
Fernando Maia Assunção

SHART-Web:
Um Sistema Tutor em Harmonia Tradicional na
Web

Dissertação apresentada ao Curso de Mestrado em Informática da Universidade Federal da Paraíba, como exigência parcial para a obtenção do Grau de Mestre.

Área de Concentração: Ciência da Computação

Orientador: Prof. Evandro de Barros Costa

Campina Grande – PB

Outubro de 2001

À minha esposa:
Cláudia,
e nossos filhos:
Larissa e Victor

Agradecimentos

Ao meu orientador Evandro.

A Edilson Ferneda.

Aos amigos Edilson Cabral e Nené Liberalquino.

Ao pessoal da UFAL: Hygoo, Emerson e Ricardo.

À Cláudia, pelo grande incentivo.

E a Deus.

Capítulo 1 - Introdução

1.1 Apresentação Geral -----	1
1.2 Motivações -----	2
1.3 Objetivos e Contribuição do Trabalho -----	3
1.4 Estrutura da Dissertação -----	3

Capítulo 2 - Abordagens em Computação e Música

2.1 Introdução -----	5
2.2 Visão Geral da Computação e Música -----	5
2.2.1 Música e Inteligência Artificial -----	6
2.2.1.1 Áreas de Pesquisa em IA e Música -----	6
2.3 O Conhecimento Musical e sua Representação -----	7
2.3.1 Estrutura, Hierarquia, Tempo e outros Atributos -----	8
2.3.2 Regras como Representações -----	9
2.4 Informática e Educação Musical -----	10
2.4.1 Sistemas Tutores Inteligentes em Música -----	10
2.4.2 <i>Softwares</i> Educacionais de Música -----	11
2.5 Trabalhos Correlatos -----	12
2.6 Conclusão -----	15

Capítulo 3 - Sistemas Computacionais de Apoio

3.1 Introdução -----	16
3.2 Sistemas Baseados em Conhecimento -----	16
3.2.1 Definição e Características -----	16
3.2.2 Arquitetura de SBC -----	16
3.2.3 Áreas e Aplicações -----	17
3.2.4 Representação do Conhecimento -----	17
3.2.4.1 Tipos de Representação -----	18
3.2.4.2 Métodos de Inferência e Controle -----	19
3.3 Sistemas Multiagentes -----	19
3.3.1 O Contexto -----	19
3.3.2 Definições de Agente -----	20

3.3.3	Classificação dos Agentes -----	21
3.3.4	Definição e Características dos Sistemas Multiagentes -----	22
3.3.5	Arquiteturas de Sistemas Multiagentes -----	22
3.3.6	Comunicação entre Agentes -----	24
3.3.6.1	Coordenação e Tipos de Mensagens -----	24
3.3.6.2	Cooperação e Distribuição de Tarefas -----	24
3.3.6.3	Protocolos de Interação -----	25
3.3.6.4	Linguagens -----	26
3.4	Sistemas Tutores Inteligentes e Hipermídia -----	27
3.4.1	Sistemas Tutores Inteligentes -----	27
3.4.1.1	Breve histórico -----	27
3.4.1.2	Definições e Características -----	28
3.4.1.3	Componentes -----	29
3.4.1.4	Arquiteturas -----	33
3.4.1.5	Arcabouços e Sistemas de Autoria -----	34
3.4.2	Evolução para os Ambientes Interativos de Aprendizagem -----	35
3.4.2.1	Definições e Características -----	35
3.4.2.2	Classificação -----	36
3.4.2.3	Multimídia, Hipertexto e Hipermídia -----	37
3.5	Sistemas de Educação a Distância via Internet -----	38
3.5.1	Definição e Características -----	38
3.5.2	Aspectos de Ensino e Aprendizagem -----	38
3.6	Conclusão -----	39

Capítulo 4 - O Sistema SHART-Web e o MATHEMA

4.1	Introdução -----	40
4.2	O Ambiente MATHEMA -----	40
4.2.1	Visão Externa do Domínio -----	40
4.2.2	Visão Interna do Domínio -----	41
4.2.3	Arquitetura Geral do MATHEMA -----	41
4.2.4	Arquitetura do Agente MATHEMA -----	42
4.2.5	Interações entre Agentes -----	44
4.2.6	Interações Aprendiz-Agente -----	45
4.3	O Sistema SHART-Web e o MATHEMA -----	46
4.3.1	Aspectos do Domínio -----	46
4.3.2	Visão Externa do Domínio -----	47
4.3.3	Visão Interna do Domínio -----	48
4.3.4	Identificação dos Agentes -----	51

4.4 Conclusão	-----53
----------------------	---------

Capítulo 5 - O Sistema SHART-Web: Requisitos, Arquitetura e Funcionamento

5.1 Introdução	-----54
5.2 Requisitos	-----54
5.2.1 Características Gerais	----- 54
5.2.2 Sobre os Usuários do Sistema	----- 54
5.3 Arquitetura e Componentes	-----55
5.3.1 Sistema Tutor Inteligente Multiagentes (STIMA)	-----56
5.3.2 Banco de Dados de Perfil e Administração	-----59
5.4 Funcionamento Geral	-----59
5.4.1 Funcionamento de um Agente	-----60
5.4.1.1 Execução de uma Estratégia Pedagógica	-----64
5.4.1.2 Diagnóstico Cognitivo	----- 64
5.4.2 Funcionamento da Arquitetura Multiagentes	-----65
5.4.2.1 Validação de Resposta	----- 65
5.4.3 Cenários de Interação	----- 66
5.5 Conclusão	----- 70

Capítulo 6 - Modelagem do Sistema SHART-Web

6.1 Introdução	----- 71
6.2 Análise	----- 71
6.2.1 Diagrama de Pacotes	----- 71
6.2.2 Modelo Conceitual	----- 72
6.2.3 Casos de Uso Essenciais	----- 73
6.2.4 Diagramas de Interação	----- 74
6.2.4.1 Diagramas de Seqüência	----- 74
6.2.4.2 Diagrama de Colaboração	-----79
6.2.5 Diagramas de Estados	----- 80
6.3 Projeto	----- 81
6.3.1 Funções Básicas do Sistema	----- 81
6.3.2 Casos de Uso Reais	-----81
6.3.3 Diagramas de Classes	-----82
6.3.4 Diagrama de Implantação	----- 84
6.3.5 Diagrama de Entidade-Relacionamento	-----84

6.4 Conclusão	86
----------------------	----

Capítulo 7 - Aspectos de Implementação e Experimentação

7.1 Introdução	87
7.2 Aspectos de Implementação	87
7.2.1 Sobre a Linguagem e Tecnologias Java	87
7.2.2 Ferramentas Utilizadas	88
7.2.3 As Bases de Conhecimento	88
7.2.4 Interface do Sistema	89
7.3 Experimentação	92
7.3.1 Objetivos	92
7.3.2 Metodologia Adotada	92
7.3.3 Sujeitos da Pesquisa	93
7.3.4 Análise dos Resultados	93
7.4 Conclusão	94

Capítulo 8 - Conclusões e Trabalhos Futuros

8.1 Considerações Finais	95
8.2 Trabalhos Futuros	95
Referências Bibliográficas	97

Lista de Figuras

Figura 2.1 - Interdisciplinaridade de Computação e Música -----	5
Figura 2.2 - Representações discreta e contínua do tempo musical -----	9
Figura 2.3 - Tela inicial do <i>software</i> Auralia -----	11
Figura 2.4 - <i>Software</i> Ear Master Pro -----	12
Figura 3.1 - Arquitetura de SBC -----	17
Figura 3.2 - SMA, IAD e IA -----	20
Figura 3.3 - Domínio dos STIs (ICAI) -----	29
Figura 3.4 - Interação entre os componentes de um ITS -----	34
Figura 4.1 - Visão externa do domínio -----	41
Figura 4.2 - Arquitetura geral do ambiente MATHEMA -----	41
Figura 4.3 - Arquitetura do agente MATHEMA -----	44
Figura 4.4 - A comunicação aprendiz-agente -----	46
Figura 4.5 - Visão externa do domínio da harmonia tradicional -----	48
Figura 4.6 - Uma estrutura pedagógica -----	50
Figura 5.1 - Arquitetura do sistema SHART-Web -----	55
Figura 5.2 - Documento hipermídia -----	59
Figura 5.3 - Descrição detalhada do Sistema Tutor de um agente da SATA--	61
Figura 5.4 - Interpretação da estratégia pedagógica -----	64
Figura 5.5 - Diagnóstico cognitivo -----	65
Figura 5.6 - Passo a passo na validação de resposta -----	65
Figura 5.7 - Exercício do tipo <i>múltipla escolha</i> -----	66
Figura 5.8 - Exercício do tipo <i>validação de resposta</i> -----	67
Figura 5.9 - Exercício sobre o assunto <i>distribuição das vozes</i> -----	67
Figura 5.10 - Dica -----	68
Figura 5.11 - Exercício sobre o assunto <i>tipos de movimento das vozes</i> -----	68
Figura 5.12 - Exercício sobre o assunto <i>distância entre as vozes</i> -----	69
Figura 5.13 - Exercício sobre o assunto <i>extensão das vozes</i> -----	69
Figura 5.14 - Resposta do aluno -----	70
Figura 6.1 - Diagrama de Pacotes -----	72
Figura 6.2 - Diagrama conceitual do sistema -----	72
Figura 6.3 - Diagrama de <i>Casos de Uso</i> Essenciais -----	73
Figura 6.4 - Diagrama Inicia Sessão de Estudos -----	74

Figura 6.5 - Diagrama de Seqüência (cadastro)-----	75
Figura 6.6 - Diagrama de Seqüência (login) -----	75
Figura 6.7 - Diagrama de Seqüência (Inicia Sessão de Estudos) -----	76
Figura 6.8 - Diagrama de Seqüência (Exibe Definição) -----	76
Figura 6.9 - Diagrama de Seqüência (Exibe Exemplo) -----	77
Figura 6.10 - Diagrama de Seqüência (Inicia Chat) -----	77
Figura 6.11 - Diagrama de Seqüência (Insere Recursos) -----	78
Figura 6.12 - Diagrama de Seqüência (Insere Estratégias) -----	78
Figura 6.13 - Diagrama de Colaboração (Exibe Exercício) -----	79
Figura 6.14 - Diagrama de Colaboração entre dois agentes -----	80
Figura 6.15 - Diagrama de Estados para a classe Motor de Inferência -----	80
Figura 6.16 - Diagrama de Classes 1 (Aplicação) -----	83
Figura 6.17 - Diagrama de Classes 2 (Aplicação) -----	83
Figura 6.18 - Diagrama de Implantação -----	84
Figura 6.19 - Diagrama de Entidade-Relacionamento (domínio) -----	85
Figura 6.20 - Diagrama de Entidade-Relacionamento (usuário) -----	85
Figura 7.1 - Tela inicial do sistema -----	89
Figura 7.2 - Definição e Exemplo sobre Vozes (navegação estática) -----	90
Figura 7.3 - Exposição sobre o assunto Vozes (navegação guiada) -----	90
Figura 7.4 - Exercício para o sistema validar a resposta -----	91
Figura 7.5 - Exercício com dica disponível -----	91

Lista de Tabelas

Tabela 3.1 - Conectivos e significados -----	18
Tabela 4.1 – Pares e lateralidades -----	48
Tabela 4.2 - Curricula -----	49
Tabela 4.3 - Unidades pedagógicas de Vozes -----	49
Tabela 4.4 – Extensão das Vozes -----	49
Tabela 4.5 - Distância entre as Vozes -----	50
Tabela 4.6 - Distribuição das Vozes -----	50
Tabela 4.7 - Tipos de Movimento das Vozes -----	50
Tabela 5.1 – Estratégias pedagógicas -----	57
Tabela 5.2 – Regras para as estratégias pedagógicas -----	58

Glossário musical

Acorde: Emissão simultânea de três ou mais notas. Na Harmonia tradicional, os acordes são formados por terças superpostas.

Alteração: Símbolos utilizados para elevar ou abaixar a nota $\frac{1}{2}$ tom, 1 tom, ou fazer voltar à posição natural (sem alteração). Os símbolos são: sustenido, bemol, dobrado-sustenido, dobrado-bemol e bequadro.

Armadura: Símbolos (sustenidos ou bemóis) colocados no início do pentagrama e que indicam a tonalidade de uma peça musical.

Cadência: Encadeamento de acordes com sentido mais ou menos conclusivo.

Cifra: É um símbolo que identifica relações entre notas dentro de um acorde.

Clave: Símbolo acoplado ao pentagrama a partir do qual são identificados os nomes das notas.

Compasso: Demarca o pulso rítmico da música. É colocado no pentagrama depois da armadura de clave. Pode ser binário, ternário ou quaternário.

Enarmonia: Ocorre quando duas notas têm a mesma altura, porém com funções harmônicas diferentes. Exemplo: dó sustenido e ré bemol.

Encadeamento: Sucessão de acordes dispostos com uma certa ordem dos graus.

Escala: Sucessão de notas com uma relação intervalar pré-estabelecida.

Frase: Ciclo completo de uma idéia melódica formando um sentido conclusivo.

Grau: Nome usado para classificar as notas de uma escala ou acorde em relação à tônica de um determinado tom. Os graus são referenciados não somente por algarismos romanos, mas também pelos nomes: I^o grau (tônica), II^o grau (supertônica), III^o grau (mediante), IV^o grau (subdominante), V^o grau (dominante), VI^o grau (submediante), VII^o grau (sensível).

Intervalo: É a distância entre duas notas. Pode ser simples (se as notas estão dentro de uma oitava, ou composto (se as notas estão em oitavas diferentes). Pode ser melódico (sentido horizontal) ou harmônico (sentido vertical), ascendente (se a 2^a nota é mais aguda que a 1^a), ou descendente (se a 2^a nota é mais grave que a 1^a).

Modo: Forma como é feita a distribuição dos intervalos numa escala. Na harmonia tonal usa-se dois modos: maior e menor.

Nota: Quatro atributos definem nota. Nome, oitava, alteração e duração. Os tres primeiros definem a frequência e o último diz respeito ao tempo.

Nota fundamental: 1^a nota quando se forma um acorde. Em seguida vem a terça e a quinta.

Oitava: Quando uma nota tem dobrada a sua frequência têm-se uma oitava. Um intervalo de oitava tem 12 semitons.

Pentagrama: Conjunto de 5 linhas onde são colocadas as notas e pausas.

Semitom: Menor intervalo usado na Harmonia tonal.

Tom: Intervalo de dois semitons ou $\frac{1}{2}$ tom.

Tonalidade: É uma relação hierárquica entre os diversos graus, onde o Iº grau é o pólo principal de atração.

Triade: Acorde de três notas formado por terças superpostas. Pode ser maior, menor, aumentada e diminuta.

Tétrade: Acorde de quatro notas formado por terças superpostas.

Voz: Está relacionada ao registro (agudo, médio ou grave) da nota. São quatro o número de vozes: soprano, contralto, tenor e baixo. Pode ser tocada por um instrumento ou cantada.

Este trabalho é parte de um projeto de pesquisa que propõe um amplo modelo de ambiente para Educação a distância via Internet, adotando uma abordagem multiagentes artificiais e humanos, envolvidos principalmente em interações cooperativas baseadas em resolução de problemas e seus desdobramentos em outras tarefas pedagógicas. Trata-se do ambiente Math_Net, que tem o modelo MATHEMA como uma de suas bases. Neste contexto, o presente trabalho trata de um Sistema Tutor Inteligente cooperativo com uma abordagem multiagentes integrado a um componente hipermídia na Web e voltado para o ensino de Harmonia musical tradicional. Denominamos este sistema de SHART-Web (Sistema tutor em HARmonia Tradicional na Web). Além de interações adaptativas durante o processo de resolução de problemas, e como consequência desta, com o SHART-Web visa-se prover o Aprendiz humano com uma navegação personalizada sobre um documento hipermídia, contendo o material de ensino. O SHART-Web, como um protótipo, foi desenvolvido e experimentado com dois grupos distintos e distribuídos na Web.

This work is a part of big research project proposing an environment model for distance learning. This environment adopts a multiagent approach involving human and artificial agents. These agents work on problem solving situations associated with other pedagogical tasks through cooperative interactions. This environment is mainly based on MATHEMA environment and it is named Math_Net. In this context, the present work propose a Web-based cooperative multiagent Intelligent Tutoring System (ITS) in Musical Harmony domain, named SHART-Web. This ITS integrates a hypermedia component. The system provides learners with adaptive interactions during problem solving activities and some cases leading these learners to the hypermedia teaching material by providing them with individualised navigation. The SHART-Web prototype was developed and experimented with distinct groups distributed along the Web structure.

Introdução

1.1 Apresentação Geral

O domínio da Harmonia tradicional tem sido uma base para o ensino da composição musical. Ultimamente, o ensino da música como um todo e, em particular, de Harmonia tradicional tem se beneficiado com a construção e disponibilização de ferramentas computacionais voltadas para apoiar o processo de ensino-aprendizagem neste domínio de conhecimento.

A dinâmica tecnológica atual, notadamente quando se considera a Internet, tem trazido para a educação novos desafios. Em plena era da informação, o fluxo de informação não apenas aproxima áreas as mais diversas como também acelera o seu meio de transmissão. Neste cenário, o ensino a distância propicia mudanças significativas no âmbito institucional e na relação aluno-professor.

Esta relação pode assumir um novo perfil, inserindo-se aí os *softwares* educacionais como pivô, na busca de soluções que eventualmente podem ser mais cooperativas, onde o professor, o aluno e o computador possam em papéis diferentes, ter objetivos comuns, visando a aprendizagem do aluno.

Por sua vez, e buscando entender os processos de aprendizagem, a pesquisa em Inteligência Artificial na Educação (IA-ED), tem trazido há algum tempo contribuições da Psicologia Cognitiva, Educação e da Ciência da Computação para o desenvolvimento de Sistemas Tutores Inteligentes (STI) [Kearsley 87].

A área de Computação e Música, de sua parte, vem se firmando cada vez mais como área de pesquisa que se pode constatar nos anais dos SBC & M (Simpósio Brasileiro de Computação e Música), que é um evento que ocorre anualmente junto ao Congresso da Sociedade Brasileira de Computação. E, a Harmonia musical, mais especificamente a Harmonia tradicional, tem sido igualmente objeto de pesquisa sob diferentes abordagens [Ebcioğlu 92] e [Holland 94].

Nesse contexto situamos nosso trabalho, voltado para o ensino em Harmonia tradicional sob a óptica dos Sistemas Tutores Inteligentes. Nesse sentido concebemos um STI para ensino/aprendizagem em Harmonia tradicional presencial ou a distância, com base numa arquitetura multiagentes, tirando proveito da infraestrutura da Internet/Web e de recursos hipermídia. Este trabalho tem sido desenvolvido no âmbito da pesquisa realizada no projeto MathNet [Costa 2000], financiado pelo programa Protem-CC/CNPq, tendo o orientador desta dissertação como o seu Coordenador geral. A proposta central do MathNet é o de ser uma plataforma para Educação a distância via Internet baseada numa abordagem com múltiplos agentes artificiais e humanos.

1.2 Motivações

A importância da Harmonia tradicional na formação do músico e a necessidade de ferramentas computacionais que apliquem a Inteligência Artificial visando captar as ações do aprendiz com vistas à melhoria de seu processo de aprendizagem foram as motivações iniciais neste trabalho.

O surgimento de tecnologias interativas, especialmente as que são via Internet, tem motivado as instituições de ensino a investirem em ambientes computacionais com objetivo de melhorar o processo de ensino-aprendizagem. Considerando estas tendências, têm-se investido no âmbito do Mestrado de Informática da UFPB, e com base no modelo de ambiente MATHEMA [Costa 97], em trabalhos nesta linha de pesquisa. Estes trabalhos são: “Da representação do conhecimento musical ao esboço conceitual de uma sociedade de agentes em Harmonia” [Teixeira 97] e “MHITS – Um Sistema Tutor Inteligente em Harmonia Musical” [Caminha 2000], que tratam também do desenvolvimento de Sistemas Tutores Inteligentes no domínio da Harmonia Musical. Dando seqüência a esses investimentos, propomos nesta dissertação o sistema Shart-Web, descrito mais adiante e por vários trechos do presente documento.

Uma terceira motivação foi a busca de novas metodologias e abordagens de ensino, numa relação não mais bipolarizada de professor-aluno, mas sim criando outro tipo de relação entre aluno, sistema e professor, desempenhando neste caso um papel de mediador.

1.3 Objetivos e Contribuições do Trabalho

O objetivo geral de nosso trabalho é a modelagem e a implementação de um sistema de ensino/aprendizagem de Harmonia tradicional a distância, com base numa arquitetura multiagentes, com características interativas e hipermídia, e funcionando na Internet/Web.

Especificamente este trabalho visa:

1. Realizar a modelagem do domínio segundo o modelo dimensional do ambiente MATHEMA [Costa 97].
2. Integrar ao Sistema Tutor definido no Agente, um componente hipermídia que contém o material didático que apoia as atividades pedagógicas. Aqui cabe a importante observação de que o Sistema provê o Aprendiz com uma navegação personalizada sobre o documento hipermídia.
3. Implementar Interfaces que permitam a comunicação dos Aprendizes e Professores com o Sistema Tutor.
4. Implementar os componentes principais do Sistema Tutor, a saber: módulo do Especialista, módulo do Tutor e módulo de modelagem do Aprendiz.
5. Tornar disponível na Internet/Web o Sistema com as opções de navegação guiada ou livre.

Estes objetivos serão melhor detalhados a partir do capítulo 4, ocasião em que apresentamos o Sistema e sua arquitetura.

Com a realização destes objetivos esperamos dar as seguintes contribuições:

- Tornar disponível na Web uma ferramenta com material didático no domínio da Harmonia tradicional que pode ser usada por alunos e professores em atividades de ensino a distância.
- No âmbito do projeto MathNet [Costa 2000] e seguindo o modelo MATHEMA, que é uma parte ponderável do MathNet, concretizar uma implementação de um sistema tutor multiagentes, no domínio de conhecimento em pauta.

1.4 Estrutura da Dissertação

Além desta introdução, o presente trabalho está organizado da seguinte forma:

No Capítulo 2, mostramos um panorama geral da Computação e Música, ressaltando os Sistemas Tutores Inteligentes no domínio da Música em geral, e a Informática na Educação musical.

No Capítulo 3, abordamos os sistemas computacionais que deram suporte teórico ao nosso trabalho. Primeiramente os Sistemas Baseados em Conhecimento (SBC) e os Sistemas Multiagentes. Posteriormente os Sistemas Tutores Inteligentes e sua evolução em direção aos Ambientes Interativos de Aprendizagem, e por fim o Ensino a distância via Internet.

No Capítulo 4, delineamos a ligação entre o ambiente MATHEMA e o nosso sistema. A partir do modelo do agente MATHEMA, mostramos como serão os agentes no domínio da Harmonia tradicional.

No Capítulo 5, descrevemos nosso sistema mais detalhadamente, incluindo sua arquitetura e funcionamento.

No Capítulo 6, tratamos da modelagem de nosso sistema da fase de Análise à fase de Projeto.

No Capítulo 7, apresentamos os aspectos relativos à sua implementação, os experimentos do SHART-Web com uma turma de alunos e a análise dos resultados.

No Capítulo 8, apresentamos as conclusões e o que pode ser feito futuramente a partir de nosso trabalho.

Capítulo 2

Abordagens em Computação e Música

2.1 Introdução

Neste capítulo, apresentamos inicialmente uma visão geral da Computação e Música, incluindo as áreas de pesquisa em Inteligência Artificial e Música. Em seguida abordaremos alguns aspectos de Representação musical, a Informática na Educação musical, que é o foco principal de nosso trabalho, e por último, os trabalhos correlatos ao nosso.

2.2 Visão Geral da Computação e Música

Moore coloca a Computação e Música como uma área interdisciplinar envolvendo as seguintes disciplinas: Música, Ciência da Computação, Psicologia, Engenharia e Física (Figura 2.1) [Moore 90].

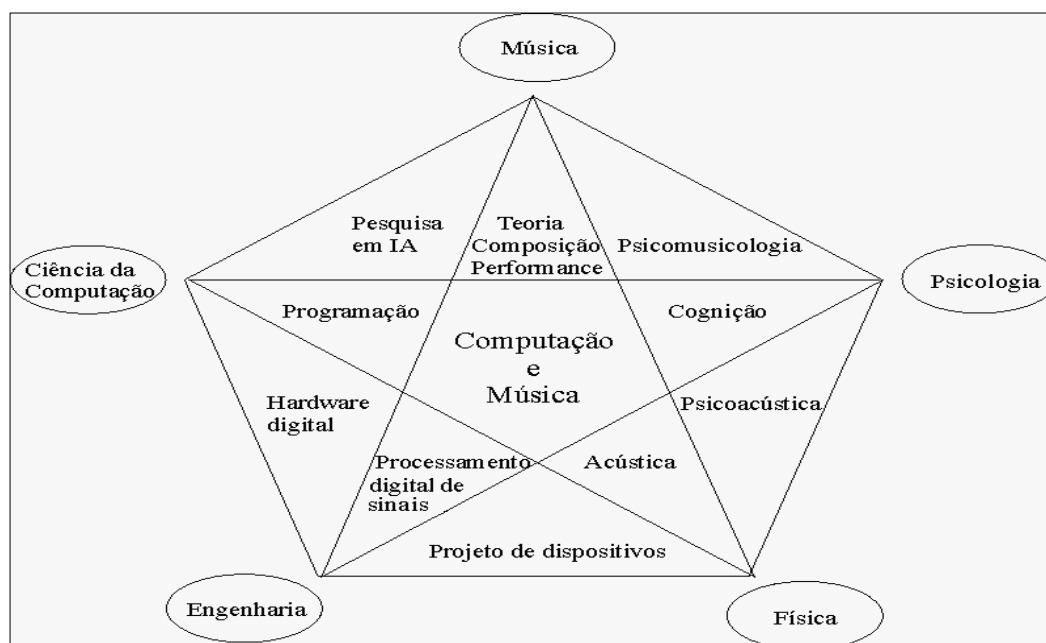


Figura 2.1 Interdisciplinaridade de Computação e Música [Moore 90].

Desde as primeiras experiências de controle e geração de sons por computador nos anos 50, até o presente, a Computação e Música tem uma

longa trajetória de evolução que engloba áreas como: Reconhecimento automático de partituras, Análise musical auxiliada por computador, Composição assistida por computador, Processamento de sinais e síntese sonora, Educação musical computadorizada, entre outras.

2.2.1 Música e Inteligência Artificial

Inteligência Artificial (IA) é definida como o estudo científico e a simulação de comportamentos inteligentes, e a Música, como uma atividade intelectual, inclui entre seus atributos, a habilidade de reconhecer padrões e modificá-los por ações [Wiggins et alii 93] e [Miranda 98].

As pesquisas em IA e Música reúnem aspectos de duas categorias:

- *Ciência Cognitiva*, que engloba o desenvolvimento de teorias ligadas à inteligência humana;
- *IA aplicada*, voltada para o desenvolvimento de programas que exibam comportamento inteligente [Roads 85].

