo inicialmente desenvolvidos para ambientes presenciais e desejava-se aplicá-los em ambiente virtuais, enfrentando assim dificuldades adicionais.

O primeiro cenário, descrito a seguir, é um método desenvolvido exclusivamente para uso em ambientes virtuais e com recursos de informática, porém não teve sucesso completo em sua aplicação, possivelmente por não dispor de um ambiente virtual adequado no qual fosse inserido.

5.1.1 **Cenário A: Projeto de Aprendizagem**

Este cenário já foi discutido em outro trabalho sobre o MOrFEU, em um fórum nacional (Menezes *et al*, 2008). Trata-se da arquitetura pedagógica chamada “Projeto de Aprendizagem” (Fagundes *et al*, 1999), descrito por Carvalho *et al* (2005) do seguinte modo:

A sistematização desta arquitetura compreende o lançamento de problemas e formulações a partir de suas "Certezas Provisórias" e "Dúvidas Temporárias". Em termos de metodologia, o primeiro passo é selecionar uma curiosidade, que para fins didáticos, denomina-se de "Questão de Investigação". A seguir é feito um inventário dos conhecimentos (sistemas nocionais, ou conceituais dos aprendizes) sobre a questão. Esse conhecimento pode ser classificado em dúvidas e certezas. As certezas para as quais não se conheça os fundamentos que a sustentem são denominadas de provisórias. As dúvidas são sempre temporárias. O processo de investigação consiste no esclarecimento das dúvidas e na validação das certezas.

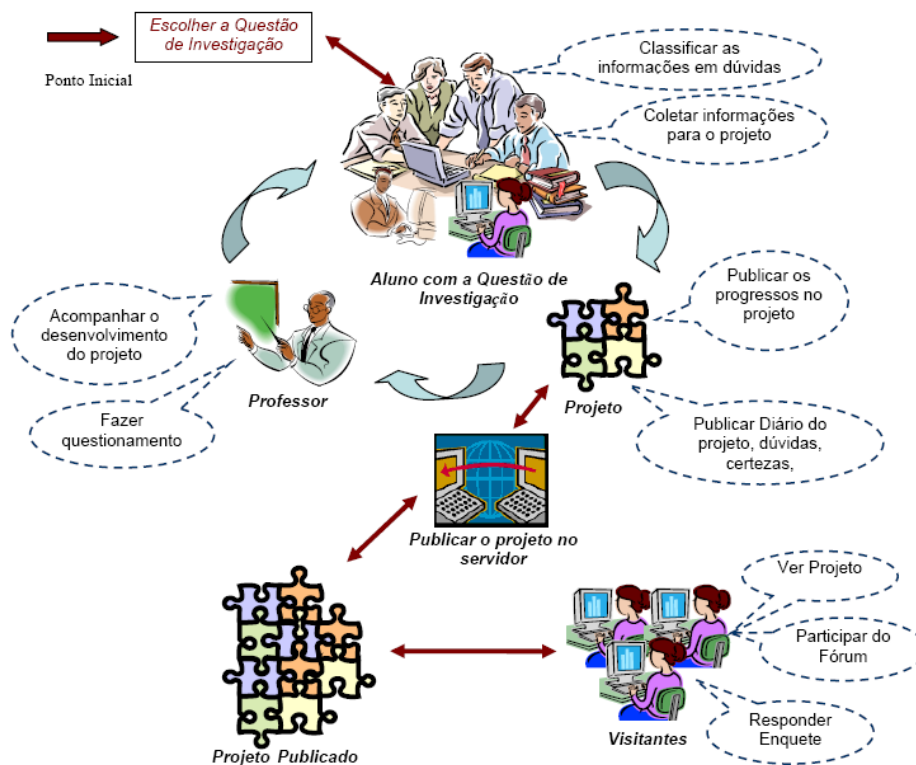


Figura 15. Processo de desenvolvimento de Projetos de Aprendizagem. Fonte: (Monteiro, 2006).

As dificuldades em trabalhar projetos de aprendizagem em ambientes virtuais tradicionais levou os autores à criação de um ambiente virtual específico, o AMADIS (Fagundes *et al*, 2006), (Monteiro, 2006) e (Monteiro *et al*, 2005).

Modelando suas características no MOrFEU, toda a produção relacionada com o desenvolvimento de um projeto pode ser estruturada a partir da composição de UPIs. Cada página de produção do projeto é uma UPI, que possuirá versões gerenciadas pelo ambiente. Os comentários dos professores, as sugestões de colegas, as discussões entre os protagonistas de um projeto de aprendizagem, são modelados através de UPIs. Na Figura 15 apresenta-se o fluxo de autorias e interações, necessário para o desenvolvimento de um projeto de

aprendizagem. Em cada nodo da figura, podemos observar uma nova situação de autoria. No MORFEU, todas estas manifestações são registradas através de uma UPI, que pode estar respondendo a uma outra. Neste sentido, tem-se uma classe especializada do veículo Projeto de Aprendizagem, e o projeto de cada grupo de investigação é uma instância desta Classe.

Na Figura 16, apresenta-se a modelagem de um veículo de comunicação capaz de organizar a autoria cooperativa do sítio de um grupo de trabalho desenvolvendo projetos de aprendizagem. Desta forma, um Projeto de Aprendizagem é concebido como um veículo de comunicação, de construção cooperativa, composto por sub-veículos: Desenvolvimento do Projeto, Diário de Bordo, Fórum de Orientação e Livro de Visitas. No caso do sub-veículo Desenvolvimento do Projeto, podemos observar que ele possui uma coleção de UPIs, onde cada uma delas pode referenciar outras do mesmo conjunto e podem receber comentários através de novas UPIs.

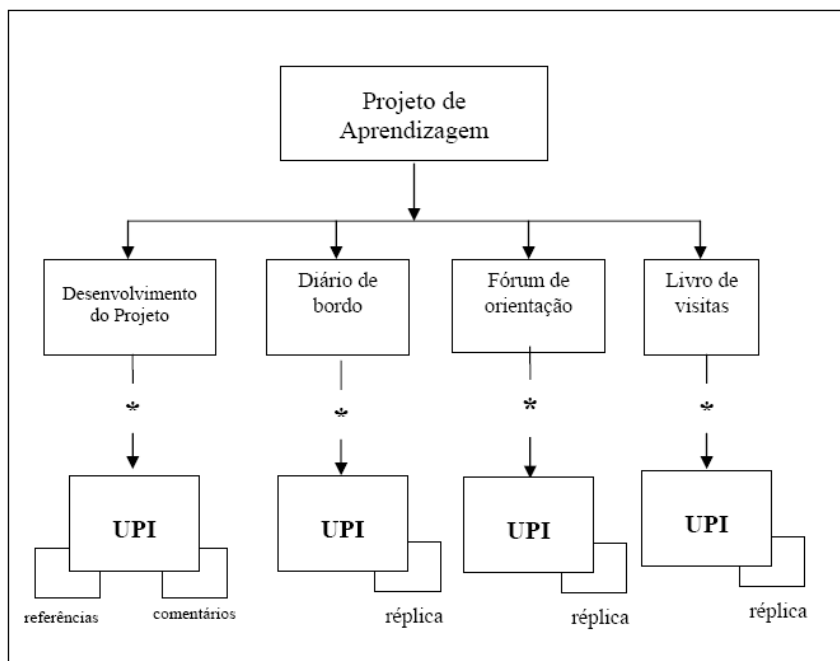


Figura 16. Modelagem do Veículo de Comunicação para Projetos de Aprendizagem. Uma classe especializada que pode ser instanciada para suportar a construção de Projetos de Aprendizagem específicos.

Portanto, o método de Projetos de Aprendizagem pode ser modelado dentro da arquitetura do MORFEU. Ressaltamos a unificação do conhecimento em um único artefato,

assim como sugerido por (Anttila, 2006) (ver a Seção 2.3 Ambientes de Trabalho e Conhecimento).

No entanto, vale mencionar que este cenário não pode ainda ser implementado na versão apresentada do protótipo, já que o protótipo não possui implementadas algumas características como a referência a outras UPIs, somente possui a relação de “resposta”, além de não conseguir agregar sub-veículos para formar um único artefato.

