

Dioraci Corrêa Junior

**EDITOR DE CONTEÚDOS PARA O SISTEMA DE  
ENSINO BASEADO EM COMPUTADOR RTUTOR**

Monografia apresentada ao Departamento de Ciências de Computação e Estatística do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como parte dos requisitos necessários para aprovação na disciplina Projeto Final.

São José do Rio Preto

Ano 2005

Dioraci Corrêa Junior

**EDITOR DE CONTEÚDOS PARA O SISTEMA DE  
ENSINO BASEADO EM COMPUTADOR TUTOR**

Monografia apresentada ao Departamento de Ciências de Computação e Estatística do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como parte dos requisitos necessários para aprovação na disciplina Projeto Final.

Orientador:  
Prof. Dr. Aleardo Manacero Junior

São José do Rio Preto

Ano 2005

Dioraci Corrêa Junior

**EDITOR DE CONTEÚDOS PARA O SISTEMA DE  
ENSINO BASEADO EM COMPUTADOR RTUTOR**

Monografia apresentada ao Departamento de Ciências de Computação e Estatística do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como parte dos requisitos necessários para aprovação na disciplina Projeto Final.

---

Prof. Dr. Alcardo Manacero Junior

---

Dioraci Corrêa Junior

Banca Examinadora:

Prof.<sup>a</sup> Dr.<sup>a</sup> Heloísa Helena Marino Silva

Prof.<sup>a</sup> Inês Aparecida Gasparotto Boaventura

São José do Rio Preto

Ano 2005

## **Dedicatória**

Dedico aos meus pais (Dioraci e Ana Maria), ao meu irmão (Jorge), a todos os amigos que me ajudaram com esse projeto e acima de tudo a Deus.

## **AGRADECIMENTOS**

Aos docentes deste instituto, em especial ao Prof. Aleardo Manacero Junior, pela atenção e orientação durante o período de desenvolvimento desse projeto e acima de tudo pela amizade construída durante todo o curso.

Aos meus pais pelos ensinamentos, preocupações e todo o esforço feito para me oferecer um suporte durante toda minha vida.

A Carine pelo apoio e compreensão, mesmo nos momentos mais difíceis.

A todos os amigos que conheci no Ibilce com os quais sempre pude contar, em especial ao Lucas Fugisawa.

Agradeço a todos que contribuíram direta e indiretamente para que fosse possível a conclusão deste projeto.

Enfim, agradeço a Deus pela minha vida, por todas as oportunidades a mim dadas e por todas as vitórias que Ele tem me permitido alcançar.

## RESUMO

Um dos principais problemas em sistemas de Ensino Baseado em Computador – EBC é que eles sobrecarregam o aluno com informações, desconsiderando suas preferências e seu estilo pessoal de aprender. Devido a este fato, dispomos hoje de ferramentas de EBC que se baseiam em Estilos de Aprendizagem, como o *RTtutor*, uma ferramenta de auxílio ao ensino de sistemas de tempo-real que utiliza o modelo de Kolb. Neste modelo os indivíduos são classificados em quatro estilos de aprendizagem, definidos a partir dos perfis individuais do aluno – Experimentação Ativa / Observação Reflexiva e Experiência Concreta / Conceitualização Abstrata.

No entanto, como acontece com a maioria das ferramentas que utilizam estilos de aprendizagem, a inserção de conteúdo no *RTtutor* é uma tarefa complexa. Essa inserção exige que o autor tenha conhecimento dos detalhes da implementação da ferramenta e da estruturação dos dados, para poder gerar corretamente um arquivo de conteúdo compatível com a ferramenta. Além disso, grande parte dos conteúdos gerados geralmente possuem redundâncias, pois existem estilos que compartilham características e conseqüentemente conteúdos.

O objetivo principal desse projeto foi desenvolver uma ferramenta de autoria e edição de conteúdo para o *RTtutor*, sabendo-se que nele os conteúdos são estruturados por padrões *XML* para evitar redundâncias enquanto se personalizam estilos. Como a base desse projeto foi o *RTtutor*, foram herdadas suas tecnologias de implementação (linguagem *Java*) e de estruturação de dados (estruturação do conteúdo em *XML*), que são tecnologias amplamente utilizadas atualmente.

As principais características desta ferramenta de editoração são sua portabilidade e facilidade de utilização e manutenção, proporcionadas pelas tecnologias empregadas. Isto simplifica, de uma forma geral, a edição de conteúdos no *RTtutor*. Testes realizados mostram uma boa eficiência na geração de conteúdos personalizados.

## **ABSTRACT**

A major issue with Computer Based Learning systems - EBC is that they overload the pupil with informations, ignoring its preferences and its personal style to learn. Nowadays, there are available EBC tools based on Learning Styles, such as the RTtutor, a teaching assistant tool for real-time systems courses that uses the Kolb's model. In this model the individuals are classified in four learning styles, defined from their profiles - Active Experimentation/Reflexive Observation and Concrete Experience/Abstract Conceptualization.

However, as it happens with the majority of the tools that use learning styles, the content insertion in RTtutor is a complex task. This insertion demands that the author has knowledge on details about the tool implementation and its data structure, in order to correctly generate content archives compatible with the tool. Moreover, some part of generated content generally contains redundancies, since styles share some characteristics.

The main goal on this project was the development of an authoring and editorial tool for the RTtutor, keeping in mind that it uses XML patterns to organize contents to avoid redundancies while customizing styles. Since this editorial tool is aimed to RTtutor, it inherited its technological specifications (as the Java programming language) and data structures (content organization in XML), which are well known technologies.

The main characteristics of this authoring tool are its portability and easiness of use and maintenance, which are results of the used technologies. This simplifies the RTtutor's content edition. Conducted tests show a good efficiency in the generation of customized contents.

## ÍNDICE

Lista de Figuras .....	iii
Lista de Tabelas.....	iv
Lista de Abreviaturas e Siglas.....	v
Capítulo 1 – Introdução.....	1
1.1 Objetivos.....	2
1.2 Descrição do texto.....	3
Capítulo 2 – Revisão Bibliográfica.....	4
2.1 Ensino Baseado em Computador – EBC.....	5
2.1.1 Tipos de EBC.....	6
2.1.2 O EBC e suas perspectivas.....	8
2.2 Estilos de Aprendizagem.....	9
2.2.1 Modelos de Estilo de Aprendizagem.....	10
2.2.2 O modelo Curry.....	11
2.2.3 Modelo de Kolb.....	12
2.2.4 Descrição dos Estilos.....	14
2.2.5 Projetando métodos de ensino - Modelo de Bernice McCarthy.....	17
2.3 A Ferramenta de Ensino <i>RTutor</i> .....	17
2.3.1 Tecnologia para implementação.....	19
2.3.2 Definição do tipo de documento (DTD).....	20
2.3.3 O documento <i>XML</i> .....	21
2.3.4 Linguagem extensível de folha de estilo ( <i>XSL</i> ) .....	21
2.4 Considerações finais.....	21

Capítulo 3 - Desenvolvimento do <i>StyleEditor</i> .....	23
3.1 Modelos de formatação para os estilos de Kolb.....	24
3.1.1 Divergentes – Reflexivo / Concreto.....	24
3.1.2 Assimiladores – Reflexivo / Abstrato.....	25
3.1.3 Convergentes – Abstrato/Ativo.....	26
3.1.4 Acomodadores – Ativo/Concreto.....	26
3.2 Construindo o questionário para levantamento de dados.....	27
3.3 A estruturação do conteúdo.....	29
3.3.1 O arquivo <i>XML</i> .....	30
3.4 A estrutura do <i>StyleEditor</i> .....	34
3.5 Considerações finais.....	37
Capítulo 4 – Resultados Obtidos.....	38
4.1 Funcionalidades Acrescentadas.....	39
4.2 Facilidade de uso e manutenção.....	39
4.3 Testes Realizados.....	40
4.4 Portabilidade.....	42
Capítulo 5 – Conclusões e perspectivas futuras.....	44
5.1 Conclusões.....	44
5.2 Perspectivas Futuras.....	45
Referências Bibliográficas.....	47

## LISTA DE FIGURAS

2.1	O esquema do <i>RTutor</i> .....	19
2.2	Exemplo de um DTD.....	20
3.1	Trecho do DTD especificado no <i>RTutor</i> .....	31
3.2	Trecho do documento <i>XML</i> .....	33
3.3	Trecho do <i>XSL</i> .....	34
3.4	Organização das classes do <i>Style Editor</i> .....	35
3.5	Ilustração da interface do <i>StyleEditor</i> .....	36
3.6	Ilustração da continuação da interface do <i>StyleEditor</i> .....	36
3.7	Atuação do <i>StyleEditor</i> no <i>RTutor</i> .....	37
4.1	Tela inicial do <i>StyleEditor</i> .....	40
4.2	Inserindo o conteúdo – Primeira questão.....	41
4.3	Inserindo o conteúdo – Segunda questão.....	41
4.4	Inserindo o conteúdo – Terceira questão.....	42
4.5	Trecho do documento <i>XML</i> .....	42

## **LISTA DE TABELAS**

2.1	Métodos de recepção e instrução e as novas tecnologias.....	8
-----	---	---

## LISTA DE ABREVIATURAS E SIGLAS

CSS	–	<i>Cascading Style Sheets</i>
DTD	–	Definição de Tipo de Documento
EBC	–	Ensino Baseado em Computador
GUI	–	<i>Graphical User Interface</i>
HTML	–	<i>Hyper Text Markup Language</i>
IDE	–	<i>Integrate Development Environment</i>
KDE	–	<i>K Desktop Environment</i>
LES	–	Laboratório de Engenharia de <i>Software</i>
LMS	–	<i>Learning Management System</i>
MS	–	<i>Microsoft</i>
XML	–	<i>Extensible Markup Language</i>
XSL	–	<i>Extensible Stylesheet La</i>

## Capítulo 1 – Introdução

Atualmente utilizam-se modernas tecnologias de comunicação e de informação no relacionamento humano e, de modo especial, no relacionamento professor-aluno. Um exemplo disso é a instrumentalização do processo de ensino/aprendizagem, com materiais informativos e/ou ambientes educacionais virtuais de apoio à aprendizagem, tendo como objetivo enriquecer o processo educacional.

Essa tecnologia, no entanto, não tem evitado que se transporte vários dos problemas dos métodos de ensino convencionais para as ferramentas de ensino baseadas em computador. Como exemplo, essas ferramentas empregam uma única abordagem sobre o conteúdo, abordagem esta que geralmente segue o modo que o professor gosta de aprender. Por outro lado, pode-se incorrer em outro erro tentando-se exibir um volume exagerado de informações, que abrangessem todas as preferências e padrões de aprendizagem encontrados nos estudantes.

Os profissionais de computação podem adotar posturas que alimentem este modelo vigente ou buscar novas posturas. Isso implica em encontrar alternativas técnicas para implementar métodos orientados por estudos sobre o comportamento humano, substituindo o grande fluxo de informações atual por um fluxo orientado de conhecimento.

Um grande desafio para desenvolvedores de sistemas de Ensino Baseado em Computador (EBC) é criar ambientes de aprendizado utilizando um fluxo orientado de conhecimento, que leve em consideração as individualidades do aluno, integrando conteúdos e abordagens diferentes para os vários estilos de aprendizagem. Isto gera um tipo de personalização na forma de apresentação do conteúdo, que se adequa ao estilo de aprendizagem de cada usuário/aluno.

É neste contexto que se insere o *RTtutor* [Cas02], um sistema de EBC criado para ser utilizado na disciplina de Sistemas de Tempo-Real, como ferramenta auxiliar ao ensino, juntamente com um simulador de escalonadores de sistemas de tempo-real – *RTsim*. O *RTtutor* está fundamentado no entendimento dos fatores humanos e, portanto oferecendo abordagens diferentes para estilos diversos, segundo o inventário de estilos de aprendizagem de Kolb [Kol76] que são os estilos Convergentes, Divergentes, Acomodadores e Assimiladores.

## 1.1 Objetivos

O objetivo deste projeto foi desenvolver uma ferramenta de autoria e edição de conteúdos – *StyleEditor* - que possa ser utilizado pelo professor para a inserção de conteúdo no sistema de ensino baseado em computador *RTtutor*. Esta ferramenta recebe de uma forma genérica e didática o conteúdo de um tópico, sem que o usuário seja um profundo conhecedor da arquitetura interna do sistema *RTtutor* e gera então um arquivo estruturado sobre o tópico definido, designando o conteúdo genérico, o conteúdo específico de cada estilo e o conteúdo comum a mais de um estilo.

Nesse projeto fez-se um aperfeiçoamento no protótipo *TutorEditor* desenvolvido em [Leo04], acrescentando mais informações e um melhor tratamento de conteúdo para os estilos já implementados, além do desenvolvimento dos conteúdos para os estilos acomodador e assimilador, completando assim os quatro estilos do modelo de Kolb.