2.2.1.1 Áreas de Pesquisa em IA e Música

As pesquisas em IA e Música englobam áreas como:

- 1) *Assistentes inteligentes de composição*;
- 2) *Análise e modelagem geradora de música*;
- 3) *Reconhecimento e entendimento de som*;
- 4) *Harmonização de melodias*;
- 5) *Reconhecimento de timbres* [Roads 85] e [Miranda 90].

O principal objetivo de um *Assistente inteligente de composição* é dar suporte ao compositor nas diversas fases do processo composicional, incluindo criação de um plano e arquitetura da composição [Roads 96].

No campo de *Análise musical* duas abordagens se destacam: *análise por síntese* e *análise por descoberta*. No primeiro caso um programa gera música de alguma estrutura inicial usando princípios teóricos. No segundo caso busca-

se descobrir características de uma peça musical ou estilo de composição por meio de analogia [Balaban 92].

No campo de *Reconhecimento e entendimento de som*, as tarefas podem estar classificadas em tempo real (ao vivo) ou não. A vantagem de sistemas que não são em tempo real é que pode-se examinar uma peça musical do começo ao fim antes de se tirar qualquer conclusão. O campo de *reconhecimento e entendimento de som* engloba *transcrição automática de música* (sinal de áudio como entrada e partitura impressa como saída), e *entendimento musical em performance* (humanos e computador interagindo) [Roads 85] e [Dannenberg 87].

Em *Harmonização de melodias*, sistemas especialistas têm sido utilizados para recriar um determinado estilo musical, servindo a propósitos de análise. Já as harmonizações em tempo real requerem um maior grau de complexidade, e entre as técnicas mais usadas estão *raciocínio baseado em casos* e *redes neurais* [Gang et alii 97].

Por último, no *Reconhecimento de timbres*, um dos mais complexos campos do conhecimento em música, redes neurais têm sido utilizadas para tal fim. Em [Pérez et alii 95], uma rede neural auto-organizável de Kohonen é empregada com 12×8 neurônios, para reconhecimento de quatro padrões (cordas, madeiras, metais, percussão).

2.3 O Conhecimento Musical e sua Representação

Devido à complexidade dos elementos que compõem o discurso musical, diferentes níveis de representação podem ser adotados. Passando de um nível mais simbólico (uma partitura impressa, por exemplo), a um nível mais concreto (um sinal de áudio), existem particularidades na informação contida em cada um destes níveis [Dannenberg 93].

Às vezes a forma de representação pode se adequar a uma área específica, como regras de produção aplicadas à harmonia e contraponto, ou o uso de gramáticas em análise musical e composição.

2.3.1. Estrutura, Hierarquia, Tempo e outros Atributos

Uma estrutura musical pode ser classificada em dois tipos: como *subdivisão* (composta num primeiro nível de elementos como notas, acordes, frases, compassos, vozes, etc., e num segundo nível, de elementos como seções, movimentos, etc.), e como *restrição* (identificação das características de uma peça musical, de um estilo de composição, etc.) [Howell et alii 91].

Em [Honing 93] são definidas três tipos de estruturas: *tácita*, quando existe apenas as primitivas de uma representação, *implícita*, quando as relações estruturais precisam ser calculadas da representação, e *explícita*, quando as relações entre entidades musicais são claramente representadas.

O tempo ainda pode ser investigado sob duas abordagens (Figura 2.2): numa, o vê como fenômeno *contínuo*. Um exemplo típico é a amostragem digital. Uma outra abordagem, o analisa como fenômeno *discreto*, ou seja, os dados musicais são representados como eventos separados, cada um ocupando um ponto ao longo do eixo do tempo. Um exemplo é a representação MIDI, onde uma nota num evento tem delimitada seu tempo inicial e duração [Dannenberg 93].

O conceito de hierarquia como uma forma de representar estruturas musicais tem sido amplamente usado por vários sistemas de representação, e até em conjunto com o conceito de orientação a objeto. Tomemos como exemplo: vozes, frases e seções. Uma frase pode ser cortada ao longo de uma seção, pode estar distribuída em várias vozes, ou uma voz pode conter várias frases. Assim, é natural que múltiplas hierarquias coexistam [Dannenberg 93].

Intercalando hierarquia e tempo, Balaban cria o conceito de *peça-de-música*. Estes dois aspectos (hierarquia e tempo), podem ser encontrados em todo processo musical [Balaban 92].

Há variantes também na forma como o tempo é medido. Na representação MIDI a medição é relativa, ou seja, é feita de acordo com o pulso rítmico da música (em batidas por minuto), enquanto que outros sistemas utilizam uma

medição absoluta. Um exemplo é o padrão SMPTE, no qual a contagem é feita em horas, minutos, segundos e *frames* (quadros) [Roads 96].

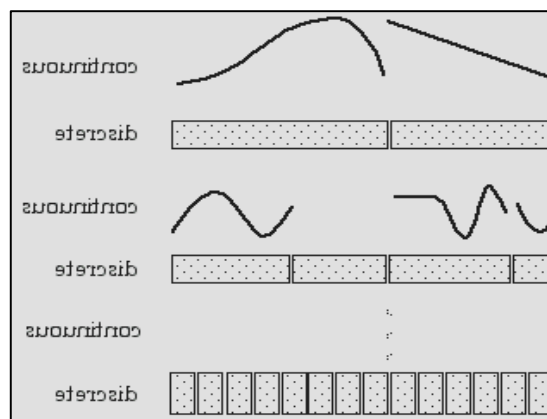


Figura 2.2 Representações discreta e contínua do tempo musical [Dannenberg et alii 97].

Além do tempo, outros componentes básicos de uma estrutura musical são complexos em sua análise. A *altura* (de uma nota) por exemplo, pode ser uma propriedade acústica, uma percepção psicológica ou uma abstração simbólica relacionada a intervalos [Dannenberg 93].

No sistema bem-temperado, as *alturas* (frequências) das notas têm valores fixos com um nome e uma oitava associados às notas. Como existem apenas doze frequências fixas para uma oitava, uma nota pode ser representada por mais de um nome. A representação MIDI por exemplo, segue este sistema e estabelece números para cada uma dessas frequências [Selfridge-Field 97].

2.3.2 Regras como Representações

O uso de *regras de produção* é adequado para expressar o conhecimento em domínios como harmonia musical e Contraponto. As regras são geralmente do tipo: se *condição* então *ação*, onde a condição testa a memória de trabalho para uma propriedade, e a ação executa alguma modificação nesta memória. Técnicas de busca podem ser acrescentadas às regras para melhorar a avaliação na resolução de problemas [Dannenberg et alii 97].

2.4 Informática e Educação Musical

A forma como o computador é utilizado em educação musical pode ter um enfoque mais passivo (no caso de Tutoriais que têm como objetivo transmitir aos alunos conceitos e habilidades específicas), ou mais criativo (onde os alunos podem exercer atividades de experimentação e improvisação) [Repsold 93].

Segundo Swanwick, é importante em educação musical os alunos terem diferentes tipos de experiências musicais em múltiplos ambientes, priorizando-se o envolvimento *com música*, como atividades de composição e performance, e não apenas o conhecimento *sobre música*, por exemplo, atividades de aquisição de habilidades (instrumental, auditiva, etc...) ou estudos da literatura musical [Swanwick 79].

Na elaboração de um produto de *software* educacional de música, segundo Smith, algumas questões devem ser consideradas: 1) Quais tarefas vão ser executadas pelo aluno ? 2) Como a música vai ser mostrada? 3) Como a música vai ser representada para a máquina? 4) Como o computador vai prover retroalimentação para o aluno? Com base nestas perguntas, o autor sugere uma classificação de *design* que enfatiza um dos seguintes aspectos: definição de tarefas do aluno, *design* da interface, representação do conhecimento e retroalimentação [Smith 2000].

2.4.1 Sistemas Tutores Inteligentes (STIs) em Música

Duas abordagens ligadas à Inteligência Artificial na Educação musical (no período de 1970 a 1987), os tradicionais Sistemas Tutores Inteligentes (STIs) e os *Micromundos*, influenciam ainda hoje muitos sistemas. As limitações constatadas são: os tradicionais STIs não são muito adequados para domínios voltados para “procura de problemas”, ou seja, áreas como composição musical por exemplo (o aprendiz de composição precisa criar objetivos ou problemas para serem resolvidos, para depois então avaliar, revisar, aproveitar ou rejeitar), enquanto que na segunda abordagem (*Micromundos*), é preciso haver mais guia para o aprendiz [Holland 2000].

Os STIs são adequados em geral para áreas que lidam com regras e identificação de erros, tais como harmonia a quatro partes (vozes) ou Contraponto. Exemplos destes sistemas são:

1) *Vivace*, um sistema especialista baseado em regras para escrever corais a quatro vozes, que toma como entrada uma melodia coral e escreve o baixo e as duas vozes intermediárias [Holland 2000].

2) *LASSO*, um STI para escrever contraponto do séc. XVII a duas vozes [Holland 2000].

Outra abordagem em STIs, inclui a negociação e diálogo como parte do ensino. Ou seja, se aplica quando o conhecimento do tutor é incompleto e o conhecimento do aprendiz pode ajudar na escolha de soluções. Um exemplo desse tipo de abordagem é o sistema INTERPRET, que trata da expressão musical em performance instrumental. A idéia principal aqui é de que sistema e aprendiz negociem um conjunto de possíveis soluções para a relação entre a estrutura analisada da peça e sua interpretação [Baker 92].

2.4.2 Softwares Educacionais de Música

Auralia (para Windows) - Produzido pela *Rising Software*

É um programa para treinamento de percepção musical para intervalos, acordes, ritmos e melodias e cadências harmônicas, com tutoriais para cada um dos tópicos (Figura 2.3). São colocados cinco níveis de dificuldade e ao final do treinamento é mostrado o *score* do aluno.

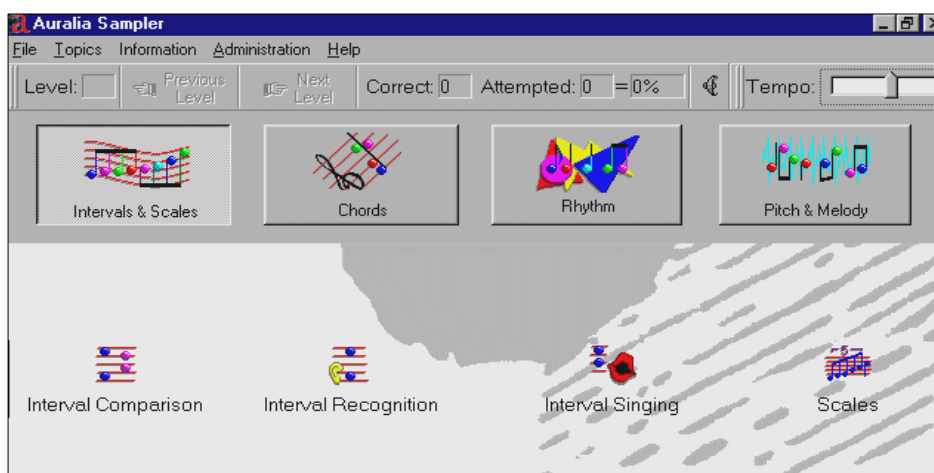


Figura 2.3 Tela inicial do *software* Auralia [Auralia 2001].

Ear Master Pro 4.0 (para Windows) - Produzido pela MidiTec.

É um programa para treinamento auditivo que inclui percepção de intervalos, escalas e acordes, ditado melódico e rítmico (Figura 2.4). Permite ao aluno quatro formas de entrada: pela notação tradicional, pelo nome das notas, pelo instrumento (via MIDI) ou por voz (microfone).

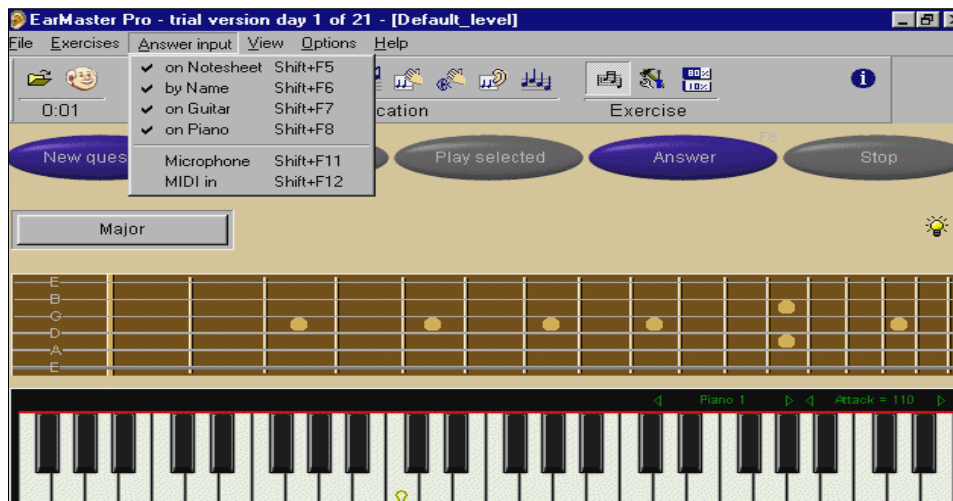


Figura 2.4 Software Ear Master Pro [Ear master 2001].

Band-in-a-Box (para Windows) - Produzido pela PG Music Inc.

É um programa para acompanhamento musical. A harmonia é introduzida na forma de cifras, com acompanhamento de baixo, bateria, piano, guitarra e cordas. O programa dispõe também de um seqüenciador [Band-in-a-Box 2001].

2.5 Trabalhos Correlatos

Apresentamos aqui trabalhos que de alguma forma estão relacionados ao nosso, seja com objetivos de ensino no domínio musical ou no próprio domínio da harmonia tradicional com uma outra abordagem, como por exemplo, programas que geram harmonizações.

1. Expert Piano (Um Ambiente de Auxilio à Aprendizagem Musical)

É um sistema tutor voltado para o ensino de Piano. O sistema tenta simular o comportamento de um professor assistindo à interpretação de um aluno e seu desempenho. O módulo principal do sistema (Sistema Tutorial Inteligente),

é responsável pelo diagnóstico e tratamentos de erros cometidos pelo aluno. Remete também sugestões e remediações contidas na base de conhecimento.

O módulo principal do sistema contém:

- Módulo *Instrutor* – responsável pela escolha do material a ser empregado para cada aluno.
- Módulo *Especialista* - é um sistema especialista que contém as bases de conhecimento e o mecanismo que manipula este conhecimento.
- Módulo *Modelo do Aluno* - verifica o conhecimento de cada aluno gerando hipóteses sobre suas concepções e estratégias de raciocínio. Este módulo compreende o conjunto de erros cometidos pelo aluno e o contexto em que foram identificados.

Na parte de implementação, a ferramenta utilizada pelo módulo principal do sistema (Toolbook 3.0), contém uma linguagem de script que pode chamar DLLs. Através de rotinas em C++ pode-se acessar às funções do Windows. O Expert Piano utiliza o modelo tradicional na modelagem do aprendiz. Os erros do aprendiz são confrontados com a base de conhecimento do especialista, e a partir daí são feitas as remediações e sugestões necessárias [Glanzmann 95].

A utilização da representação MIDI usada no Expert Piano para captação e comparação das informações pode ser insuficiente em termos de representação musical.

2. SETMUS (Sistema Especialista para Teoria Musical)

Implementado em HyperCard, o sistema possui um banco de dados formado por escalas e arpejos dos vinte e quatro modos maiores e menores. Por meio de uma calculadora musical o aluno pode formar escalas com base nos nomes das notas e suas alterações.

Caso queira consultar a calculadora musical o aluno deve escolher a nota, o modo e finalmente a opção de escala ou arpejo. A interface permite a entrada de notas pelo usuário usando o *mouse* diretamente na tela. Tem também um recurso didático de visualização das notas com seus nomes nos locais onde são escritas na partitura.

O sistema fornece ainda justificativas e explicações mediante perguntas pré-determinadas, e pode fornecer diagnóstico de erro com a função de *playback*, comparando escalas e arpejos tocados pelo aluno com os do sistema. Com isso o aluno pode visualizar na partitura os erros cometidos. O sistema usa um algoritmo de reconhecimento de escalas e arpejos para obter os intervalos entre as notas. A partir daí uma série de regras determinam a estrutura das escalas e arpejos (ascendente, descendente, maior, menor).

O SETMUS serve como uma ferramenta para auxílio de teoria musical que é a base para o ensino de harmonia [Fritsch & Viccari 95].

Dois aspectos a comentar sobre o SETMUS:

- 1) Não são representadas as notas alteradas com dobrado-sustenido e dobrado-bemol, o que em termos de representação harmônica é insuficiente.
- 2) O sistema é monofônico, ou seja, o aluno não pode escutar várias notas tocadas simultaneamente.

3. An Expert System for Harmonizing Chorales in the Style of Bach

É um sistema especialista usado para harmonização. Neste exemplo, o autor criou uma nova linguagem de programação chamada BSL (Backtracking Specification Language). Há um total de 350 regras de produção, restrições e heurísticas no programa. O objetivo é harmonizar uma melodia de um coral (Bach), e dar uma análise para esta harmonização. O programa toma um código alfanumérico da melodia coral como entrada, e a saída é em notação convencional de música. Uma interface interativa é inserida através do compilador BSL, que explica as escolhas feitas.

O sistema usa um processo de *backtracking* (retorno), para implementar os vários “pontos de vistas” do coral citado acima : vista do “esqueleto do acorde”. Observa o acorde como uma seqüência sem ritmo. Isto permite referenciar atributos tais como “altura” e “alterações” de uma nota ou voz de qualquer acorde na seqüência. Vista da “cadeia melódica”: observa a seqüência de notas individuais de um ponto de vista puramente melódico. Referencia a “altura” e a “alteração” de qualquer nota de uma voz. Vista da

“cadeia melódica reunida”. Similar à anterior, exceto que observa notas de “alturas” repetidas, e vista do “tempo fatiado”: observa o coral como uma seqüência vertical de pequenas unidades de tempo e impõe restrições harmônicas. Uma questão importante também é como introduzir na base de conhecimento heurísticas adequadas. O propósito destas heurísticas aqui é fazer uma estimativa, a cada passo, entre os possíveis caminhos, a partir de uma visão parcial da obra, para se chegar ao melhor acabamento, já que um programa baseado apenas em regras absolutas e seleção randômica pode gerar um processo confuso [Ebcioğlu 92].

4. Harmony Space

É um conjunto de ferramentas que serve para composição musical, análise harmônica e para aprendizagem de harmonia musical. Baseia-se na idéia de que relações harmônicas podem ser expressas como uma grade espacial na qual notas, intervalos, acordes e seqüências de acordes podem ser manipulados e visualizados pelo usuário. A interface é desenhada segundo o princípio de que relações harmônicas na escala cromática podem ser representadas como uma matriz bidimensional com terças menores no eixo vertical e terças maiores no eixo horizontal, e com isso vários modelos podem ser tirados da matriz para formar seqüências de acordes [Holland 94].

2.6 Conclusão

Neste capítulo apresentamos subsídios para uma melhor compreensão na área interdisciplinar de Computação e Música. Entre os sistemas analisados nos trabalhos correlatos, nenhum atende inteiramente aos requisitos que estão presentes no sistema SHART-Web tais como: apresentação do domínio da Harmonia tradicional voltado especificamente para o ensino, integração adequada do Sistema Tutor Inteligente (STI) com hipermídia e disponível na Internet/Web.

No próximo capítulo, daremos o suporte teórico para o desenvolvimento de nosso sistema que será exposto em detalhes a partir do capítulo 4.

Capítulo 3

Sistemas Computacionais de Apoio

3.1 Introdução

Neste capítulo trataremos dos diversos sistemas computacionais que têm uma relação direta com o nosso sistema. Inicialmente abordaremos os Sistemas Baseados em Conhecimento (SBC), na sua concepção mais geral. Em seguida, abordaremos os Sistemas multiagentes, suas características e aplicações, os Sistemas Tutores Inteligentes (STIs) e sua evolução em direção aos Ambientes Interativos de Aprendizagem no contexto da hipermídia, e finalmente o Ensino a distância.

3.2 Sistemas Baseados em Conhecimento

3.2.1 Definição e Características

Como uma das áreas da Inteligência Artificial, os Sistemas Baseados em Conhecimento (SBC), são sistemas que utilizam conhecimento e o representam explícita e separadamente do módulo principal do sistema [Stefik 95].

Um conhecimento, no sentido em que é usado em Sistemas de Conhecimento, refere-se à experiência codificada de agentes, sendo a experiência a fonte da informação para resolver problemas. A experiência precisa ser articulada para se tornar conhecimento explícito, e a experiência codificada precisa ser organizada e generalizada para guiar uma futura ação [Stefik 95].

3.2.2 Arquitetura de SBC

A arquitetura básica de um SBC contém uma *base de conhecimento*, duas *interfaces homem-máquina* (*usuário e especialista*) e um *mecanismo de inferência*. A *base de conhecimento* é o repositório para o conhecimento usado pelo sistema e serve de guia na resolução de problemas. A *interface do*

usuário responde pela comunicação com o sistema (consultas, perguntas, respostas e mostras de resultados).

A *interface do especialista* provê acesso para atualizações, testes e correções da base de conhecimento. O *mecanismo de inferência* independente do domínio e raciocina sobre a resolução de problemas segundo o conteúdo da *base de conhecimento*.

A seguir a arquitetura característica clássica de um SBC:

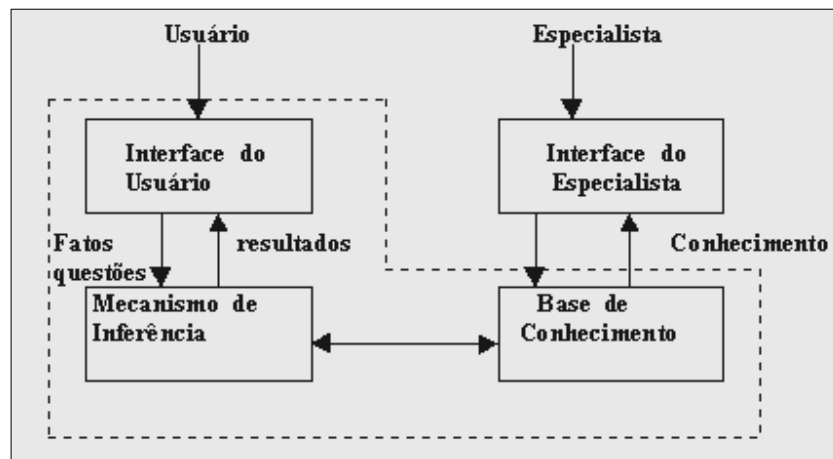


Figura 3.1 Arquitetura de SBC [Stefik 95].

3.2.3 Áreas e Aplicações

Diferentes áreas de aplicações são estudadas hoje nos Sistemas Baseados em Conhecimento, tais como: processamento de linguagem natural, diagnóstico médico, previsão financeira, sistemas tutores, etc. Cada uma destas áreas tem suas particularidades, porém com um princípio comum, que é uma base de conhecimento a partir da qual são feitas as inferências usadas pelo sistema [KBS 99].

3.2.4 Representação do Conhecimento

Uma representação pode ser definida como um conjunto de convenções sintáticas e semânticas para descrever coisas. A sintaxe especifica os símbolos usados e o modo que eles podem ser arranjados. A semântica especifica como o significado é ligado ao sistema [Stefik 95].

Para se resolver os complexos problemas encontrados em Inteligência Artificial, se precisa tanto de uma grande base de conhecimento, como

mecanismos para manipular este conhecimento e criar soluções para novos problemas. As coisas que queremos representar são os fatos. E, para representar estes fatos, precisamos de algum tipo de formalismo [Rich 91].

3.2.4.1 Tipos de Representação

Entre os diversos tipos de formalismos para representar o conhecimento, mencionamos: *Regras de Produção*, *Lógica*, *Redes Semânticas*, e *Quadros*.

1) *Regras de Produção* - uma regra de produção consiste de duas partes : um antecedente, expressado por uma condição, e um conseqüente. Exemplo: “*Se há fumaça então há fogo*”. Uma das vantagens em se usar regras é que são *modulares*, ou seja, encapsulam conhecimento e podem ser facilmente expandidas.

2) *Lógica* - o sistema lógico trata com a manipulação de variáveis lógicas que representam proposições. Estas proposições podem ser classificadas como verdadeiras ou falsas e podem ser combinadas através de conectivos (conjunção, disjunção, negação, condicional e bicondicional).

Conectivo	Significado
\wedge	e ; conjunção
\vee	ou ; disjunção
\sim	não ; negação
\rightarrow	se...então ; condicional
\leftrightarrow	se e somente se ; bicondicional

Tabela 3.1 Conectivos e significados

Uma extensão da lógica proposicional é o cálculo de predicados, e sua forma mais simples é a lógica de 1ª ordem.

3) *Redes Semânticas* - a estrutura de uma rede semântica é mostrada graficamente em termos de nós e arcos conectando estes nós. Os nós são geralmente usados para representar objetos, conceitos ou situações. Os arcos são usados para expressar relações binárias.

4) *Quadros (Frames)* - proposto como um método para o entendimento em visão e linguagem natural, um *quadro* é basicamente uma coleção de atributos (*slots*) e valores associados que descrevem alguma entidade no mundo. É útil principalmente para representar conhecimento causal, associando por exemplo determinadas situações às ações correspondentes. A principal dificuldade no uso de *quadros* é que não provêm um modo de definir *slots* inalteráveis.

Desde que qualquer *slot* pode ser mudado, as propriedades que um *quadro* herda, podem ser alteradas ou canceladas em qualquer lugar na hierarquia [Giarratano & Riley 89] e [Rich 91].

3.2.4.2 Métodos de Inferência e Controle

O mecanismo de inferência é o componente que trabalha diretamente com a base de conhecimento do SBC. Contém os métodos de inferência a serem usados pelo controle para a resolução de problemas. Alguns métodos de inferência são:

1) *Encadeamento progressivo* - este método raciocina a partir de um fato, para as conclusões resultantes deste fato.

2) *Encadeamento regressivo* - envolve um raciocínio reverso à hipótese, ou seja, uma conclusão potencial a ser provada para os fatos, os quais suportam a hipótese [Giarratano & Riley 89] e [Rich 91].

3.3 Sistemas Multi-Agentes

3.3.1 O Contexto

Os sistemas multiagentes são um tipo de SBC, vinculados a uma subárea da Inteligência Artificial Distribuída (IAD). A IAD diz respeito ao estudo e concepção de sistemas com várias entidades interagindo, logicamente e espacialmente distribuídas e que podem ser autônomas e inteligentes [Weiss 99].

A IAD pode ser subdividida em duas outras áreas:

- Sistemas Multiagentes (SMA)

- Resolução Distribuída de Problemas (RDP).

Tanto nos Sistemas Multiagentes como na Resolução Distribuída de Problemas os agentes compartilham um mesmo ambiente. A diferença está mais na forma de interação entre os agentes. Nos modelos RDP, os agentes buscam mais a cooperação. Nos modelos SMA, os agentes competem por recursos e precisam muitas vezes resolver conflitos [Scherer & Schlageter 95].