5.1.2 Cenário B: Investigação em Grupo

Investigação em Grupo é um método de aprendizagem cooperativa utilizado em sala de aula, no qual

os estudantes trabalham colaborativamente em pequenos grupos para examinar, experimentar e compreender temas centrais de estudo. A Investigação em Grupo é projetada para lidar com todas as habilidades dos estudantes e experiências relevantes ao processo de aprendizagem. Esse método proporciona meios para que os educadores conduzam o ensino e a aprendizagem na escola que diferem significativamente da instrução tradicional (Silva *et al*, 2002)

Este método é composto de seis estágios, assim como descrito no Quadro 3, e é importante mencionar que possui várias similaridades com o método de Projetos de Aprendizagem. Após uma reflexão sobre essas similaridades, consideramos que Projetos de Aprendizagem é um método mais geral que Investigação em Grupo, e portanto, as estratégias propostas para o primeiro são perfeitamente aplicáveis também para o segundo. É importante ressaltar que este método foi proposto inicialmente para aplicação em ambiente presencial e que se deseja aplicá-lo em um ambiente virtual, e que o Projeto de Aprendizagem parece já ter sido criado para ser utilizado com apoio de recursos de informática e a Internet.

Estágios da Investigação em Grupo
(1) Identificação do tema de pesquisa e organização dos alunos em grupos; <ul style="list-style-type: none"> • Pesquisa de fontes bibliográficas, proposição de tópicos e lista de sugestões; • Associação em grupos de acordo com o tópico de pesquisa escolhido; • Composição dos grupos heterogênea e baseada em interesses; • Professor auxilia na aquisição de informações e facilita a organização.
(2) Planejamento das tarefas de aprendizagem em grupo; <ul style="list-style-type: none"> • Planejamento em conjunto do que deve ser estudado e como deve ser estudado; • Divisão de tarefas entre os componentes do grupo; • Determinação dos objetivos da investigação em grupo.
(3) Execução da investigação pelo grupo; <ul style="list-style-type: none"> • Coleta de informações, análise de dados, e formação de conclusões; • Cada membro do grupo contribui para o esforço do grupo; • Troca, discussão, esclarecimento e síntese de idéias.
(4) Preparação do relatório final; <ul style="list-style-type: none"> • Determinação da mensagem essencial de seus projetos pelos membros do grupo; • Planejamento do relatório e das apresentações pelos membros do grupo; • Comissão de orientação formada pelos representantes de cada grupo para coordenação dos planos para a apresentação.
(5) Apresentação do relatório final; <ul style="list-style-type: none"> • Apresentação realizada para toda a classe de formas variadas; • Envolvimento da audiência ativamente em parte da apresentação; • Avaliação da clareza e os recursos da apresentação pela audiência de acordo com os critérios determinados previamente por toda a classe.
(6) Avaliação dos projetos. <ul style="list-style-type: none"> • Troca de informações sobre o tema, sobre o trabalho e sobre experiências; • Colaboração na avaliação da aprendizagem dos estudantes pelos professores e alunos.

Quadro 3. Os seis estágios da Investigação em Grupo e um esboço de suas principais atividades. Fonte: (Silva *et al*, 2002)

5.1.3 Cenário C: Controvérsia Acadêmica

O método de aprendizagem cooperativa Controvérsia Acadêmica (Johson *et al*, 1999) consiste em debater temas polêmicos, em um primeiro instante, defendendo uma posição favorável, em um segundo instante, uma posição contrária, passando para um terceiro momento quando é feita a síntese das informações e opiniões. Primeiramente foi desenhado para ser utilizado em ambiente presencial, em sala de aula. No Quadro 4 são descritas as etapas deste método.

Etapas da Controvérsia Acadêmica
1. A turma é organizada em grupos de quatro estudantes, os quais são posteriormente divididos em dois pares. Cada par pesquisa sobre a posição designada, organiza suas descobertas em um framework conceitual buscando construir argumentos persuasivos e convincentes para validar a sua posição.
2. Os estudantes apresentam persuasivamente o melhor argumento possível para a sua posição, ouvem cuidadosamente a apresentação oposta, e tentam aprender os dados e lógica sobre os quais eles se basearam.
3. Os estudantes se engajam em uma discussão aberta, continuam a advogar suas posições enquanto tentam aprender sobre a posição oposta. Eles analisam criticamente a evidência e a lógica da posição contrária e tentam refutá-los. Ao mesmo tempo, eles rebatem os ataques sobre suas evidências e lógica em um esforço para persuadir os oponentes a concordarem como eles.
4. Os estudantes invertem as perspectivas e passam a advogar a posição oposta tão sincera, completa, precisa e persuasivamente quanto possível. Para libertar os estudantes de suas antigas convicções, os mesmos devem investir em pesquisa, recorrer às anotações feitas durante os passos 2 e 3 e desenvolver um framework conceitual contendo os melhores argumentos possíveis para validar esta nova posição e persuadir os oponentes.
5. Por fim, os estudantes voltam à composição inicial do grupo, com quatro integrantes, e desenvolvem uma síntese integrando diferentes idéias e fatos em uma única posição. Para fazer isso, os estudantes devem contar com as melhores evidências e raciocínio de ambos os lados. O propósito dual da síntese é chegar à melhor posição sobre o assunto e encontrar argumentos que todos os membros do grupo possam concordar e comprometer-se.

Quadro 4. Estágios do método de aprendizagem cooperativa Controvérsia Acadêmica. Fonte: (Mendonça *et al*, 2002).

Estando interessados em como seria a aplicação deste método em um ambiente virtual, foram levantados os requisitos e desenvolvido um ambiente com esse propósito específico, o de mediar a controvérsia acadêmica (Mendonça *et al*, 2003). Naquele trabalho, foi desenvolvido um fórum, onde só é possível responder a um tópico com um dos três tipos de respostas: “Concordo”, “Discordo” ou “Depende”. Cada tipo de resposta normalmente corresponde a uma etapa do processo.

Foi necessária a construção de um fórum específico para este método, mas isto não seria necessário no MOrFEU, dadas suas características. Pois seria apenas uma pequena modificação nas configurações do veículo de comunicação “Fórum” da seguinte maneira: três tópicos seriam criados no nível 1 da árvore de respostas (“Concordo”, “Discordo” e “Depende”) e somente seria permitido postar mensagens no nível 2 da árvore. Assim teríamos o veículo de comunicação “Fórum para Controvérsia Acadêmica”. Portanto, esta solução já é viável no protótipo desenvolvido. É importante destacar que isso evidencia que o MOrFEU,

não tendo sido criado para atender especificamente a esta necessidade, atende aos requisitos especificados em consequência do aspecto de “flexibilidade” buscado em seu projeto.

5.1.4 Cenário D: Jigsaw

O método de aprendizagem cooperativa Jigsaw (Slavin, 1995) consiste na formação de grupos para estudo de um tema específico, onde grupos especialistas estudam partes isoladas do assunto e, em um segundo momento, são formados outros grupos com cada membro sendo um anterior membro de um dos grupos especialistas, montando assim o assunto todo. As etapas do Jigsaw são melhor descritas no Quadro 5.

A exemplo de outros dois métodos de aprendizagem cooperativa citados neste trabalho, o Jigsaw também foi desenvolvido para ser utilizado em um ambiente presencial, mas se buscou adaptar o método ao uso em ambientes virtuais. Pereira *et al* (2002) sugerem algumas estratégias e ferramentas para tal propósito. Assim, dividiram-se as ferramentas necessárias de acordo com a etapa do método:

- **Introdução:** nesta etapa é necessário prover ferramentas para acesso ao material, como repositórios e lista de discussões sobre este material;
- **Exploração:** neste momento é necessário que haja interação síncrona entre os participantes e ferramentas de produção da síntese, que podem ser um texto;
- **Relato:** os especialistas voltam aos seus grupos e devem compartilhar o conhecimento especializado adquirido. Este compartilhamento pode ser feito de forma mais organizada em um fórum, de forma que dê espaço à discussão dos detalhes.
- **Integração:** nesta última etapa é então produzida uma síntese de todo o assunto, que deve ser feita de forma conjunta e síncrona;

Estágios do Jigsaw
1. Introdução. Nesta fase o professor organiza os grupos <i>Jigsaw</i> , introduz os tópicos, textos, informações ou materiais para ajudar os estudantes no entendimento dos tópicos que vão ser trabalhados e verificar como eles se encaixam com o que foi estudado e como serão importantes no futuro.
2. Exploração. Os estudantes se reorganizam em outros grupos, chamados de especialistas, para estudar os tópicos em maior profundidade. Nesta fase o professor deve utilizar ferramentas para incentivar os alunos a interagir com os outros.
3. Relato e transformação. Os estudantes voltam ao grupo original para explicar os tópicos para os companheiros. Deve-se entender primeiro as partes, para ter uma compreensão melhor do todo.
4. Integração e avaliação. O que foi obtido pelos alunos em grupos pequenos é integrado com as outras pessoas envolvidas no ambiente de estudo. O resultado é então avaliado.