## 1.2 Descrição do Texto

Apresenta-se inicialmente, no capítulo 2, os fundamentos teóricos sobre modelos de aprendizagem, que tratam as características individuais do aluno no momento de seu aprendizado, e diretrizes de como utilizar tais modelos para ensinar. Vê-se ainda nesse capítulo um panorama sobre EBC, expondo o cenário atual do EBC no mundo, explicitando os tipos de EBC utilizados nas ferramentas e as perspectivas futuras destes sistemas.

A seguir, no capítulo 3, é feito um levantamento das características apresentadas nos quatro estilos do inventário de Kolb: Convergentes, Divergentes, Assimiladores e Acomodadores. Em seguida, é detalhada a construção do questionário para levantamento de dados, segundo as características mencionadas anteriormente. Tal questionário deve ser respondido pelo usuário para que o sistema faça o levantamento dos dados a serem inseridos. Por fim é detalhada a implementação da ferramenta *StyleEditor*, a estruturação do conteúdo, como deve ser o arquivo do tipo Extensible Markup Language (*XML*) gerado e a estrutura das classes da aplicação.

No capítulo 4, são expostos os testes realizados com a ferramenta construída nesse projeto e os resultados obtidos na execução nas plataformas conhecidas, verificando como a ferramenta se comporta em cada uma, reforçando as características apresentadas pela mesma com relação à facilidade de uso e manutenção, portabilidade e desempenho.

Finalmente, no capítulo 5, apresenta-se as conclusões sobre o desenvolvimento desse projeto bem como perspectivas de novos projetos complementares, aprimorando ainda mais o uso das ferramentas *StyleEditor* e *RTutor*.

## Capítulo 2 - Revisão Bibliográfica

O crescimento da utilização de computação no ensino desafia os educadores a entender quais estudantes irão se adaptar e se beneficiar e quais estudantes terão dificuldade quando confrontados com um sistema de ensino aprimorado por métodos computacionais. Isso ocorre porque a maioria das ferramentas de ensino baseadas em computador existentes no mercado praticamente desconsideram as individualidades de cada aluno no momento do aprendizado. Desse modo o uso de tais sistemas recai sobre um problema comum em salas de aula, o professor ministra a disciplina da mesma forma como ele gostaria de que fosse passado para ele, seguindo o seu estilo de aprendizagem. No caso do ensino baseado em computador, este problema é agravado pelo fato de que, não é possível interromper o professor e exigir um novo enfoque do assunto, de acordo com a necessidade do aluno. Logo, alguns estudantes podem enfrentar dificuldades ao lidar com tais sistemas.

Neste capítulo, na seção 2.1, apresenta-se o cenário atual do ensino baseado em computador, mostrando-se alguns exemplos e apresentando algumas críticas a esse formato de aprendizado. Na seção 2.2, serão apresentados alguns modelos que são provenientes de pesquisa na área de estilos de aprendizagem. Na seção 2.3, serão apresentadas algumas tecnologias que tornam possível e eficiente a implementação

destes modelos. Nesse contexto há um enfoque no estudo da ferramenta de ensino de sistemas de tempo-real *RTtutor*, desenvolvida por Casagrande e Lopes [Cas02], que implementa tais estilos de aprendizagem e é a base da ferramenta que foi construída no atual projeto.

## 2.1 Ensino baseado em computador (EBC)

A utilização da informática na educação é um desafio permanente e acompanha o desenvolvimento dos computadores e dos sistemas computacionais. A utilização do computador como ferramenta auxiliar no ensino permeia todas as áreas da ciência da computação. O uso da informática no ensino de outras disciplinas acompanha a evolução das aplicações nas mesmas. O computador como ferramenta educacional complementar, e até autônoma, continua sendo objeto de pesquisas. Sistemas de ensino-aprendizagem baseados em computador (tutoriais, enciclopédias, livros, ou manuais eletrônicos) proliferam no mercado, prescindindo ainda de desenvolvimentos que os torne mais "amigáveis". A proliferação de redes de computadores coloca à disposição dos usuários e pesquisadores recursos computacionais abundantes, tanto locais como remotamente.[Mac05]

Algumas ferramentas de ensino baseado em computador já são empregadas hoje em dia no auxílio ao aprendizado, como por exemplo o Aulanet, que é um software LMS (*Learning Management System*) que possui uma eficiente plataforma de ensino. A ferramenta foi desenvolvida no Laboratório de Engenharia de Software - LES - do Departamento de Informática da PUC-Rio, em 1997.

O AulaNet se apóia nas premissas de que os cursos criados com essas ferramentas possuem grande capacidade de interatividade, de forma a atrair a participação intensa do aluno no processo de aprendizado (*learningware*); o autor do curso não precisa ser necessariamente um especialista em Internet, o AulaNet é uma ferramenta amigável; os recursos oferecidos para a criação de cursos devem corresponder aos de uma sala de aula convencional, acrescidos de outros normalmente disponíveis no ambiente *Web*; deve ser possível a reutilização de conteúdos já existentes em mídia digital, através, por exemplo, da importação de arquivos dos softwares padrão de mercado (ex. *Word, Power Point, Real*).[Edu05]

Outro exemplo de ferramenta de EBC que encontramos no mercado é o TelEduc. Um ambiente de ensino a distância pelo qual se pode realizar cursos através da Internet. Está sendo desenvolvido conjuntamente pelo Núcleo de Informática Aplicada à Educação (Nied) e pelo Instituto de Computação (IC) da Universidade Estadual de Campinas (Unicamp). O TelEduc foi concebido tendo como elemento central a ferramenta que disponibiliza atividades. Isso possibilita a ação onde o aprendizado de conceitos em qualquer domínio do conhecimento é feito a partir da resolução de problemas, com o subsídio de diferentes materiais didáticos como textos, software, referências na Internet, dentre outros, que podem ser colocadas para o aluno usando ferramentas como: Material de Apoio, Leituras, Perguntas Frequentes, etc.[Tel05]

Esses dois exemplos e muitos outros tais como o *WebCT* [Gol97], o *Lotus Learning Space* [Lot98] dentre outros disponíveis no mercado, se caracterizam por serem ferramentas que tratam o conteúdo de uma forma estática, não considerando os estilos de aprendizagem de cada aluno, apenas trabalhando o mesmo conteúdo com formas diferentes de exibição e interatividade.

Neste documento é usado o termo CBE ou EBC (Educação Baseada em Computador) para se referir ao método de ensino em que o assunto ou tema é apresentado principalmente por meio de um computador, em oposição ao ensino tradicional conduzido por um professor.

### 2.1.1 Tipos de EBC

Existem quatro níveis de ensino baseado em computador, cada qual definido pela complexidade da aplicação e seu nível de interatividade com o usuário [Dul96]:

- **Nível 1: Apresentação linear customizada:** Tipo de ensino similar a uma apresentação baseada no padrão *PowerPoint*, com alguma interatividade;
- **Nível 2: Apresentação não-linear conduzida por professor:** Ensino oferecido por um instrutor acompanhado pela navegação por todo o

conteúdo no computador, sem o uso de multimídia;

- **Nível 3: Ensino conduzido por um instrutor:** Apresentação multimídia acompanhada por educação baseada em sala de aula;
- **Nível 4: Ensino individualizado:** Apresentação multimídia que os praticantes usam independentemente com mínima assistência (também conhecida como educação *stand-alone*). Neste tipo de ensino os indivíduos podem aprender segundo seu próprio ritmo, tanto em laboratórios quanto em seus próprios computadores pessoais, e completar o teste provido pelo programa.

O enquadramento em um destes níveis é feito de acordo com quais dos critérios definidos abaixo a ferramenta de EBC consegue atender:

- Ser fácil de entrar e de sair;
- Prover um meio simples de navegação, isto é, um modo simples para seguir para frente e voltar;
- Ser consistente em suas convenções chave;
- Oferecer ajuda e indicadores sensíveis ao contexto;
- Prover mecanismos de informação de histórico (por exemplo, "Onde eu já estive?" "Onde estou agora?" "Quanto falta para terminar?");
- Oferecer marcadores (isto é, se sair agora é possível reiniciar mais tarde);
- Sempre oferecer uma saída.

De modo geral, o desenvolvimento de uma aplicação em EBC pode ser agrupada, de acordo com a classificação ambiental de Romiszowski [Rom81], como mostrado na tabela 2.1 a seguir.

Romiszowski propõe uma classificação para os métodos instrucionais de recepção e de descobrimento [Rom81] que podem, hoje, ser exemplificados com o uso das novas tecnologias. Desta forma estamos relacionando, na tabela 2.1, os tipos de aprendizagem descritos com o uso das redes, hipermídia, multimídia, tutores ferramentas de trabalho cooperativo, na busca de associar estas formas de aprendizagem com os recursos disponíveis para a educação.

Tabela 2.1 - Métodos de recepção e instrução e as novas tecnologias

<b>Tipo de Aprendizagem</b>	<b>Descrição</b>	<b>Uso de novas tecnologias</b>
<b>Descoberta imprevista</b>	<ul style="list-style-type: none"> <li>aprendizagem não planejada, nenhuma instrução é envolvida diretamente</li> </ul>	<ul style="list-style-type: none"> <li>busca livre na Internet.</li> </ul>
<b>Descoberta por livre exploração</b>	<ul style="list-style-type: none"> <li>abordagem de Bruner. Os macro objetivos são fixados, e os alunos ficam livres para explorar métodos, sub-objetivos, etc.</li> </ul>	<ul style="list-style-type: none"> <li>busca em programas hipermídia ou rede com tema definido.</li> </ul>
<b>Descoberta guiada</b>	<ul style="list-style-type: none"> <li>abordagem de Gagné. Os objetivos de cada passo da aprendizagem são fixados. O aprendiz é livre para explorar métodos, mas com guia e ajuda em cada estágio.</li> </ul>	<ul style="list-style-type: none"> <li>hipermídias adaptativas que privilegiam os interesses dos alunos</li> </ul>
<b>Descoberta adaptativa</b>	<ul style="list-style-type: none"> <li>abordagem cibernética<sup>1</sup>, guia e reforço corretivo s com estrutura de diagnóstico.</li> </ul>	<ul style="list-style-type: none"> <li>hipermídias adaptativas baseadas no estereótipo dos alunos e definição de nós a serem visitados.</li> </ul>
<b>Descoberta linear/intrínseca</b>	<ul style="list-style-type: none"> <li>direcionada rigidamente, guia e reforço são pré programados, baseados no estudante típico.</li> </ul>	<ul style="list-style-type: none"> <li>sistemas hipermídia com excursão definida.</li> </ul>
<b>Exposição indutiva</b>	<ul style="list-style-type: none"> <li>o aluno recebe o argumento, não tem que descobrir a regra. Pode ser programado.</li> </ul>	<ul style="list-style-type: none"> <li>multimídia.</li> </ul>
<b>Exposição dedutiva</b>	<ul style="list-style-type: none"> <li>a compreensão do problema é mostrada pela habilidade de aplicá-lo a exemplos. Pode ser programado. Abordagem baseada em Ausubel</li> </ul>	<ul style="list-style-type: none"> <li>multimídia adaptativa.</li> </ul>
<b>Aprendizagem de recepção direcionada (exercício e prática)</b>	<ul style="list-style-type: none"> <li>aprendizagem de fatos, sentenças e operações sem entender os conceitos envolvidos. Pode ser programado. Memorização</li> </ul>	<ul style="list-style-type: none"> <li>apresentação multimídia.</li> </ul>
<b>Aprendizagem de recepção imprevista</b>	<ul style="list-style-type: none"> <li>fatos e observações, originalmente não planejados, fornecidas por professores, outras fontes e estudantes.</li> </ul>	<ul style="list-style-type: none"> <li>ferramentas de trabalho cooperativo.</li> </ul>

### 2.1.2 O EBC e suas perspectivas

Dois outros métodos novos e bastante conhecidos de se oferecer ensino computadorizado merecem ser mencionados.

Um é o ensino à distância, que, em seu nível mais básico, ocorre quando professor e aluno estão separados por distância física, e a tecnologia é usada para

servir como ponte. Um professor, através de *e-mail* ou salas de bate-papo na Internet, pode auxiliar no progresso do ensino baseado em computador em um método de ensino à distância. Em um ambiente acadêmico, esta disposição é conhecida como uma "universidade virtual", utilizando-se de tecnologias de vídeo-conferência (*streaming*), o que permite uma interatividade muito alta entre ministrante (professor) e aprendizes (alunos).