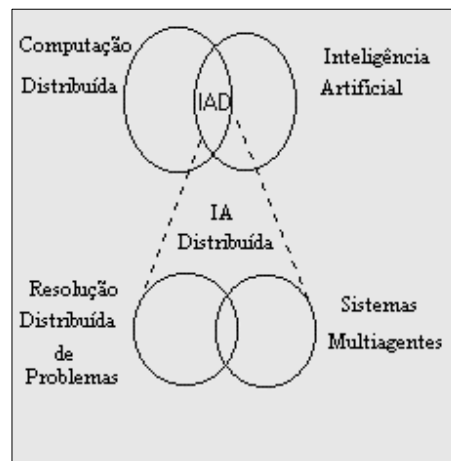


Figura 3.2 - SMA, IAD e IA [Stone & Veloso 97].

3.3.2 Definições de Agente

Antes de entrarmos em detalhes sobre os sistemas multiagentes, é preciso conhecermos o que vem a ser o termo *agente*. Um *agente* é um sistema computacional situado em algum ambiente, e que é capaz de ações autônomas neste ambiente, de forma a encontrar seus objetivos [Weiss 99].

Encontramos em [Russel & Norvig 95] o conceito de *agente* como sendo algo com a capacidade de atuar sobre um ambiente, através de sua percepção sobre este ambiente.

[Genesereth & Ketchpel 94] descrevem *agentes* como componentes de “software” que se comunicam entre si através da troca de mensagens numa linguagem de comunicação de agentes.

Segundo [Wooldridge & Jennings 95], um sistema pode ser visto como um agente, se ele possui as seguintes propriedades: autonomia (pode funcionar sem

intervenção humana, com suas ações baseadas em seu conhecimento sobre o ambiente), habilidade social (interação com outros agentes através de uma linguagem comum), reatividade (percebe mudanças em seu ambiente e atua de acordo com estas mudanças), pró-atividade (atuação do agente não apenas por percepção, mas também apresentando iniciativa própria).

3.3.3 Classificação dos Agentes

Em [Nwana 96] é proposta a seguinte classificação para os agentes:

1) *Agentes colaborativos* - cada agente contribui com sua própria técnica para a solução de um problema. A negociação é necessária neste ambiente, para estabelecer acordos entre eles.

2) *Agentes de interface* - são conhecidos também como assistentes pessoais. Suportam e providenciam assistência para o usuário aprender uma aplicação. Estes agentes podem aprender tanto com usuário (observando e imitando, recebendo retorno positivo ou negativo ou recebendo instruções do mesmo), como também podem aprender com outros agentes, solicitando orientações.

3) *Agentes móveis* - podem se mover em redes como a Internet, interagindo com outros *hosts*, coletando e retornando com informações. O principal problema com relação a esse tipo de agente é a segurança, já que podem transportar vírus.

4) *Agentes reativos* - não têm um modelo interno simbólico de ambiente. Reagem a um estímulo de entrada que é controlado por algum estado ou evento no seu ambiente.

5) *Agentes híbridos* - são a união de diferentes classes de agentes com uma unidade de controle no agente, permitindo cada tipo interoperar efetivamente.

6) *Agentes heterogêneos* - nos sistemas de agentes heterogêneos há uma interoperabilidade. O problema maior é a comunicação entre eles.

7) *Agentes cognitivos* - são agentes que coordenam suas atividades trabalhando cooperativamente.

3.3.4 Definição e Características dos Sistemas Multiagentes

Várias definições têm sido propostas para o termo “sistemas multiagentes”. Do ponto de vista da Inteligência Artificial Distribuída, um sistema multiagentes é uma rede fracamente acoplada de entidades de resolução-de-problemas que trabalham juntas para achar respostas aos problemas que estão além das capacidades individuais ou conhecimento de cada entidade [Durfee & Rosenschein 94].

Segundo [Weiss 99], os ambientes multiagentes possuem as seguintes características:

- Especificação de uma infra-estrutura de comunicação e protocolos de interação;
- São tipicamente abertos e têm uma estrutura não centralizada;
- Contêm agentes que são autônomos e distribuídos e podem ser centrados em seu próprio interesse ou cooperativos.

3.3.5 Arquiteturas de Sistemas Multiagentes

Segundo Maes, uma arquitetura de um agente é uma metodologia particular para construir agentes, especificando como um agente pode ser decomposto na construção de um conjunto de módulos componentes, e como estes módulos podem ser feitos para interagirem. Uma arquitetura abarca técnicas e algoritmos que suportam esta metodologia [Maes 91].

De acordo com o grau de complexidade de uma arquitetura (que inclui as necessidades específicas da aplicação e o nível de inteligência dos agentes), Knapik propõe a seguinte classificação : *Arquitetura simples*, composta por apenas um agente, *Arquitetura moderada*, composta por agentes que realizam tarefas iguais, mas têm usuários diferentes e residem em diversas máquinas, e *Arquitetura complexa*, composta por tipos diferentes de agentes, com as características de autonomia e cooperação, podendo ainda estar em plataformas diferentes [Knapik & Johnson 98].

Encontramos em [Weiss 99] quatro abordagens para as arquiteturas de sistemas multiagentes:

1) *Arquitetura Baseada em Lógica* - nesta abordagem, a tomada de decisão (de um agente) é vista como uma dedução. Um programa de um agente (a estratégia de tomada de decisão) é codificado como uma teoria lógica, e o processo de selecionar uma ação se reduz a um problema de prova (de teorema). A principal desvantagem desta abordagem é que a representação e o raciocínio sobre propriedades dinâmicas (mudanças sobre o tempo, por exemplo) tornam-se extremamente difíceis.

2) *Arquitetura Reativa* - o termo *reativo* é devido ao fato de que tais sistemas são freqüentemente percebidos como simplesmente reagindo ao ambiente, sem raciocinar sobre este. A arquitetura mais conhecida desse tipo é a arquitetura “*subsumption*”. Há duas definições sobre esta. A primeira afirma que a tomada de decisão de um agente é realizada através de um conjunto de tarefas acompanhando comportamentos. Cada comportamento é visto como uma função de uma ação individual, que toma a percepção como entrada e mapeia esta para uma ação a ser executada. Cada um destes módulos de comportamento tem a intenção de concluir alguma tarefa particular. A segunda definição afirma que muitos comportamentos podem ser ativados simultaneamente, sendo necessário um mecanismo para escolher entre as diferentes ações selecionadas.

3) *Arquitetura BDI (Belief-Desire-Intention)* - baseada no raciocínio prático que envolve dois processos: decidir qual objetivo que se quer concluir, e como concluí-lo. O processo interno do agente nesta arquitetura tem os seguintes passos : 1) Um conjunto de crenças, representando informações que o agente tem sobre o ambiente, 2) Uma função de revisão, que pode determinar novos conjuntos de crenças a partir das atuais, 3) Uma função de opções, que determina as opções disponíveis para o agente (seus desejos) com base na crenças correntes sobre seu ambiente e nas intenções correntes.

4) *Arquitetura em Camadas* - vários subsistemas são arranjados numa hierarquia de camadas interagindo entre si. A idéia básica aqui é criar subsistemas separados para tratar com os diferentes tipos de comportamento dos agentes (reativo e pró-ativo). Existem dois modelos básicos neste tipo de arquitetura:

Horizontal e *Vertical*. No primeiro modelo, cada uma das camadas são diretamente conectadas a um sensor (percepção) de entrada, e uma ação de saída. Cada camada age como um agente, criando sugestões de como agir. No segundo modelo as entradas (percepções) e saídas (ações) passam por cada uma das camadas. Este modelo tem dois tipos de controle: controle de um passo e controle de dois passos.

3.3.6 Comunicação entre Agentes

3.3.6.1 Coordenação e Tipos de Mensagens

Os agentes se comunicam de forma a alcançar melhor seus objetivos ou os objetivos do sistema ao qual pertencem. Para se executar alguma atividade num ambiente compartilhado é necessário coordenação. A coordenação engloba tanto a cooperação (entre agentes não antagônicos), como a competição (agentes que agem em seu próprio interesse) [Weiss 99].

Ferber distingue quatro tipos de ações de coordenação:

- *Coordenação por sincronização* - gerencia várias ações concorrentes e verifica se os resultados são coerentes;
- *Coordenação por planejamento* - as ações são divididas em duas fases: a primeira, considera um conjunto de ações para concluir um certo objetivo, e a segunda, seleciona um dos planos para posterior execução;
- *Coordenação reativa* - esta técnica considera que é mais fácil implementar mecanismos de coordenação baseados em agentes reativos do que o planejamento prévio de ações;
- *Coordenação por regulação* - o princípio desta técnica é colocar regras de comportamento que tenham como alvo eliminar possíveis conflitos [Ferber 99].

Existem dois tipos básicos de mensagens: afirmações e perguntas. Outros tipos de mensagens incluem: réplica, requerimento, explanação, comando, permissão, recusa, concordância, confirmação, retração, negação [Weiss 99].

3.3.6.2 Cooperação e Distribuição de Tarefas

Quando os agentes decidem cooperar em uma determinada tarefa ou durante um certo tempo, firmam uma espécie de *compromisso* (commitment).

Stone & Veloso definem três tipos de *compromissos*: *interno*, quando o agente assume um compromisso para si próprio ; *social*, quando concorda com outro agente em realizar alguma tarefa, e *coletivo*, quando assume um certo papel em meio aos outros agentes [Stone & Veloso 97].

Weiss faz uma distinção entre *compromisso* (commitment), que é visto como uma garantia para se empreender um curso especificado de uma ação, e uma *convenção* (convention), que tem um sentido de gerenciamento de *compromissos* quando mudam as circunstâncias [Weiss 99].

Na distribuição de tarefas, os agentes podem assumir papéis de *clientes* (agentes precisando de informação), ou de *servidores* (agentes capazes de fornecer um serviço). A forma de distribuição de tarefas pode ser : *centralizada* ou *distribuída*. Na forma *centralizada*, se a estrutura é hierárquica, um agente superior ordena um subordinado para levar adiante uma tarefa. Se a estrutura é igualitária, a distribuição de tarefas é feita por agentes especiais que gerenciam todo o procedimento de alocação. Na forma *distribuída*, cada agente individualmente tenta obter os serviços que ele precisa através dos servidores [Ferber 99].

3.3.6.3 Protocolos de Interação

Protocolos de interação estabelecem uma conversação entre agentes, ou seja, controlam a troca de uma série de mensagens entre eles. No caso de agentes que têm objetivos similares ou problemas comuns, por exemplo, o objetivo do protocolo é manter a estrutura global coerente, sem violar a autonomia de cada um. Aspectos importantes neste caso são: determinar objetivos compartilhados, determinar tarefas comuns, evitar conflitos desnecessários [Weiss 99].

Os protocolos utilizados podem ser:

1) *Protocolos de coordenação* - um exemplo de coordenação é a sincronização das ações dos agentes, de modo a assegurar que não haja redundância numa resolução de problema. A distribuição de controle e dados produz também uma melhor coordenação nesses protocolos. Um controle distribuído significa que os agentes têm um grau maior de autonomia. A

desvantagem de controle e dados distribuídos é que o conhecimento do estado geral do sistema é disperso podendo levar a um grau de incerteza sobre as ações de cada agente.

2) *Protocolos de cooperação* - uma estratégia básica de muitos protocolos de cooperação é decompor para depois distribuir uma determinada tarefa. Isto pode ser feito pelo projetista do sistema, ou pelos agentes usando planejamento hierárquico. A distribuição das tarefas segue alguns critérios: evitar sobrecarregar os recursos, fazer um agente com uma visão mais ampla assinalar tarefas para outros agentes, assinalar tarefas altamente interdependentes com uma proximidade espacial ou semântica e reassinalar tarefas para completar outras mais urgentes [Weiss 99].

A forma mais usada entre os protocolos de cooperação é o *Contract Net*. Este protocolo é voltado para a resolução de problemas cooperativos entre os agentes. As partes envolvidas são: O proponente de uma tarefa a ser resolvida, chamado *gerenciador* e os potenciais candidatos a resolver a tarefa, chamados *contratantes*. Funciona da seguinte maneira: um agente *gerenciador* anuncia para os outros agentes, uma tarefa a ser resolvida. Os agentes *contratantes* avaliam o anúncio e alguns se propõem a resolver a tarefa. O gerenciador então indica o agente mais apropriado [Weiss 99].

3.3.6.4 Linguagens

Existem duas abordagens principais para linguagens de comunicação entre agentes. A primeira é *procedural*, onde a comunicação é baseada sobre um conteúdo executável. A segunda abordagem é declarativa, onde a comunicação é baseada em definições, asserções e outras [Flores-Mendez 2000].

Um aspecto fundamental para a interação de agentes é separar a semântica do protocolo de comunicação (que precisa ser independente do domínio), da semântica da mensagem embutida (que pode depender do domínio) [Weiss 99].

A seguir, algumas das linguagens usadas para comunicação entre agentes:

KQML (Knowledge Query and Manipulation Language) - foi concebida tanto como um formato de mensagem como um protocolo de manipulação de mensagem para suportar conhecimento compartilhado entre agentes em tempo de execução. Consiste de três camadas : *camada de comunicação*, que descreve os parâmetros de comunicação de baixo nível tais como o remetente, o receptor e identificadores de comunicação, *camada de mensagem*, que contém uma “performative” e indica o protocolo de interpretação, e *camada de conteúdo*, que contém a informação relativa à “performative” submetida. KQML precisa operar dentro de uma infraestrutura de comunicação que permita aos agentes se localizarem. Esta infraestrutura não faz parte da especificação da linguagem, e para isto, sistemas implementados em KQML utilizam programas chamados roteadores [Flores-Mendez 2000] e [Weiss 99].

KIF (Knowledge Interchange Format) - baseada no cálculo de predicados, tem sido proposta como padrão para uso em sistemas especialistas, banco de dados, agentes inteligentes, etc.. A semântica do núcleo de KIF (sem as regras e definições) é semelhante à lógica de 1ª ordem. Há uma extensão para manipular operadores não-padrões e há também uma restrição, que os modelos precisam satisfazer a vários esquemas de axiomas, para dar significado ao vocabulário no formato [Genesereth & Fikes 94].

3.4 Sistemas Tutores Inteligentes e Hiperídia

Apresentaremos nesse tópico uma visão geral dos Sistemas Tutores Inteligentes, sua evolução em direção aos Ambientes Interativos de Aprendizagem, e os recursos de hiperídia que foram incorporados a estes sistemas, assim como um maior controle no processo de aprendizagem por parte do usuário.

3.4.1 Sistemas Tutores Inteligentes (STIs)

3.4.1.1 Breve Histórico

Os primeiros programas de instrução assistida por computador (CAI – Computer Aided Instruction) foram concebidos para melhorar a performance do estudante em domínios baseados em habilidades. Estes programas tinham

uma visão comportamental do processo de aprendizado. Não davam os resultados de como as pessoas aprendem. Assumiam que, dada uma informação ao estudante, este poderia absorvê-la [Urban-Lurain 96].

Ainda na década de 60 surgem os *Micromundos*, tendo uma proposta diametralmente oposta aos CAIs, ou seja, o aluno passa a construir seu próprio conhecimento. A transição para os ITSs (Intelligent Tutoring Systems - Sistemas Tutores Inteligentes), também chamados de ICAIs, se dá no início da década de 80. Nos ITSs a noção de modelo é de central importância: modelo do domínio, modelo do estudante e modelo do processo de comunicação. Enfatizando este aspecto da comunicação, Wenger redireciona as pesquisas dos ITSs focando a parte cognitiva como preponderante em relação aos modelos computacionais pedagógicos e do domínio [Urban-Lurain 96].

Da combinação dos ITSs e *Micromundos* surgem os ILEs (Interactive/Intelligent Learning Environments – Ambientes Interativos de Aprendizagem). Enquanto que nos ITSs o sistema detém o controle da interação, esta passa a ser mais flexível nos ILEs.

3.4.1.2 Definições e Características

Um Sistema Tutor Inteligente (STI) é definido, segundo Auberger, como um sistema instrucional baseado em computador, que ensina o estudante de uma maneira interativa, usando os conceitos de Inteligência Artificial [Auberger 98].

Murray ressalta como um dos objetivos dos STIs, ser capaz de modelar complexos comportamentos de ensino, os quais se adaptam às necessidades do estudante, à situação de aprendizagem e ao assunto da instrução [Murray 99].

Para Kearsley, os STIs fazem parte de um campo interdisciplinar envolvendo Ciência da Computação, Psicologia cognitiva e Educação (Figura 3.3) [Kearsley 87].

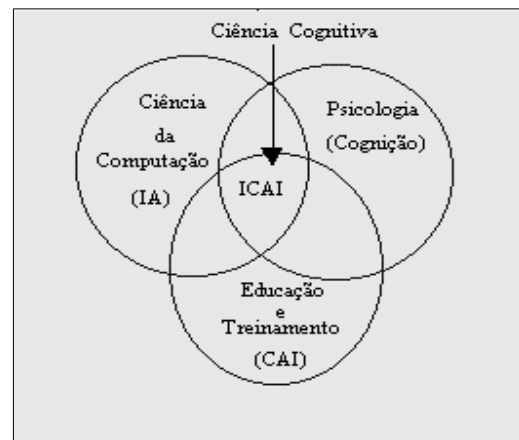


Figura 3.3 - Domínio dos STIs (ICAI) [Kearsley 87].

3.4.1.3 Componentes

Apesar de haver diversas arquiteturas para STIs, há um consenso quanto aos componentes básicos: *Módulo Especialista*, *Módulo do Estudante*, *Módulo Pedagógico (Tutor)* e *Módulo da Interface* [Woolf 88], [Wenger 87] e [Kearsley 87].

Módulo Especialista:

O módulo do modelo do especialista contém o conhecimento do domínio do sistema, e os mecanismos de inferência. Este modelo tem duas tarefas: serve como o conhecimento a ser apresentado, e ao mesmo tempo como um padrão para avaliar a performance do estudante [Auberger 98].

O módulo do especialista pode ser classificado em dois tipos de arquitetura, com relação à representação: *caixa-preta* (o sistema avalia o que o estudante produz, e não como isto foi alcançado. Somente os resultados finais são disponíveis), e *transparente* (cada passo no raciocínio do sistema, pode ser inspecionado e interpretado) [Wenger 87].

Wenger faz uma distinção entre conhecimento compilado e conhecimento articulado. O conhecimento compilado pode ser visto como tão especializado a ponto de ter perdido a transparência e generalidade. Já o conhecimento articulado tende a incluir justificativas, podendo ser revisto dinamicamente e adaptado a novas situações [Wenger 87].

Módulo do Estudante:

Este módulo descreve o conhecimento e crenças do estudante. É na verdade, uma representação abstrata do estudante no sistema. A principal função do modelo do estudante é permitir que o conteúdo instrucional seja designado para um aprendiz individual. Com conhecimento sobre o estudante, o tutor pode controlar a ordem e a dificuldade do material a ser apresentado, bem como prover remediações [Greer & McCalla 91a].

Na construção deste modelo, três fases podem ser percebidas: *fase de modelagem*, que envolve o entendimento do comportamento do estudante no contexto do ambiente no qual ocorre a aprendizagem, *fase de intervenção*, que engloba assistência para preencher falhas no conhecimento do estudante assim como correção de concepções errôneas, e *fase de avaliação*, que serve para validar o modelo original do conhecimento do estudante, bem como fazer os ajustes necessários [Greer & McCalla 91b].

Usualmente o conhecimento do estudante é modelado em comparação com o modelo do especialista. De acordo com esta relação pode-se distinguir vários tipos de modelos:

1) *Modelo de Sobreposição (Overlay)* - a hipótese deste modelo é que todas as diferenças entre os modelos do estudante e especialista são devidas à falta de conhecimento do estudante. O modelo do estudante é visto então como um subconjunto do modelo do especialista. Uma falha deste modelo é que não trata do conhecimento do estudante que difere do modelo do especialista. Por exemplo, se o estudante tem uma estratégia alternativa que funciona bem para um determinado problema, o tutor provavelmente tentará corrigi-la.

2) *Modelo de Perturbação (Buggy)* - este modelo não considera o conhecimento do estudante como um subconjunto do modelo do especialista, mas um subconjunto de uma união do domínio e outro conjunto com todas as concepções errôneas que o estudante poderia ter. Neste modelo é construída uma biblioteca de erros (bugs), e com o progresso do estudante, o modelo pode ser atualizado com vistas à presença ou ausência de erros conhecidos nesta biblioteca.

3) *Modelo de Incerteza (Fuzzy)* - este modelo emprega os conceitos de imprecisão e incerteza. Cada variável de conhecimento é associada com uma distribuição que possibilita ser atualizada em diferentes graus, dependendo do tipo e peso de evidência que aparece num determinado problema dado para o estudante resolver. Esta distribuição probabilística da variável de conhecimento é feita em vários níveis, abrangendo desde o nível de “não conhecimento” até o “pleno conhecimento”.

4) *Modelo de Simulação* - este modelo explica o comportamento do estudante em termos de um modelo geral e um procedimento de inferência (como o estudante interpreta observações e o modelo geral) [Greer & McCalla 91a].

Alguns autores diferenciam o processo de *modelagem do estudante* do processo de *modelagem do usuário*. Enquanto que o primeiro trata mais do que incluir no modelo e como diagnosticar concepções errôneas, o segundo trata mais de como adquirir modelos baseados em incerteza e como adquirir informações práticas [Paiva 95].

Existem ainda vários obstáculos com relação à modelagem do estudante que resultam do problema de inferir conhecimento do comportamento do mesmo. Alguns destes obstáculos são:

- O ambiente contém uma grande quantidade de incerteza e ruído;
- A inferência do estudante pode ser baseada sobre conhecimento inconsistente;
- A criatividade do estudante pode gerar novos comportamentos que requerem mais sofisticação para serem interpretados [Greer & MacCalla 91a].

Módulo Pedagógico:

O módulo pedagógico, também chamado de modelo tutor ou instrucional, usa as informações do modelo do estudante para determinar quais aspectos do conhecimento do domínio poderiam ser apresentados para o aprendiz. Estas informações, poderiam conter novos materiais, uma revisão de tópicos anteriores ou do atual. Halfff define instrução, como a forma de apresentação do conteúdo a ser ensinado, e currículo, como a seleção e o seqüenciamento

do conteúdo a ser apresentado ao aluno. Ainda segundo Halff, um STI, mesmo utilizando diferentes técnicas pedagógicas, deve apresentar as seguintes características: 1) Ter controle sobre o currículo, 2) Saber responder às questões do aluno, quanto ao conteúdo, 3) Ser capaz de determinar as necessidades do aluno [Halff 88].

Wenger, identifica neste módulo, dois tipos de suporte:

1) **Diagnóstico** - são feitas inferências sobre o estado do aprendiz em três níveis: *Comportamental* (ignora o conhecimento do Aprendiz, e foca apenas o comportamento observável), *Epistêmico* (trata com o estado de conhecimento do aprendiz, e tenta inferir aquele estado, baseado no comportamento observado) e *Individual* (são as motivações e conceitos que o estudante tem sobre o domínio).

2) **Didático** - trata da entrega do material instrucional para o estudante. É organizado em torno de quatro princípios:

Planos de ação - é um mini-curriculum usado para guiar o estudante, e prover as operações de diagnóstico.

Contextos pedagógicos - aqui os planos de ação são implementados. Podem ser: oportunistas (alguma pergunta que o estudante faz, por exemplo), ou planejados (dinamicamente controlado pelo sistema, o qual organiza a atividade e a interação).

Base de decisão - quando e como o sistema deve intervir em um objetivo didático implica às vezes a resolução de conflitos.

Níveis de objetivos - o nível do objetivo de uma operação didática é definido como o nível do modelo do estudante (comportamental, epistêmico ou individual), no qual uma operação busca modificações imediatas. No nível comportamental, as intervenções didáticas guiam a performance de uma tarefa, sem endereçar conhecimento internalizado de maneira organizada. No nível epistêmico, as operações didáticas procuram modificar o estado de conhecimento do estudante. As explanações são de central importância. No nível individual são explorados aspectos como motivações (por exemplo, mostrar um determinado conteúdo de uma forma mais atrativa, abandonar um tópico temporariamente, ou congratular o estudante por algum acerto) [Wenger 87].

Entre as estratégias de ensino mais usadas pelos STIs estão:

- *Método Socrático* - o tutor aprofunda o conteúdo do domínio a partir de um tópico no qual o aluno já tenha algum conhecimento, e conduz o estudante a sucessivas questões, para formular princípios gerais sobre casos individuais. Faz uso de diálogos entre o tutor e o estudante.
- *Método de treinamento (coaching)* - o sistema monitora as atividades do aluno e interrompe para dar conselhos. Algumas vezes o aluno lidera o processo de resolução e o tutor indica apenas sua concordância ou não. Em outras vezes o tutor orienta o aluno nas partes mais difíceis.
- *Método baseado em casos* - este modelo é usado para apoiar o ensino em domínios de conhecimento vasto. Neste caso, é preciso uma grande quantidade de regras para descrever toda a área, e uma alternativa é apresentar o domínio na forma de casos-exemplos. Raciocínio baseado em casos é uma técnica de Inteligência Artificial que procura em uma base de casos, um caso mais próximo (análogo), para aplicar a solução.
- *Método de simulação* - os ambientes que utilizam simulação, se diferenciam dos demais na natureza do modelo do domínio, que é modelado em termos quantitativos para executar cálculos das variáveis de simulação envolvidas. É um método bastante usado em sistemas de treinamento profissional, tais como: simuladores de voo, simuladores de tratamento de pacientes, etc.[Halff 88].

Módulo da Interface:

Este módulo é responsável por processar o fluxo de informações entre o sistema e o usuário ou aluno. Traduz a representação interna do sistema numa linguagem compreensível para o aluno. Embora a interface opere em cooperação com os outros módulos, suas decisões requerem um tipo diferente de conhecimento [Wenger 87].

3.4.1.4 Arquiteturas de STIs

Quanto à arquitetura dos STIs atuais (Figura 3.4), a sua concepção modular continua a ter a mesma importância. Porém esta inerente modularidade e acesso ao código é apenas disponível para os desenvolvedores de sistemas [Murray 99].

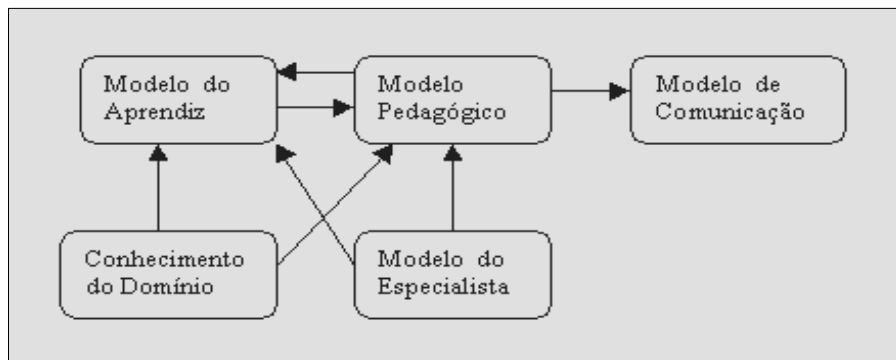


Figura 3.4 Interação entre os componentes de um STI [Beck et alli 96].