Quadro 5. Estágios do método de aprendizagem cooperativa Jigsaw. Fonte: (Pereira *et al*, 2002).

Na Figura 17 vê-se a representação de um veículo de comunicação para o Jigsaw modelado no MOrFEU, com cada etapa representada por um sub-veículo, e estes últimos compostos por outros sub-veículos. Os veículos Exploração, Relato e Integração podem ter mais de uma instância, uma para cada grupo, de acordo com os grupos de cada etapa. Há ainda uma seta representando uma relação de “referência” entre dois veículos, que acontece quando se possibilita referências às sínteses dos grupos especialistas.

Assim, o Jigsaw pode ser realizado dentro do MOrFEU. Mas como acontece com os Projetos de Aprendizagem, o protótipo atual não dá o suporte para tal, tendo em vista que ele não implementa relações de sub-veículos e de referências.

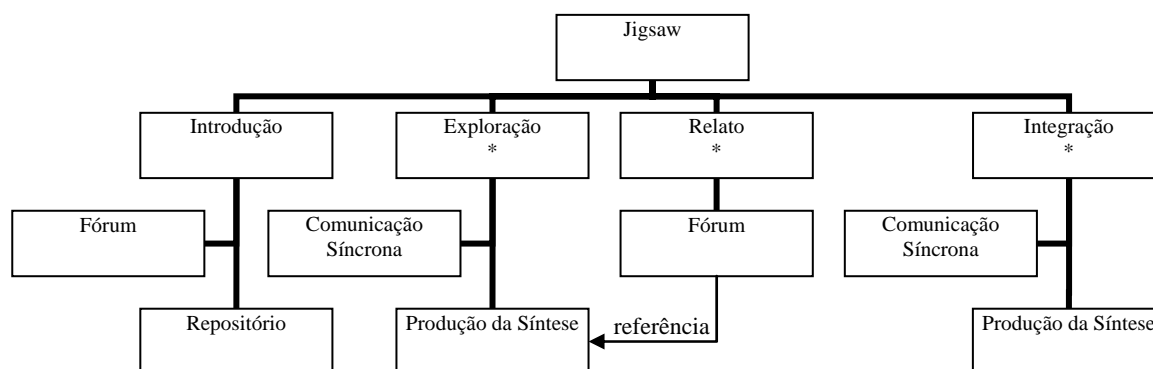


Figura 17. Modelagem do Veículo de Comunicação para o Jigsaw. Sub-veículos de cada etapa compõe o veículo principal, que por sua vez são compostos por outros sub-veículos. Os veículos Exploração, Relato e Integração podem ter mais de uma instância, uma para cada grupo.

5.1.5 Síntese dos Cenários

Vimos que o MOrFEU foi capaz de modelar veículos de comunicação para suprir os requisitos dos cenários apresentados, e mesmo tratando-se de uma prospecção inicial, é possível perceber sua adequação onde antes não havia sido identificada uma solução satisfatória.

O protótipo desenvolvido não implementa todas as características descritas no projeto conceitual do MOrFEU, mas mostra que mesmo com poucas destas características foi capaz de atender aos requisitos de um dos cenários, a Controvérsia Acadêmica.

Nossa análise sugere uma visão promissora no aprofundamento e diversificação das investigações acerca do MOrFEU.

5.2 Avaliação do Protótipo

Nesta seção são descritas algumas verificações realizadas com relação ao funcionamento do protótipo: seu correto funcionamento; se atendeu aos requisitos da modelagem conceitual; e se aquilo que foi modelado foi corretamente implementado.

5.2.1 Funcionamento do Módulo do Moodle

A preocupação inicial é se o módulo do Moodle em que foi feito o protótipo está funcionando de maneira correta e se apresenta as características apontadas como vantagens por ser um módulo do Moodle. O segundo ponto a ser verificado é se o protótipo não é apenas um incremento do Moodle, ou seja, se ele está independente o suficiente do Moodle a ponto de não herdar as características não desejáveis do Moodle.

Durante a criação do módulo, foi sempre utilizada a funcionalidade de fazer uma atualização no momento da instalação de uma nova versão do módulo. Sempre foram passadas instruções sobre alterações nos dados já inseridos no módulo, assim como alterações no esquema do banco de dados. Em algumas versões e também na versão corrente do módulo,

foi feita uma instalação deste em um sistema do Moodle que ainda não possuísse este módulo instalado, visando verificar se sua funcionalidade de instalação estava correta. Não foram feitas instalações em versões diferentes do Moodle, somente foi utilizada a versão 1.9, e todos os testes de instalação e atualização da versão corrente do módulo foram bem sucedidos.

Outra característica gerenciada pelo Moodle e descrita o seu procedimento no módulo é a funcionalidade de *cron*, a execução periódica de um procedimento. Foi utilizada esta funcionalidade para o envio de *e-mails*, prática comum nos demais módulos do Moodle. Foram enviados *e-mails* aos participantes em um fórum, e se tudo no Moodle estivesse configurado de forma apropriada (eventos do *cron* e servidor de envios de *e-mails*), sairia tudo como esperado, enviando *e-mails* com as mais novas postagens no fórum, ou em qualquer veículo do tipo Listagem configurado para tal, o que aconteceu de fato.

Uma característica interessante apresentada, devido à forma como o protótipo foi construído, foi a permanência de todas as características do sistema Moodle onde foi instalado o módulo. Ou seja, o ambiente Moodle onde foi instalado o módulo continua funcionando normalmente, com todos os usuários, cursos, etc. O Moodle e o MOrFEU, do ponto de vista do usuário, funcionam como dois sistemas diferentes, compartilhando apenas algumas características: os usuários e os elementos visuais de sua aparência. Assim, é possível que haja uma maior aceitação pelas pessoas interessadas em testar o MOrFEU e que já possuam instalado em seu servidor uma versão do sistema Moodle, pois não terão que fazer a instalação de outro sistema completo e nem terão que refazer o cadastro dos usuários. Esta característica é descrita por Black *et al* (2007) como *Triability*, o grau com que uma inovação pode ser testada em uma base limitada antes da adoção em larga escala (Rogers, 2003 apud Black *et al*, 2007).

5.2.2 Funcionalidades Comuns a Outros Ambientes

Segundo Black *et al* (2007) um ambiente virtual novo deve ter como uma de suas características a Compatibilidade, ou seja, o grau com que uma inovação é percebida como consistente com valores existentes, experiências passadas e necessidade de potenciais adotantes (Rogers, 2003, p.15 apud Black *et al*, 2007). Investigou-se se o protótipo possuía as funcionalidades mais comuns aos ambientes virtuais e quais eram suas principais limitações nessa direção. Dentre as funcionalidades mais comuns estão a gerência de usuários, a gerências de comunidades/grupos/cursos, a gerência de arquivos, ferramentas de comunicação e ferramentas de produção coletiva, como o Wiki.

Como foi especificado, foi aproveitada a gerência de usuários do Moodle, com cadastro e *login* como suas principais funcionalidades. E esses usuários do Moodle foram perfeitamente encaixados como usuário do MOrFEU no protótipo. Não houve perda das informações contidas no Moodle, nem houve acréscimo de informações. No entanto, a página do perfil do usuário foi um pouco modificada para acrescentar informações sobre as produções do usuário e de seus grupos. É importante salientar que esta modificação não foi feita na página que existe no Moodle, mas foi feita uma cópia dela para o MOrFEU.