Uma outra opção popular é o ensino em rede. Organizações que já tem feito investimentos pesados em redes de computadores estão estabelecendo ensino baseado em computador em suas redes para os seus funcionários acessarem em suas áreas de trabalho, através de cursos *on-line* auto-explicativos, que permitem uma grande interatividade entre o aluno e o computador, despertando a sua curiosidade e criatividade.

## **2.2 Estilos de aprendizagem**

A escola que freqüentamos, baseia-se no modelo educacional ainda predominante no nosso país, o da educação homogênea. À primeira vista esta visão pode parecer justa, mas se refletirmos um pouco mais, lembrando de algumas situações vividas por nós mesmos durante a vida escolar, podemos encontrar indícios de que a educação homogênea não atinge a todos de forma igual e equitativa. Ao contrário, lutamos o tempo todo para nos adaptar a um modelo de aprendizagem que freqüentemente não nos serve; tentamos desenvolver estratégias para lidar com essa dificuldade e aprendemos a conviver com o fracasso e os rótulos impostos pelas avaliações.[Lia05] Pesquisas sobre os estilos de aprendizagem deram aos educadores um novo sentido para realizar mudanças em suas salas de aula, nas formas de abordagem para trabalhos intelectuais, atividades alternativas, introduzindo projetos complexos como um esforço para criar oportunidades para os estudantes avaliarem o material usando o potencial de cada um.

Apesar da variedade de modelos, os conceitos de estilos de aprendizado vêm ganhando crescente atenção dos educadores. Estes fornecem uma caracterização suficientemente estável para planejar estratégias pedagógicas mais responsáveis em relação às necessidades dos estudantes, e parecem fornecer oportunidades melhores

de aprendizado, dando assim, um sentido novo ao ensino alternativo. A seguir estão listadas algumas recomendações gerais aos professores que aplicarem os estilos de aprendizagem em suas salas de aula:

- Cada pessoa é única, pode aprender, e tem um estilo de aprendizagem individual;
- Os estilos de aprendizado devem ser conhecidos e respeitados, são uma função da hereditariedade e experiência, impõem barreiras e limitações;
- Os estudantes absorvem conhecimentos de seus próprios estilos e dos outros;
- Os estudantes aprenderão melhor quando forem trabalhadas as preferências em que eles são bem sucedidos;
- Os estudantes serão mais completos e, portanto, melhores quando puderem expandir suas preferências;
- Quando o ensino acomodar várias preferências, mais alunos serão bem sucedidos;
- Os professores podem construir atividades que incluam preferências específicas e múltiplas de aprendizado

### **2.2.1 Modelos de estilos de aprendizagem**

Todos os modelos partem do princípio de que os estudantes aprendem melhor quando os ensinamentos são apresentados de maneira que eles se identifiquem e se sintam seguros. Se na sua orientação extraem a teoria de experiências concretas, então o seu aprendizado será melhor "fazendo" algo do que "refletindo" sobre os conceitos. Se seu estilo pessoal é orientado à abstração, então aprenderão melhor de forma abstrata. De fato, um indivíduo não pode fixar seus conhecimentos enquanto estes não lhe sejam apresentados na forma que ele confia.

Estes são alguns exemplos de individualidades no aprendizado, no entanto quando as experiências de aprendizagem se limitam a estas abordagens, os estudantes que se sentem seguros com os outros estilos certamente não serão tão bem sucedidos quanto os que se sentem seguros com estes estilos. Ambientes de aula

limitados inibem um ou mais grupos de alunos que preferem estilos que não são usados, prejudicando-os.

Uma instituição de ensino pode melhorar as possibilidades de aprendizado dos estudantes apenas aumentando o contato dos mesmos com variadas abordagens com as quais os estudantes possam se identificar, para isso é necessário que seus educadores identifiquem e conheçam as peculiaridades dos estilos de aprendizagem que cada aluno se encaixa. Neste caminho, o uso de tecnologia fornece novas potencialidades para a construção de ambientes de aprendizado em torno de estilos específicos.

Na seqüência, expõem-se dois estudos de estilos de aprendizagem de uma forma mais detalhada, o modelo de Curry que classifica os vários modelos de estilos de aprendizagem, de acordo com o que cada um considera para se identificar os estilos de cada indivíduo e o modelo de Kolb, que foi o modelo adotado para a implementação deste projeto.

### 2.2.2 O modelo Curry

Devido à variedade dos modelos usados para caracterizar os estilos de aprendizagem, o modelo de Curry [Cur87], que pesquisa as diferenças da aprendizagem humana, é útil. Curry divide estes estudos em três níveis, que podem ser examinados como as diferentes camadas de uma "cebola", utilizando-se da combinação das características nos diversos níveis para explicar as variações humanas. Recentemente, a "cebola" foi re-dividida [Cur90] nos quatro níveis descritos abaixo:

- **Nível de personalidade:** é o centro da "cebola", representando como a personalidade, em nível mais profundo, fornece a forma e a orientação com as quais o indivíduo examina o mundo. Um dos modelos mais conhecidos é o de Myers-Briggs [Mye62]. Este modelo categoriza os indivíduos como extrovertidos/introvertidos, sensoriais/intuitivos, racionais/sentimentais, e decididos/indecisos. A classificação dos indivíduos ao longo da escala indica suas tendências e atitudes perante o mundo. Modelos desse nível enquadram as

preferências pessoais na forma de ser. A questão não se resume simplesmente a treinar habilidades ou atitudes, mas reconhecer que os indivíduos agem de modo fundamentalmente diferente na sala de aula. Deste modo, não é possível obter o sucesso no ensino sem considerar os motivos que objetivaram o ingresso do estudante no curso ou disciplina.

- **Nível de processamento de informação:** é a camada intermediária da "cebola", e representa como as informações são assimiladas, ou seja, classificadas, armazenadas e utilizadas. A idéia mais conhecida sobre as diferenças no processamento de informações é a discussão sobre os lados direito/esquerdo do cérebro. Um modelo mais complexo é o de Kolb [Kol76], que é um estudo experimental sobre o aprendizado, sendo também um dos modelos mais utilizados.
- **Nível de interação social:** consideram-se as maneiras como pessoas inseridas em contextos sociais específicos usam certas estratégias. Um modelo bem conhecido é o de William Perry [Ctl05], que mostra como os estudantes universitários desenvolvem-se através de diferentes níveis de maturação intelectual de acordo com o que vivem na faculdade.
- **Nível de preferências instrutivas e ambientais:** é referente aos traços mais visíveis, ou seja, as preferências ambientais do estudante no momento do aprendizado. O modelo de Dunn & Dunn [Ctl05], por exemplo, identifica cinco dimensões que marcam as várias preferências, que são as preferências ambientais, emocionais, sociais, fisiológicas e psicológicas.

### 2.2.3 Modelo de Kolb

O modelo de Kolb [Kol76] trabalha como um inventário de estilos de aprendizagem para fazer a identificação desses, que é composto de algumas sentenças às quais estão associadas alternativas tais como:

---

1. Quando aprendo	a. Gosto de trabalhar meus sentidos b. Gosto de ouvir e escutar c. Gosto de pensar nos conceitos d. Gosto de estar fazendo coisas
2. Aprendo melhor a partir de	a. Relacionamentos pessoais b. Observação c. Teorias racionais d. Uma chance de praticar e experimentar

---

O estudante deve responder estas sentenças atribuindo a cada alternativa um peso de acordo com o que ele acredita que melhor descreve suas atitudes e sentimentos no momento em que ele está aprendendo. A partir dos pesos são calculados quatro índices, as quatro dimensões do inventário de estilos de aprendizagem de Kolb, que são: experiência concreta, conceitualização abstrata, observação reflexiva e experimentação ativa. O significado destes índices é visto a seguir:

- **Experiência concreta ou Concreto:** Kolb estabelece que um alto índice em experiência concreta representa uma receptividade à abordagem baseada em experiências, de modo que o aprendizado se baseia em ponderações baseadas em sentimentos. Os indivíduos deste estilo tendem a ser empáticos. Eles geralmente acham abordagens teóricas inúteis e preferem tratar cada situação como um caso único. Aprendem melhor por meio de exemplos específicos nos quais se sintam envolvidos. Estes estudantes tendem a se relacionar melhor com outros estudantes, do que com uma autoridade como o professor. Aulas teóricas não são muito úteis, enquanto que o trabalho em grupo e em pares conduzem frequentemente ao sucesso. As atividades devem ser planejadas para desenvolver as capacidades que devem ser aprendidas. O instrutor deve agir como um treinador/técnico para este aprendiz autodidata.

- **Conceitualização abstrata ou Abstrato:** um alto índice em conceitualização abstrata indica um modo de aprendizado analítico e conceitual, que se baseia pesadamente em raciocínio lógico. Estes indivíduos tendem a serem mais orientados a coisas e a símbolos, do que as outras pessoas. Aprendem melhor quando orientados por uma autoridade de modo impessoal, com ênfase teórica e análise sistemática. Eles se sentem frustrados e aprendem pouco pelo aprendizado através de descobertas de modo desestruturado, como em exercícios e simulações. Estudos de caso, aulas teóricas e exercícios de reflexão ajudam este estudante, de resto, pouca coisa pode ajudá-lo.
- **Observação reflexiva ou Reflexivo:** um alto índice em observação reflexiva indica uma abordagem por tentativas, imparcial e reflexiva. Estes indivíduos aprendem baseando-se fortemente em cuidadosas observações e fazendo julgamento das mesmas. Eles preferem aprender assistindo aulas, o que lhes dá a possibilidade de exercer o seu papel de observador e juiz imparcial; tendem a serem introvertidos. Aulas são úteis a este estudante que espera que o instrutor atue como um especialista, expondo seu ponto de vista e fornecendo atividades e orientações. Este estudante quer que seu desempenho seja medido por critérios diferentes dos seus.
- **Experimentação ativa ou Ativo:** um alto índice em experimentação ativa indica uma disposição forte em realizar atividades práticas. Estes indivíduos aprendem mais facilmente quando participam de projetos práticos, discussões em grupo e fazendo tarefas em casa. Eles não gostam de situações de aprendizado passivo como assistir a aulas, e tendem a ser extrovertidos. Resolver problemas, pequenos grupos de discussões ou jogos, discussões em dupla e estudos autodirecionados ajudam este estudante, que gosta de ver tudo e de determinar seus próprios critérios para a relevância dos materiais.

#### 2.2.4 Descrição dos estilos [Cas02]

Kolb denomina os estilos de aprendizagem como: Divergente ("Diverger"), Assimilador ("Assimilator"), Convergente ("Converger") e Acomodador ("Accomodator"), que são indexados segundo as dimensões definidas acima. Segue abaixo uma descrição dos estilos de aprendizagem baseando-se nesses índices:

- **Divergentes (concreto, abstrato):** a questão típica deste estilo de aprendizagem é "Por quê?". Os estudantes deste estilo respondem bem à explicação de como o material estudado se refere às suas experiências, interesses e carreiras futuras. Divergentes captam as informações através de experiência concreta, baseando-se nos sentimentos, precisam expressar seus sentimentos quando estão aprendendo, procurando por significados pessoais sobre o que aprendem, e querem interagir com professores e colegas. Eles precisam ser convencidos de que o material estudado é importante para eles como pessoas e/ou para outras pessoas as quais eles sintam empatia ou se identifiquem. Aprendem através de discussões e são interessados em como as informações afetam as pessoas. Trabalham para manter o grupo harmonioso e são chamados divergentes porque podem ver as coisas de diferentes perspectivas e geram novas idéias facilmente. Assim saem-se muito bem em reuniões do tipo "*brainstorming*". Para obter sucesso com estes estudantes deve-se atuar como motivador, motivando-lhes a aprender o conteúdo e mostrando a eles como o conteúdo se encaixa em suas vidas. Divergentes querem interagir pessoalmente com seus instrutores e serem reconhecidos como indivíduos. Assim, o instrutor deve testemunhar o crescimento pessoal do estudante.
- **Assimiladores (abstrato, reflexivo):** a questão típica deste estilo de aprendizagem é "O que?". Estudantes deste estilo são bem sucedidos quando as informações são apresentadas de maneira organizada e lógica, e obtêm aproveitamento melhor se tiverem tempo para refletir. Assimiladores captam as informações através de conceitualização abstrata e as processam através de observação reflexiva. Eles gostam de informações que justifiquem o que estão aprendendo, querem saber o que

os especialistas pensam e procuram compreender os conceitos. São bem sucedidos na escola tradicional. De fato, a maioria dos estudantes, apesar de terem seus estilos próprios, se adaptam ao estilo do assimilador para obter sucesso na escola. Eles são chamados assimiladores porque se utilizam de pedaços de informações, analisam, organizam, e assimilamos como um todo. Gostam de organizar, e têm tendência a serem detalhistas. Tem sucesso nos procedimentos quando se baseiam na literatura. Nas suas abordagens são cuidadosos, metódicos e cautelosos, tentando evitar erros. Eles aprendem bem lendo sobre os tópicos e preferem não trabalhar em grupos, exceto para seguir normas. Para ser efetivo, o instrutor deve unicamente ser uma autoridade no assunto e uma fonte de informações. Assimiladores gostam de assistir aulas e/ou palestras e preferem que o professor explique o conteúdo de forma organizada e precisa.