3.4.1.5 “Arcabouços” e Sistemas de Autoria

Um “arcabouço” para STIs é um ambiente geral para se construir STIs, enquanto que um sistema de autoria para STIs é um “arcabouço” para STIs juntamente com uma interface do usuário que permite não programadores formalizar e visualizar seu conhecimento [Murray 99].

Murray divide em várias categorias os STIs que utilizam sistemas de autoria:

1) *Seqüenciamento de curriculum e planejamento*: sistemas de autoria nesta categoria, organizam as unidades de instrução numa hierarquia de cursos, lições, apresentações, etc., que são relacionadas por pré-requisitos.

2) *Estratégias de ensino*: estes sistemas tendem a ser similares aos descritos acima, entretanto codificam estratégias de fina granularidade e têm a habilidade para representar múltiplas estratégias de ensino, selecionando qual a mais apropriada para uma dada situação.

3) *Simulação e treinamento*: a diferença destes sistemas em relação aos descritos acima, é que o estudante é colocado numa situação de “aprender fazendo”.

4) *Sistemas especialistas e tutores cognitivos*: estes sistemas incluem modelos cognitivos baseados em regras do domínio do especialista. Aqui os estudantes trabalham na resolução de problemas e sub-problemas dentro de

um objetivo, e recebem um retorno quando seus comportamentos divergem do modelo do especialista.

5) *Hipermídia inteligente*: em muitos dos sistemas baseados na Web hoje, o nível de interatividade e fidelidade disponível para o estudante ainda é baixo. Ao contrário das outras categorias, estes sistemas precisam gerenciar os “hiperlinks” entre unidades de conteúdo diversas, bem como a forma e a seqüência do próprio conteúdo [Murray 99].

O principal objetivo então, em se usar sistemas de autoria nos STIs, é que torna o processo de construção mais fácil e reduz os custos. Uma outra questão importante que vem sendo tratada por pesquisadores é o reuso. Isto pode ser feito em vários níveis:

- *Base de conhecimento* (utilizando partes de um domínio particular);
- *Bibliotecas de ensino* (exercícios e testes desenvolvidos no mesmo domínio);
- *Arquiteturas* (arquiteturas genéricas para uma vasta gama de domínios)

[Murray 99].

Os sistemas de autoria, segundo Murray, podem ser colocados em duas categorias: *sistemas de orientação pedagógica*, e *sistemas orientados à performance*. O primeiro tipo foca a atenção em como seqüenciar e ensinar um conteúdo pré-determinado, e o segundo tipo foca a atenção num ambiente no qual os estudantes possam aprender habilidades, praticando eles mesmos, por meio de interações com o sistema [Murray 99].

3.4.2 Evolução para os Ambientes Interativos de Aprendizagem

3.4.2.1 Definições e Características

As críticas de vários autores em relação aos STIs dos anos 70 e 80, sugere que estes sistemas davam demasiado controle ao tutor, assim como o fato de que eles eram modelados numa abordagem instrucionista, enfatizando a transferência para o estudante, do conhecimento que o tutor possuía [Kearsley 87].

Em [McLoughlin & Oliver 95], Ambiente Interativo de Aprendizagem é definido como um ambiente em que o controle do processo de aprendizagem é feito pelo aprendiz, incluindo aí a escolha da seqüência e quantidade da informação a ser processada.

O conceito de interação é vista, segundo Giardina, como um complexo conceito que envolve controle, adaptabilidade e inteligência. Interatividade não é somente definida em função dos aspectos técnicos do ambiente, mas também em relação às ações e decisões do aprendiz e ajustes às diferenças individuais [Giardina 92].

A questão central passa a ser então: como manter o controle por parte do aprendiz (usuário) e ao mesmo tempo ter um sistema que oriente implicitamente a sua navegação pelo conteúdo de um hipertexto. Entre as alternativas propostas em [Kommers et alii 96] estão:

- 1) Prover o aprendiz de um mecanismo de orientação que mostre o que é central e periférico no nível de informação.
- 2) Especificar a função do hipertexto criando uma série de lições.
- 3) Restringir o número de estratégias de aprendizagem fazendo com que o aprendiz escolha o estilo mais adequado.

3.4.2.2 Classificação

Segundo Wilson, os Ambientes de Aprendizagem são opostos aos ambientes instrucionais, no sentido de promover uma idéia mais flexível de aprendizagem na qual o aprendiz pode desenvolver habilidades mais relevantes para a resolução de problemas [Wilson 96].

Em [Murphy 97] são propostas algumas abordagens para os Ambientes de Aprendizagem:

- *Ambientes de Aprendizagem Baseados em Computador* - ambientes nos quais as atividades propiciadas pelo computador são integradas a outras atividades, havendo uma mútua influência;
- *Ambientes de Aprendizagem com Multimídia Interativa* - ambientes não estáticos, com papéis não fixados para o professor e o aprendiz, com

equilíbrio dinâmico, onde a natureza e a mudança da informação depende da situação, do contexto do aprendizado e das necessidades individuais;

- *Ambientes de Aprendizagem Construtivista* - se baseiam nos seguintes princípios : prover experiência no processo de construção do conhecimento, prover colaboração e múltiplos modos de representação, e encorajar atividades reflexivas;
- *Ambientes de Aprendizagem Construcionista* - se baseiam na conceito de se produzir o máximo de aprendizagem com o mínimo de ensino. Representar frações por meio de jogos, por exemplo, ilustra a idéia de construção mental (construções no mundo como suporte para as da mente).

Wilson classifica em três categorias os ambientes de aprendizagem:

Micromundos, Ambientes de aprendizagem baseados em sala de aula e Ambiente virtual [Wilson 96].

3.4.2.3 Multimídia, Hipertexto e Hipermissão

Multimídia é a integração de várias mídias tais como texto, gráficos, animação, som e vídeo. Schwier distingue três níveis de interação para instrução multimídia: *reativa* (resposta a um dado estímulo), *pró-ativa* (geração de construções únicas), e *mutual* (uso de Inteligência Artificial). Neste último nível o sistema se adapta ao progresso do aprendiz, além de construir e refinar o ambiente para este [Schwier & Misanchuk 93].

Hipertexto é um método não linear e não seqüencial de mostrar texto, gráficos, animação, som e vídeo e que tem as seguintes características: nós de informação, ligações entre os nós, estrutura organizada como uma rede de idéias, controle dinâmico por parte do usuário e acesso de múltiplos usuários. *Hipermissão* é a união de hipertexto e multimídia [Davies 97].

Segundo Padgett, entre as vantagens em se usar hipermissão estão: o favorecimento da memória associativa, a apresentação de uma grande variedade de perspectivas nos tópicos e controle por parte do aprendiz. Alerta contudo os *designers* de hipermissão para que não haja uma sobrecarga de informação e desvios pelo excesso de *links* [Padgett 97].

A estrutura básica de um sistema multimídia é composta de nós e ligações (*links*). Esta forma não linear de dispor a informação é chamada de modular. Um nó pode ser construído de duas maneiras: pelo projetista ou pelo usuário. O primeiro caso é mais apropriado para sistemas nos quais o objetivo e conteúdo não devam ser mudados pelo usuário. O segundo caso é usado quando há entrada de dados pelo usuário, como respostas a questões, pedidos de informações ou comentários [Kommers et alii 96].

Já os *links* podem ter a seguinte classificação:

Link Contextual - junta as várias partes do hipertexto e permite o usuário achar a informação que precisa. O *link* contextual se subdivide em dois tipos:

- *Seqüencial* - cria um caminho pré-determinado onde o usuário tem apenas duas opções: próximo nó ou nó anterior;
- *Relacional* - permite ao usuário buscar a informação ligada por elementos comuns, embora não de uma forma seqüencial. Existem três tipos: *associativo* (procura pela informação relativa a uma palavra, frase ou nó específico), *elaborativo* (tipo de link seqüencial que provê informação mais detalhada sobre um tópico específico), e *hierárquico* (liga a informação de uma maneira progressiva e estruturada).

Link de Suporte - são meta-*links* que provêm ajuda para o usuário acessar e aprender do hipertexto [Kommers et alii 96].

3.5 Sistemas de Educação a Distância via Internet

3.5.1 Definição e Características

Educação a distância é o processo instrucional que não exige a presença física do aluno no local onde é dada a instrução. Os termos educação a distância e aprendizagem a distância são freqüentemente intercambiáveis. Pode-se dizer que a aprendizagem a distância é o resultado da educação a distância [Steiner 95].

3.5.2 Aspectos de Ensino e Aprendizagem

No processo de ensino e aprendizagem, pode-se distinguir quatro elementos básicos: *O professor, o conteúdo, o aluno e o contexto*. A partir destes quatro elementos, vários modelos podem ser considerados de acordo com o nível de interação entre as partes. O mais freqüente é o modelo em que um dos elementos se encontra num nível mais baixo quanto à integração e tecnologias de informação em relação aos outros elementos. Tendendo para uma máxima interação “online” entre as partes encontramos uma situação em que o *professor* passa de um simples transmissor de um conhecimento pré-organizado para uma posição de orientador ou facilitador, o *conteúdo* curricular passa de pré-organizado a uma posição de ser construído pelos alunos, os *alunos* têm livre acesso aos recursos “online”, e o *contexto* suporta o uso e expansão destes recursos [Bracewell et alii 98].

3.6 Conclusão

Neste capítulo vimos que os SBC abrangem várias sub-áreas. Vimos também como o conhecimento de um domínio pode ser construído de forma distribuída num sistema multiagentes.

O atual estado da arte dos Sistemas Tutores Inteligentes incorpora a mudança de enfoque em relação ao processo de aprendizagem, dando mais autonomia ao aprendiz, ao mesmo tempo que as novas tecnologias de ensino a distância têm um leque de possibilidades que incluem o acesso individual a cursos na *Web*.

Nos próximos capítulos vamos expor nosso sistema, desde a sua concepção aos aspectos de implementação.

Capítulo 4

O Sistema SHART-Web e o MATHEMA

4.1 Introdução

Neste capítulo abordaremos o sistema SHART-Web a partir do contexto do ambiente MATHEMA. Inicialmente daremos uma visão geral do ambiente MATHEMA, a arquitetura interna do agente MATHEMA, e como se procedem as interações entre estes agentes. Em seguida, mostraremos o domínio da Harmonia tradicional do ponto de vista do SHART-Web instanciado do MATHEMA, e a partir daí, a identificação dos agentes que serão utilizados pelo sistema.

4.2 O Ambiente MATHEMA

O MATHEMA é um modelo conceitual de ambiente computacional de ensino-aprendizagem baseado em uma abordagem multiagentes. O processo cooperativo entre aprendiz-tutor e as interações daí decorrentes têm como princípio básico a atividade de resolução de problemas. A sistematização e representação do conhecimento é feita de duas formas: visão externa e visão interna.

4.2.1 Visão Externa do Domínio

Na visão externa, o domínio (D) é abordado num esquema de visões em dimensão que disciplinam o particionamento de D, em vários subdomínios onde o conhecimento é visto em múltiplas dimensões na seguinte forma:

Contexto - é uma determinada abordagem do domínio.

Profundidade - estabelece os graus de dificuldades para os problemas num determinado *contexto*.

Lateralidades - conjunto de conhecimento de suporte definido para cada par $\langle \text{contexto}, \text{profundidade} \rangle$.

Sendo assim, um *domínio* D qualquer no MATHEMA, passa a ser abordado de acordo com o esquema de dimensão indicada na Figura 4.1. Neste, escolhem-

se primeiramente os contextos identificados em D . Em seguida, para cada contexto, identificam-se profundidades. Com base nessa visão, particiona-se D em subdomínios ($D = d_{11}, d_{12}, \dots, d_{nm}$), onde d_{ij} é definido a partir de um par $\langle C_i, P_{ij} \rangle$, onde C_i é um contexto particular e P_{ij} uma profundidade associada a este contexto. E, a cada par $\langle C_i, P_{ij} \rangle$, é associado um conjunto de lateralidades: $\langle C_i, P_{ij} \rangle \rightarrow \{L_{ij1}, L_{ij2}, \dots, L_{ijt}\}$ [Costa 97].

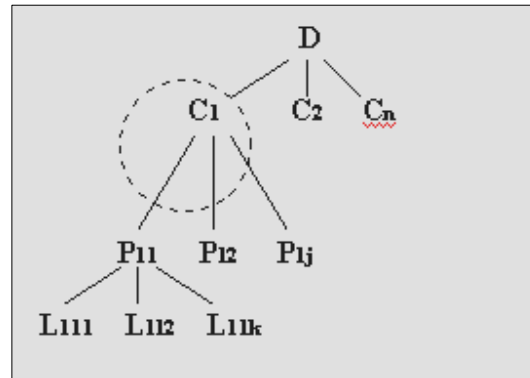


Figura 4.1 Visão externa do domínio [Costa 97].

4.2.2 Visão Interna do Domínio

Na visão interna, a cada subdomínio é associado um *currículo*, que contém um conjunto de *unidades pedagógicas*: $Curriculum = \{up_1, up_2, \dots, up_n\}$, dispostas segundo uma ordem, normalmente definida em função do nível de dificuldade. A cada *unidade pedagógica* associa-se um conjunto de problemas. E, para cada um destes problemas existem unidades de conhecimento (suporte), que auxiliam na resolução dos problemas.

4.2.3 Arquitetura Geral do MATHEMA

Vê-se na Figura 4.2 a arquitetura geral do ambiente MATHEMA.

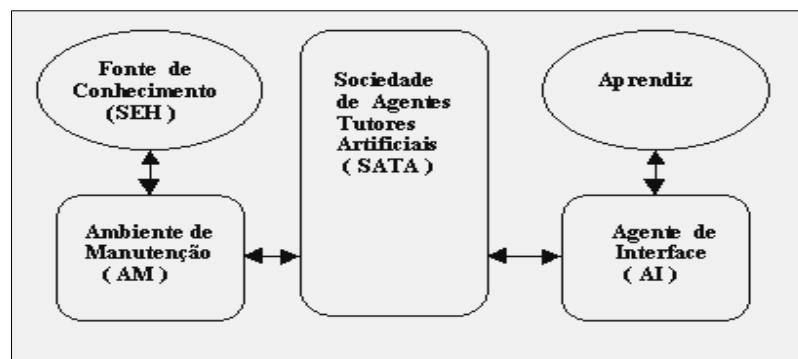


Figura 4.2 Arquitetura geral do ambiente MATHEMA [Costa 97].

SATA - Conjunto de agentes que cooperam entre si com o intuito de promover a aprendizagem de um aprendiz. Cada agente na SATA está associado a um subdomínio de D.

Aprendiz Humano - é o aluno que interage com o sistema com o objetivo de aprender um determinado domínio.

Agente de Interface - é o agente responsável pela ligação entre o aprendiz humano e a SATA. Logo na primeira sessão, ajuda o aprendiz a escolher o agente supervisor (encarregado de acompanhar mais de perto as atividades do aluno).

SEH (Sociedade de Especialistas Humanos) - é a fonte externa de conhecimento, integrada à SATA. Responde pela criação e manutenção da SATA (entrada de novos agentes, exclusão de agentes, e alterações no conhecimento destes).

Ambiente de Manutenção - faz o elo entre a SEH e a SATA. Provê meios que facilitam a aquisição de conhecimento pela SATA.

4.2.4 Arquitetura do Agente MATHEMA

A arquitetura do agente MATHEMA (Figura 4.3) é composta de três sistemas:

1. **Sistema Tutor** - módulo das atividades tutoriais do sub-domínio ao qual o agente está associado.
2. **Sistema Social** - módulo responsável pelas atividades cooperativas entre os agentes. Neste módulo são guardadas as bases de conhecimento sobre as habilidades dos outros agentes e o conhecimento sobre si mesmo.
3. **Sistema de Distribuição** - módulo responsável pelo tráfego de mensagens enviadas e recebidas pelo agente.

Internamente o **Sistema Tutor** é composto de três módulos:

- **Bases de conhecimento** - este módulo inclui o conhecimento da estrutura pedagógica, do domínio e do aprendiz. Tem uma relação estreita com o

módulo *raciocinadores* e é constituído de três bases: base do especialista, base do tutor e modelo do aprendiz.

- *Raciocinadores* - este módulo provê os mecanismos de raciocínio do sistema e inclui: resolução de problemas, diagnóstico cognitivo, explicação e instrução. É composto também de três sub-módulos:
 1. *Especialista* - desempenha papéis como resolução de problemas, explicações, diagnóstico cognitivo e remediação.
 2. *Tutor* - é responsável pela seleção de recursos pedagógicos escolhidos com base no curriculum.
 3. *Modelagem do Aprendiz* - este sub-módulo é encarregado de construir o modelo do aprendiz.
- *Mediador* - este módulo exerce a função de controle geral da execução das funções pedagógicas no sistema tutor.

O *Sistema Social* é composto dos seguintes módulos:

- *Autoconhecimento (AC)* - neste módulo estão representadas as próprias habilidades do agente.
- *Conhecimento Social (CS)* - neste módulo está o conhecimento que o agente possui sobre as habilidades de outros agentes da SATA.
- *Alocação* - módulo que se encarrega da seleção dos agentes aptos a resolver uma determinada tarefa submetida pelo *Sistema Tutor*.
- *Coordenação* - é o módulo que interpreta a estrutura de tarefas pedagógicas, determinando a cada momento, qual a tarefa a ser repassada para o próximo módulo (módulo *Cooperação*). O módulo de coordenação sincroniza as execuções das subtarefas.
- *Cooperação* - é o módulo que cumpre a execução de uma tarefa pedagógica. Uma instância de cooperação esgota todas as possibilidades de cooperação e dispara os protocolos necessários para isto. No caso de sucesso, o agente tutor que realiza a tarefa envia o seu resultado até chegar na instância de cooperação. No caso de insucesso, a SEH e o aprendiz são notificados sobre a situação ocorrida.
- *Manutenção Social* - é o módulo responsável pela entrada, saída e atualização de agentes, que são feitos por meio de protocolos de manutenção. Para cada situação de manutenção (entrada, saída e atualização, existe um algoritmo específico).

- *Protocolos* - é o módulo responsável pela criação e ativação de um diálogo com base nos protocolos disponíveis. Um protocolo define uma interação, e uma instância de diálogo é responsável pela execução desta interação.

O *Sistema de Distribuição* é composto dos seguintes módulos:

- *Controle* - é o módulo intermediário entre o sistema social e o sistema de distribuição. Suas funções são: encaminhar as mensagens recebidas para serem tratadas pela instância de diálogo apropriada., verificar a consistência das mensagens recebidas e eliminar as mensagens atrasadas.
- *Comunicação* - este módulo implementa um modelo de comunicação que se encarrega das atividades de envio e recebimento de mensagens através do ambiente de comunicação, comum a todos os agentes da SATA.

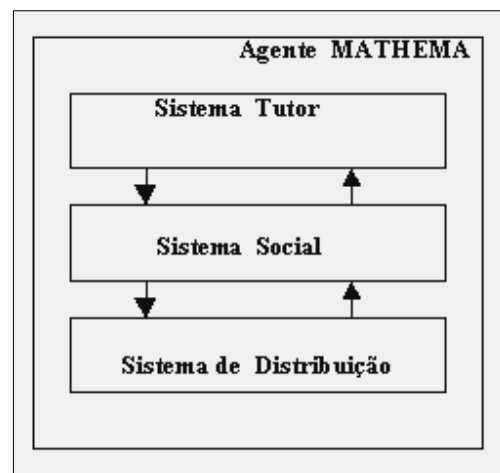


Figura 4.3 Arquitetura do agente Mathema [Costa 97].

4.2.5 Interações entre Agentes

Tendo sido já exposta a arquitetura de um agente MATHEMA, veremos agora como se processa as interações entre estes agentes. A sua organização é baseada num modelo de complementaridade em que os agentes para atingirem seus objetivos, podem pedir ajuda uns aos outros, e ao mesmo tempo há uma dinâmica no processo de formação de uma coalizão entre agentes, algo que não é pré-estabelecido, mas ocorre em função de uma demanda de acordo com o tipo de interação.

O controle das atividades cooperativas, por sua vez, passa a ser exercido pelo agente que pede cooperação. Ou seja, não está a cargo apenas de um agente específico, mas em princípio todos podem exercer este controle.

Existem três situações possíveis de acontecer quando um determinado agente está diante de uma tarefa para resolvê-la:

- 1) Tem o conhecimento necessário para resolver uma tarefa. Neste caso não precisa interagir com outros agentes.
- 2) Tem ignorância parcial para resolver uma tarefa. Neste caso, ele precisa da cooperação de outros agentes para concluí-la.
- 3) Não tem conhecimento algum para resolver uma tarefa, nem conhece outro agente que possa fazê-lo. Neste caso anuncia-se a tarefa para todos os agentes na sociedade.

O modelo de cooperação entre os agentes no ambiente MATHEMA é híbrido, permitindo tanto o protocolo do tipo *mestre-escravo* quanto o de *licitação*. O tipo *mestre-escravo* quando se tem a situação de existir apenas um agente para resolver a tarefa. Já o tipo *licitação* ocorre quando existe mais de um agente que se habilita a resolver a tarefa requisitada.

Finalmente, para que se opere adequadamente o processo de interação, tanto em nível interno do agente (entre as camadas do sistema social e tutor), quanto em nível externo (entre as camadas sociais dos agentes), é preciso uma linguagem que padronize o vocabulário utilizado.

4.2.6 Interações Aprendiz-Agente

O processo de interação entre o aprendiz e um agente (Figura 4.4) pode ser visto como uma troca de mensagens representadas por X (para o agente) e Y (para o aprendiz). Por parte do agente pode haver os seguintes tipos de mensagens:

1. Problema = apresenta um problema (exercício) ao aprendiz.
2. Diagnóstico = mostra o passo a passo na resolução de um problema.
3. Dica = apresenta uma dica ao aprendiz.

4. Instrução = apresenta uma Instrução (aula) em resposta a um pedido do aprendiz.

Por parte do aprendiz pode haver os seguintes tipos de mensagens:

1. Solução = resolução de um problema (exercício) proposto pelo agente.
2. Consulta = pedido para esclarecer uma determinada dúvida.
3. Dica = pedido de uma dica sobre um determinado tópico.
4. Instrução = pedido de uma Instrução (aula) sobre um determinado tópico.

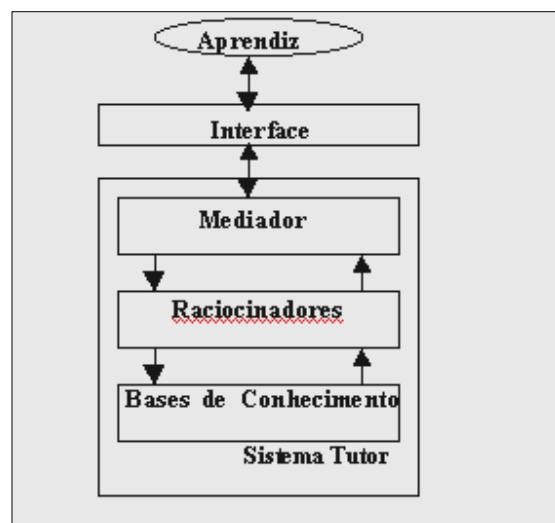


Figura 4.4 A comunicação aprendiz-agente [Costa 97].

4.3 O Sistema SHART-Web e o MATHEMA

4.3.1 Aspectos do Domínio

O domínio da Harmonia musical engloba a Harmonia tradicional (normativa) e a Harmonia funcional. Na Harmonia tradicional, que é o objeto de nosso trabalho, diversos conceitos preliminares, que são pertinentes à teoria musical, formam a base para o entendimento dos demais conceitos e do conjunto de regras¹ que a caracterizam. Estas regras são encontradas na literatura da Harmonia tradicional [Piston 83], [Benjamin et alli 98], [Caron 99].

¹ Este conjunto de regras é o resultado de um processo histórico empírico, no qual estas regras foram catalogadas por diversos teóricos, abrangendo aproximadamente dois séculos: início do século XVIII ao final do século XIX, período conhecido como “Common Practice”.

A Harmonia funcional é baseada na “teoria funcional” de Hugo Riemann, segundo a qual, as funções de todos os acordes de uma determinada tonalidade são reduzidas a apenas três funções principais: tônica, subdominante e dominante [Brisolla 79].

Com relação ao aspecto pedagógico, a Harmonia tradicional tem os seguintes objetivos:

- Capacitar o aluno(a) a analisar uma obra do ponto de vista de sua estrutura harmônica (tipos de acordes, progressões harmônicas, cadências), e relacionar esta estrutura ao período histórico correspondente;
- Tornar o aluno(a) apto(a) a fazer harmonizações de exercícios a partir de um baixo dado ou de um canto dado;
- Ser uma base para o ensino de Composição musical.

4.3.2 Visão Externa do Domínio

Os conceitos em Harmonia tradicional seguem um nível crescente de complexidade em que é visível a relação de dependência entre eles. Por exemplo, o conceito de *acorde* depende de *intervalo*, que por sua vez depende de *nota*. Assim, não é difícil ser aplicada a idéia de profundidade na estrutura do domínio. Na Figura 4.5, temos a organização da Harmonia musical segundo os contextos da Harmonia tradicional (C1) e da Harmonia funcional (C2). O contexto da Harmonia tradicional desdobra-se nas seguintes profundidades:

P₁₁: Vozes Nível 1 (Conceitos Básicos de Vozes)

P₁₂: Acordes Nível 1 (Posição dos Acordes)

P₁₃: Intervalos Nível 1 (Intervalos Melódicos Proibidos e Permitidos)

P₁₄: Intervalos Nível 2 (Intervalos Harmônicos Proibidos e Permitidos)

P₁₅: Vozes Nível 2 (Condução das Vozes)

P₁₆: Acordes Nível 2 (Acordes Diatônicos no Estado Fundamental)

P₁₇: Acordes Nível 3 (Dobramento e Supressão nos Acordes Diatônicos)

P₁₈: Tonalidades (Função Tonal dos Graus)

P₁₉: Fraseologia e Cadências

P₂₀: Procedimentos para o Baixo dado

A partir destas profundidades formam-se os pares (10) e suas respectivas lateralidades (Tabela 4.1).