O usuário possui uma área particular, onde são listados a sua produção de UPIs e Veículos, e também são listadas as produções dos seus grupos. Há ainda um menu que sempre o acompanha nas páginas, dando acesso aos principais módulos, e também pode possuir sub-menus, de acordo com a necessidade da página. A partir desse lugar é possível ter acesso aos recursos e ferramentas do ambiente.

É possível acessar os grupos através do menu, que dará acesso a uma lista de todos os grupos do ambiente, indicando aqueles ao qual o usuário pertence. Como foi dito na especificação do protótipo não há um grande suporte à gerência de grupos, mas é possível criar, participar e deixar de participar de um grupo.

À gerência de arquivos não foi dada tanta ênfase na construção do protótipo. Foi reaproveitada a gerência de arquivos do Moodle, mas a organização desses arquivos é um pouco atrapalhada, pelo fato de todos os usuários manipularem os mesmos diretórios e arquivos. Além de esses arquivos só serem acessíveis via UPIs.

Entre as ferramentas de comunicação disponíveis tem-se o Blog, o Fórum e o Chat, implementados através da Superclasse de Veículos de Comunicação Listagem. Um sistema de notícias simples também é possível ser utilizado através de um fórum. O Blog pode ser utilmente utilizado tanto para um grupo quanto para um indivíduo. O envio de emails ainda pode estar associado a qualquer estas ferramentas, enviando as mensagens postadas nelas aos seus usuários.

Uma ferramenta de produção coletiva de texto muito popular é o Wiki. Um Wiki é um produtor de páginas que podem possuir *links* para outras páginas dentro do Wiki. Não foi implementada no protótipo uma ferramenta de Wiki, especificamente, mas uma página simples pode ser feita através de uma UPI. No entanto, os *links* só são possíveis se feitos de forma manual.

Assim, o protótipo abrange uma boa amostra das funcionalidades mais comuns entre os ambientes virtuais.

5.2.3 A Flexibilidade no Protótipo

Como foi demonstrado, a flexibilidade deve ser a principal característica que difere o MOrFEU dos demais ambientes padronizados. Isso é uma Vantagem Relativa, descrita por Rogers (2003, apud Black *et al*, 2007) como a superioridade de uma inovação quando comparada aos seus predecessores. Alguns cenários foram analisados na Seção 5.1 visando demonstrar o caráter flexível do projeto conceitual do MOrFEU. Mas o que o protótipo já é capaz de fazer?

Como foi visto na Seção 4.4, não foram implementadas várias funcionalidades do projeto conceitual do MOrFEU. Uma funcionalidade muito utilizada para descrever as soluções utilizando o MOrFEU na Seção 5.1 foram os sub-veículos, isto é, a capacidade de um veículo possuir outros veículos agregados a eles de forma subordinada. Esta funcionalidade remonta a idéia de Artefatos de Conhecimento de Anttila (2006) (ver a Seção 2.3 Ambientes de Trabalho e Conhecimento).

Assim, três dos quatro cenários estudados na Seção 5.1 não são passíveis de implementação na versão atual do protótipo. Mas o Cenário C: Controvérsia Acadêmica, na Seção 5.1.3, é um bom exemplo da flexibilidade já implementada pelo protótipo.

A solução proposta para o método da Controvérsia Acadêmica foi a criação de um Fórum com três respostas possíveis, “Concordo”, “Discordo” e “Depende” (Mendonça *et al*, 2003). No protótipo construído, ainda não há uma Classe de Veículos “Fórum da Controvérsia Acadêmica”, mas este é possível de ser construído com alguns poucos passos:

1. Criar um Veículo “Fórum” comum para o grupo de todos os alunos;
2. Criar três respostas no Fórum, “Concordo”, “Discordo” e “Depende”, no Nível 1;
3. Editar as configurações do veículo para permitir respostas no Nível 2 (trocar o valor da configuração “*configurations/permission/treeLevel*” de “*all*” para “*2*”).

Na Figura 18, pode-se ver o fórum resultante deste procedimento, acima descrito. Como se pode observar no destaque, só há *links* para resposta aos três tópicos “Concordo”, “Discordo” e “Depende”.

Outros cenários já foram exemplificados no decorrer deste trabalho, como: um Chat que se transforma num Fórum para uma discussão mais organizada dos temas abordados na seção inicial de interação; e um veículo do tipo Codificação pode transformar-se num fórum para uma discussão das falhas, dificuldades e soluções interessantes encontradas na execução daquela atividade.



Figura 18. Um Fórum para a Controvérsia Acadêmica implementado no protótipo do MORFEU-UFAM. Em destaque nas elipses azuis, os *links* para responder aparecem apenas para os três tópicos principais, em destaque com os retângulos vermelhos.

5.2.4 Próximos Passos

Uma ação ainda em andamento na avaliação do protótipo (e conseqüentemente da proposta), é sua aplicação em situação real de uso com usuários e atividades “importados” de outro ambiente. Questionários e entrevistas serão instrumentos utilizados, e os resultados serão cruzados com informações dos registros no *log* do sistema, para verificar a consistência das informações.

6 Conclusão

O trabalho e a aprendizagem construídos individual e coletivamente tomaram novo e determinante impulso a partir da disseminação das ferramentas de *software* que atualmente compõem os ambientes virtuais baseados na Web. Após um período de reconhecimento e apropriação desses recursos, o trabalho, o ensino, a aprendizagem e as relações sociais como um todo passam a apresentar demandas que os ambientes virtuais ainda não conseguem atender, em parte devido à conformação desses ambientes a certos aspectos funcionais mantidos principalmente por tradição. O desenvolvimento da primeira versão de um ambiente concebido sob um paradigma diferenciado por sua inovadora simplicidade cumpre seu papel na busca por soluções para esse problema, servindo como agregador para um esforço multi-institucional de pesquisa no tema.

6.1 Contribuições do Trabalho

Além de ter alcançado seu objetivo geral – a concepção de um ambiente virtual segundo um paradigma diferenciado para a organização da produção intelectual de seus usuários – acredita-se que o projeto relatado nesta dissertação contribuiu à área nos seguintes aspectos:

- Foi organizado um conjunto inicial de requisitos para ambientes a serem desenvolvidos segundo o paradigma proposto;
- A modelagem conceitual para o MOrFEU foi submetida a seu primeiro ciclo de discussões, tendo em vista a implementação de alguns de seus elementos;
- Descrição de uso do Moodle como plataforma de desenvolvimento para outro ambiente virtual;

- Análise conceitual da proposta do MOrFEU a partir de cenários de aplicação.

O protótipo do MOrFEU demonstra a factibilidade da proposta, além de apresentar uma forma alternativa para a construção de sistemas Web, com produtos desenvolvidos de forma rápida e relativamente robusta, e que pode ser melhor explorada e sistematizada.

Pode-se observar que o protótipo atendeu a vários requisitos listados no projeto conceitual, enquanto outros já possuem uma boa estruturação para solução. Um exemplo disto é o requisito não-funcional de Suportabilidade (ver Seção 3.3 Requisitos Não-Funcionais), pois por tratar-se de um módulo do Moodle, é possível encaminhar a formação de uma comunidade de desenvolvedores voltada ao suporte, manutenção e desenvolvimento seguindo os padrões do *software* livre.

6.2 Trabalhos Futuros

Ao longo deste trabalho, alguns possíveis desdobramentos foram oportunamente mencionados, e a seguir, revisitamos algumas dessas idéias além de apresentar outras tantas.

Um primeiro ponto a se trabalhar é na melhoria do protótipo. Já são sabidas várias preocupações acerca de construção de ambientes virtuais. Dimitracopoulou (2005) destaca algumas funcionalidades fundamentais para ambientes virtuais de aprendizagem. Dentre elas podem-se citar: funcionalidades de percepção (*awareness*); papéis de atuação para os usuários; ferramentas para auxílio do gerenciamento de grupos e comunidades; e facilidades para o gerente de grupos. E outro ponto que se deve dar grande importância é quanto à usabilidade do sistema e sua interface humano-máquina. E ainda é importante citar que o protótipo não passou por nenhum experimento formal de utilização por usuários finais.