- **Convergentes (abstrato, ativo):** a questão típica deste estilo de aprendizagem é "Como?". Os estudantes deste estilo gostam de ter a oportunidade de trabalhar ativamente em tarefas bem definidas, e de aprender por tentativa e erro em um ambiente que permita-lhes errar com segurança. Os convergentes captam as informações através de conceitualização abstrata e as processam de forma ativa. Gostam de testar as informações, tentar coisas, perceber coisas à parte, ver como as coisas funcionam, e aprender fazendo. Testam a viabilidade e utilidade das informações. Assim, não gostam de permanecerem sentados por um longo período, assistindo aulas ou lendo uma grande quantidade de material. Eles dão ênfase a fazer as coisas sem perder tempo; são aprendizes que convergem mais rapidamente ou chegam rapidamente a tomar decisões, que procuram por uma resposta correta, e que cortam caminho, passando somente pela parte mais essencial da matéria. Gostam de exemplos a seguir, em detrimento de assistir a aulas ou palestras. O ambiente de laboratório é ideal para eles. Tendem a ver o trabalho em grupo e as discussões como perda de tempo, porque podem fazer o trabalho mais rapidamente de forma individual. Para obter sucesso com os convergentes, o instrutor deve trabalhar como um técnico, provendo

práticas orientadas e respostas. Esses estudantes preferem que o professor os ajude a colocar as lições em prática. O instrutor precisa se tornar menos ativamente envolvido, agindo como uma fonte de informações e permitir ao convergente ter um papel mais ativo.

- **Acomodadores (concreto, ativo):** a questão típica deste estilo de aprendizagem é "E se?". Os estudantes deste estilo gostam de aplicar as informações em novas situações para resolver problemas reais. Os acomodadores captam as informações através de experiências concretas e processam-nas através de experimentação ativa. Eles são entusiastas e preferem aprender através de descobertas próprias. Seguem seus próprios horários e ordem quando estudam e não gostam de muitos procedimentos e regras. Por outro lado, gostam de interação com outras pessoas, de discussões em grupo, de obter informações e criar algo novo com elas, e ainda de descobrir coisas por sua própria conta. O papel do instrutor é o de avaliador e revisor. Esses estudantes precisam do professor para encorajá-los às descobertas, permitindo-lhes se auto-instruírem. Eles preferem alguma interação com o instrutor, mas pouca supervisão e querem que o instrutor permaneça em segundo plano, para servir como um recurso extra, e para avaliar o que estão fazendo.

### **2.2.5 Projetando métodos de ensino - Modelo de Bernice McCarthy [Mcc05]**

Bernice McCarthy usou o inventário de estilo de aprendizagem de Kolb para criar um modelo que descrevesse os estilos de ensinar e as atividades requeridas para satisfazer aos estilos de aprendizagem descritos por Kolb. Quando a maioria dos estudantes de uma classe são divergentes e assimiladores, o instrutor deve assumir um papel ativo de liderança e quando a maioria dos estudantes são convergentes e acomodadores o instrutor deve deixá-los assumir um papel mais ativo.

McCarthy explica também que o papel do instrutor é diferente para cada estilo de aprendizagem.

Para o divergente, o instrutor deve trabalhar motivando o aluno, mas ao trabalhar com assimiladores ele deve trabalhar expondo conceitos. O convergente requer

práticas e acompanhamento pessoal, enquanto que o acomodador necessita integrar aplicações com experiências.

### **2.3 A ferramenta de ensino *RTutor***

O *RTutor* [Cas02] é composto por dois módulos, que são o servidor e o cliente (arquitetura vista na figura 2.1). O servidor armazena e recupera os dados dos usuários e do conteúdo do curso. Já o cliente cuida da apresentação do curso ao usuário. Na descrição de seu funcionamento nos próximos parágrafos, os números entre parênteses se referem aos momentos de execução apresentados na figura 2.1.

O servidor é responsável por recuperar informações referentes ao aluno e ao conteúdo do curso, enviando-os quando forem requisitados. O conteúdo do curso é enviado por partes (seções) de acordo com as requisições do aluno efetuadas pela "árvore de seleção" (8) da janela do gerenciador de conteúdo (13).

Para o conteúdo ser utilizado pelo gerenciador ele deve estar no formato *Hyper Text Markup Language (HTML)*, portanto o servidor tem como uma de suas tarefas efetuar a transformação e a seleção do que será visualizado pelo aluno, a partir de um documento *XML*, em um documento *HTML*.

Este processo, por ser um servidor, é caracterizado pela escuta de requisições (2) através da rede, atendendo-as e instanciando classes que assumirão as conexões (9). A partir da instanciação desta classe o servidor volta a escutar por novas requisições.

O cliente é responsável por requisitar a validação do usuário no servidor (1 e 2), recuperando os dados sobre o andamento do curso, os quais serão utilizados pelo gerenciador de conteúdo (12). Ele é responsável também por recuperar e exibir o conteúdo do curso segundo as necessidades e o estilo do usuário.

Iniciando o servidor é possível que os clientes efetuem conexões e iniciem o uso da ferramenta de ensino. A partir da janela de *login* (1) é possível configurar o endereço e a porta em que o servidor atende. Em seguida com a entrada de nome de usuário e senha instancia-se um objeto da classe que será utilizada para efetuar o pedido de conexão ao servidor.

Ao receber o pedido de conexão o servidor instancia um processo (*thread*), que efetua uma chamada a um objeto para recuperação dos dados do cliente (10). Este *thread* também assumirá a negociação de serviços com o cliente.

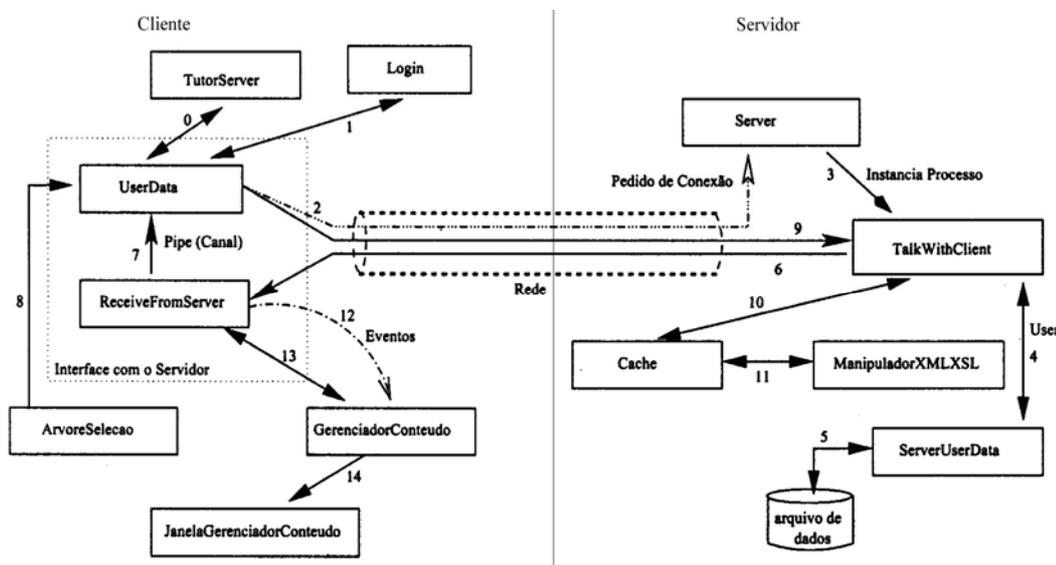


Figura 2.1: O esquema do RTtutor

### 2.3.1 Tecnologia para implementação

Como se viu anteriormente, fazendo uso de estilos de aprendizagem tem-se características comuns entre estilos diferentes. Como por exemplo, no modelo de Kolb, entre os estilos acomodador e divergente há em comum a preferência por experiências concretas. Assim, uma tecnologia de implementação do conteúdo que não ofereça suporte à definição de trechos de conteúdo comuns para estes casos, certamente ocasionaria redundâncias, dificultando tanto a criação quanto a manutenção do conteúdo.

Deste modo, é necessário a criação de uma estrutura que possibilite a definição dos elementos genéricos, elementos de grupos e específicos. Isso pode ser feito através da criação de uma linguagem para essa estruturação ou poderia se utilizar uma linguagem que permita a extensão de conteúdo de modo conveniente.

Nesse contexto, se insere a linguagem de marcação extensível (*XML*) [Dec00] [Ray00] [Arm02]. Vê-se brevemente a seguir como a linguagem *XML* é formada, apresentando alguns conceitos importantes para o entendimento do projeto.

A linguagem *XML* possui uma construção diferente das outras linguagens de marcação, tal como a *Hyper Text Markup Language (HTML)*, pois estas linguagens não definem nenhuma estrutura para o armazenamento de conteúdo, apenas armazenam a formatação com que o conteúdo será apresentado. Como exemplo de formatação cita-se cores e fontes utilizadas para a apresentação. Já o *XML* segue o conceito de definir uma gramática que deve ser obedecida ao se escrever o conteúdo.

Para que isso seja alcançado os conteúdos apresentados com essa linguagem são divididos em três partes, definição de estruturas, dados e regras de transformação, podendo estar em um mesmo arquivo ou em arquivos separados. A seguir é feita uma descrição destas partes e alguns de seus componentes importantes.

### 2.3.2 Definição do tipo de documento (DTD)

A DTD é uma parte da definição da linguagem *XML* com a qual se pode especificar a estrutura do documento *XML*. Não é necessário que alguma estrutura seja definida, porém se isso não ocorrer é impossível validar o documento *XML*. É com a DTD que se especifica quais elementos estarão presentes no documento e quais as estruturas desses elementos, sua ordem de ocorrência, seu conteúdo e atributos. Deve-se perceber, no entanto, que a DTD não define nada sobre a exibição de informações.

As declarações presentes na estruturação de um documento podem ser: os elementos e suas respectivas estruturas, os atributos de cada elemento, entidades presentes no documento e as notações.

Na figura 2.2 a seguir temos um trecho de um DTD exemplo.

```
<!ELEMENT Capitulo (TítuloCap, DescriçãoCap, Seção+)>
(!ATTLIST Capitulo
  id ID #REQUIRED
  proprietario(genérico | assimilador | divergente | acomodador |
               convergente | acomodador_divergente |
               divergente_assimilador | assimilador_convergente |
               convergente_acomodador) #REQUIRED>
<!ELEMENT TítuloCap (#PCDATA | %textohtml;)*>
<CompSec proprietario="genérico">
  <dl>
    <dt>Crítica</dt>
```

```
<dd>em que o atraso em seu atendimento pode acarretar graves  
conseqüências para o sistema.</dd>  
</dl>  
</CompSec>
```

Figura 2.2. Exemplo de um DTD

### 2.3.3 - O documento *XML*

É no documento *XML* que as informações estão armazenadas, e este documento deve ser escrito de acordo com o que foi especificado na DTD. Caso isso não ocorra o documento *XML* não será válido. Porém o documento *XML* pode ser escrito sem a presença de uma DTD e neste caso ele somente será inválido se houver quebra da hierarquia das marcações. Deve-se notar ainda que mesmo com a presença da DTD é necessário que não haja quebra da hierarquia.

### 2.3.4 Linguagem extensível de folha de estilo (*XSL*)

O processo de filtragem das informações que serão apresentadas, como também as transformações necessárias para o controle do processo de apresentação das informações são efetuadas através da linguagem extensível de formatação de páginas (*XSL*).

Existe um outro modo de apresentar as informações contidas no *XML*, utilizando-se folhas de estilo em cascata – *Cascade Style Sheets (CSS)* que atua percorrendo o documento e aplicando os formatos definidos sem a possibilidade de incluir qualquer alteração.

Já o *XSL* processa o documento de maneira programável, ou seja, é possível pela aplicação de regras formatar as informações contidas nos elementos e nos atributos presentes no documento *XML*, bem como incluir informações novas, ignorar parte das informações e filtrá-las baseando-se em desvios de fluxo de acordo com testes aplicados sobre as informações.

## 2.4 Considerações Finais

Para se ter eficiência ao ensinar é preciso utilizar metodologias de ensino que levem em consideração o estilo pessoal de aprendizagem do aluno. Assim, buscou-se na literatura modelos que fossem adequados à proposta de uma ferramenta computacional para o auxílio ao ensino.