Pares	Lateralidades				
	Notas	Intervalos	Escalas	Acordes	Cifras
<C1, P11>	L111	L112			L113
<C1, P12>	L121	L122		L123	L124
<C1, P13>	L131	L132			
<C1, P14>	L141	L142			L143
<C1, P15>	L151	L152		L153	L154
<C1, P16>	L161	L162	L163		L164
<C1, P17>	L171	L172	L173	L174	L175
<C1, P18>	L181	L182	L183	L184	L185
<C1, P19>	L191	L192	L193	L194	L195
<C1, P20>	L201	L202	L203	L204	L205

Tabela 4.1 Pares e lateralidades

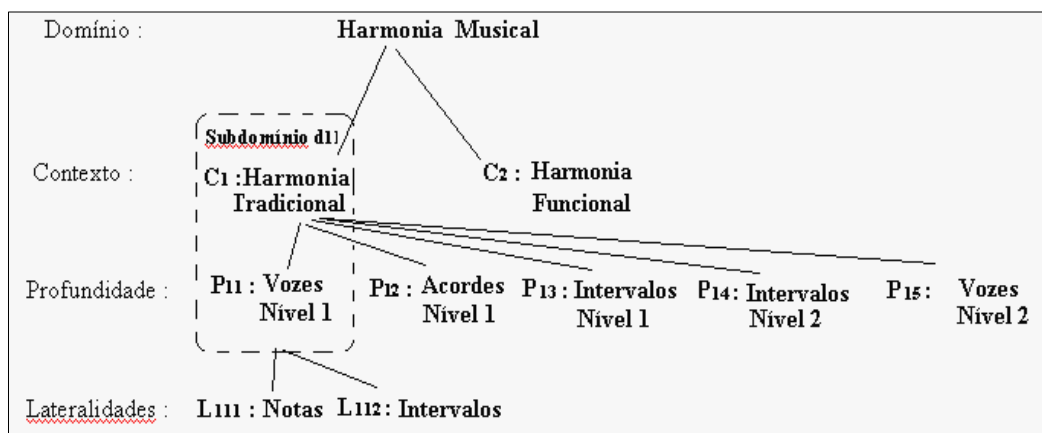


Figura 4.5 Visão externa do domínio da Harmonia tradicional.

4.3.3 Visão Interna do Domínio

Aqui, cada subdomínio é composto de um conjunto de unidades pedagógicas (Up), que contém unidades de conhecimento (Uc), tudo isto formando o *currículum*. Três planos são colocados:

1. *Plano pedagógico* - as unidades pedagógicas são colocadas seguindo um nível crescente de dificuldade.

2. *Plano de problemas* - cada unidade pedagógica tem um conjunto de problemas seguindo também uma ordem.
3. *Plano de suporte* - conhecimento necessário à resolução dos problemas que incluem: contra-exemplos, dicas, problemas resolvidos, etc.

A seguir a tabela com os 10 subdomínios (*Curricula*), e as tabelas com as unidades pedagógicas e unidades de conhecimento de *Vozes Nível 1*:

<i>Curricula</i>	
Identificador	Nome do subdomínio
1	<i>Vozes Nível 1 (Conceitos Básicos)</i>
2	<i>Acordes Nível 1 (Posição dos Acordes)</i>
3	<i>Intervalos Nível 1 (Intervalos Melódicos Proibidos e Permitidos)</i>
4	<i>Intervalos Nível 2 (Intervalos Harmônicos Proibidos e Permitidos)</i>
5	<i>Vozes Nível 2 (Condução das Vozes)</i>
6	<i>Acordes Nível 2 (Acordes Diatônicos no Estado Fundamental)</i>
7	<i>Acordes Nível 3 (Dobramento e Supressão nos Acordes Diatônicos)</i>
8	<i>Tonalidades (Função Tonal dos Graus)</i>
9	<i>Fraseologia e Cadências</i>
10	<i>Procedimentos para o Baixo Dado</i>

Tabela 4.2 Curricula.

Vozes Nível 1 (Conceitos Básicos):

<i>Unidades pedagógicas de Vozes Nível 1:</i>	
Identificador	Nome da unidade pedagógica
1	<i>Extensão das Vozes</i>
2	<i>Distância entre as Vozes</i>
3	<i>Distribuição das Vozes</i>
4	<i>Tipos de Movimento das Vozes</i>

Tabela 4.3 Unidades pedagógicas de Vozes Nível 1.

<i>Unidades de conhecimento de Extensão das Vozes:</i>	
Identificador	Nome da unidade de conhecimento
1	<i>Soprano</i>
2	<i>Contralto</i>
3	<i>Tenor</i>
4	<i>Baixo</i>

Tabela 4.4 Extensão das Vozes.

<i>Unidades de conhecimento de Distância entre as Vozes:</i>	
Identificador	Nome da unidade de conhecimento
1	<i>Distância entre Soprano e Contralto</i>
2	<i>Distância entre Contralto e Tenor</i>
3	<i>Distância entre Tenor e Baixo</i>

Tabela 4.5 Distância entre as Vozes.

<i>Unidades de conhecimento de Distribuição das Vozes:</i>	
Identificador	Nome da unidade de conhecimento
1	<i>Posição das hastes das figuras e as Vozes</i>
2	<i>Distribuição das Vozes nos dois pentagramas</i>

Tabela 4.6 Distribuição das Vozes.

<i>Unidades de conhecimento de Tipos de Movimento das Vozes:</i>	
Identificador	Nome da unidade de conhecimento
1	<i>Movimento Contrário</i>
2	<i>Movimento Oblíquo</i>
3	<i>Movimento Similar</i>
4	<i>Movimento Paralelo</i>

Tabela 4.7 Tipos de Movimento das Vozes.

A partir desta organização do *Curriculum*, podemos estabelecer uma ordem ou estrutura, que quando aplicada, pode gerar uma série de seqüências pedagógicas. Podemos ver na Figura 4.6 um exemplo de uma estrutura pedagógica no subdomínio Vozes (Nível 1). Iniciando-se no curriculum pela unidade pedagógica Extensão das Vozes (Up1), esta é pré-requisito para as demais unidades pedagógicas. Distância entre Vozes (Up2) e Distribuição das Vozes (Up3) estão no mesmo nível de dificuldade e são pré-requisitos para Tipos de movimento (Up4).

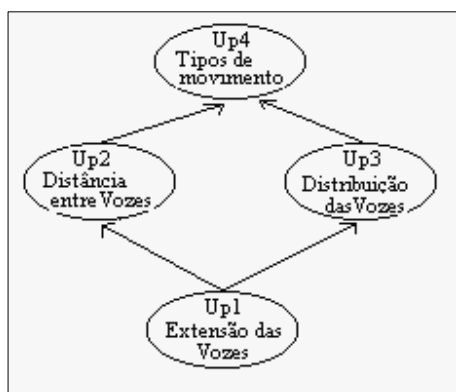


Figura 4.6 Uma estrutura pedagógica.

Seqüências pedagógicas geradas a partir desta estrutura:

Seqüência 1: { Up₁, Up₂, Up₃, Up₄ }

Seqüência 2: { Up₁, Up₃, Up₂, Up₄ }

4.3.4 Identificação dos Agentes

A partir da definição dos subdomínios e suas lateralidades, são identificados os agentes [Costa 97].

Conforme vimos anteriormente (Figura 4.5), o particionamento do domínio ficou estabelecido da seguinte forma:

d₁₁: Vozes Nível 1 (Conceitos Básicos de Vozes)

d₁₂: Acordes Nível 1 (Posição dos Acordes)

d₁₃: Intervalos Nível 1 (Intervalos Melódicos Proibidos e Permitidos)

d₁₄: Intervalos Nível 2 (Intervalos Harmônicos Proibidos e Permitidos)

d₁₅: Vozes Nível 2 (Condução das Vozes)

d₁₆: Acordes Nível 2 (Acordes Diatônicos no Estado Fundamental)

d₁₇: Acordes Nível 3 (Dobramento e Supressão nos Acordes Diatônicos)

d₁₈: Tonalidades (Função Tonal dos Graus)

d₁₉: Fraseologia e Cadências

d₂₀: Procedimentos para o Baixo dado

Todos os agentes acima resolvem problemas (validam respostas) e aplicam estratégias pedagógicas referentes aos seus respectivos subdomínios. Estes subdomínios vão gerar os seguintes agentes:

AT₁₁ (Agente Vozes Nível 1):

Responsável pela identificação dos registros de cada voz (Soprano, Contralto, Tenor e Baixo), verificando se as notas das vozes estão dentro da extensão, se a distância entre as vozes e a distribuição estão corretas e define também os tipos de movimento das vozes.

AT₁₂ (Agente Acordes Nível 1):

Verifica a posição do acorde (aberta ou fechada) e o estado dos acordes (se está na posição fundamental ou invertido)

AT13 (Agente Intervalos Nível 1):

Verifica se ocorre intervalos melódicos proibidos e permitidos nas vozes. Tem a colaboração do agente lateral Intervalos.

AT14 (Agente Intervalos Nível 2):

Verifica se ocorre intervalos harmônicos, que são classificados como proibidos ou permitidos.

AT15 (Agente Vozes Nível 2):

Este agente tem o conhecimento sobre a condução das vozes, se dois acordes têm ou não notas em comum e os procedimentos para a resolução do VII grau (sensível).

AT16 (Agente Tonalidades):

Este agente analisa as funções tonais dos graus. Para realizar suas tarefas precisa da colaboração dos agentes laterais Tonalidades e Cifras.

AT17 (Agente Acordes Nível 2):

Este agente tem o conhecimento sobre os acordes diatônicos e a relação entre os modos das escalas e as tríades correspondentes.

AT18 (Agente Acordes Nível 3):

Este agente tem o conhecimento sobre as notas que podem ser dobradas (duplicadas) ou suprimidas.

AT19 (Agente Fraseologia e Cadências):

Identifica uma frase musical a partir de um intervalo melódico, quais os pontos (notas) da frase onde serão colocados os graus da cadência, e os diversos tipos de cadência.

Além destes, temos os seguintes agentes relativos às lateralidades:

Agente Notas:

Verifica se um objeto é uma nota (com seus respectivos atributos: oitavas, nomes, alterações e durações), ou pausa. Converte também a nota na representação MIDI equivalente.

Agente Intervalos:

A partir de duas notas de entrada, classifica o número de semitons do intervalo, se um intervalo é melódico ou harmônico, se for intervalo harmônico, identifica também se são uníssonos paralelos, quintas paralelas, oitavas paralelas, quintas diretas ou oitavas diretas.

Agente Escalas:

Reconhece e forma quatro tipos de escala (uma maior e três menores).

Agente Acordes:

Define o que seja acorde e classifica os tipos de tríade (maiores, menores, aumentadas e diminutas). Reconhece e forma um acorde. Verifica se um acorde está em posição fundamental, 1^a, 2^a ou 3^a inversão. Define e verifica qual o tipo de dobramento que um acorde tem : fundamental, terça ou quinta.

Agente Tonalidades:

Classifica todas as tonalidades maiores e menores, associadas às respectivas armaduras de clave. Identifica o tipo de encadeamento entre dois acordes. Reconhece em qual tonalidade está uma escala ou um encadeamento de acordes. Define os nomes dos graus de uma tonalidade e descreve as funções dos graus : tônica, subdominante e dominante.

Agente Cifras:

A partir dos acordes formados gera as cifras equivalentes, e a partir das cifras gera os acordes correspondentes.

4.4 Conclusão

Neste capítulo tratamos da concepção do sistema SHART-Web a partir do ambiente MATHEMA. Vimos a relação entre os subdomínios e a identificação dos agentes e que um agente pode ser criado a partir dos subdomínios e/ou de suas lateralidades.

Abordaremos nos próximos dois capítulos a arquitetura e o funcionamento do sistema e sua modelagem.

Capítulo 5

O Sistema SHART-Web: Requisitos, Arquitetura e Funcionamento

5.1 Introdução

Neste capítulo são apresentados os requisitos, a arquitetura, seus componentes e o funcionamento do sistema SHART-Web: um sistema de apoio à aprendizagem baseado num processo que envolve interações cooperativas entre agentes.

5.2 Requisitos

5.2.1 Características Gerais

Trata-se de um sistema tutor hipermídia voltado para ensino de harmonia tradicional, prestando uma assistência constante ao aprendiz com vistas à sua aprendizagem. O sistema fornece um modo de navegação guiado por estratégias pedagógicas, servindo de orientação ao aprendiz, exibindo diversos recursos tais como: conceitos, exemplos, exercícios, dicas, entre outros. Esta orientação é feita a partir da análise cognitiva do aprendiz, criando um perfil do mesmo, com base nas informações de suas interações com o sistema, e que servirá para direcionar as futuras interações.

5.2.2 Sobre os Usuários do Sistema

Os usuários do sistema podem ser tanto os que já tenham algum conhecimento prévio de teoria musical, como também os usuários que queiram navegar livremente sem qualquer conhecimento de música. No caso de usuário optar pela navegação com guia, é feito um questionário no qual ele(ela) diz em que nível se encontra, ou mesmo pode optar por um teste. O usuário pode ser remetido a *links* externos se não tem nenhuma noção de teoria musical.

5.3 Arquitetura e Componentes

O SHART-Web é um sistema de apoio à aprendizagem baseado num processo que envolve interações cooperativas entre agentes. A arquitetura do SHART-Web é composta do Sistema Tutor Inteligente Multiagentes (STIMA), e dos bancos de dados de Perfil do aprendiz e Administração do sistema. Integram o STIMA os agentes da SATA, o documento hipermídia e as bases de conhecimento. Estes elementos compõem a visão geral do sistema.

Na Figura 5.1, vê-se a arquitetura do SHART-Web. Nesta figura observam-se sete componentes: aprendiz, professor, suas respectivas interfaces, a SEH (Sociedade de Especialistas Humanos) e sua interface, e o sistema.

As interações¹ entre aprendiz e professor (representadas pela seta pontilhada), incluem e-mail e *chat*. Aprendiz e professor interagem com o sistema através de suas respectivas interfaces.

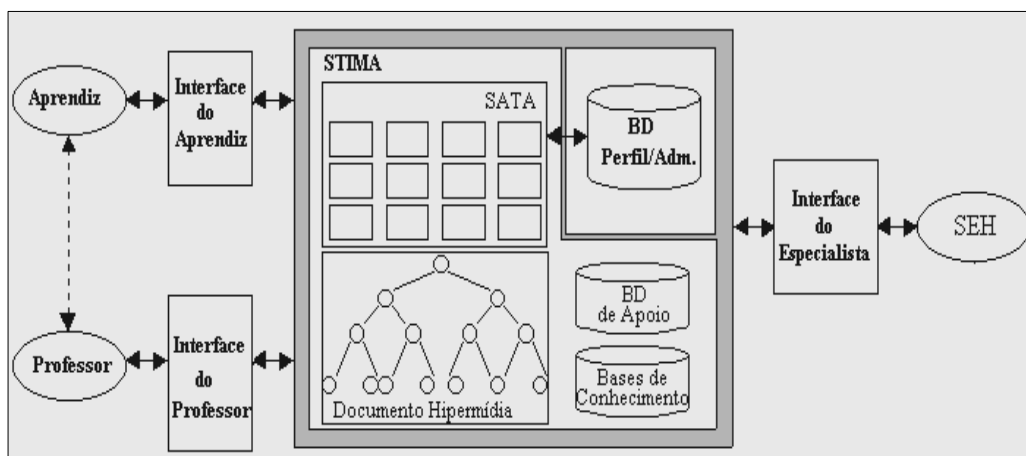


Figura 5.1 Arquitetura do sistema SHART-Web.

A seguir as entidades humanas e os componentes da arquitetura do SHART-Web:

- **Aprendiz:**

O aprendiz ou usuário do sistema tem que se cadastrar inicialmente. Além da sua comunicação com o sistema via interface, pode ter uma comunicação direta com o professor.

¹ A ferramenta que torna possível estas interações chama-se GRAW e está sendo implementada dentro do Projeto MathNet.

- **Professor:**

O professor é o responsável pelo papel de auxiliar o aprendiz tendo uma comunicação direta com ele.

- **Interface (Aprendiz):**

A interface permite a comunicação entre o aprendiz e o sistema. A visualização é feita geralmente através de um navegador (*browser*) *web*. É por meio da interface que o aprendiz manipula os objetos gráficos musicais, são mostrados os arquivos de imagem com os problemas e são enviadas as respostas para o sistema.

- **Interface (Professor):**

Da mesma forma que a interface do aprendiz, toda a comunicação entre professor e sistema é feita por esta interface.

- **SEH (Sociedade de Especialistas Humanos):**

Encarregada de dar o suporte de conhecimento do domínio. Qualquer modificação na base de conhecimento é feita via SEH.

- **Interface do Especialista:**

Viabiliza a comunicação entre especialistas da SEH e da STIMA.

- **STIMA:**

O STIMA é composto da SATA, do documento hipermídia, do banco de dados de apoio e das bases de conhecimento.

5.3.1 Sistema Tutor Inteligente Multiagentes (STIMA)

O STIMA tem os seguintes componentes:

1) **SATA (Sociedade de Agentes Tutores Artificiais):**

Cada agente da SATA tem associados a si mesmo, uma parte do documento hipermídia, das bases de conhecimento e do banco de dados de apoio.

Os agentes da SATA buscam na cooperação a melhor forma de resolver os problemas. Conforme vimos no Capítulo 4, o agente MATHEMA (modelo de agente da SATA), é composto de três sistemas: sistema tutor, sistema social e sistema de distribuição. Abordaremos aqui mais detalhadamente os componentes do sistema tutor que são:

- **Módulo do Especialista:**

Neste módulo está o conhecimento que o agente tem sobre o assunto a ser ensinado. Faz parte deste módulo o *Motor de Inferência*. O mecanismo de inferência usado é o encadeamento regressivo (*backward chaining*).

▪ **Módulo Tutor:**

Neste módulo estão contidas as estratégias pedagógicas e o curriculum relacionado a cada agente. Este módulo tem a função de definir quais as estratégias pedagógicas a serem usadas, por meio da análise do perfil do aluno, armazenado no módulo do modelo do aprendiz. Como há uma variação grande na forma como os alunos aprendem, é preciso haver uma adaptação do sistema na escolha de estratégias mais adequadas. As estratégias pedagógicas usadas aqui são na verdade um conjunto de táticas, escolhidas de acordo com o contexto da interação e do histórico do aprendiz. Os recursos que compõem uma estratégia pedagógica são: *definições*, *exemplos* e *exercícios*.

Tendo-se uma quantidade razoável destes elementos, pode-se combiná-los adequadamente, resultando em diferentes níveis de estratégias. A escolha da próxima estratégia é feita de acordo com o nível que o aprendiz se encontra desde a última estratégia. Este nível é medido pelo grau de pontuação que o aprendiz obtém a partir da resposta a um ou mais exercícios. Na tabela 5.1 vê-se um conjunto de 18 estratégias pedagógicas. Estas estratégias podem ter um ou mais exercícios sobre um determinado assunto, sendo algumas delas compostas apenas de exercícios.

Estratégias pedagógicas								
Subdomínio: Vozes Nível 1 (Conceitos Básicos)								
Número da estratégia	Recursos da estratégia	Referência de arquivo (.gif)	Número da estratégia	Recursos da estratégia	Referência de arquivo (.gif)	Número da estratégia	Recursos da estratégia	Referência de arquivo (.gif)
1	Definição Exemplo Exercício	Def 1.1.2 Ep 1.1.2 Ex 1.1.2	7	Definição Exemplo Exercício	Def1.4.4.2 Ep 1.4.4.2 Ex 1.4.4.2	13	Definição Exemplo Exercício	Def1.4.3.1 Ep 1.4.3.1 Ex 1.4.3.1
2	Definição Exemplo Exercício	Def 1.2.2 Ep 1.2.2 Ex 1.2.2	8	Definição Exemplo Exercício	Def 1.1.1 Ep 1.1.1 Ex 1.1.1	14	Definição Exemplo Exercício	Def1.4.4.1 Ep 1.4.4.1 Ex 1.4.4.1
3	Definição Exemplo Exercício	Def 1.3.2 Ep 1.3.2 Ex 1.3.2	9	Definição Exemplo Exercício	Def 1.2.1 Ep 1.2.1 Ex 1.2.1	15	Definição Exemplo Exercício	Def 1.1.3 Ep 1.1.3 Ex 1.1.3 Ex 1.1.4
4	Definição Exemplo Exercício	Def1.4.1.2 Ep 1.4.1.2 Ex 1.4.1.2	10	Definição Exemplo Exercício	Def 1.3.1 Ep 1.3.1.1 Ex 1.3.1.1	16	Definição Exemplo Exercício	Def 1.2.2 Ep 1.2.2 Ex 1.2.3 Ex 1.2.4
5	Definição Exemplo Exercício	Def1.4.2.2 Ep 1.4.2.2 Ex 1.4.2.2	11	Definição Exemplo Exercício	Def1.4.1.1 Ep 1.4.1.1 Ex 1.4.1.1	17	Definição Exemplo Exercício	Def 1.3.2 Ep 1.3.2.2 Ex 1.3.2
6	Definição Exemplo Exercício	Def1.4.3.2 Ep 1.4.3.2 Ex 1.4.3.2	12	Definição Exemplo Exercício	Def1.4.2.1 Ep 1.4.2.1 Ex 1.4.2.1	18	Exercício Exercício Exercício Exercício	Ex 1.4.5 Ex 1.4.6 Ex 1.4.7 Ex 1.4.8

Tabela 5.1 Estratégias pedagógicas.

Na Tabela 5.2 vê-se algumas das regras para manipular estas estratégias pedagógicas (da Tabela 5.1). Na regra 1 por exemplo, se o aluno tiver uma pontuação maior ou igual a 8 na estratégia 1 então é apresentada a estratégia 2. Na regra 2 se a pontuação estiver entre 5 e 8 então é apresentada a estratégia 8.

Regras para as estratégias pedagógicas
1. Se estratégia 1 nível ≥ 8 então faça estratégia 2
2. Se estratégia 1 nível < 8 e > 5 então faça estratégia 8
3. Se estratégia 2 nível ≥ 8 então faça estratégia 3
4. Se estratégia 2 nível < 8 e > 5 então faça estratégia 9
5. Se estratégia 3 nível ≥ 8 então faça estratégia 4
6. Se estratégia 3 nível < 8 e > 5 então faça estratégia 10
7. Se estratégia 4 nível ≥ 8 então faça estratégia 5
8. Se estratégia 4 nível < 8 e > 5 então faça estratégia 11
9. Se estratégia 5 nível ≥ 8 então faça estratégia 6
10. Se estratégia 5 nível < 8 e > 5 então faça estratégia 12

Tabela 5.2 Regras para as estratégias pedagógicas.

▪ Módulo da Modelagem do Aprendiz:

Este módulo guarda as interações do aprendiz com o sistema, mantendo informações sobre seu desempenho e histórico das estratégias e recursos já utilizados no aprendizado de determinado tópico. Este módulo é responsável também pelo processamento dos dados armazenados com o objetivo de se ter uma inspeção das ações do aprendiz. Este modo inspecionado é um diagnóstico de erros casuais (pequenos erros cometidos pelo aprendiz e que pode-se dar uma segunda chance) nas resoluções de exercícios. Se em qualquer estratégia o aluno(a) não tiver um nível satisfatório, por exemplo, estará à sua disposição um exercício do mesmo nível para ser novamente respondido.

2) Documento Hipermissão:

Composto das páginas em html, tem subjacente a estrutura de uma árvore. Os vários nós que o compõem são objetos dispostos numa relação hierárquica. Na forma de navegação não guiada, o aprendiz tem total controle de decisão.

Na navegação guiada o sistema determina qual o próximo recurso pedagógico a ser exibido. Na Figura 5.2 temos uma visão do documento hipermissão.

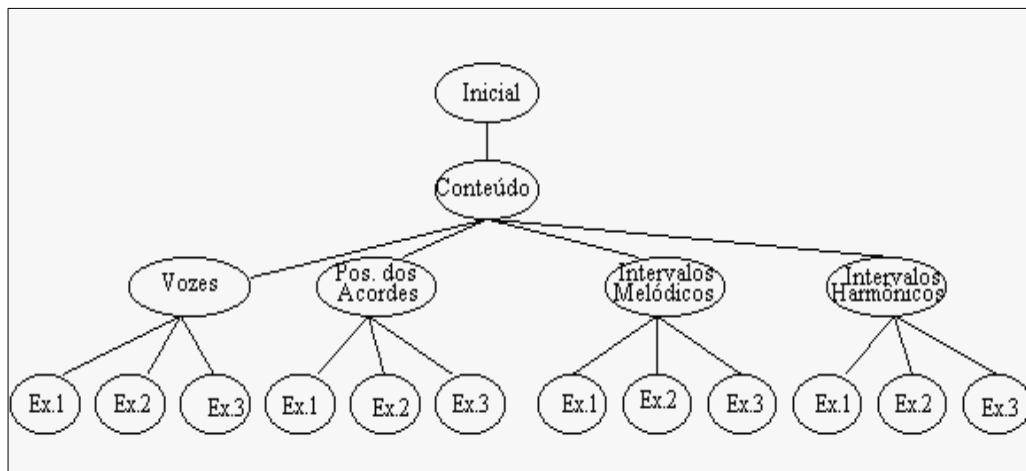


Figura 5.2 Documento hipermedia.

3) Banco de Dados de Apoio:

Serve para guardar (num banco de dados relacional) o material didático (definições, exemplos, exercícios).

4) Bases de Conhecimento:

É composta das regras (bem-formadas e mal-formadas) relacionadas ao conhecimento do domínio, e das estratégias pedagógicas. As regras bem-formadas fornecem o conhecimento correto, enquanto que as mal-formadas representam o conhecimento incorreto ou ruidoso fornecido pelo especialista (servindo para auxiliar na detecção de erros).

5.3.2 Banco de Dados de Perfil e Administração

Aqui são armazenados os dados relativos ao perfil do aprendiz e da administração do sistema. Os dados do aprendiz são constantemente atualizados, e a cada vez que o aprendiz se conecta ao sistema, é recuperada sua última interação.

5.4 Funcionamento Geral

Diversos aspectos envolvem o funcionamento do sistema. Entre estes estão: como é feita a escolha de estratégias pedagógicas, como é o procedimento de cooperação entre os agentes na resolução de um problema,

como é feita a avaliação do aprendiz, entre outros. Uma vez tendo escolhido a navegação guiada, o aluno se cadastra e faz um teste inicial para o sistema analisar em que nível se encontra.

5.4.1 Funcionamento de um Agente

A Figura 5.3 apresenta mais um nível de detalhamento do Sistema Tutor, enfatizando especificidades de seus principais módulos ao destacar submódulos que implementam suas funcionalidades.

Módulo Tutor:

Este módulo consta dos submódulos descritos abaixo:

Mediador: Responsável pela interação com o aprendiz, consta de uma interface gráfica.

Guia Hiperídia: Responsável pela navegação personalizada. Direciona o aprendiz para o documento html que será gerado pelo Gerenciador de Tarefas Pedagógicas.

Gerenciador de Tarefas Pedagógicas: Gerencia o conhecimento alocando os recursos necessários ao aprendizado do aluno.

Validador de Respostas: Responsável por validar as respostas dos alunos em exercícios que não sejam de múltipla escolha.

Conhecimento Pedagógico: Repositório do conhecimento pedagógico. Banco de dados contendo conceitos, exemplos, exercícios... etc.

Componente Hiperídia:

Módulo da Modelagem do Aprendiz:

Este módulo consta dos submódulos descritos abaixo:

Remediador: Responsável pela próxima ação a ser tomada pelo sistema de acordo com o diagnóstico cognitivo do aluno.