Uma das questões pendentes entre o conceito e o protótipo recai sobre os veículos-padrão do sistema, como: a caixa de produções individual do usuário e do grupo; a página inicial do usuário; a página do perfil do usuário; a página inicial do grupo; e a página de perfil do grupo. Todos estão implementados de forma diferente da arquitetura de veículos de

comunicação, e não lhes foram dadas funcionalidades flexíveis adequadas. Se eles fossem implementados de acordo com a arquitetura de veículos de comunicação proposta, eles poderiam ter maior flexibilidade, o que seria ideal para o sistema.

Um ponto que ficou a ser implementado que faz parte da estrutura conceitual do MORFEU, é a biblioteca de mídias. Na verdade, ela foi pouco explorada quando se apresentava o MORFEU e implementada precariamente no protótipo. Essa biblioteca deve conter vários tipos de mídias, textos, imagens, sons. Há ainda que se considerar que uma UPI e um Veículo de Comunicação também são mídias. Isto leva a outra questão, a de como seria a organização dessas mídias, pois se acredita que a organização tradicional de diretórios e subdiretórios pode ser melhor explorada para se tornar mais flexível.

Com a idéia mais ampla sobre biblioteca de mídias, é conseqüência pensar que os elementos dessas bibliotecas de mídias possuam relações entre si. Duas dessas relações já foram consideradas: a relação entre um Artigo publicado em um Veículo e este próprio Veículo, que foi chamada de Publicação; e a relação entre os Artigos publicados num Veículo, que foi chamada “resposta”, formando uma árvore de Artigos. Porém, como se considera que a árvore de diretórios não é a idéia final e perfeita para organização de mídias, imagina-se que apenas este tipo de relação “resposta” entre artigos publicados em um veículo não seja a mais adequada em relação à flexibilidade do sistema, também.

Assim como no mundo real definimos elementos e sub-elementos que compõem um dado objeto, pode-se imaginar também sub-artigos e sub-veículos, como já foi um pouco explorado na definição conceitual inicial. Ou seja, deseja-se definir relações do tipo “parte-de” entre artigos ou entre veículos. Este contexto traz outras questões que precisam ser consideradas em conjunto, tais como: herança de autoria entre os elementos e os sub-elementos; herança de características ou não; e controle de papéis de atuação nos elementos. Assim, se tivéssemos um Veículo do tipo Jornal, onde cada seção dele é responsabilidade de

uma equipe diferente, mas todos são subordinados ao editor-chefe, as seções poderiam ser sub-veículos, mas seria preciso considerar, por exemplo, se essas seções teriam o mesmo estilo de apresentação ou se cada sub-veículo poderia atribuir seu próprio estilo de apresentação, e ainda se uma equipe poderia intervir no trabalho de outra equipe, e que tipo de intervenção seria essa, se edição, ou revisão ou não poderia existir este tipo de intervenção. Questões desse tipo precisam ser bem melhor estudadas e aprofundadas.

Uma das funcionalidades que é ainda desejada é a anotação pessoal sobre objetos componentes do sistema (Grupos, UPIs ou Veículos de Comunicação), de modo que o usuário seja capaz de registrar suas observações e considerações a respeito deles. Estas observações ainda podem ser compartilhadas com os outros usuários de forma que se possibilite uma troca de idéias e opiniões sobre determinado elemento considerado de forma isolada. Como é de se esperar, esta funcionalidade pode ser feita através de um veículo próprio para tal.

Com relação aos papéis de usuários, é preciso definir se os papéis serão fixos, com número e atribuições limitadas, o que não é desejável, ou se serão livres para criação e definição de suas atribuições, o que resultaria em maior flexibilidade. E que atribuições seriam estas e se teriam um efeito global ou local, dentro do ambiente ou do grupo. Mas a questão principal é de que forma isto poderia ser feito. Inicialmente, pode-se tomar como exemplo a solução adotada pelo Moodle, que possui um conjunto estático de “atividades” e podem ser definidos papéis que têm permissão ou não para realizar uma dada atividade. Mas esse conjunto estático de “atividades” não traz a flexibilidade adequada, por isso é preciso investigar outra solução.

Como foi visto, apenas foram feitos dois tipos de Superclasse de Veículo, a Listagem e a Codificação. Essas abrangem apenas uma parte muito pequena do conjunto das ferramentas de comunicação existentes na Web. Para demonstrar os limites de flexibilidade da arquitetura atual do MOrFEU seria interessante a definição de veículos que modelem outras ferramentas

existentes. E o esperado de tal processo seria a descoberta de outros tipos ferramentas de comunicação ainda não formalizados.

As configurações de um Veículo são feitas em XML, mas é responsabilidade do usuário dominar esta sintaxe e tecnologia para poder fazer uma configuração adequada de seu veículo. Assim, é desejável que existam ferramentas CASE para edição das configurações de um Veículo.

Em sua concepção conceitual inicial, o MOrFEU admite *templates* diferentes para cada Veículo. É razoável pensar que estes *templates* sejam atribuídos individualmente por usuário do Veículo ou de forma geral, para todas as pessoas. E de que forma esta customização pode ser feita e como fazer isto dentro da arquitetura atual do MOrFEU é uma questão a ser pesquisada.

E aos agentes de *software* inteligentes foi reservado um lugar, como autores de UPIs. Porém, resta investigar se apenas este papel é suficiente para ferramentas inteligentes desejáveis de inclusão, como aquelas que tentam fazer a junção da *Web Social* (O'Reilly, 2005) e da *Web Semântica* (Berners-Lee *et al*, 2001), (Gruber, 2008), (Bojárs *et al*, 2008) e (Greaves & Mika, 2008) para gestão do conhecimento, especialmente levando em consideração que o MOrFEU é um ambiente mais organizado que a *Web*.

E desse modo, já é possível vislumbrar um caminho longo de descobertas envolvendo o MOrFEU.

Referências Bibliográficas

ALMEIDA NETO, F. A. Um Ambiente de Acompanhamento do Processo de Desenvolvimento de Programas. *Dissertação de Mestrado em Informática*. Universidade Federal do Amazonas, 2007.

ANTTILA, J. Modern Approach of Information Society to Knowledge Work Environment for Management. *ICIT 2006 IEEE International Conference on Industrial Technology*, 2006. p 2113-2118. ISBN 1-4244-0726-5.

ÁVILA, L. *Una Propuesta para la Clasificación de Herramientas “Groupware” y el Sintegrador como un Desarrollo Groupware*. Disponível em: <<http://hdl.handle.net/1992/592>>, DSpace, 1999.

BEDER, D. M. *et al.* A Case Study of the Development of e-Learning Systems Following a Component-based Layered Architecture. *Seventh IEEE International Conference on Advanced Learning Technologies (ICALT 2007)*, 2007. p.21-25.

BERNERS-LEE, T. *Weaving the Web: the original design of the World Wide Web by its inventor*. Harper Business, 1st Edition, 2000

BERNERS-LEE, T., HENDLER, J.A., LASSILA, O. The Semantic Web. *Scientific American*, vol. 284, n.5, 2001. p34-43. BIANCHINI, S. L. Avaliação de Métodos de Desenvolvimento de aplicações Web. *Dissertação de Mestrado em Ciências de Computação e Matemática Computacional*. Instituto de Ciências Matemáticas e de Computação da USP São Carlos, 2008.

BLACK, E. W; BECK, D.; DAWSON, K.; JINKS, S.; DIPIETRO, M. The other side of the LMS: Considering implementation and use in the adoption of an LMS in online and blended learning environments. *TechTrends*. Springer Boston, 2007. p.35-39. ISSN 8756-3894 (Print) 1559-7075 (Online).

BOJÁRS, U.; BRESLIN, J.; PERISTERAS, V.; TUMMARELLO, G.; DECKER, S. Interlinking the Social Web with Semantics, *IEEE Intelligent Systems*, 2008. p29-40.