Para colocar em prática as abordagens propostas pelos modelos presentes na literatura buscou-se tecnologias que fornecessem suporte para os diversos estilos mapeados por estes modelos, atendendo os requisitos de implementação de modo a evitar a redundância de informações, visto que há parcelas de informações que são comuns entre os estilos definidos pelo mesmo modelo.

Com base nestas informações, implementou-se a ferramenta de EBC *RTutor* [Cas02], que considera os estilos de aprendizagem do aluno, adaptando o conteúdo ministrado de forma individualizada, considerando o perfil do aluno segundo o inventário de Kolb [Kol76]. Na seqüência, implementou-se um protótipo de uma Ferramenta de Inserção de Conteúdo no *RTutor* [Leo04]. Este inseria apenas o conteúdo dos estilos Convergentes e Divergentes. Nesse projeto foi reformulada a implementação desta ferramenta adicionando-se recursos para a inserção de conteúdo para todos os estilos de aprendizagem além de aumentar e melhorar a forma de inserção de conteúdo implementada em [Leo04]. Esta ferramenta, denominada Style Editor é descrita no próximo capítulo.

## Capítulo 3 - Desenvolvimento do *StyleEditor*

Neste capítulo é realizada a descrição do desenvolvimento e implementação da ferramenta de geração e edição de conteúdos para o sistema de ensino baseado em computador que utiliza estilos de aprendizagem *RTutor* [Cas02] – o *StyleEditor*. Inicialmente, descreve-se a modelagem para aplicação de estilos de aprendizagem no ensino baseado em computador segundo a forma de conteúdo suportado pela ferramenta *RTutor*.

Tendo o modelo de aplicação, descreve-se o desenvolvimento do questionário para levantamento de dados, que será respondido pelo professor, levando em consideração o modelo de estilo de aprendizado definido na primeira fase.

A seguir descreve-se a geração do arquivo estruturado de conteúdo em *XML*, arquivo responsável pela interface entre a ferramenta desenvolvida e o sistema *RTutor*.

Finaliza-se este capítulo com a descrição do processo de desenvolvimento das interfaces gráficas – *Graphical User Interface (GUI's)* – do sistema de Geração de Conteúdo, que está diretamente ligada ao arquivo *XML* já mencionado.

### 3.1. Modelos de formatação para os estilos de Kolb

A ferramenta desenvolvida, o *StyleEditor*, seguindo o paradigma da ferramenta de ensino *RTtutor*, foi implementada na linguagem *Java* que oferece diversas vantagens, a principal delas é a portabilidade.

O aplicativo *StyleEditor* constitui-se de uma ferramenta que é representada principalmente por uma interface de entrada de informações. O objetivo principal deste aplicativo é obter os dados de conteúdo do usuário (o autor da matéria da disciplina), sendo a inserção de dados dirigida por um questionário sobre um determinado tópico. O programa obtém as informações através dos campos correspondentes a cada questão, e constrói um arquivo estruturado sobre o tópico definido, designando o conteúdo genérico, o conteúdo específico de cada estilo e conteúdo comum a mais de um estilo. Por questões de compatibilidade de conteúdo com a ferramenta *RTtutor*, o arquivo estruturado é um arquivo escrito em *XML*, que permite exatamente esta estruturação definida de conteúdo.

O projeto da ferramenta *StyleEditor* é direcionado para a geração de conteúdo baseando-se no inventário de estilos de Kolb, que é utilizado pela ferramenta de ensino baseado em computador *RTtutor*.

O modelo de Kolb foi seguido na implementação da ferramenta, para que os arquivos de conteúdo gerados por ela possam ser reconhecidos e utilizados pela ferramenta *RTtutor*.

Através de um estudo do inventário de Kolb foram identificadas as individualidades de cada estilo de seu inventário de estilos, que nos forneceu uma base para o desenvolvimento das questões utilizadas para o levantamento dos dados. Esse estudo é explicitado nas subseções abaixo:

#### 3.1.1 Divergentes – *Reflexivo / Concreto*

Para este estilo de aprendizado o instrutor deve atuar motivando os estudantes. Motivá-los a aprender o material e mostrar-lhes como este se enquadra com o que esperam. Deve-se mostrar aos alunos a utilidade do que eles aprenderam ou vão aprender, sempre com o intuito de entusiasamá-los e motivá-los.

Os alunos que possuem este estilo de aprendizado querem interagir pessoalmente com o instrutor e serem reconhecidos como únicos. O educador precisa entusiasmar o crescimento pessoal dos estudantes. Além disso, aprendem bem através de discussões e costumam ver as coisas de forma diferente nas discussões em grupos.

Segundo estas características, o material deve ser apresentado juntamente com questões que façam os alunos pararem para pensar. O material deve apresentar, de acordo com as decisões (respostas) dos alunos, as justificativas das respostas apresentadas. Em um grupo de discussão, esta poderia ser a explicação de outras pessoas.

Como esses alunos gostam de ver as coisas de perspectivas diferentes é importante que o material apresente estas perspectivas, fazendo-os pensarem melhor sobre o assunto. Assim, nos testes, pode-se apresentar afirmações seguidas de uma questão e várias alternativas, justificando as afirmações de modo que, para escolher uma resposta, eles tenham que avaliar os vários pontos de vista da questão.

Devido ao fato de alunos deste estilo gostarem de trabalhos em grupos, é necessário que haja uma maneira deles interagirem com outros estudantes do mesmo estilo durante o estudo e principalmente durante a resolução de problemas. E desta forma, ao decidir em grupo quem possui a razão sobre determinado assunto, eles estarão observando todas as perspectivas, solidificando assim seus conhecimentos.

### **3.1.2 Assimiladores - *Reflexivo /Abstrato***

Assimiladores aprendem da forma tradicional de ensino, em que a matéria deve ser bem organizada e detalhada. O instrutor deve agir como um especialista, sendo autoritário e fornecendo informações. Grupos devem ser evitados. O educador deve ser cauteloso para a forma de abordagem não fugir ao estilo de aprendizado dos alunos, pois como estes alunos seguem procedimentos, eles podem executar as tarefas mesmo que estas não sejam ideais para seu estilo, tornando difícil identificar abordagens incorretas.

O material deve ser apresentado de forma sequencial, completa e bem organizada, e deve trazer vários exemplos e justificativas para os mesmos, pois

representam o ponto de vista de especialistas. Exercícios podem ser apresentados ao final de cada sessão, mas estes exercícios não devem ser extensos, complexos, ou muito práticos, devem apenas ajudar a assimilar o conteúdo. Exercícios tipo teste serão bem aceitos aqui, pois este tipo de aluno gosta de ler e observar, processando a informação pensando sobre ela e não colocando-a em prática.

Ao final dos capítulos podem ser apresentados exercícios mais complexos, que façam os alunos refletirem sobre o que eles acabaram de aprender.

### **3.1.3 Convergentes - *Abstrato /Ativo***

Os convergentes precisam que o educador trabalhe como um técnico, propondo trabalhos práticos e exigindo um retorno ou resposta dos alunos. Os estudantes precisam ter um papel mais ativo, pois não gostam de ler muito, preferindo laboratórios onde possam aprender por tentativa e erro, aprendendo ao fazer. Acreditam que grupos são desperdício de tempo.

O material deve ser apresentado de forma resumida com apenas o conteúdo que é essencial, porém deve ser possível aos alunos buscarem informações complementares em um material mais completo, que não será utilizado como texto principal e sim como um texto de consulta.

### **3.1.4 Acomodadores – *Ativo/Concreto***

Para trabalhar com este estilo o instrutor deve permitir que os alunos desenvolvam o auto-aprendizado. O instrutor deve avaliar, orientar, e incentivar a auto-descoberta.

Estudantes desse estilo gostam de grupos e de resolver problemas, tem sua própria ordem e horário para as tarefas e não gostam de muitos procedimentos e regras. Costumam correr riscos e aprendem a partir dos erros.

Desta forma a abordagem para este grupo deve ser como um material que não siga uma sequência fixa, apesar de poder ser sugerida. Assim os alunos podem estudar os módulos de acordo com sua própria vontade, até que o material seja totalmente completado. Como alunos deste estilo gostam de grupos, pode-se criar

grupos de discussão fictícios que discutam a matéria, tal como foi sugerido ao estilo divergente. Porém, como esse estilo processa a informação de forma diferente dos divergentes, a forma das discussões deve ser diferente. Além disso, deve haver exercícios práticos, como foi sugerido ao tipo convergente. Os exercícios para este tipo de aluno devem mesclar grupos de discussões com trabalhos individuais, devendo ser os mais práticos e reais possíveis.[Leo04]

### 3.2 Construindo o questionário para levantamento de dados

Os estilos possuem características em comum (por exemplo, o divergente compartilha a preferência por observação reflexiva com o assimilador e a preferência por experimentação concreta com o acomodador), logo é preciso apresentar partes do material que são iguais a dois ou mais estilos. Nesse contexto tem-se a necessidade de fazer com que as informações comuns a estilos adjacentes apareçam como uma abordagem comum aos dois grupos. Existem ainda características comuns a todos os estilos, ou seja, genérica, que também deve ser exibida da mesma forma a todos os estilos.

Nesse projeto foi feito um aperfeiçoamento no protótipo desenvolvido em [Leo04], acrescentando mais informações e um melhor tratamento de conteúdo para os estilos já implementados, além do desenvolvimento e implementação dos conteúdos para os outros estilos do inventário de Kolb. Dessa forma, construiu-se vinte questões que abrangem todos os estilos do inventário e compõem a interface do *StyleEditor*, sendo que a ferramenta extrai o conteúdo a ser inserido no *RTutor* através das respostas a estas questões. Tais questões estão listadas a seguir, exemplificando em algumas como elas se encaixam no modelo em questão:

- *Qual é o tópico?*
- *Qual o motivo que originou o tópico?*
- *Descreva o tópico sucintamente:*

Estilos Divergentes e Acomodadores, pois em ambos os estilos o professor deve agir em segundo plano, permitindo que os estudantes raciocinem por si mesmos.

- *Um “overview” completo do tópico em questão:*

Estilos Assimiladores e Convergentes, pois em ambos os estilos o professor deve ser uma autoridade no assunto, com grande conhecimento técnico.

- *Uma descrição conceitual rápida:*

Estilo Divergente, onde para este estilo de aprendizado o instrutor deve atuar motivando os estudantes.

- *Uma descrição conceitual detalhada:*

- *Justifique a descrição do item acima:*

- *Um exemplo de aplicação prática do tópico:*

Estilos Divergentes e Acomodadores, ambos os estilos gostam de interagir com o grupo a fim de aplicar à prática o conteúdo que está sendo visto.

- *Uma segunda abordagem para este exemplo:*

- *Um segundo exemplo de aplicação prática para este tópico:*

- *Uma questão para a discussão coletiva:*

Estilos Divergentes e Acomodadores, pois ambos preferem discussões em grupos.

- *Um parecer sobre a questão discutida acima, uma opinião:*

- *Uma segunda questão para a discussão coletiva:*

- *As respostas corretas para as duas questões coletivas acima:*

- *Um exercício prático a ser trabalhado individualmente:*

Esse item está relacionado com os Convergentes, onde os estudantes precisam ter um papel mais ativo, não gostam de ler muito, preferindo laboratórios onde possam aprender por tentativa e erro e acreditam que grupos são desperdício de tempo.

- *A conclusão desse exercício prático:*

- *Um segundo exercício prático a ser trabalhado individualmente:*

- *A conclusão desse segundo exercício prático:*

- *Uma conclusão sobre a abordagem do tópico:*

Essa questão, por exemplo, está relacionada com o estilo Divergente, pois alunos deste estilo precisam ser convencidos de que o material

apresentado é útil, necessitando assim de uma conclusão das idéias mostrando isso.

- *Uma bibliografia alternativa para o tópico em questão:*

Foi definido também que a aplicação permitiria ao autor do conteúdo criar e editar informações referentes a um tópico, sendo que esta informação seria armazenada em um arquivo de dados comum (como por exemplo um arquivo temporário) que seria capaz de armazenar e recuperar as informações inseridas, para após as modificações feitas, os dados serem recolocados no arquivo *XML*.

### 3.3. A estruturação do conteúdo

Como visto anteriormente, os diversos estilos devem possuir conteúdos adequados às suas necessidades, juntamente com conteúdos comuns a dois ou mais estilos (genéricos).

Para a ferramenta *StyleEditor* suportar a implementação destas características e evitar problemas, tais como a redundância de informações e incompatibilidade entre as duas ferramentas, seguiu-se o paradigma utilizado pelo *RTutor* para a estruturação do conteúdo do curso. A eliminação de redundâncias é proveitosa pois a necessidade de geração de conteúdo e manutenção deste, ficaria multiplicada por quatro com a sua existência. O conteúdo do curso é estruturado em elementos genéricos, mistos e específicos:

**Elementos genéricos:** Representam partes do conteúdo que serão apresentadas a todos os estilos.

**Elementos mistos:** Apresentam conteúdo comum das regiões de fronteira entre os diversos estilos. Por exemplo, o "assimilador" e o "divergente" possuem em comum a característica de serem ambos reflexivos, enquanto que "assimilador" e "convergente" têm em comum a preferência por conceitualização abstrata.