Gerenciador de Histórico: Responsável pela organização do conhecimento pedagógico já visto pelo aluno durante as sessões.

Gerenciador de Perfil: Responsável pela atualização do nível do aluno, levando em consideração os novos recursos e seu desempenho nos exercícios apresentados.

Diagnóstico Cognitivo: Responsável por “captar” o nível atual de conhecimento do aluno em relação a determinado assunto ou domínio ensinado pelo sistema.

Inspecção: Responsável pelo modo inspecionado do aluno. Se o aluno quiser rever ou contestar seu atual nível de aprendizado, o sistema deverá mostrar os exercícios cujas respostas não foram satisfatórias e propor a resolução de um novo exercício do mesmo nível.

Modelo do Aprendiz: Repositório dos dados de perfil e histórico do aprendiz.

Módulo do Especialista:

Este módulo consta dos submódulos descritos abaixo:

Máquina de inferência: é responsável pelo raciocínio sobre as bases de conhecimento: *Base de Conhecimento Correto* e *Base de Conhecimento Ruidoso* (que consta de regras contendo algum tipo de mal entendimento sobre elementos do domínio de conhecimento).

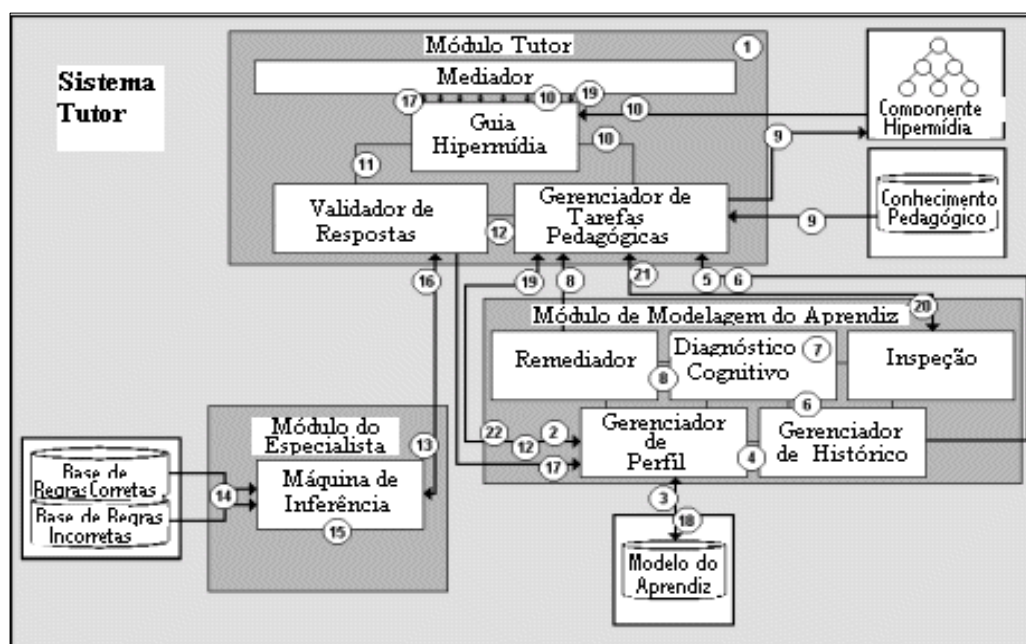


Figura 5.3 Descrição detalhada do Sistema Tutor de um agente da SATA.

1 – Considerando que o agente que contém o **Sistema Tutor** acima esteja com o controle da execução do sistema, o **Módulo Tutor** irá iniciar sua execução com a entrada (login) do aprendiz no sistema.

- 2 – O **Módulo Tutor** por meio do **Gerenciador de Tarefas Pedagógicas** irá solicitar ao **Módulo de Modelagem do Aprendiz** a próxima seqüência de recursos pedagógicos a ser apresentada ao aprendiz.
- 3 – O **Módulo de Modelagem do Aprendiz** por meio do **Gerenciador de Perfil** irá recuperar o **Modelo do Aprendiz** do banco de dados.
- 4 – O **Gerenciador de Perfil** repassará as informações para o **Gerenciador de Histórico** que verificará se o aprendiz, em sua última sessão de ensino, viu todo o conteúdo da seqüência de recursos a que foi submetido.
- 5 – Caso o conteúdo não tenha sido visto por completo, o **Gerenciador de Histórico** retorna ao **Gerenciador de Tarefas Pedagógicas** a seqüência de recursos a ser apresentada em continuação a que já havia sido mostrada anteriormente, retornando também uma lista dos recursos já apresentados ao aprendiz anteriormente para que o **Gerenciador de Tarefas Pedagógicas** possa gerar o conteúdo de revisão.
- 6 – Caso não haja nenhuma seqüência incompleta de recursos apresentados ao aprendiz, o **Gerenciador de Histórico** repassa as informações de perfil para o **Diagnóstico Cognitivo** e retorna uma lista dos recursos já apresentados ao aprendiz anteriormente para o **Gerenciador de Tarefas Pedagógicas**, que irá gerar o conteúdo de revisão.
- 7 – O **Diagnóstico Cognitivo** processará as informações baseado num conjunto de regras definidas pelo especialista e obterá o nível de aprendizado atual do aprendiz.
- 8 – Em seguida, o **Diagnóstico Cognitivo** repassará esse nível de aprendizado para o **Remediador** que, finalmente, retornará para o **Gerenciador de Tarefas Pedagógicas** a seqüência de recursos a serem apresentadas a partir de agora ao aprendiz.
- 9 – De posse da seqüência de recursos a serem apresentados ao aprendiz e da lista de recursos já apresentados anteriormente, o **Gerenciador de Tarefas Pedagógicas** gera os documentos html (**Componente Hipermídia**) necessários a apresentação dos novos recursos (**Conhecimento Pedagógico**) buscando-os no banco de dados. Caso o aprendiz solicite uma revisão, os documentos para revisão dos recursos já apresentados anteriormente (definições, exemplos e exercícios com a resposta do aprendiz e a resposta certa) são gerados.
- 10 – Uma vez gerados os documentos html, o **Guia Hipermídia** direciona o aprendiz através do **Mediador** para o documento a ser apresentado. O aprendiz

então irá interagir com o **Mediador** de acordo com o tipo de recurso. Caso o recurso seja um exercício, o aprendiz deverá respondê-lo.

11 – Após responder o exercício, a resposta é repassada para o **Validador de Respostas**.

12 – Se o exercício for de múltipla escolha, o **Validador de Respostas** busca a resposta certa no **Gerenciador de Tarefas Pedagógicas**, compara com a resposta do aprendiz e repassa seu desempenho ao **Gerenciador de Perfil** no **Módulo de Modelagem do Aprendiz**.

13 – Se o exercício for do tipo aberto, no qual a resposta do aprendiz deve ser validada pelo sistema, o **Validador de Respostas** repassa para o **Módulo do Especialista** as informações sobre o exercício e a resposta do aluno.

14 – O **Módulo do Especialista** por meio da **Máquina de Inferência** busca nas **Base de Conhecimento Correto** e **Base de Conhecimento Incorreto** as regras necessárias para validação da resposta.

15 – Caso a resposta, ou parte dela, não possa ser validada por este agente, a **Máquina de Inferência** solicita através da **Camada Social** que outro agente da sociedade valide a resposta.

16 – De posse da validação da resposta, a **Máquina de Inferência** retorna o desempenho do aprendiz e o caminho (passo a passo) percorrido pelo sistema para provar se a resposta do aluno estava certa ou não, ao **Validador de Respostas**.

17 – O **Validador de Respostas** repassará para o **Gerenciador de Perfil** o desempenho obtido na avaliação e para o **Mediador**, através do **Guia Hiperímida**, o desempenho e o caminho de obtenção desse desempenho (prova).

18 – O **Gerenciador de Perfil** atualiza o **Modelo do Aprendiz** no banco de dados.

19 – Ao fim de cada seqüência de recursos, o **Gerenciador de Perfil** solicita ao **Gerenciador de Tarefas Pedagógicas** que mostre ao aprendiz seu nível atual de aprendizado e seu desempenho na seqüência de recursos que acabou de ser apresentada a ele. Caso o aprendiz questione seu desempenho ou nível de aprendizado, poderá solicitar uma inspeção.

20 – Caso isto aconteça, o **Gerenciador de Tarefas Pedagógicas** solicita inspeção ao **Módulo de Modelagem do Aprendiz** que, por meio da **Inspeção** verificará se o aprendiz, em algum momento, teve uma diferença de desempenho satisfatória

(nível de satisfação definido pelo especialista) ao resolver exercícios de um mesmo nível de dificuldade, o que poderia significar descuidos.

21 – Caso a **Inspeção** encontre alguma ocorrência desses exercícios, uma nova seqüência de recursos é repassada ao **Gerenciador de Tarefas Pedagógicas** para apresentação ao aprendiz, que assim terá uma segunda chance de resolvê-los (serão apresentados exercícios similares) e aumentar seu nível de aprendizado.

22 – Caso o aprendiz concorde com os resultados mostrados, uma nova seqüência é requisitada ao **Módulo de Modelagem do Aprendiz**. (Voltar para 2).

5.4.1.1 Execução de uma Estratégia Pedagógica

Em relação à execução da estratégia (geração de html's , busca de arquivos de estratégias), ou seja, navegação guiada, cada agente quando requisitado passa a ter o controle das ações do sistema (Figura 5.4). Também não há saltos entre agentes enquanto está em execução uma estratégia pedagógica. A escolha de uma estratégia pedagógica a ser apresentada ao aprendiz é feita segundo um mecanismo de regras que avaliam qual a mais adequada num dado momento.

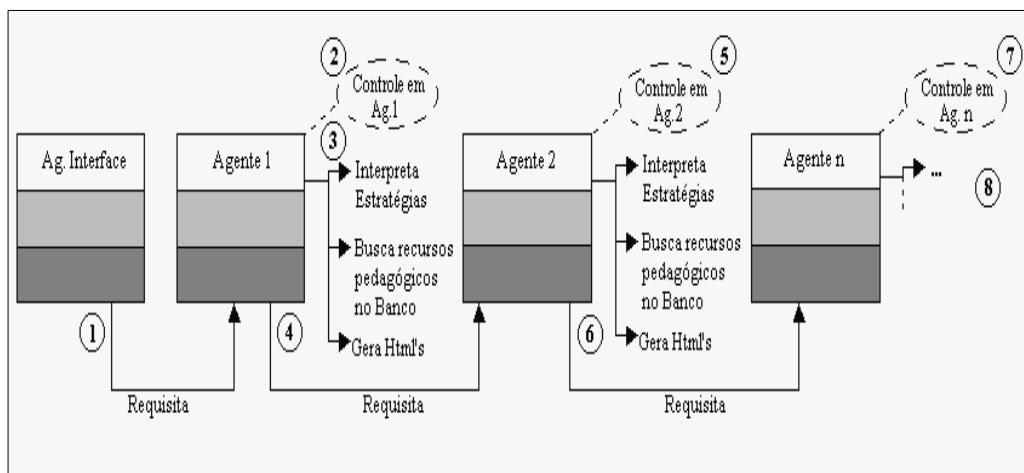


Figura 5.4 Interpretação da estratégia pedagógica.

5.4.1.2 Diagnóstico Cognitivo

O diagnóstico cognitivo do aprendiz, ou seja, a avaliação de seu estado de aprendizagem, é feito por cada agente sempre que se termina a execução de uma estratégia pedagógica.

Com o diagnóstico cognitivo tem-se uma visão a cada momento das deficiências ou aptidões do aprendiz, e conseqüentemente o que o sistema deve oferecer na interação seguinte, ou seja, qual a próxima estratégia a ser usada (Figura 5.5).

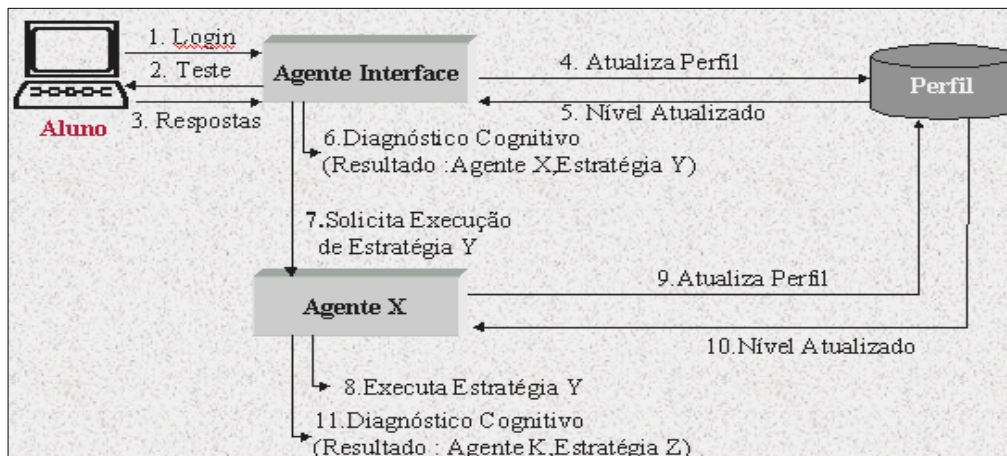


Figura 5.5 Diagnóstico cognitivo.

5.4.2 Funcionamento da Arquitetura Multiagentes

5.4.2.1 Validação de resposta

Em relação à validação de uma resposta dada pelo aprendiz (Figura 5.6), o agente controlador no momento (atual) requisita apenas um serviço de outro agente da sociedade, que retorna o que foi solicitado ao controlador. Então o agente controlador continua com sua execução.

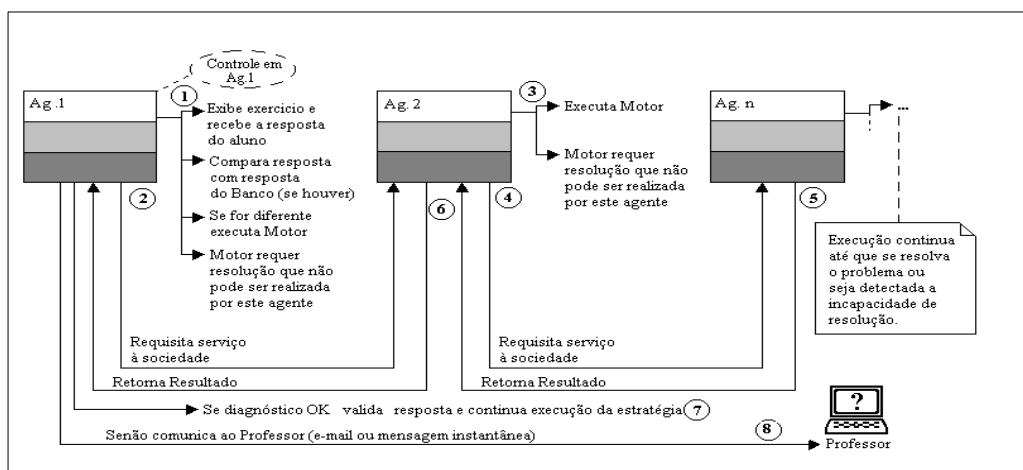


Figura 5.6 Para o passo a passo na validação de resposta.

5.4.3 Cenários de Interação

Cenário 1: Diagnóstico

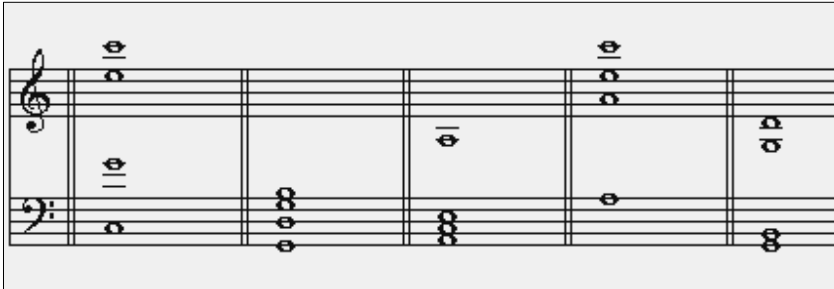


Figura 5.7 Exercício do tipo *múltipla escolha*.

1. Sistema

Identifique em cada compasso a voz (baixo, tenor, contralto, soprano) que está com erro quanto à extensão (Figura 5.7).

2. Aprendiz

Consulta o assunto (Extensão das Vozes) novamente na navegação estática e envia a seguinte resposta:

Compasso 1: tenor, contralto e soprano, compasso 2: soprano, compasso 3: contralto e soprano, compasso 4: tenor, contralto e soprano, compasso 5: tenor.

3. Sistema

Verifica que a resposta está correta e exhibe o próximo recurso (uma definição de distância entre as vozes).

Os procedimentos de avaliação da resposta são:

1. A partir da resposta do aprendiz, o sistema compara esta resposta de acordo com os tipos de exercícios disponíveis:
 - a) No exercício do tipo *múltipla escolha*, existe apenas uma resposta certa, guardada no banco de dados, e é dado ao aprendiz uma série de opções para ser escolhida uma delas.
 - b) No exercício do tipo *validação de resposta*, é usado o motor de inferência, que busca nas bases de conhecimento as regras relativas ao assunto (no caso aqui extensão das vozes) e então tentará validar a resposta do aprendiz.
2. Como o exercício é do tipo *múltipla escolha*, a resposta do sistema é buscada diretamente no banco de dados, comparada com a resposta (do aprendiz), e em seguida o perfil (desempenho) do aprendiz é atualizado.

Cenário 2: Diagnóstico

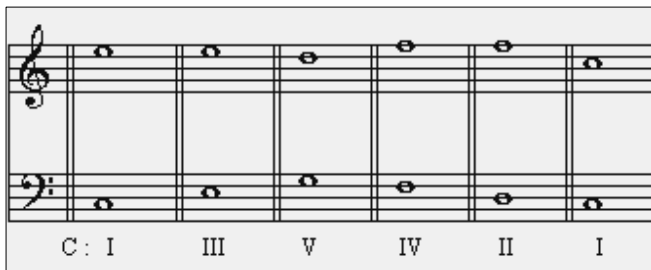


Figura 5.8 Exercício do tipo *validação de resposta*.

1. Sistema

Preencha e distribua as vozes contralto e tenor, com dobramento da nota do Baixo, observando as regras de extensão das vozes (Figura 5.8).

2. Aprendiz

Preenche as vozes que faltam e envia a resposta.

3. Sistema

Verifica erros em mais de um compasso. Exibe resposta ao aprendiz e torna disponível o passo a passo da resolução do exercício caso o aprendiz queira consultar.

4. Aprendiz

Decide ver o passo a passo na validação de resposta do exercício.

5. Sistema

Exibe o passo a passo e torna disponível a revisão do assunto.

6. Aprendiz

Encerra a sessão.

Neste caso, é usado o motor de inferência, já que o exercício é do tipo *validação de resposta*. A atualização do perfil (desempenho) do aprendiz é idêntica ao primeiro caso.

Cenário 3: Instrução (dica)

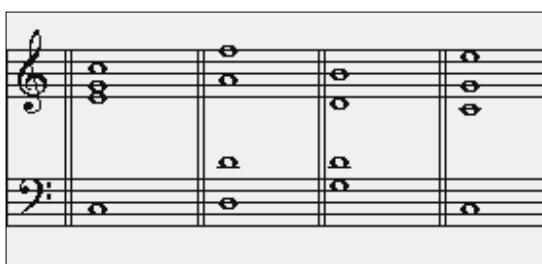


Figura 5.9 Exercício sobre o assunto *distribuição das vozes*.

A Dica é um recurso que está sempre disponível para o aluno quando se está resolvendo um problema.

1. Sistema

Identifique em cada compasso, onde há erro de escrita na distribuição das vozes (Figura 5.9).

2. Aprendiz

Consulta uma Dica disponível (Figura 5.10):

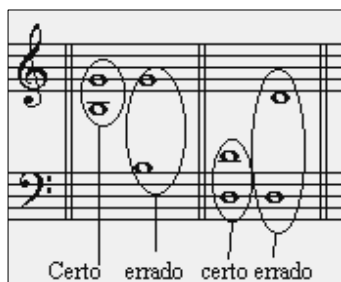


Figura 5.10 Dica

Cenário 4: Diagnóstico (com cooperação entre agentes)



Figura 5.11 Exercício sobre o assunto *tipos de movimento das vozes*.

1. Sistema

Complete as vozes (tenor, contralto e soprano) da seguinte forma: 1º compasso e 3º compasso com movimento oblíquo entre as vozes contralto e soprano, e 2º compasso e 4º compasso com movimento paralelo entre as vozes tenor e contralto (Figura 5.11).

2. Aprendiz

Preenche as vozes, e envia a resposta.

3. Sistema

O agente Vozes chama o agente Acordes para verificar se não há erro na formação de cada acorde preenchido pelo aprendiz, chama o agente Cifras para verificar se as cifras estão corretas, e verifica se a resposta está de acordo com o enunciado do problema.

Cenário 5: Diagnóstico (com cooperação entre agentes)

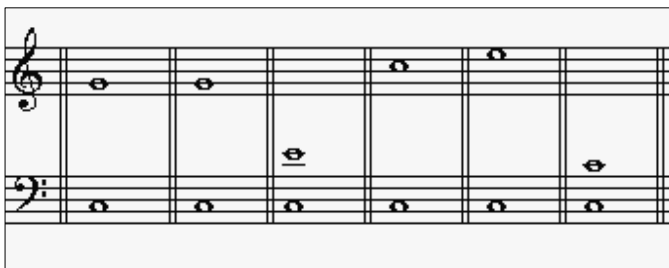


Figura 5.12 Exercício sobre o assunto *distância entre as vozes*.

1. *Sistema*

Preencha as vozes tenor e soprano observando as regras de distância entre as vozes (Figura 5.12).

2. *Aprendiz*

Preenche e envia a resposta.

3. *Sistema*

A resposta do aluno é primeiramente analisada sob vários aspectos: barra dupla de compasso, quantidade de notas por compasso e a distância entre as vozes. Após validar a extensão das vozes (gerando o passo a passo da validação), verifica o intervalo entre as notas para saber se o intervalo entre elas está correto. Como esta tarefa não faz parte deste agente (Vozes), ela é repassada para o agente Intervalo, que retorna a resposta da validação e o passo a passo de sua validação para o agente Vozes. De posse desta resposta o agente Vozes valida por inteiro a resposta do aluno.

Cenário 6: Modo inspecionado

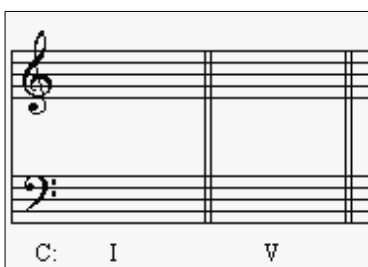


Figura 5.13 Exercício sobre o assunto *extensão das vozes*.

1. *Sistema*

Preencha os acordes com as quatro vozes (baixo, tenor, contralto e soprano) de acordo com as cifras, observando as regras de extensão das vozes (Figura 5.13).

2. Aprendiz

Envia a resposta contendo erros (Figura 5.14).

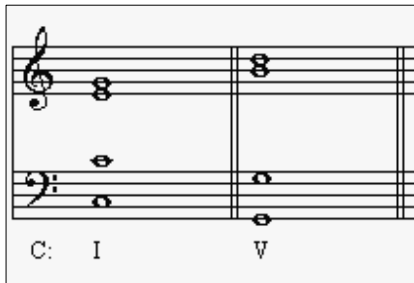


Figura 5.14 Resposta do aluno.

3. Sistema

Avalia a resposta e disponibiliza um outro exercício no mesmo nível e no mesmo assunto para o aluno responder.

5.5 Conclusão

Tivemos neste capítulo, uma visão da arquitetura do sistema SHART-Web bem como de seu funcionamento. Mostramos como é o funcionamento de cooperação entre os agentes e como se interrelacionam os vários componentes do sistema a partir da visão do agente. No próximo capítulo mostraremos como foi feita a modelagem do sistema.

Capítulo 6

Modelagem do Sistema SHART-Web

6.1 Introdução

Neste capítulo abordamos a modelagem do sistema. Para isso, é necessário seguir um método de desenvolvimento. Um método de desenvolvimento é composto pelo menos de uma linguagem de modelagem e um processo de desenvolvimento. Utilizamos uma linguagem específica para modelagem, a UML (Unified Modeling Language - Linguagem de Modelagem Unificada), que trata de todas as etapas de desenvolvimento de *softwares*, desde a fase de análise até a fase de implementação [Booch et alii 2000].

6.2 Análise

A fase de Análise compreende o estudo das funcionalidades do sistema. O diagrama de Pacotes, o modelo Conceitual, os *Casos de Uso* essenciais, os diagramas de Interação e os diagramas de Estados fazem parte desta fase.

6.2.1 Diagrama de Pacotes

Os pacotes descrevem um conjunto ou reunião de classes afins. Ou seja, que têm o mesmo objetivo ou pertencem à mesma família. Os diagramas de pacotes mostram de uma forma superficial as interações e dependências entre as camadas de um determinado sistema, omitindo as classes que compõem os pacotes. O diagrama de pacotes da Figura 6.1 descreve as dependências entre as camadas da arquitetura do sistema (ver capítulo 5).

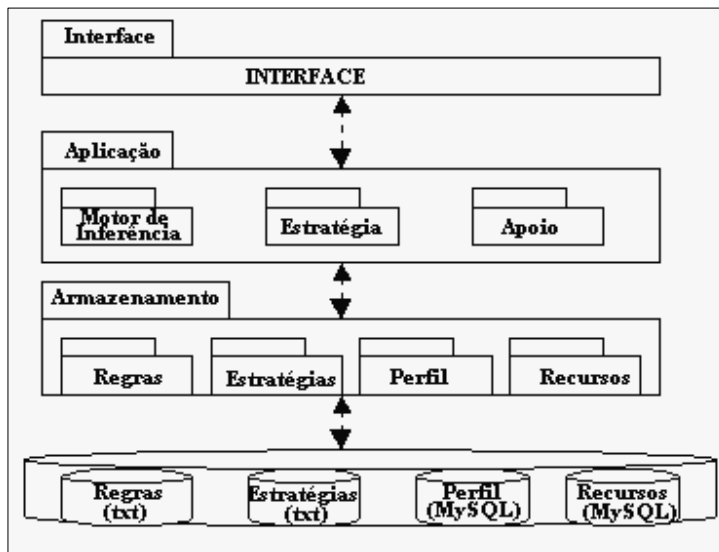


Figura 6.1 Diagrama de Pacotes.

6.2.2 Modelo Conceitual

No modelo conceitual (Figura 6.2), têm-se uma visão estática e superficial da organização das classes num determinado pacote ou sistema num todo. Também descreve as relações de dependência e associações entre as classes. Relações de herança, composição e métodos nas classes não são levados em conta neste diagrama.