CARVALHO, M. J.; NEVADO, R. A.; MENEZES, C.S. Arquiteturas Pedagógicas para Educação a Distância: Concepções e Suporte Telemático. *Anais – XVI Simpósio Brasileiro de Informática na Educação*, p.362-372, 2005.

CÁTEDRA UNESCO DE EDUCACIÓN A DISTANCIA. *Entornos y plataformas para virtualizar cursos*. Disponível em: <<http://www.uned.es/catedraunesco-ead/plataformas.htm>>, acessado em 9 fev 2009.

CERI, S.; FRATERNALI, P.; BONGIO, A. Web Modeling Language (WebML): a modeling language for designing Web sites. *Computer Networks*, 2000. p.137-157.

CHIEU, V. M. COFALE: An Authoring System for Supporting Cognitive Flexibility. *Proceedings of the Sixth International Conference on Advanced Learning Technologies (ICALT'06)*, 2006.

DIMITRACOPOULOU, A. Designing Collaborative Learning Systems: Current Trends & Future Research Agenda. *Conference on Computer support for Collaborative Learning*, 2005. p.115-124.

DOUGIAMAS, M.; TAYLOR, P. C. Moodle: Using Learning Communities to Create an Open Source Course Management System. *Proceedings of the EDMEDIA Conference*, 2003.

EDUTOOLS. *Archive CMS: Product List*, 2009a. Disponível em: <http://www.edutools.info/item_list.jsp?pj=8>, acessado em 10 fev 2009.

EDUTOOLS. *CMS: Product List*, 2009b. Disponível em: <http://www.edutools.info/item_list.jsp?pj=4>, acessado em 10 fev 2009.

ELLIS, C.A.; GIBBS, S.J; REIN, G.L. Groupware - Some Issues and Experiences. *Communications of the ACM*, January 1991, Vol. 34, N. 1, p. 38-58.

FAGUNDES, L.C., SATO, L.S., MAÇADA, D. L., *Aprendizes do Futuro, as Inovações já começaram*, Brasília, MEC, 1999.

FAGUNDES, L.; NEVADO, R.; BASSO, M.; BITENCOURT, J.; MENEZES, C.; MONTEIRO, V. Projetos de Aprendizagem – Uma experiência mediada por ambientes Telemáticos. *RBIE Revista Brasileira de Informática na Educação*, 2006.

FUKS, H. Aprendizagem e Trabalho Cooperativo no Ambiente AulaNet. *Revista Brasileira de Informática na Educação*, SBC, N6, 2000. p.53-73.

FUKS, H.; RAPOSO, A.B.; GEROSA, M.A. Engenharia de Groupware: Desenvolvimento de Aplicações Colaborativas. *XXI Jornada de Atualização em Informática, Anais do XXII Congresso da Sociedade Brasileira de Computação*, 2002, V2, Cap. 3, ISBN 85-88442-24-8, pp. 89-128.

GOOGLE DIRECTORY. *Google Directory: Course Website Software*. Disponível em: <http://www.google.com/Top/Reference/Education/Instructional_Technology/Course_Website_Software/>, acessado em: 18 mar 2009.

GREAVES, M., MIKA, P. Semantic Web and Web 2.0 (Editorial). *Web Semantics: Science, Services and Agents on the World Wide Web* 6, 2008. p1-3.

GRUBER, Tom. Collective knowledge systems: Where the Social Web meets the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web* 6, 2008. p4-13.

HAMMER, M. Reengineering Work: Don't Automate, Obliterate. *Harvard Business Review*. July-August, 1990, p.104-112.

HENNICKER, R.; KOCH, N. Systematic design of Web applications with UML. *Unified modeling language: systems analysis, design and development issues*. IGI Publishing Hershey, PA, USA, 2001. p. 1-20. ISBN 1-930708-05-X.

HORNBAEK, K. Current practice in measuring usability: Challenges to usability studies and research. *Int. J. Human-Computer Studies* 64, 2006, p. 79–102.

ITMAZI, J. Sistema Flexible de Gestión Del eLearning para Soportar El Aprendizaje en las Universidades Tradicionales y Abiertas. *Tesis Doctoral en Métodos y Técnicas Avanzadas de Desarrollo de Software*. Universidad de Granada, 2005.

JOHNSON, D. W.; JOHNSON, R. T.; SMITH, K. A. *Academic Controversy: Enriching College Instruction through Intellectual Conflict*. ASHEERIC Higher Education Report Volume 25, Nº 3. Washington, D. C.: The George Washington University, Graduate School of Education and Human Development, 1999.

LIMA, P. S. R. Um Ambiente Colaborativo de Aprendizagem Interdisciplinar Apoiado por Interfaces Adaptativas. *Tese de Doutorado em Engenharia Elétrica*. Universidade Federal do Pará, 2006.

LYYTINEN, K. J.; NGWENYAMA, O. K. What does Computer Support for Cooperative Work mean? A Structural Analysis of Computer Supported Cooperative Work. *Accounting, Management and Information Technology*, Vol. 2, N. 1, p. 19-37, 1992. ISSN 0959-8022/92.

MAÇADA, D. L.; TIJIBOY, A. V. Aprendizagem Cooperativa em Ambientes Telemáticos. *IV Congresso da Rede Iberoamericana de Informática Educativa (RIBIE)*, 1998.

MENDONÇA, A. P.; CASTRO Jr, A. N.; MOTA, E. S.; SOUZA, F. F.; SILVA, L. S.; PEREIRA, V. L.. Uma Experiência com o uso dos Mapas Conceituais para apoiar o Método da Controvérsia Acadêmica. *VIII WIE - Workshop de Informática na Educação*. XXII Congresso da Sociedade Brasileira da Computação. Florianópolis, v. 5, p. 99-107, 2002.

MENDONÇA, A. P. *et al.* Um Ambiente Telemático para mediar a Controvérsia Acadêmica. *Simpósio Brasileiro de Informática na Educação*, 2003.

MENEZES, C. S.; CURY, D.; CASTRO Jr, A. N. An Architecture of an Environment for Cooperative Learning (AmCorA). *Proceedings of ICECE 2000 - International Conference on Engineering and Computer Education*, 2000, São Paulo.

MENEZES, C.; NEVADO, R.; CASTRO Jr., A.; SANTOS, L. MOrFEU – Multi-Organizador Flexível de Espaços Virtuais para Apoiar a Inovação Pedagógica em EAD. *Anais do XIX Simpósio Brasileiro de Informática na Educação*. SBC, Fortaleza-CE, 2008.

MINISTÉRIO DA EDUCAÇÃO. *e-Proinfo: Ambiente Colaborativo de Aprendizagem*. Disponível em: <<http://www.eproinfo.mec.gov.br/>>, 2000.

MOODLE.ORG. *Online Learning History*. Disponível em: <http://docs.moodle.org/en/Online_Learning_History>, acessado em: 13 mar 2009.

MOODLE.ORG. *Moodle Statistics*. Disponível em: <<http://moodle.org/stats/>>, acessado em: 18 mar 2009.

MONTEIRO, V.; MENEZES, C.; NEVADO, R.; FAGUNDES, L. Ferramenta de Autoria e Interação para apoio ao desenvolvimento de Projetos de Aprendizagem. *Renote Revista Novas Tecnologias na Educação* V3, v. 3, n. 2, 2005.

MONTEIRO, Valéria Cristina Pelinzzar Cauper; Um ambiente de apoio ao desenvolvimento de Projetos de Aprendizagem, *Dissertação de Mestrado*, PPGIUFES, Vitória-ES, junho, 2006.

MURRAY, T.; HILTZ, S. R. Computer Support for Group Versus Individual Decisions. *IEEE Transactions on Communications*, Vol. COM-30, N. 1, p.82-91. Jan.1982. ISSN 0090-6778/82.

NEVADO, R. A.; CARVALHO, M. J. S.; MENEZES, C. S. (Org.) Aprendizagem em Rede da Educação a Distância: Estudos para Formação de Professores. Ricardo Lens Editor, 2007.

O'REILLY, T. *What is web 2.0: Design Patterns and Business Models for the Next Generation of Software*, 2005. Disponível em: <<http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-isweb-20.html>>, acessado em 10 fev 2009.