**Elementos específicos:** São partes do conteúdo que somente serão apresentados para o estilo em questão, ou seja, um elemento que possui a

propriedade *assimilator* será exibido somente para alunos que são do estilo assimilador.

### 3.3.1 O arquivo XML

A estruturação do conteúdo é obtida basicamente pela definição de uma gramática livre de contexto que serve de base para a análise sintática do documento XML. Esta gramática fica armazenada em um arquivo de extensão .DTD - (Definição de Tipo de Documento) - no caso, utilizou-se a gramática da ferramenta *RTutor*, que fica em um arquivo nomeado *conteudo.dtd*, para que haja compatibilidade entre as duas ferramentas, pois este é um dos principais objetivos do *StyleEditor*.

Nota-se que esta DTD segue a proposição realizada na subseção 3.3., que sugere a especificação de alguns elementos para determinados estilos (elementos específicos) e de outros para grupos (elementos mistos) ou ainda elementos comuns a todos os estilos (genéricos). Esta classificação é realizada pela presença do atributo *proprietario*, que pode assumir os valores:

- diverger
- assimilator
- converger
- acomodador
- generico
- diverger\_assimilator
- assimilator\_converger
- converger\_acomodador
- acomodador\_diverger

Sendo os quatro primeiros relacionados aos estilos definidos pelo modelo, seguidos pelo qualificador genérico e seguido ainda pelos qualificadores mistos (grupos).

A seguir, na figura 3.1, vê-se um trecho do arquivo .DTD, que define a gramática de validação do conteúdo, em que se pode observar as estruturas principais que a compõem.

```

<!ELEMENT Capitulo (TituloCap, DescricaoCap, Secao+)>
<!ATTLIST Capitulo
    id ID #REQUIRED
    proprietario (generico | assimilator | diverger |
accommodator |
                converger | accommodator_diverger |
                diverger_assimilator | assimilator_converger |
                converger_accommodator) #REQUIRED>
<!ELEMENT TituloCap (#PCDATA|%textohtml;)*>
<!ELEMENT DescricaoCap (p)>

<!ELEMENT Secao (TituloSec, DescricaoSec, (CompSec | SubSec)+,
Teste*)>
<!ATTLIST Secao
    id ID #REQUIRED
    proprietario (generico | assimilator | diverger |
accommodator |
                converger | accommodator_diverger |
                diverger_assimilator | assimilator_converger |
                converger_accommodator) #REQUIRED>

<!ELEMENT TituloSec (#PCDATA|%textohtml;)*>
<!ELEMENT DescricaoSec (p)>

<!ELEMENT CompSec (%textohtml;|%estrutura;)*>
<!ATTLIST CompSec
    proprietario (generico | assimilator | diverger |
accommodator |
                converger | accommodator_diverger |
                diverger_assimilator | assimilator_converger |
                converger_accommodator) #REQUIRED>

```

Figura 3.1 - Trecho do DTD especificado no *RTutor*

O elemento *Capitulo* possui respectivamente um título do capítulo (*TituloCap*), uma descrição do capítulo (*DescricaoCap*) e uma ou mais seções (*Secao*). Além disso, possui como atributo um identificador (*id ID*) que deve ser o único em todo o arquivo *XML* e será utilizado para recuperar o conteúdo do elemento correspondente ao identificador *id*, durante a interação do usuário.

O elemento *Secao* refere-se às seções que compõem cada capítulo. Este elemento possui os mesmos atributos do elemento *Capitulo*. As diferenças surgem, pois os elementos *Secao* podem possuir uma ou mais subseções entremeadas por componentes de seção (*compSec*). Além disso, no final de cada seção pode aparecer um ou mais testes (*Teste*).

Os componentes *CompSec* por sua vez, são compostos por elementos definidos pelas duas entidades presentes na DTD (*textohtml* e *estrutura*). A entidade *textohtml* referenciada na DTD da figura 3.1 é definida da forma:

```
<!ENTITY % textohtml p|em|b|i|u|s|big|small|sub|sup|cite|code|pre|font|a|br ">
```

ou seja, ela representa os elementos definidos pela DTD de nome e significado semântico iguais às *tags HTML*. A outra entidade *estrutura* define as estruturas do *HTML*: tabelas, listas, imagens, linha horizontal. Assim, consegue-se reutilizar os conhecimentos sobre a linguagem *HTML* na autoria do conteúdo, ou seja, pode-se inserir *tags HTML* no conteúdo a fim de se melhorar a formatação do texto, realçar um trecho do conteúdo, etc.

O conteúdo do curso é finalmente escrito, em um arquivo *XML*, de acordo com a gramática definida pela DTD. A seguir, na figura 3.2, vê-se um trecho deste arquivo que contém o conteúdo do curso.

```
<Capitulo id="cap4" proprietario="generico">
  <TituloCap>Escalonamento de Tarefas de Tempo-real</TituloCap>
  H-H <DescricaoCap>
    <p>Este capítulo expõe vários algoritmos de escalonamento de
      tarefas de tempo-real, e discute suas vantagens e
      deficiências.</p>
  </DescricaoCap>
  H-H <Secao id="sec4.1" proprietario="assimilator">
    <TituloSec>Escalonadores</TituloSec>
    H-H <DescricaoSec>
      <p>O que são escalonadores de tarefas?</p>
    </DescricaoSec>
    H-H <CompSec proprietario="assimilator">
      <p>O escalonador é o responsável pela otimização no uso do
        processador pelo conjunto de tarefas a serem executadas.
        Dado um conjunto de tarefas, cabe a ele decidir qual a
        ordem em que estas tarefas serão atendidas pela CPU a fim
        de tornar possível a conclusão de cada uma dentro de seu
        "deadline".</p>
    </CompSec>
  </Secao>
```

Figura 3.2 - Trecho do documento *XML*

Pode-se notar neste trecho de código que existe um componente de seção genérico e outro específico (*assimilator*). Dessa forma, para que seja possível apresentar o conteúdo corretamente para um dos estilos de aprendizagem, é preciso

selecionar todos os componentes genéricos, específicos e os dois grupos nos quais o estilo está inserido (elementos mistos), ou seja, os elementos que possuem atributo proprietário com os valores genérico, estilo e grupo, em que o estilo assume o valor de um dos estilos definido pelo modelo de Kolb e grupo assume o valor dos grupos aos quais o estilo pertence.

Para efetuar esta seleção utiliza-se um arquivo de extensão *XSL*, do qual se vê um trecho de código na figura 3.3.

```

<xsl:template match="Capitulo" />
H-H <xsl:template match="TituloCap">
  H-H <h1 align="center">
    <xsl:apply-templates
      select="text()|em|b|i|u|s|big|small|sub|sup|cite|code|
        pre|font|a|br" />
    </h1>
  </xsl:template>
H-H <xsl:template match="DescricaoCap">
  <xsl:apply-templates />
</xsl:template>
H-H <xsl:template match="p">
  <xsl:copy-of select="." />
</xsl:template>
H-H <xsl:template match="Secao">
  <xsl:apply-templates />
</xsl:template>
H-H <xsl:template match="TituloSec">
  H-H <h2>
    <xsl:apply-templates
      select="text()|em|b|i|u|s|big|small|sub|sup|strike|cite|co
        de|pre|font|a|br" />
    </h2>
  </xsl:template>
H-H <xsl:template match="DescricaoSec">
  <xsl:apply-templates />
</xsl:template>

```

Figura 3.3 - Trecho do *XSL*.

A exemplo do que ocorre no DTD, também o arquivo *XSL* é especificado uma única vez, visto que a finalidade deste é somente promover a seleção dos elementos que já foram especificados pela DTD segundo o estilo e o identificador (*id*). Desta forma, os códigos *XSL* estão inseridos no componente de manipulação

*XML/XSL*, sendo esta a razão da existência de variáveis como a *proprietario*, no trecho apresentado na figura 3.1, pois apesar do *XSL* ser único para a seleção de seções, capítulos ou do conteúdo é necessário que os atributos *proprietario* e *id* sejam ajustados para a pesquisa necessária neste instante.

### 3.4 A estrutura do *StyleEditor*

Para esta versão inicial do *StyleEditor*, foram desenvolvidos três componentes (classes). Cada componente foi criado visando que cada um tivesse funções bem definidas e que permitissem que versões futuras implementem outras diretrizes de inserção ou edição de conteúdo, como outras questões, e que assim possam gerar conteúdo mais elaborado para os estilos.

A seguir é mostrada uma descrição da funcionalidade de cada um dos três componentes.

#### *StyleEditor*

É o componente principal do aplicativo, definindo a função principal do programa. Importa as duas classes abaixo e representa a interface gráfica principal com o usuário. Contem todos os objetos da interface gráfica, instanciando-os de acordo com a necessidade do programa.

#### *StyleReg*

Esta classe define um registro de informações sobre um determinado tópico. Esta é declarada com métodos *public* que permitem que outras classes tenham livre acesso ao conjunto de variáveis nela declarado. O objeto que é gravado no arquivo de dados temporário é uma instancia dessa classe. Também foi criada fora do componente principal visando posterior incremento de funcionalidade.

#### *StyleContentReg*

Esta classe é responsável pela entrada e saída do sistema e pela geração do *XML*. Faz a interface entre a classe *StyleReg* e os arquivos gerados pela

ferramenta. Instancia objetos de *StyleReg* e permite a manipulação de arquivos de conteúdo de tópicos (leitura e gravação). É nesta classe em que se manipulam os registros do arquivo de tópico (como dito antes, um arquivo de dados comum qualquer), neste caso um arquivo que recebe, armazena e recupera objetos do tipo *StyleReg* que posteriormente serão gravados no arquivo de conteúdo *XML*.

A organização das classes e seus métodos é ilustrada na figura 3.4 abaixo.

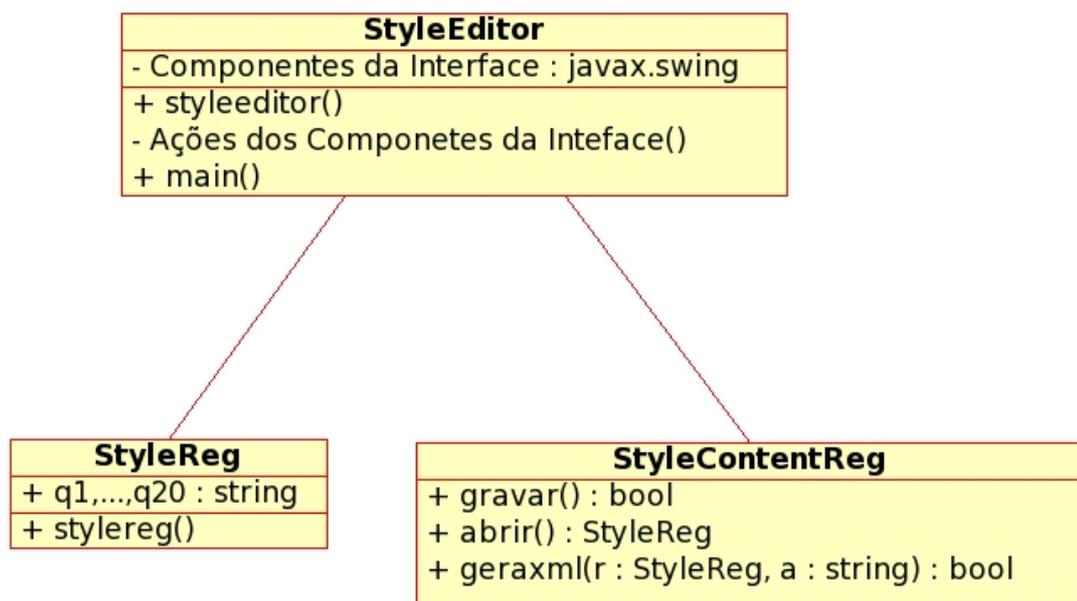


Figura 3.4 – Organização das classes do *Style Editor*

A seguir são mostradas duas ilustrações da janela principal de criação de tópicos da ferramenta desenvolvida.