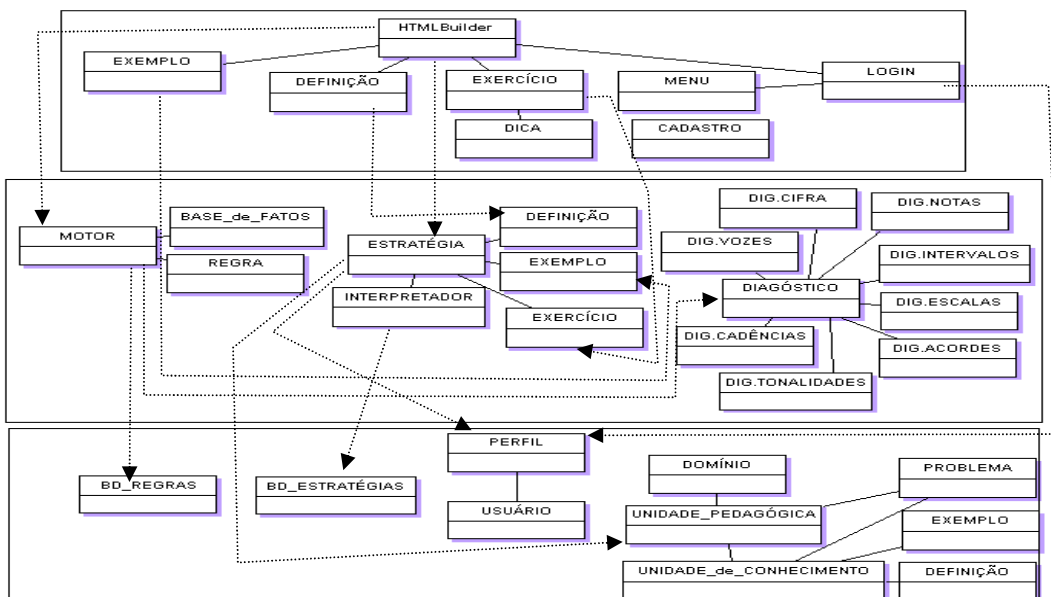


Figura 6.2 Diagrama conceitual do sistema.

6.2.3 Casos de Uso Essenciais

O diagrama de *Casos de Uso* é uma técnica usada para descrever as situações didáticas generalizadas vivenciadas pelos atores do processo ensino-aprendizagem (aluno e professor).

O diagrama de *Casos de Uso* mostra as funcionalidades do sistema. É uma visão externa do sistema. Mostra quem vai usar o sistema (os atores, no caso aqui são aluno, professor e especialista), e quais são os papéis de cada um.

Vê-se (Figura 6.3), que o aluno e o professor, ambos precisam se cadastrar e entrar com login, enquanto existem algumas funções que somente o aluno faz (Iniciar Sessão de Estudos), outras que são comuns a ambos (Inicia Chat), e outras que são de competência do especialista (Insere recursos, Insere Estratégias e Insere Base de Conhecimento).

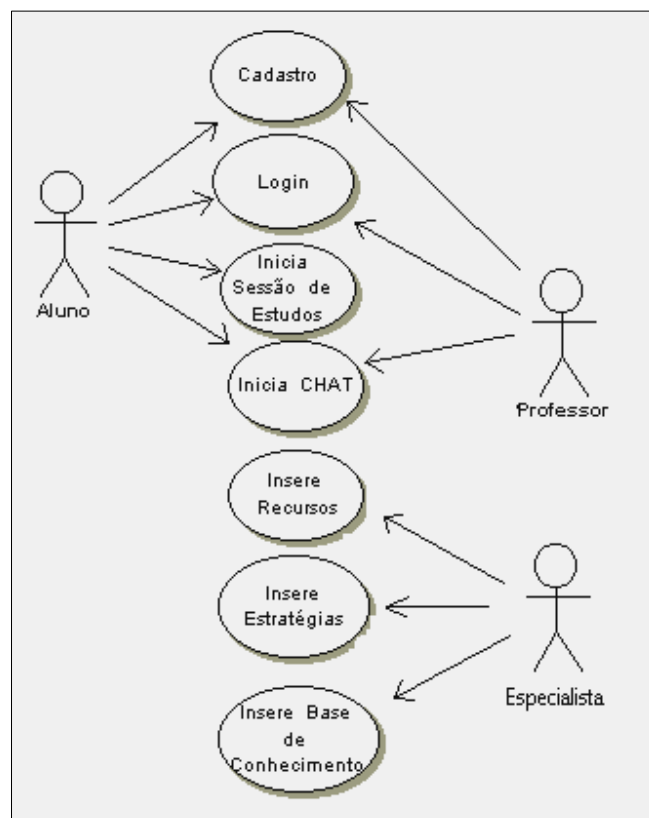


Figura 6.3 Diagrama de *Casos de Uso* Essenciais.

“Inicia Sessão de Estudos” é desdobrado em três diagramas (Figura 6.4), que mostram respectivamente as exibições de Definição, Exemplo e Exercício.

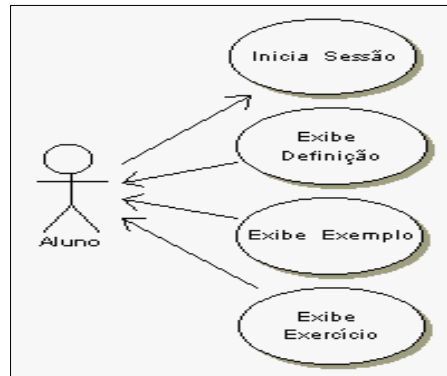


Figura 6.4 Diagrama Inicia Sessão de Estudos.

6.2.4 Diagramas de Interação

Os diagramas de Interação são de dois tipos: Seqüência e Colaboração.

6.2.4.1 Diagramas de Seqüência

A partir do Diagrama de *Casos de Uso* são elaborados os diagramas de seqüência utilizados na modelagem na fase de análise do sistema. Um diagrama de seqüência é uma ordenação temporal de mensagens entre objetos e entidades. São usados para mostrar a visão dinâmica de um sistema.

Num diagrama de seqüência a linha tracejada representa a existência de um objeto num determinado período de tempo. As setas seguem uma ordem temporal de cima para baixo [Booch et alii 2000].

1) Cadastro: Neste diagrama (Figura 6.5), vê-se inicialmente a exibição da tela para o aluno se cadastrar, em seguida os dados são enviados até a classe usuário, que faz a validação do cadastro, e logo após é feito um teste com aluno para o sistema saber em que nível se encontra, e a partir da resposta ao teste, é mostrado o conteúdo didático correspondente ao seu nível.

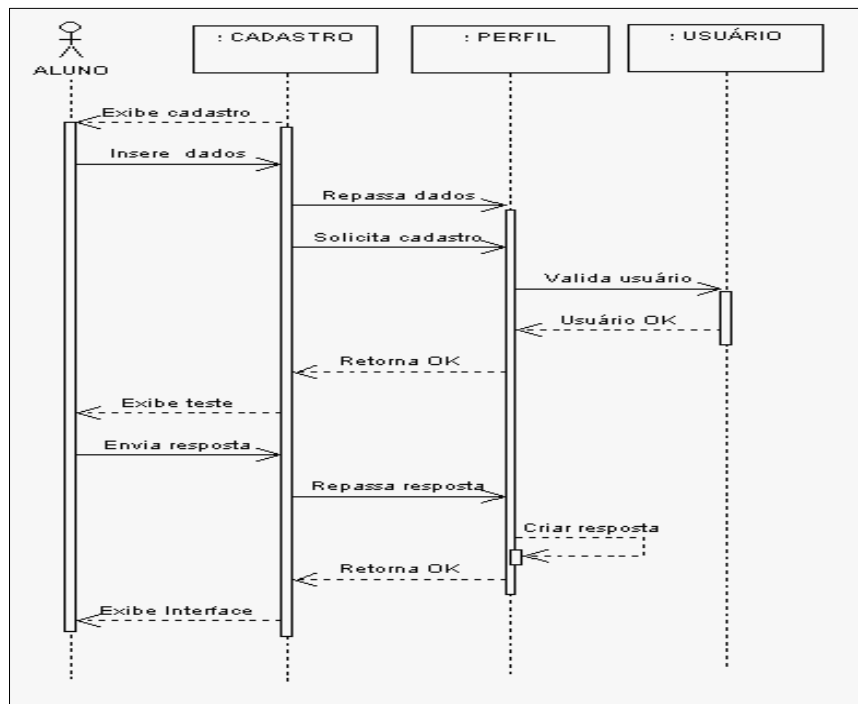


Figura 6.5 Diagrama de Seqüência (Cadastro).

2) Login: Este diagrama (Figura 6.6) mostra a seqüência desde a exibição da tela para o aluno entrar com o login/senha, até sua validação pela classe usuário. Se for validado, é exibida a tela com as opções de estudo para o aluno escolher, senão retorna a mesma tela.

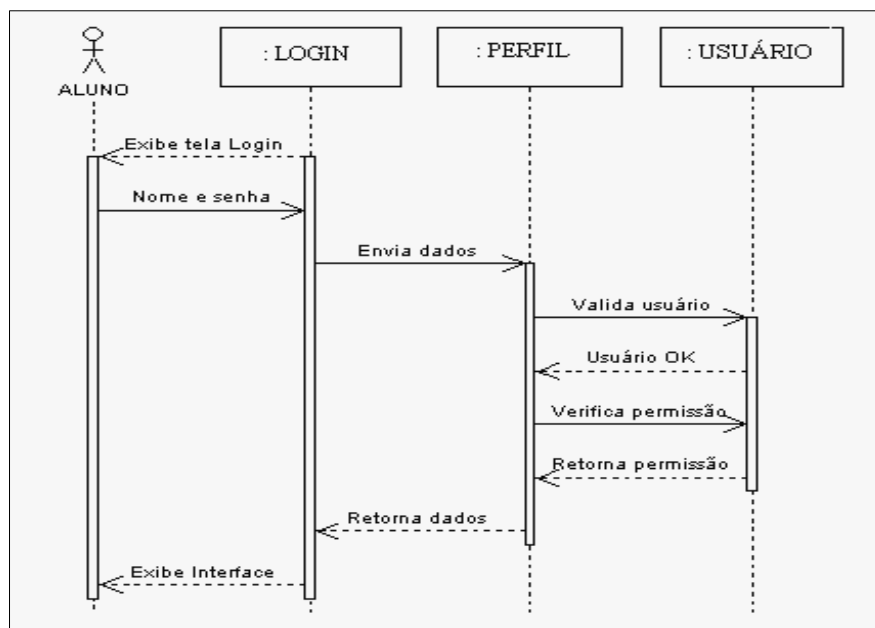


Figura 6.6 Diagrama de Seqüência (Login).

3) Inicia Sessão de Estudos: Uma vez validado o login, é mostrado ao aluno a tela com as opções (navegação guiada ou sem guia). Se o aluno escolher navegação guiada, a classe Menu envia para o Banco de Dados Perfil, onde estão armazenados os dados sobre o aluno e sua última interação. Esses dados são repassados para a classe Estratégia, que então seleciona a estratégia adequada para o momento (Figura 6.7).

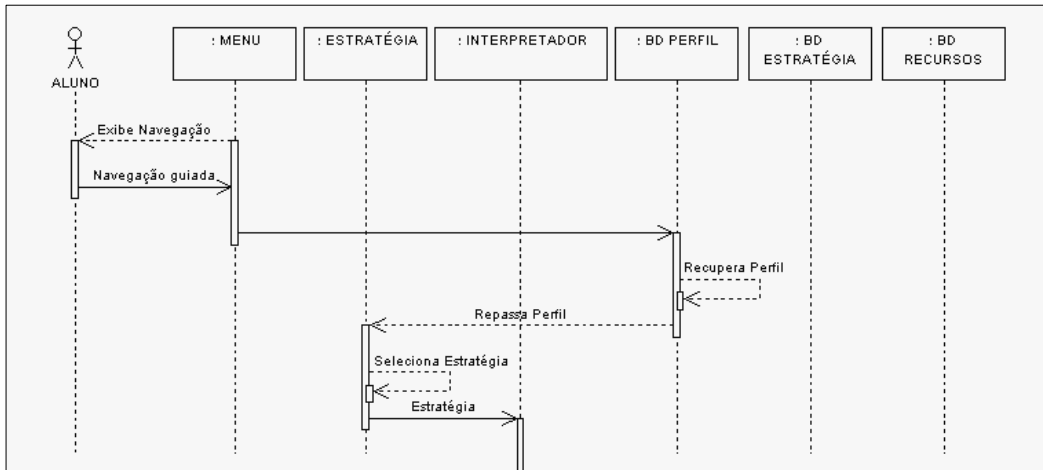


Figura 6.7 Diagrama de Seqüência (Inicia Sessão de Estudos).

4) Exibe Definição: Este diagrama (Figura 6.8) mostra o processo de exibição de uma Definição para o aluno. Começa pela classe Estratégia solicitando o próximo recurso à classe Interpretador, que devolve a informação para Estratégia, que busca Definição em Recursos, que segue vários passos, até ser exibida ao aluno.

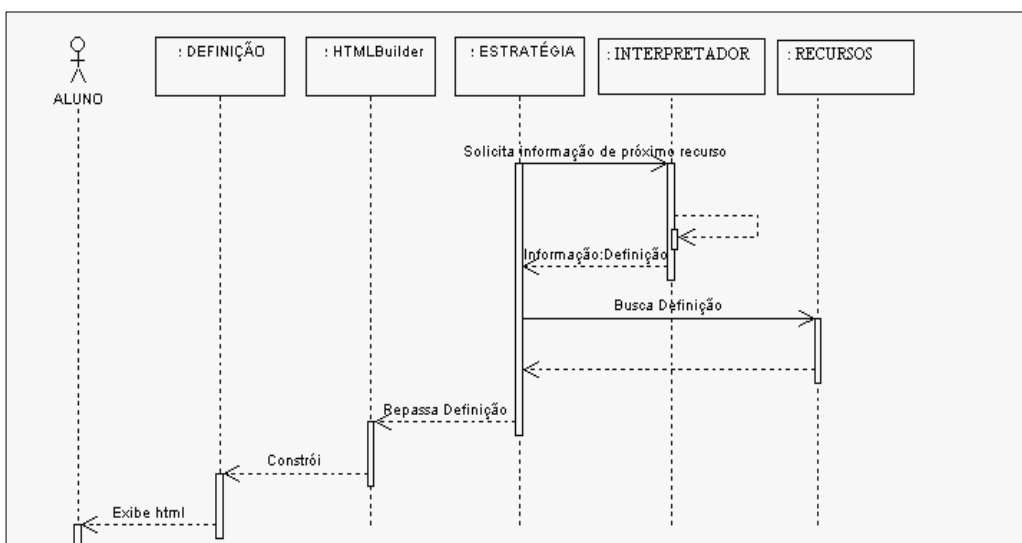


Figura 6.8 Diagrama de Seqüência (Exibe Definição).

5) Exibe Exemplo: Este diagrama (Figura 6.9) segue o mesmo processo do diagrama anterior, com a diferença do conteúdo a ser exibido (no caso aqui, Exemplo).

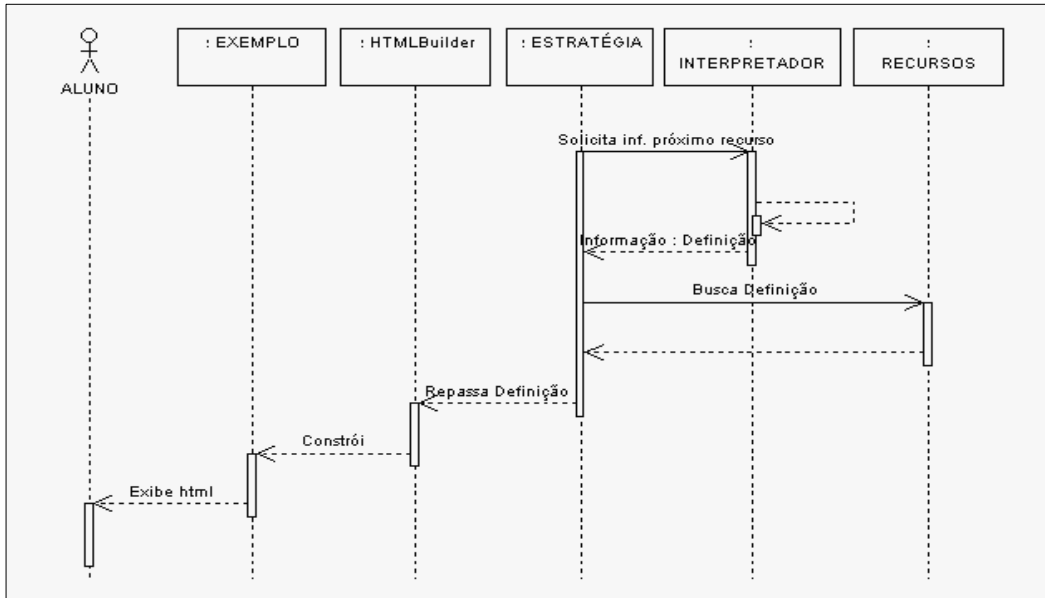


Figura 6.9 Diagrama de Sequência (Exibe Exemplo).

6) Inicia CHAT: Este diagrama (Figura 6.10) mostra a sequência de um *chat* entre aluno e professor. A classe Chat faz a intermediação entre aluno e professor na troca de mensagens.

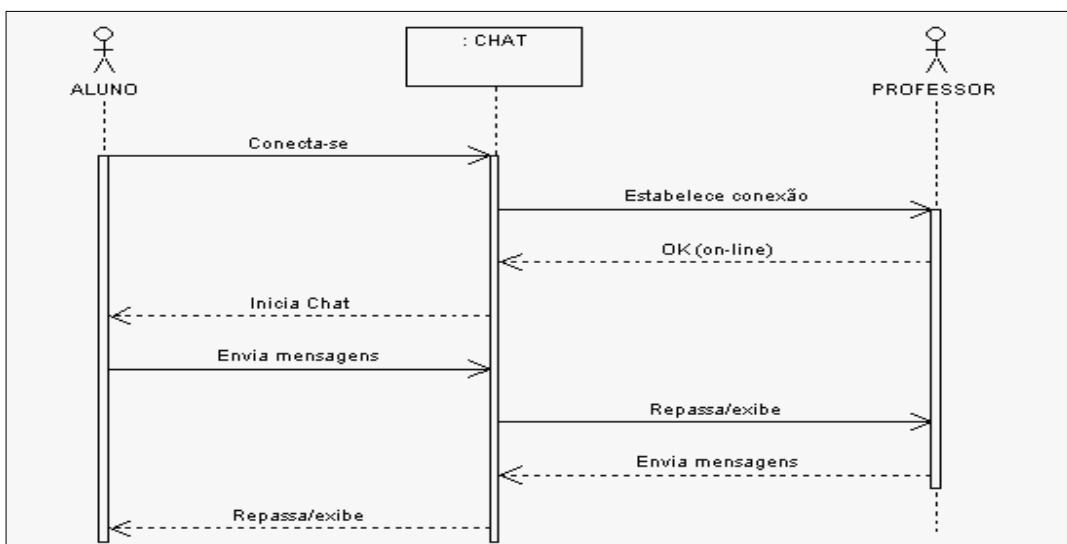


Figura 6.10 Diagrama de Sequência (Inicia Chat).

7) InseRe Recursos: Neste diagrama (Figura 6.11), vê-se como é a seqüência de inserção de recursos pedagógicos (definição, exemplo, exercício) pelo Especialista. Inicialmente a classe Menu exibe a tela com o formulário a ser preenchido pelo Especialista. Após receber o cadastro preenchido pelo Especialista, a classe Menu o repassa para a classe Cadastro, que o repassa para a classe Recursos que faz a inserção do recurso pedagógico no banco de dados.

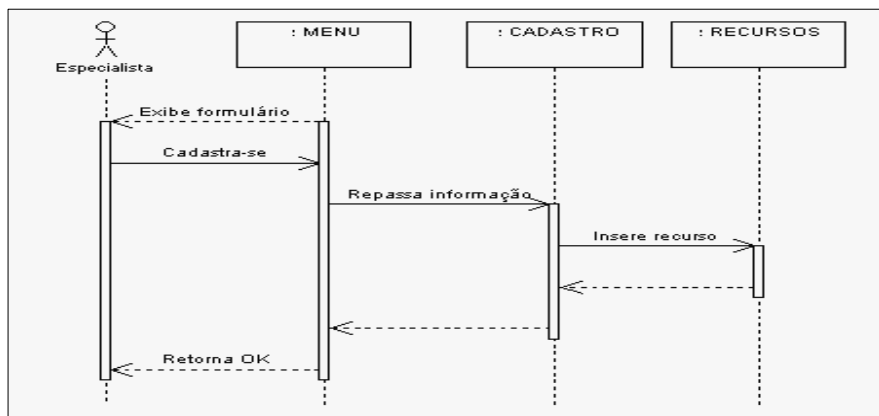


Figura 6.11 Diagrama de Seqüência (InseRe Recursos).

8) InseRe Estratégias: Vê-se neste diagrama (Figura 6.12) a seqüência de inserção de uma estratégia pedagógica pelo Especialista. Desde a exibição do formulário de inserção pela classe Menu até a confirmação feita pelo sistema.

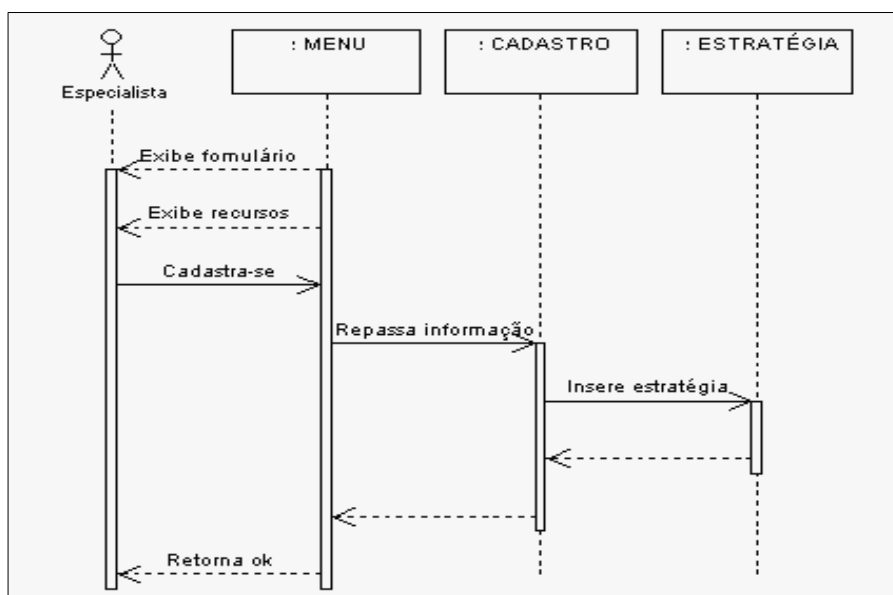


Figura 6.12 Diagrama de Seqüência (InseRe Estratégias).

6.2.4.2 Diagramas de Colaboração

Um diagrama de colaboração é um diagrama de interação que dá destaque à organização estrutural dos objetos, enviando e recebendo mensagens.

9) Exibe Exercício: Este diagrama (Figura 6.13) mostra a colaboração entre várias classes no processo de seleção de uma estratégia pedagógica até o armazenamento de uma resposta do aluno no Banco de Dados (Perfil).

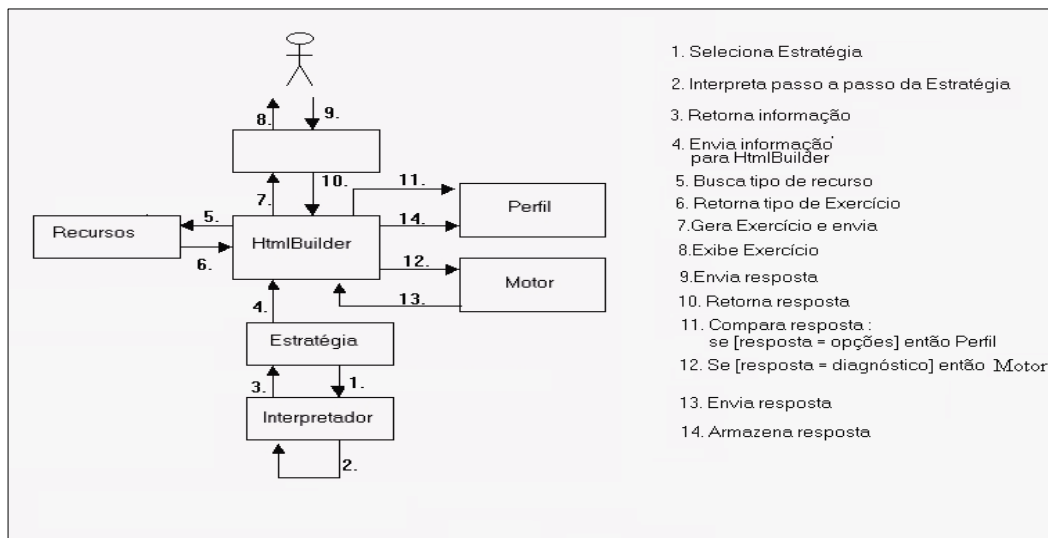


Figura 6.13 Diagrama de Colaboração (Exibe Exercício).

10) Colaboração entre dois agentes:

Na Figura 6.14 vê-se a colaboração entre dois agentes (Vozes e Intervalos). As linhas tracejadas (Banco de Dados de Apoio e Banco de Dados de Perfil) ajudam a visualizar melhor as interações dos agentes Vozes e Intervalos com estes bancos de dados. Após o aprendiz fazer o login e escolher o assunto vozes para estudar, o agente Vozes que tem o controle, executa sua estratégia e exibe um exercício para o aprendiz. O aprendiz envia uma resposta errada. Então o agente detecta que o erro é referente a um conceito que pertence a outro agente (Intervalos) para o qual passa o controle. O agente Intervalos exibe então uma definição sobre o assunto (Intervalos) para o aprendiz.

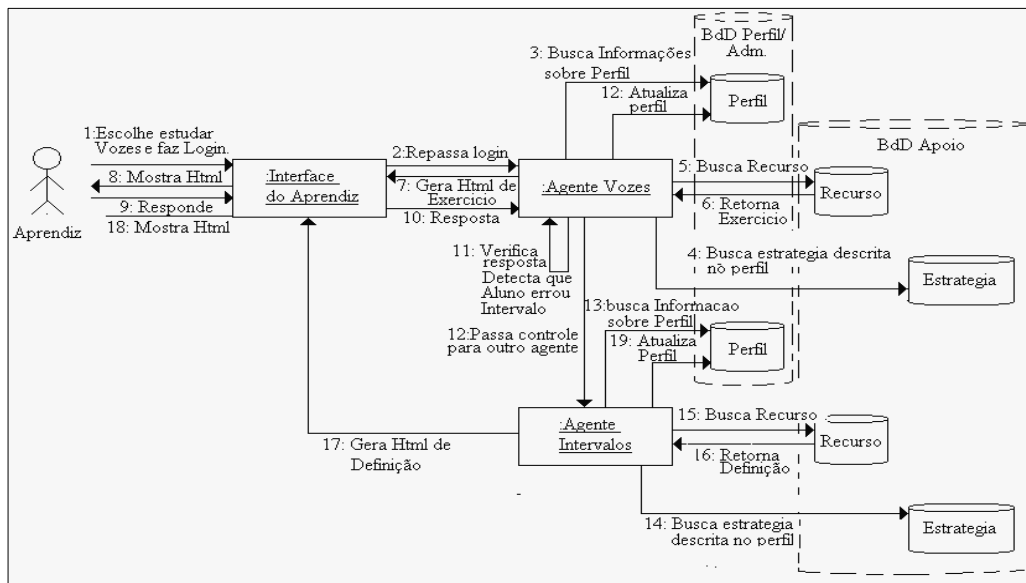


Figura 6.14 Diagrama de colaboração entre dois agentes.