PENICHET, V.M.R.; MARIN, I.; GALLUD, J.A.; LOZANO, M.D.; TESORIERO, R. A Classification Method for CSCW Systems. *Electronic Notes in Theoretical Computer Science* 168, p. 237-247, Elsevier, 2007.

PEREIRA, V. L.; CASTRO Jr, A. N.; SOUZA, F. F.; MENDONÇA, A. P.; SILVA, L. S. Análise do Método Jigsaw de Aprendizagem Cooperativa através da utilização de Mapas Conceituais. *Anais - VIII WIE - Workshop de Informática na Educação*. XXII Congresso da Sociedade Brasileira da Computação. Florianópolis, v. 5, p. 181-188, 2002.

RAMA, J; BISHOP, J. A Survey and Comparison of CSCW Groupware Applications. *Proceedings of SAICSIT*, p. 198-205, 2006.

ROCHA, H.V. TelEduc: Software livre para educação a distância. In: Silva, Marco. (Org.). *Educação On-line*. 1ª. ed. São Paulo: Edições Loyola, 2003, p. 377-396.

RÖSSLING, G. *et al.* Enhancing learning management systems to better support computer science education. *ACM SIGCSE Bulletin*, V.40 , Issue 4, 2008. p.142-166. ISSN 0097-8418.

SANTOS, L.; CASTRO Jr, A.; CASTRO, T. Alteração no Modelo de Grupos do Moodle para Apoiar a Colaboração. *Simpósio Brasileiro de Informática na Educação*, 2007.

SCHMIDT, K.; BANNON, L. Taking CSCW Seriously: Supporting Articulation Work. *Journal of Computer Supported Cooperative Work (CSCW)*, v. 1, n. 1-2, 1992. ISSN 0925-9724 (Print) 1573-7551 (Online).

SILVA, L. S.; CASTRO Jr, A. N.; SOUZA, F. F.; MENDONÇA, A. P.; PEREIRA, V. S. Mapas Conceituais como suporte à estratégia de Investigação em Grupo: Uma experiência na Universidade do Amazonas. VIII WIE - Workshop de Informática na Educação. XXII Congresso da Sociedade Brasileira da Computação. Florianópolis, v. 5, p. 163-172, 2002.

SLAVIN, Robert E. *Cooperative learning: theory, research, and practice*. - [s.l.]:Allyn & Bacon, 1995.

STAHL, G. Group cognition in computer-assisted collaborative learning. *Journal of Computer Assisted Learning* 21, 2005. pp. 79-90.

TUROFF, M.; HILTZ, S. Computer Support for Group Versus Individual Decisions. *IEEE Transactions on Communications*. V. 30, p. 82-91, 1982. ISSN 0090-6778.

WIKIPEDIA. *History of virtual learning environments*. Disponível em: <http://en.wikipedia.org/wiki/History_of_virtual_learning_environments>, acessado em: 13 mar 2009.

WATSON, W. R.; WATSON, S. L. An Argument for Clarity: What are Learning Management Systems, What are They Not, and What Should They Become? *TechTrends*. Springer Boston, Volume 51, Number 2, 2007. p. 28-34. ISSN 8756-3894 (Print) 1559-7075 (Online).

APÊNDICE I: Diagrama de Navegação

Em um curto levantamento bibliográfico realizado acerca de modelos de diagrama de navegação para páginas Web, não foi encontrado um que atendesse às necessidades no desenvolvimento do MOrFEU. Neste levantamento, pode-se citar o WebML (Ceri et al, 2000), UWE Hennicker&Koch, 2001 e (Bianchini, 2008). Este último relata vários trabalhos da área de modelagem para desenvolvimento Web e propõe uma metodologia para avaliação desses modelos. Assim, como o trabalho de Bianchini (2008) é bem recente e traz um panorama do estado da arte, pôde-se ter esse mesmo panorama.

Contudo, nenhum desses modelos atendeu às necessidades esperadas de um modelo desse tipo. Desejava-se um modelo mais baixo nível, que pudesse levar em conta os formulários HTML e as informações que são passadas de uma página para outra, além de representar as operações realizadas em cada página, não se restringindo apenas à consulta, alteração e inserção de informações no banco de dados.

Assim sendo, desenvolveu-se este simples modelo de Diagrama de Navegação para páginas Web para apoiar o desenvolvimento do protótipo. A seguir, são explicadas algumas características e os elementos que compõe este modelo:

- **PÁGINA:** uma página é representada por uma caixa dividida horizontalmente em três partes. A primeira parte contém o nome da página. A segunda parte, a do meio, é descrito o conteúdo da página em questão. E terceira parte, a mais debaixo, contém as operações realizadas naquela página. Observe a Figura 19.

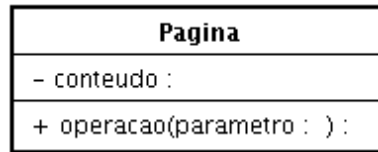


Figura 19. Uma página no Diagrama de Navegação.

- **CONTEÚDO:** O conteúdo de uma página sugere do que vai ser composta a página, que pode ser composta de várias coisas. Se escrito começando com letra minúscula, indica a descrição do que conterà. E se escrito começando com letra maiúscula, indica algum conteúdo presente no diagrama de dados. Existem ainda dois tipos especiais de conteúdos: “list” representa uma lista de elementos um elemento de dado; e “form” indica um formulário HTML.
- **OPERAÇÕES:** são métodos executados quando aquela página é acessada. Estas operações, por sua vez, podem conter parâmetros de entrada.
- **LINK:** os *links* são setas que saem de uma página com destino a outra página. Se esta seta não contiver nenhum texto lhe acompanhando, isto indica que é um *link* simples. Mas ela conter algum texto, isto indica que este leva alguma informação, um parâmetro de entrada, para outra página. Estes últimos também representam uma passagem de parâmetros que pode ocorrer com um formulário, ou seja, tratando-se de HTML, esta passagem de parâmetros pode ser do tipo *GET* ou do tipo *POST*, *links* realmente (estes somente pelo tipo *GET*) ou formulários. Um mesmo *link* pode conter vários parâmetros que devem ser separados por vírgula. Veja na Figura 20 dois exemplos de *links*.

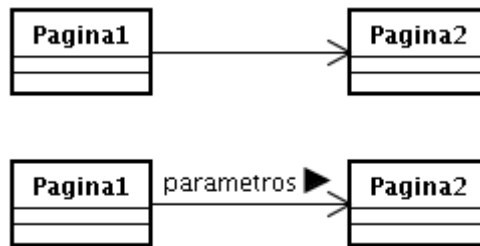


Figura 20. *Links* no diagrama de navegação.

- **INCLUSÃO:** as linguagens de construção dinâmica de páginas HTML, como PHP e JSP, permitem a inclusão de outros pedaços de códigos, que podem ser páginas. A Figura 21 apresenta este tipo de inclusão. O losango branco está do lado da página que está incluindo a página do outro lado da ligação. Esta inclusão pode ser acompanhada com uma passagem de parâmetros, porém estes parâmetros não são HTML, mas variáveis internas. O círculo negro indica uma disjunção, ou seja, ou a *Página2* ou a *Página3* será incluída na *Página1*, exclusivamente.

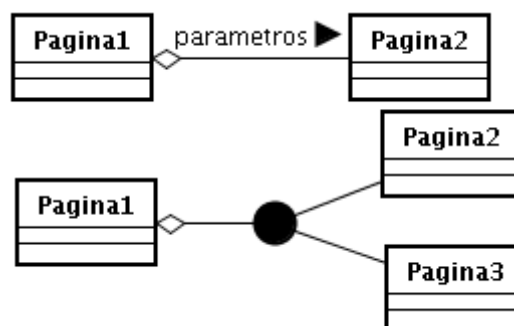


Figura 21. Inclusão no Diagrama de Navegação.

- **REDIRECIONAMENTOS:** são *links* que são utilizados imediatamente depois que a *Página1* é totalmente executada. Ou seja, a impressão que fica para o usuário é a de que a *Página1* nem foi carregada, e que ele foi diretamente para a *Página2*. Estes redirecionamentos são representados por setas pontilhadas, como pode ser visto na Figura 22.