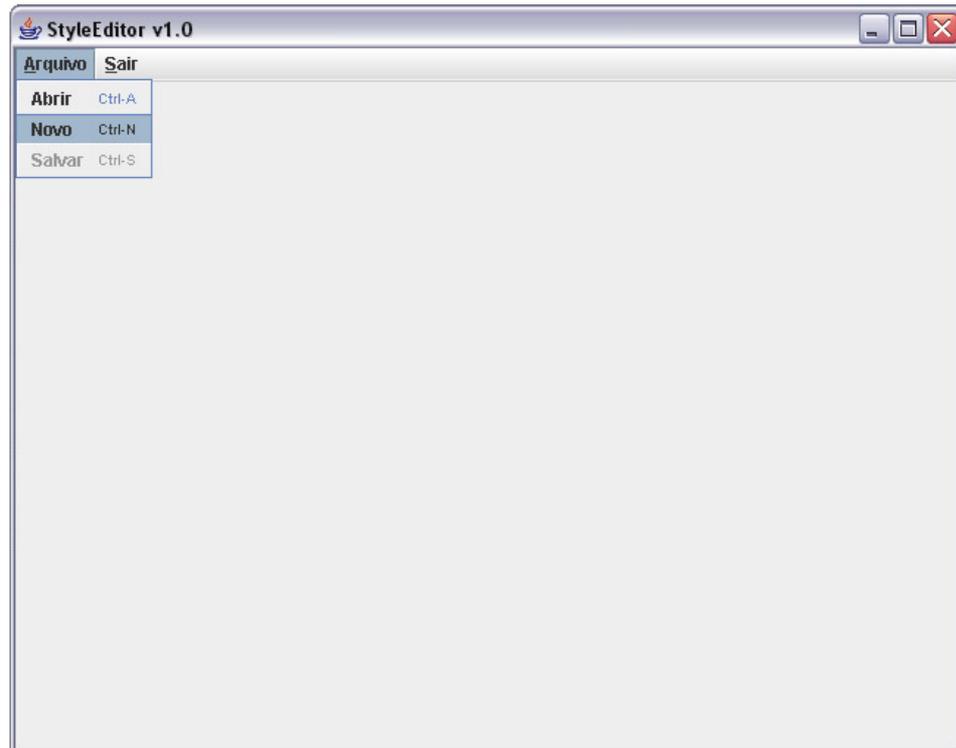


Figura 3.5 – Ilustração da interface do *StyleEditor*.



Figura 3.6 – Ilustração da continuação da interface do *StyleEditor*.

O *StyleEditor* atua na edição de conteúdos no lado servidor do *RTtutor*, como ilustrado na figura 3.7, porém fica claro que as ferramentas não são integradas, a integração destas ficará a título de um projeto futuro, esta figura é apenas ilustrativa.

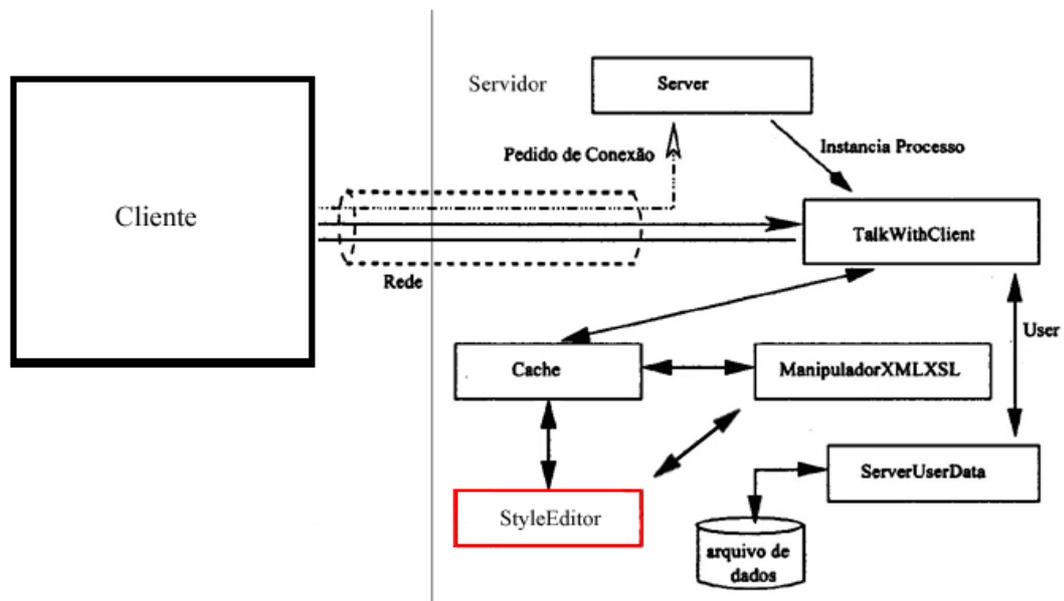


Figura 3.7: Atuação do *StyleEditor* no *RTtutor*

### 3.5 Considerações finais

Neste capítulo vimos todas as etapas do desenvolvimento da ferramenta *StyleEditor*. Inicialmente foi descrito os modelos de formatação dos estilos de *Kolb* exibindo-se as características de cada estilo de aprendizagem do modelo, foi descrita a construção do questionário para levantamento de dados seguindo as estas características.

A seguir foi descrito a forma de estruturação do conteúdo, especificando o arquivo *XML* onde os dados seriam inseridos. Por fim mostrou-se a estrutura da ferramenta *StyleEditor*, como ela foi construída e sua interface de trabalho.

No capítulo a seguir serão apresentados os testes realizados com a *StyleEditor* e os resultados obtidos com a implementação desta.

## Capítulo 4 - Resultados Obtidos

O *StyleEditor* foi desenvolvido baseando-se na ferramenta *RTtutor*, visando principalmente a autoria de conteúdo para a disciplina de Sistemas de Tempo-Real de uma forma prática e sem exigir que o usuário necessite conhecer detalhes da arquitetura interna daquela ferramenta.

Todo o conhecimento da complexidade de tal arquitetura fica a cargo do *StyleEditor*, que é responsável por transformar esse conteúdo simples e enxuto inserido, em um arquivo de conteúdo (*XML*) rico em detalhes contendo um leque de informações que garantem a compatibilidade desse conteúdo com o *RTtutor*.

Nesse capítulo serão expostos o resultados alcançados com a especificação e implementação do aplicativo *StyleEditor*. Os testes foram realizados utilizando-se a ferramenta para a autoria de um conteúdo e em seguida utilizou-se os módulos cliente e servidor do *RTtutor*. Os resultados estão divididos em quatro seções, que são as funcionalidades acrescentadas em relação ao protótipo anterior de [Leo04], a facilidade de uso e manutenção, desempenho e portabilidade.

## 4.1 Funcionalidades Acrescentadas

Como mencionado anteriormente, o *StyleEditor* foi desenvolvido com base em duas ferramentas, o *RTutor* e o *TutorEditor*. Utilizando-se as idéias do protótipo *TutorEditor*, a ferramenta foi implementada acrescentando funcionalidades, como a implementação de todos os estilos de aprendizagem do inventário de Kolb e a reconstrução do questionário de levantamento de conteúdo revisando todas as características de cada estilo para a elaboração de um questionário mais completo.

Por meio da comparação entre os conteúdos apresentados aos diversos estilos definidos por Kolb, percebeu-se que a ferramenta realmente cumpre o que se propõe, já que apresenta um conteúdo de acordo com as especificações propostas para cada estilo levantadas anteriormente.

Pode-se ver claramente as diferenças quando se compara os conteúdos gerados para cada um dos quatro estilos do modelo implementado. Por exemplo, pode-se verificar que o conteúdo do modelo Divergente é bem maior que o conteúdo gerado para o estilo Acomodador. Isso era esperado, pois este estilo gosta de diferentes perspectivas sobre o assunto e assim são apresentados exemplos e justificativas juntamente com os conceitos. Já o estilo Acomodador não gosta de muitos exemplos e justificativas, preferindo aprender com experiências práticas, assim grandes quantidades de conteúdo não são apresentados a este estilo.

## 4.2 Facilidade de uso e manutenção

Através do uso das bibliotecas gráficas fornecidas pela linguagem *Java* e de um ambiente de IDE, foi desenvolvida uma interface com o usuário amigável e intuitiva, onde o usuário pode navegar através do questionário de forma fácil (botões Próxima Página e Página Anterior), reutilizando a familiaridade do usuário com gerenciadores de janelas gráficas, como por exemplo o *KDE (Linux)* ou o *MS Windows Explorer (MS Windows)*.

Alcançou-se também a facilidade de manutenção através do emprego de técnicas propostas pela Engenharia de *Software*, tais como a reutilização de componentes, a boa definição das funções exercidas por cada uma das classes de

forma que a manutenção e futuras melhorias no código possam ser realizadas sem alterar a estrutura da ferramenta, bastando para isso manter a interface entre as classes. A tecnologia *XML* foi utilizada para a estruturação do conteúdo, visando a compatibilidade com ferramenta *RTutor*, construindo-se componentes para reuso evitando assim redundância de código e facilitando a manutenção do aplicativo.

### 4.3 Testes Realizados

A seguir é ilustrado o uso da ferramenta para a inserção de conteúdo e em seguida o trecho do documento *XML* do conteúdo inserido:

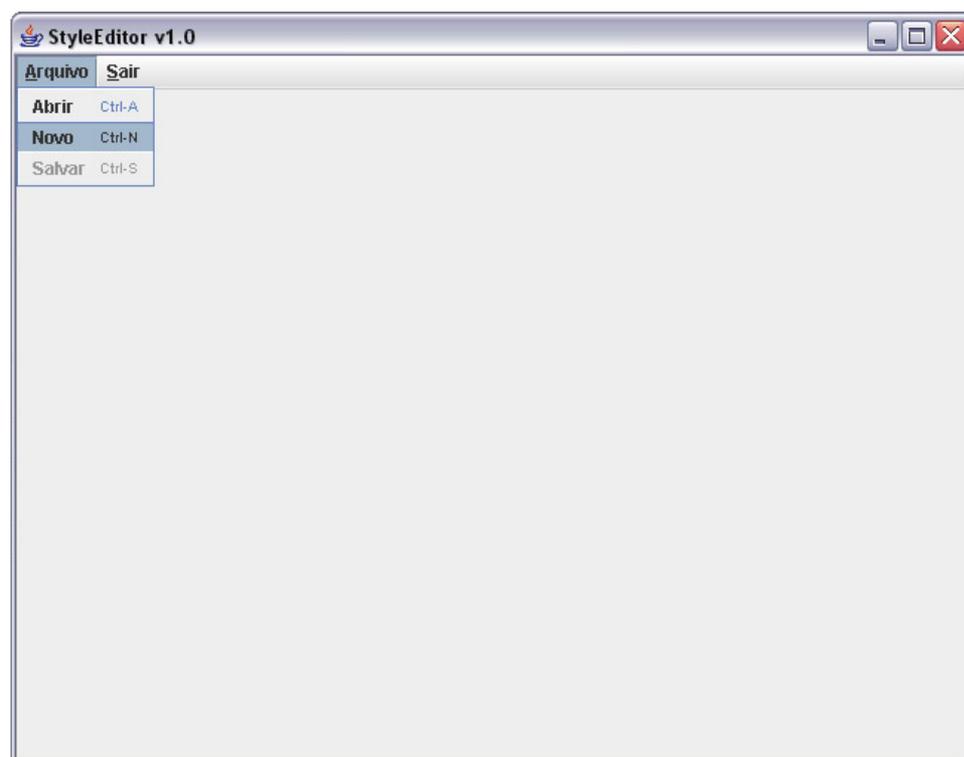


Figura 4.1: Tela inicial do StyleEditor

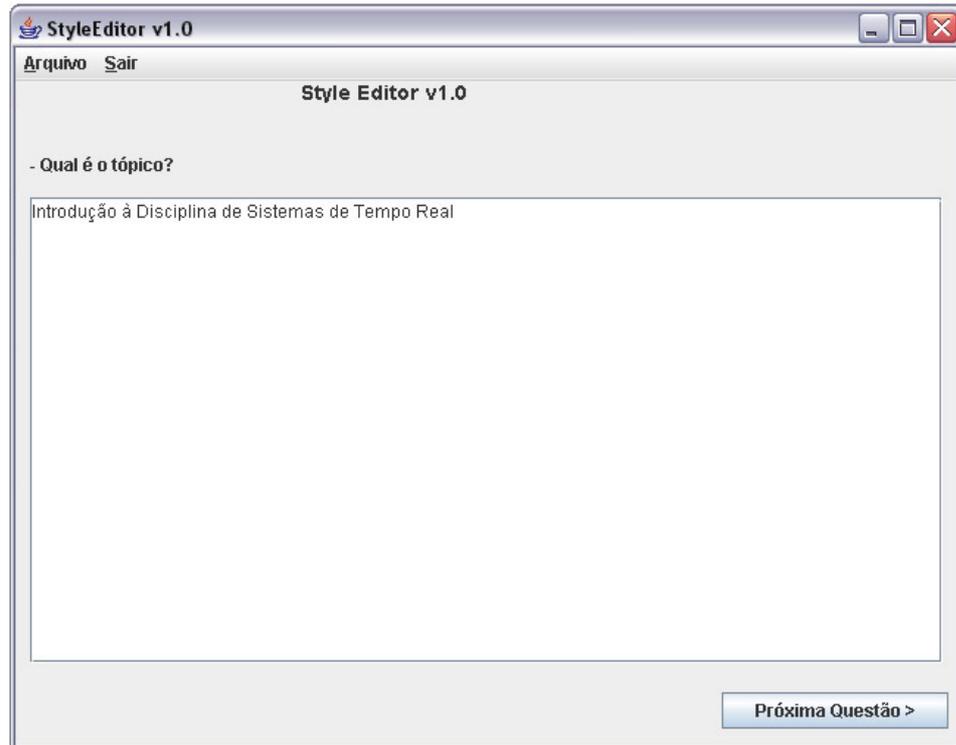


Figura 4.2: Inserindo o conteúdo – Primeira questão

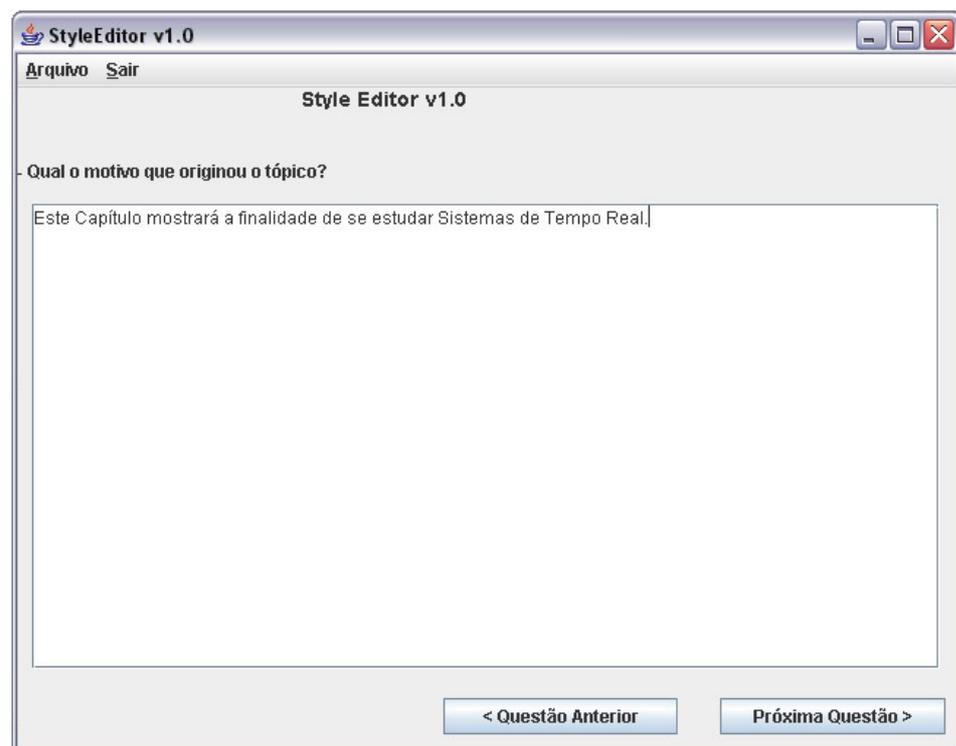


Figura 4.3: Inserindo o conteúdo – Segunda questão



Figura 4.4 : Inserindo o conteúdo – Terceira questão

```

<?xml version="1.0" encoding="ISO-8859-1" ?>
- <Capitulo id="cap1" proprietario="generico">
  <TituloCap>Introdução à Disciplina de Sistemas de Tempo Real</TituloCap>
  - <DescricaoCap>
    <p>Este Capítulo mostrará a finalidade de se estudar Sistemas de Tempo Real.</p>
  </DescricaoCap>
  - <Secao id="sec1.1" proprietario="generico">
    <TituloSec>Seção 1.1</TituloSec>
    - <DescricaoSec>
      <p>Apresentação:</p>
    </DescricaoSec>
    - <CompSec proprietario="acomodator_diverger">
      <p>Este capítulo fala sobre a história dos sistemas de tempo-real, durante a longa jornada da Computação.</p>
    </CompSec>

```

Figura 4.5: Trecho do documento XML

## 4.4 Portabilidade

A linguagem *Java*, na qual a ferramenta foi implementada, possui a tecnologia que garante a portabilidade de código e execução. Para a portabilidade das informações (o conteúdo de autoria do sistema de ensino) utilizou-se a tecnologia *XML*, que está sendo amplamente empregada para troca de informações, o que mostra que ela permite a portabilidade dos dados. Deste modo, sendo esta ferramenta totalmente implementada usando tecnologias que garantem portabilidade, alcançou-se também a

portabilidade da ferramenta, tal como o *RTutor*. Esta foi a razão principal para que se adotasse também essas tecnologias na ferramenta de geração de conteúdo.

Realizou-se diversos testes entre plataformas com o *StyleEditor* executando independentemente do sistema *RTutor*. O aplicativo foi testado nos ambientes *MS Windows XP*, *Linux* (distribuições *SlackWare 10.2* e *Red Hat*) e *Solaris*. Os testes envolveram a geração de código para uma plataforma e execução do aplicativo em outra (sendo possível seis combinações dessa maneira). Além disso houve testes de geração de dados em arquivos *XML* em uma arquitetura e processamento da informação em outra, (seis combinações também).

Em todos os testes o *StyleEditor* funcionou perfeitamente, garantindo portanto a portabilidade da ferramenta para estas plataformas.

## Capítulo 5 - Conclusões e perspectivas futuras

Neste capítulo apresenta-se as considerações finais a respeito do desenvolvimento da ferramenta *StyleEditor*. Assim, na seção 5.1 descreve-se as principais conclusões baseadas nos resultados apresentados no capítulo anterior e, a seguir, na seção 5.2 apresenta-se algumas propostas de trabalhos futuros visando aprimorar o funcionamento do *StyleEditor* e sua integração com a ferramenta *RTtutor*.

### 5.1 Conclusões

O foco principal durante o desenvolvimento desta ferramenta foi a melhoria das funcionalidades do sistema de ensino de Sistemas de Tempo-Real providas pelo *RTtutor*, baseando-se para isso no protótipo *TutorEditor*.

Como visto através dos resultados apresentados no capítulo anterior, foi disponibilizada uma interface amigável para produção de material instrutivo para os alunos considerando-se os quatro estilos de aprendizagem do modelo de Kolb, resolvendo assim a deficiência proveniente da concepção inicial do projeto da ferramenta de *RTtutor*.

Para que se alcançasse a implementação computacional de tal técnica foi necessário o emprego de uma tecnologia que tornasse viável a manutenção do conteúdo, eliminando a redundância que decorre de características comuns entre os estilos. Utilizou-se para tanto a tecnologia *XML*.

Visando a melhora de desempenho e facilidade de manutenção, estruturou-se o código do programa em componentes (classes) em *Java* reutilizáveis, além de permitir que trabalhos futuros (novas versões do *StyleEditor*) possam adicionar novas funcionalidades, ou realizar alterações no questionário de levantamento, com extrema facilidade.

Finalmente, a ferramenta aqui produzida se enquadra como solução à proposta de construção de uma ferramenta de autoria de conteúdo baseada no inventário de estilos de Kolb e que se adapte à ferramenta de auxílio de ensino de sistemas de tempo-real *RTutor*.

## 5.2 Perspectivas futuras

Este trabalho abre novas possibilidades para diversos projetos complementares visando a melhoria das ferramentas de ensino envolvidas no projeto atual. Três das principais propostas são descritas a seguir:

- **Integração das ferramentas de autoria e de ensino:** Propõe-se que se construa um aplicativo que combine o servidor do *RTutor* e o *StyleEditor*. Para isso deve-se, além de combinar as duas ferramentas, implementar novas formas de acesso dos autores do conteúdo e dos alunos, além de uma possível interação entre eles.
- **Generalização da ferramenta de autoria baseada em estilos:** A forma como a ferramenta *StyleEditor* foi construída, em que se cria e edita as informações em arquivos comuns para posterior conversão em *XML*, permite que uma nova versão seja criada para gerar conteúdo para outras disciplinas, desde que se generalizasse também a estrutura do *RTutor*.

- **Inserção de Recursos Multimídia:** Propõe-se a alteração dos sistemas *RTutor* e *StyleEditor* para que os mesmos possam dar suporte a recursos multimídia. Por exemplo, a inserção de uma imagem no conteúdo de um estilo e quando esse for visualizado, cria-se um *hyperlink* para a imagem em questão, na qual o aluno com um clique possa visualizar tal imagem.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [Arm02] ARMSTRONG, E. – *Understanding XML em The Java(tm) Web Services Tutorial*; Sun Microsystems, Palo Alto, Estados Unidos, 2002.
- [Cas02] CASAGRANDE, A. G., LOPES, O. C. - *Ferramenta para auxílio ao ensino de sistemas de tempo-real*; monografia de projeto final apresentada em 12/2002, S. J. do Rio Preto, SP.
- [Cur87] CURRY, L. (1987) - *Integrating concepts of cognitive or learning style: A review with attention o psychometric standarts*; Canadian College of Health Service Executives, Ottawa, Canada.
- [Cur90] CURRY, L. (1990) - *A critique of the research on learning styles*; Educational Leadership, October: 50-52.
- [Ctl05] *CTL Leaming Styles Site*;  
<http://www.indstate.edu/ctf/styles/inodel.html>,  
Ultimo acesso: 29/08/2005
- [Dec00] DÉCIO, O. C. - *Guia de Consulta Rápida XML*; Novatec, 2000.

- [Dul96] DULWORTH, M.R. and CARNEY, J. - *Improve training with interactive multimedia*. info-line 9601. Alexandria, VA: American Society for Training and Development, 1996.
- [Edu05] EduWeb - *Aulanet - E-learning Tecnologia*  
[http://www.eduweb.com.br/portugues/elearning\\_tecnologia.asp#](http://www.eduweb.com.br/portugues/elearning_tecnologia.asp#)  
Ultimo acesso: 29/08/2005
- [Gol97] GOLDBERG, Murray W. e SALARI, Susan, "*An Update on WebCT (World-Wide-Web Course Tools) - a Tool for the Creation of Sophisticated Web-Based Learning Environments*", Proceedings of NAUWeb '97 – Current Practices in Web-Based Course Development, June 12 - 15, 1997, Flagstaff, Arizona.
- [Kol76] KOLB, D. A. – *Learning Style Inventory*. Boston: McBer. (1976)
- [Leo04] GONÇALVES, L. J. – *Ferramenta para geração de informações em cursos de aprendizagem baseada em computador*; monografia de projeto final apresentada em 12/2004, S. J. Rio Preto, SP
- [Lia05] CAVELLUCCI, L.C.B. – *Estilos de aprendizagem: em busca das diferenças individuais*  
[http://www.iar.unicamp.br/disciplinas/am540\\_2003/lia/estilos\\_de\\_aprendizagem.pdf](http://www.iar.unicamp.br/disciplinas/am540_2003/lia/estilos_de_aprendizagem.pdf)  
Ultimo acesso: 25/08/2005
- [Lot98] LOTUS - *Learning Space 4.0: A new vision of e-learning*.  
<http://www.lotus.com/home.nsf/tabs/learnspace>  
Último acesso: 06/11/2000
- [Mac05] MACKENZIE. Programa de Pós Graduação em Engenharia Elétrica – *Educação via Internet: Avaliação Comparativa de Metodologias de Construção de Ferramentas e Aplicações*,  
[http://www.mackenzie.com.br/pos\\_graduacao/engeletrica/pesq\\_proj\\_educacao.htm](http://www.mackenzie.com.br/pos_graduacao/engeletrica/pesq_proj_educacao.htm)  
Ultimo acesso: 25/08/2005.
- [Mcc05] A LIVING LABORATORY – *Learning Styles and the 4MAT System: A Cycle of Learning*  
<http://volcano.und.nodak.edu/vwdocs/msh/llc/is/4mat.html>  
Ultimo acesso: 30/08/2005
- [Mye62] MYERS, I. - *Myers-Briggs Type Indicator*, Palo Alto, CA: Consulting Psychotogists Press, 1962.

- [Ray00] RAY, E. T. – *Aprendendo XML*; Campus, 2001.
- [Rom81] ROMISZOWISKI, A. J. - *Designing Instructional Systems*. Nichols Publishing Company. 1981.
- [Tel05] NÚCLEO DE INFORMATICA APLICADA À EDUCAÇÃO (NIED)  
- *Teleduc - Ambiente de ensino à distância*  
<http://hera.nied.unicamp.br/teleduc/>  
Último acesso: 29/08/2005