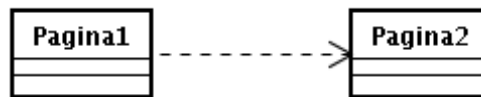


Figura 22. Redirecionamento no Diagrama de Navegação.

- **LINKS DE SÍTIOS:** o círculo branco significa uma abstração de um lugar qualquer, seja ele dentro do próprio sítio, ou um outro sítio. Ideal para abstrações de sítios externos. Um exemplo é mostrado na Figura 23.

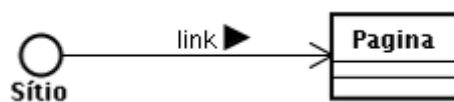


Figura 23. Links de sítios no Diagrama de Navegação.

- **CONDIÇÕES:** as condições são representadas por colchetes, “[condição]”, e indicam que a ação seguinte à condição só será executada ou exibida se a condição for satisfeita. Elas podem aparecer no conteúdos e nas operações de páginas ou nos parâmetros de *links*. Se esta condição estiver em um *link*, significa que este *link* só existirá se a condição for satisfeita. Para compor a condição ainda pode-se utilizar uma vírgula para representar uma conjunção, que é um “e”, e o símbolo *pipe* “|” para representar uma disjunção, que é um “ou”.

APÊNDICE II: Esquemas das Configurações dos Veículos

Nesta seção são apresentados os esquemas das configurações dos veículos da superclasse Listagem (Quadro 6) e da superclasse Codificação (Quadro 7), ambos feitos no padrão XMLXhema²⁵.

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="configurations">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="treeOrder">
          <xs:complexType>
            <xs:all>
              <xs:element name="criteria">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:pattern value="tree|date"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>

              <xs:element name="order">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:pattern value="ASC|DESC"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>

              <xs:element name="printType">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:pattern value="indented|flat"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
            </xs:all>
          </xs:complexType>
        </xs:element>

        <xs:element name="publication">
          <xs:complexType>
            <xs:all>
              <xs:element name="newArticle">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:pattern value="yes|no"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
            </xs:all>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

²⁵ <http://www.w3.org/2001/XMLSchema>


```

</xs:element>

<xs:element name="editArticle">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="yes|no"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="removeArticle">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="yes|no"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="sendMail" minOccurs="0" maxOccurs="1">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="yes|no"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:all>
</xs:complexType>
</xs:element>

<xs:element name="page">
<xs:complexType>
<xs:all>
  <xs:element name="allowView">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="author|all|none"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

  <xs:element name="header">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="yes|no"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

  <xs:element name="refresh">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="no|([0-9])+"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>

  <xs:element name="css" type="xs:string" />

</xs:all>
</xs:complexType>
</xs:element>

<xs:element name="permission" minOccurs="0"
maxOccurs="unbounded">
<xs:complexType>
<xs:all>
  <xs:element name="allowPublish">

```

```

        <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="author|all|none"/>
        </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="treeLevel">
        <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="all|([0-9])+|(from([0-
9]))+?(until([0-9])+)?" />
        </xs:restriction>
        </xs:simpleType>
    </xs:element>

    </xs:all>
</xs:complexType>
</xs:element>

</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Quadro 6. Esquema das configurações de veículos de comunicação da superclasse Listagem, feito em XMLSchema.

```

<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="configurations">
        <xs:complexType>
            <xs:sequence>

                <xs:element name="coding">
                    <xs:complexType>
                        <xs:all>
                            <xs:element name="mode">
                                <xs:simpleType>
                                    <xs:restriction base="xs:string">
                                        <xs:pattern value="individual|viewAll"/>
                                    </xs:restriction>
                                </xs:simpleType>
                            </xs:element>

                            <xs:element name="language">
                                <xs:simpleType>
                                    <xs:restriction base="xs:string">
                                        <xs:pattern value="haskell|prolog"/>
                                    </xs:restriction>
                                </xs:simpleType>
                            </xs:element>

                            <xs:element name="interpreter">
                                <xs:simpleType>
                                    <xs:restriction base="xs:string">
                                        <xs:pattern value="hugs|gprolog"/>
                                    </xs:restriction>
                                </xs:simpleType>
                            </xs:element>

                            <xs:element name="edit">
                                <xs:simpleType>
                                    <xs:restriction base="xs:string">
                                        <xs:pattern value="yes|no"/>
                                    </xs:restriction>
                                </xs:simpleType>
                            </xs:element>
                        </xs:all>
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

```

        </xs:all>
        </xs:complexType>
        </xs:element>

        <xs:element name="page">
        <xs:complexType>
        <xs:all>
            <xs:element name="allowView">
                <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:pattern value="author|all|none"/>
                </xs:restriction>
                </xs:simpleType>
            </xs:element>

            <xs:element name="header">
                <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:pattern value="yes|no"/>
                </xs:restriction>
                </xs:simpleType>
            </xs:element>

            <xs:element name="refresh">
                <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:pattern value="no|([0-9])+"/>
                </xs:restriction>
                </xs:simpleType>
            </xs:element>

            <xs:element name="css" type="xs:string" />

        </xs:all>
        </xs:complexType>
        </xs:element>

    </xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

Quadro 7. Esquema das configurações de veículos de comunicação da superclasse Codificação, feito em XMLSchema

APÊNCIDE III: Configurações das Classes de Veículos de Comunicação: Fórum, Blog, Chat, Código Haskell e Código Prolog

Neste apêndice são listados as configurações em XML definidas para as classes de veículos de comunicação *Fórum*, *Blog* e *Chat*, da superclasse Listagem (Quadro 8 e Quadro 9), e os veículos *CódigoHaskell* e *CódigoProlog*, da superclasse Codificação (Quadro 10).

<pre><configurations> <treeOrder> <criteria>tree</criteria> <order>ASC</order> <printType>indented</printType> </treeOrder> <publication> <newArticle>yes</newArticle> <editArticle>no</editArticle> <removeArticle>no</removeArticle> </publication> <page> <allowView>author</allowView> <header>yes</header> <refresh>no</refresh> <css></css> </page> <permission> <allowPublish>author</allowPublish> <treeLevel>all</treeLevel> </permission> </configurations></pre>	<pre><configurations> <treeOrder> <criteria>tree</criteria> <order>DESC</order> <printType>indented</printType> </treeOrder> <publication> <newArticle>yes</newArticle> <editArticle>no</editArticle> <removeArticle>no</removeArticle> </publication> <page> <allowView>all</allowView> <header>yes</header> <refresh>no</refresh> <css></css> </page> <permission> <allowPublish>author</allowPublish> <treeLevel>1</treeLevel> </permission> <permission> <allowPublish>all</allowPublish> <treeLevel>2</treeLevel> </permission> </configurations></pre>
Fórum	Blog

Quadro 8. Configuração das classes de veículos *Fórum* e *Blog*.

```
<configurations>

  <treeOrder>
    <criteria>date</criteria>
    <order>ASC</order>
    <printType>flat</printType>
  </treeOrder>
```

```

<publication>
  <newArticle>yes</newArticle>
  <editArticle>no</editArticle>
  <removeArticle>no</removeArticle>
</publication>

<page>
  <allowView>all</allowView>
  <header>yes</header>
  <refresh>30</refresh>
  <css></css>
</page>

<permission>
  <allowPublish>author</allowPublish>
  <treeLevel>1</treeLevel>
</permission>

</configurations>

```

Chat

Quadro 9. Configuração da classe de veículos *Chat*.

<pre> <configurations> <coding> <mode>individual</mode> <language>haskell</language> <interpreter>hugs</interpreter> <edit>yes</edit> </coding> <page> <allowView>author</allowView> <header>yes</header> <refresh>no</refresh> <css></css> </page> </configurations> </pre>	<pre> <configurations> <coding> <mode>individual</mode> <language>prolog</language> <interpreter>gprolog</interpreter> <edit>yes</edit> </coding> <page> <allowView>author</allowView> <header>yes</header> <refresh>no</refresh> <css></css> </page> </configurations> </pre>
Código Haskell	Código Prolog

Quadro 10. Configuração das classes de veículos *Código Haskell* e *Código Prolog*.