6.2.5 Diagramas de Estados

Um diagrama de estados mostra as reações de uma determinada classe quando solicitada por eventos. Descreve uma visão dinâmica do sistema mostrando o comportamento da classe do seu estado inicial ao seu estado final.

1) Motor de Inferência: Aqui mostramos como é o comportamento da classe Motor diante de eventos. A classe tem como evento inicial a busca por objetivo, até a validação deste objetivo num evento final (Figura 6.15).

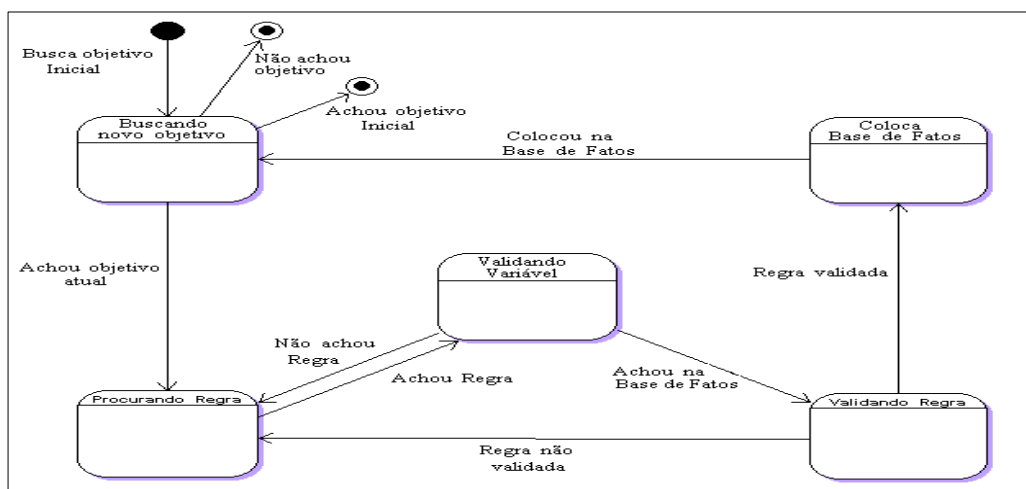


Figura 6.15 Diagrama de Estados para a classe Motor de Inferência.

6.3 Projeto

Na fase de Projeto tenta se aproximar o modelo, gerado durante a fase de Análise, da implementação do sistema. A passagem da fase de Análise para a fase de Projeto corresponde à uma expansão do modelo.

6.3.1 Funções Básicas do Sistema

As funções básicas do sistema são:

- Navegação guiada no domínio da harmonia tradicional.
- Validação de respostas (do aluno) por meio de cooperação entre os agentes.

6.3.2 Casos de Uso Reais

Os *Casos de Uso Reais* representam uma importante ferramenta da fase de projeto e estão relacionados a como as tarefas serão realizadas. A seguir serão ilustrados três *Casos de Uso Reais* (Cadastro, Inicia Sessão de Estudos e Exibe Definição, Exemplo ou Exercício):

1) Cadastro:

Atores : Aluno (Aprendiz) e Professor.

Descrição : O Aluno ou Professor cadastra dados pessoais. Após encerrada esta etapa, poderá efetuar o login no sistema.

Seqüência de Eventos:

Ator/Ação:

Resposta do Sistema:

1. Aluno/Professor
preenche.

1. Sistema exibe tela Cadastro.

3. Sistema valida usuário.

4. Retorna OK.

2) Inicia Sessão de Estudos:

Atores: Aluno.

Descrição: O Aluno tem acesso a uma Sessão de Estudos.

Seqüência de Eventos:

Ator/Ação:

2. Aluno escolhe
navegação com guia.

Resposta do Sistema:

1. Sistema valida o login e é mostrada ao Aluno tela com opções de navegação (com guia ou sem guia).
2. Sistema recupera Perfil no Banco de Dados (última interação) e em seguida seleciona a estratégia a ser usada.

3) Exibe Definição, Exemplo ou Exercício:

Atores: Aluno.

Descrição: Exibe para o aluno os recursos pedagógicos Definição, Exemplo e Exercício

Seqüência de Eventos:**Ator/Ação:****Resposta do Sistema:**

1. Sistema interpreta Estratégia (Definição, Exemplo ou Exercício)
2. Sistema busca um dos recursos no Banco de Dados.
3. Sistema gera Html.
4. Sistema exibe Html.

6.3.3 Diagramas de Classes

Um diagrama de classes mostra a visão estática do sistema. Uma classe descreve um conjunto de objetos compartilhando os mesmos atributos, operações e relacionamentos. A sua representação gráfica é um retângulo. Cada classe tem um nome, atributos e operações [Booch et alii 2000].

Os diagramas das Figuras 6.16 e 6.17 descrevem relações entre as classes que compõem a camada Aplicação. A Figura 6.16 mostra as classes do pacote Motor de Inferência e a Figura 6.17 mostra as classes do pacote Estratégia (ver Diagrama de Pacotes).

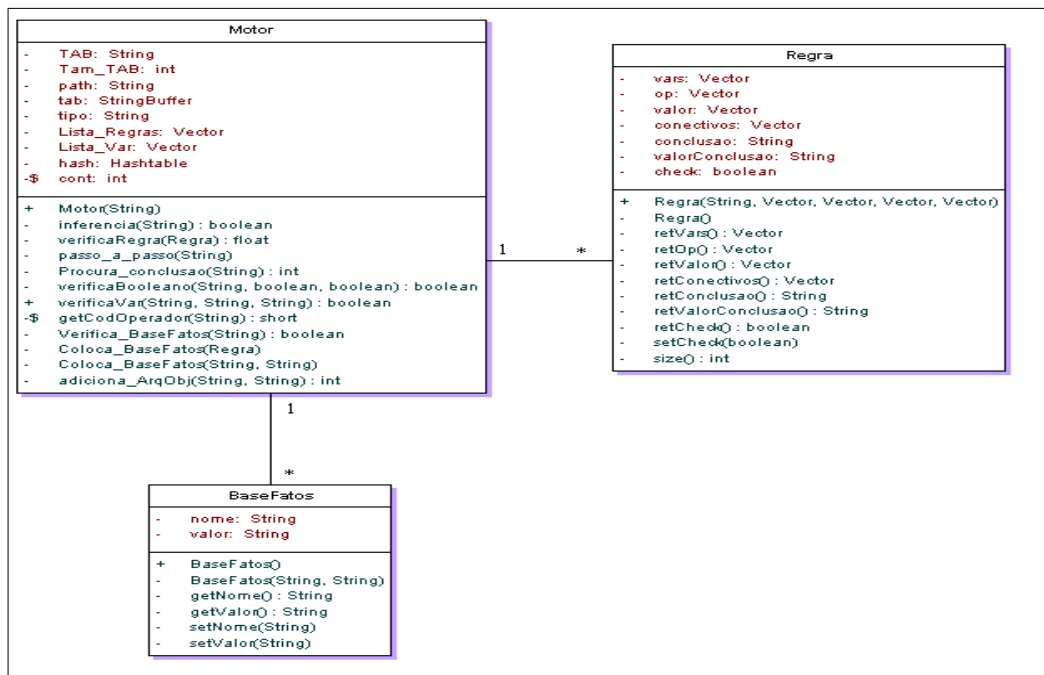


Figura 6.16 Diagrama de Classes 1 (Aplicação).

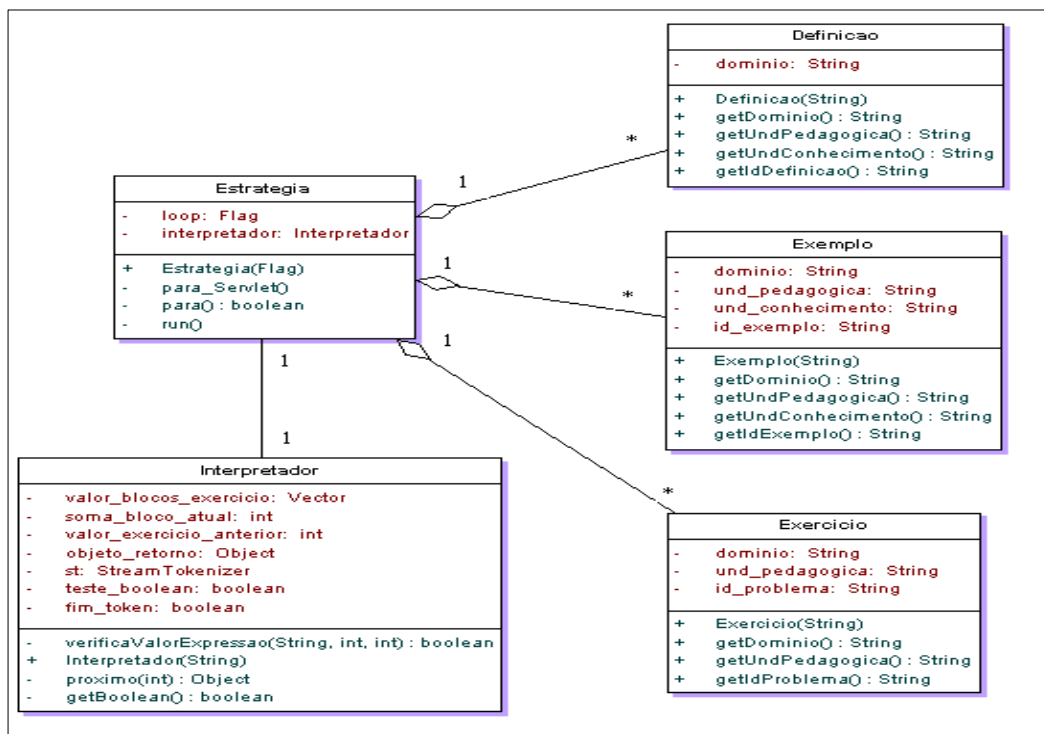


Figura 6.17 Diagrama de Classes 2 (Aplicação).

6.3.4 Diagrama de Implantação

Os diagramas de implantação tratam dos aspectos físicos de um sistema orientado a objetos. Mostram a visão estática da implantação de um sistema. Inclui os *nós*, que representam os dispositivos com capacidade de processamento. Graficamente um *nó* é representado por um cubo. Na Figura 6.18, vê-se as interações entre os componentes físicos do sistema e também a distribuição dos agentes em cada nó.

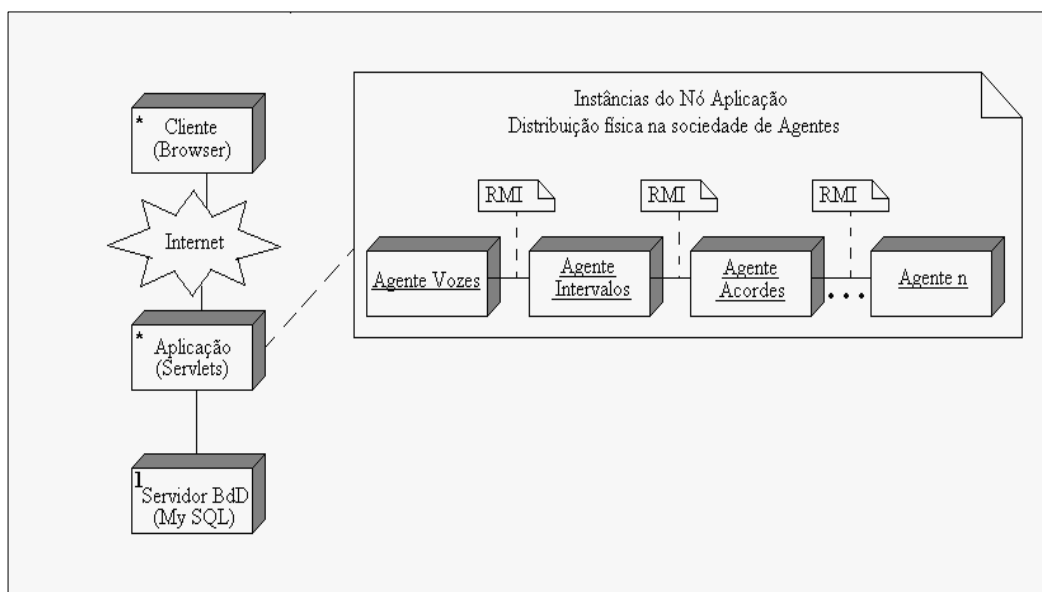


Figura 6.18 Diagrama de Implantação.

6.3.5 Diagrama de Entidade – Relacionamento (Banco de Dados)

1) Domínio: Este diagrama (Figura 6.19) mostra os tipos de relacionamento (não-hierárquicos) de cada subdomínio. Um domínio é formado de várias unidades pedagógicas (ver Capítulo 4 - visão interna do domínio). Cada unidade pedagógica é formada de mais de uma unidade de conhecimento, que é formada de vários problemas (exercícios). Um problema é composto de enunciado, solução, esclarecimento, dica e opção.

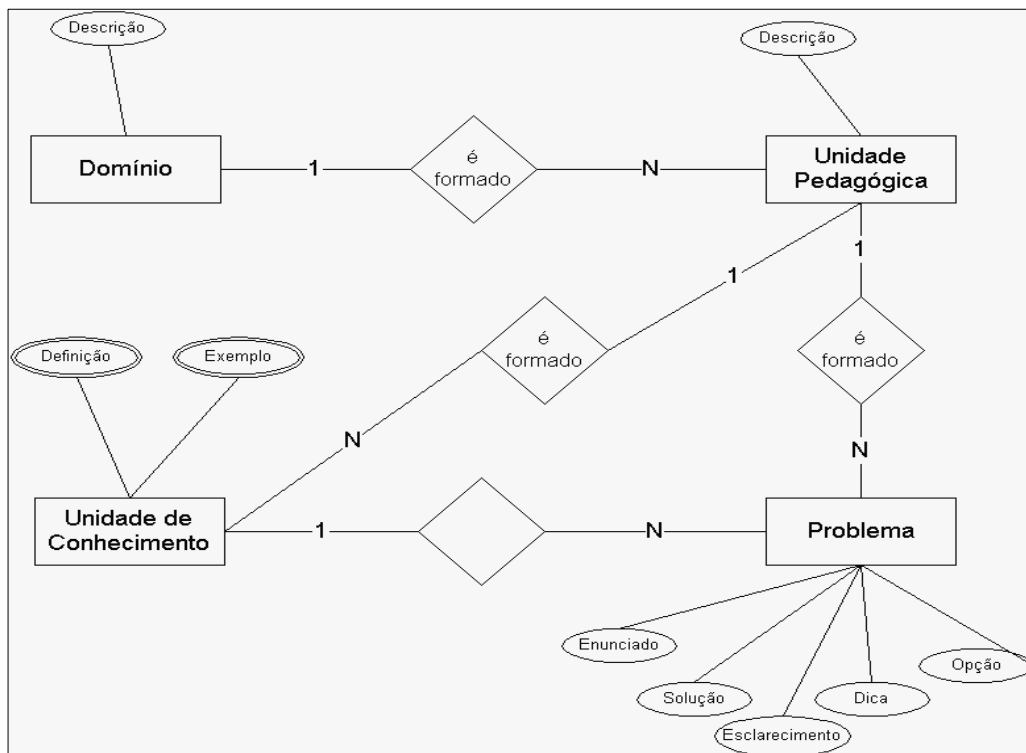


Figura 6.19 Diagrama de Entidade-Relacionamento (domínio).

2) Usuário: Este diagrama (Figura 6.20) mostra as relações entre as duas entidades do Banco de Dados (Perfil/Administrador).

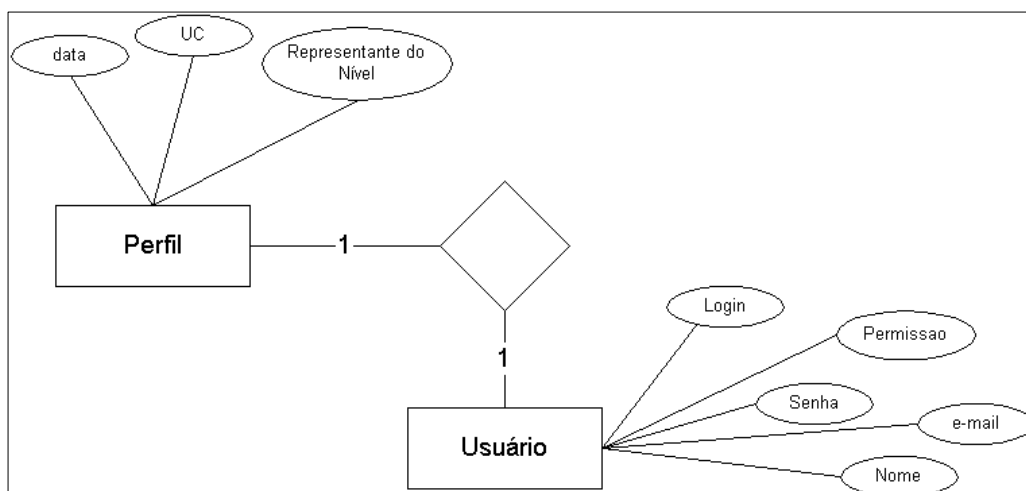


Figura 6.20 Diagrama de Entidade-Relacionamento (usuário).

6.4 Conclusão

A modelagem do sistema abordada neste capítulo, dá uma idéia abrangente de como podemos vê-lo em seus aspectos estáticos e dinâmicos, e o comportamento de cada parte do sistema num dado momento.

Capítulo 7

Aspectos de Implementação e Experimentação

7.1 Introdução

Este capítulo aborda os aspectos de implementação do sistema SHART-Web bem como dos experimentos feitos com usuários e a análise dos resultados.

7.2 Aspectos de Implementação

7.2.1 Sobre a linguagem e tecnologias Java

A linguagem Java vem se tornando o padrão para sistemas ou aplicações na Web. É uma linguagem orientada a objetos, baseada em classes, com recursos de programação concorrente e multi-plataforma. A linguagem Java especifica uma representação de *código de byte* para os programas compilados ao invés de uma representação binária. Este código é uma plataforma independente, transportável em rede, e pode ser executado numa “máquina virtual Java”, que é um programa executável disponível nas diversas plataformas existentes no mercado [Abrams 98].

Além da linguagem Java, foram utilizadas também na implementação do nosso sistema algumas tecnologias Java: *Servlets*, *JDBC* e *MySQL*.

Servlets:

Servlets são programas Java que são executados num servidor *Web* ou HTTP em resposta a um navegador (browser) *Web*. Este servidor usa uma “máquina virtual Java” para rodar o servlet e gerar uma página HTML. O servlet toma como entrada uma página HTML contendo *tags*, processa e retorna uma página HTML com os resultados [Caribou 2000].

JDBC (Java Data Base Connectivity):

É um padrão industrial para conectividade de bancos de dados independentes entre a plataforma Java e uma vasta gama de bancos de dados. A interface

de JDBC provê uma API para acesso a bancos de dados baseados em SQL (Structured Query Language) [JDBC 2001].

MySQL:

É um sistema cliente-servidor de gerenciamento de banco de dados, que consiste de um multi-concorrente servidor SQL. Conectividade, velocidade e segurança são requisitos que fazem de MySQL ideal para o acesso a banco de dados na Internet [MySQL 2001].

7.2.2 Ferramentas utilizadas

As ferramentas utilizadas para o desenvolvimento do sistema foram:

- CodeGuide 3.0 – utilizada para escrever o código-fonte e testes [Omnicores 2001];
- JDK 1.3 [Sun 2000];
- NetObjectsFusion 2.0.2 - utilizada no desenvolvimento do documento hipermídia [Netobjects 97];
- Enterprise Architect v2.00 - utilizada para a modelagem [SparxSystems 2001].

7.2.3 As Bases de Conhecimento

As bases de conhecimento envolvem as classes (com as definições dos conceitos) e as regras (bem-formadas e mal-formadas). A parte relativa às classes, aproveitamos o trabalho de [Teixeira 97], escrito na linguagem Prolog, e transportamos para Java acrescentando vários métodos e outras classes. A vantagem de se usar classes para definições de conceitos é que se tem um poder maior de generalização, porém ao se usar regras torna-se mais fácil fazer alterações ou acréscimos.

Utilizamos as regras com as características de restrições. Um exemplo, com duas regras tiradas de “Regras dos Intervalos Harmônicos Proibidos e Permitidos”:

2ª regra: Se existe intervalo harmônico = 5ª paralela (entre as vozes Soprano e Contralto, Contralto e Tenor, ou Tenor e Baixo), então “intervalo harmônico proibido”.

11ª regra: Se intervalo = “intervalo harmônico proibido” então retorne falso.

Quanto ao conteúdo didático, concebemos inicialmente um curso de Harmonia tradicional em 4 módulos, com uma parte implementada. Esta parte (módulo 1 ou nível 1) é o equivalente a um curso de 6 meses de harmonia, dado normalmente nas escolas de música. Dentro deste módulo 1 dividimos o conteúdo do curso em 10 itens, abrangendo desde a prática de vozes até os princípios básicos para se realizar um baixo dado.

7.2.4 Interface do Sistema

A seguir algumas telas capturadas:



Figura 7.1 Tela inicial do sistema.

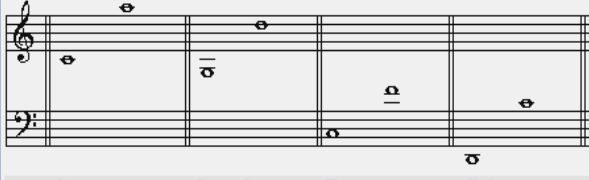
1. Vozes

2. Posição dos Acordes
3. Intervalos Melódicos Proibidos e Permitidos
4. Intervalos Harmônicos Proibidos e Permitidos
5. Condução das Vozes
6. Função Tonal dos Graus
7. Acordes Diatônicos
8. Dobramento e Supressão nos Acordes Diatônicos
9. Fraseologia e Cadências
10. Procedimentos para o Baixo Dado

Navegação Guiada Navegação Estática

1. Vozes

1.1 Extensão das vozes



Soprano Contralto Tenor Baixo

1.2 Distância entre as vozes

A distância entre Soprano e Contralto, e entre Contralto e Tenor, não deve ultrapassar uma 8ª. Entre Tenor e Baixo pode chegar no máximo a uma 12ª.



Figura 7.2 Definição e Exemplo sobre Vozes (navegação estática).

1. Conceitos Básicos de Vozes

2. Posição dos Acordes
3. Intervalos Melódicos Proibidos e Permitidos
4. Intervalos Harmônicos Proibidos e Permitidos
5. Condução das Vozes
6. Função Tonal dos Graus
7. Acordes Diatônicos no Estado Fundamental
8. Dobramento e Supressão dos Acordes Diatônicos
9. Fraseologia e Cadências
10. Procedimentos para o baixo dado

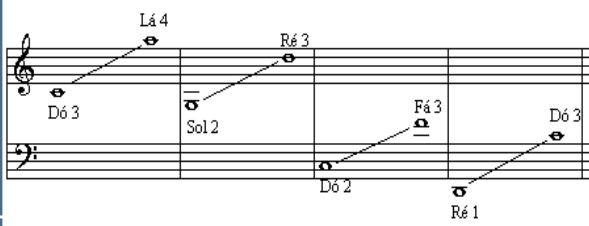
Navegação Guiada Navegação Estática

Assunto : Extensão das Vozes

Definição : Em Harmonia tradicional são quatro o número de vozes (ou partes) utilizadas: Soprano, Contralto, Tenor e Baixo. O termo voz significa neste caso que tanto pode ser cantada por uma voz humana ou tocada por um instrumento.

Assunto : Extensão das Vozes

Exemplo:



Proximo

Figura 7.3 Exposição sobre o assunto Vozes (navegação guiada).

1. Conceitos Básicos de Vozes

2. Posição dos Acordes

3. Intervalos Melódicos Proibidos e Permitidos

4. Intervalos Melódicos Proibidos e Permitidos

5. Condução das Vozes

6. Função Tonal dos Graus

7. Acordes Diatônicos no Estado Fundamental

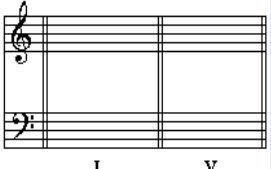
8. Dobramento e Supressão dos Acordes Diatônicos

9. Fraseologia e Cadências

10. Procedimentos para o baixo dado

Navegação Guiada Navegação Estática

Enunciado : Complete o pentagrama abaixo, com quatro vozes por compasso, levando em conta as regras de extensão entre as vozes baixo, tenor, contralto e soprano. Obs.: Reescreva todas as notas obedecendo a ordem baixo, tenor, contralto e soprano. DICA



D0
Natural
Oitava 1
Clave de Fa

Adicionar A

Figura 7.4 Exercício para o sistema validar resposta.

1. Conceitos Básicos de Vozes

2. Posição dos Acordes

3. Intervalos Melódicos Proibidos e Permitidos

4. Intervalos Melódicos Proibidos e Permitidos

5. Condução das Vozes

6. Função Tonal dos Graus

7. Acordes Diatônicos no Estado Fundamental

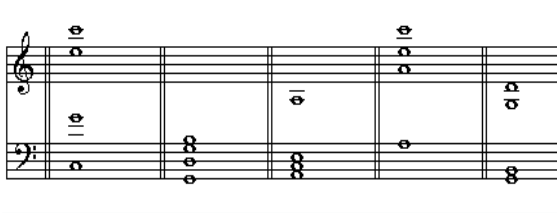
8. Dobramento e Supressão dos Acordes Diatônicos

9. Fraseologia e Cadências

10. Procedimentos para o baixo dado

Navegação Guiada Navegação Estática

Enunciado : Identifique a voz (Baixo, Tenor, Contralto, Soprano) que está com erro quanto a extensão (tessitura). DICA



Compasso 1 : Tenor, Contralto e Soprano ; Compasso 2 : Soprano ; Compasso 3 : Contralto e Soprano ; Compasso 4 : Tenor, Contralto e Soprano ; Compasso 5 : Tenor

Compasso 1 : Tenor e Soprano ; Compasso 2 : Soprano ; Compasso 3 : Contralto e Soprano ; Compasso 4 : Tenor, Contralto e Soprano ; Compasso 5 : Baixo

Compasso 1 : Tenor ; Compasso 2 : Soprano ; Compasso 3 : Contralto e Soprano ; Compasso 4 : Soprano ; Compasso 5 : Baixo

Figura 7.5 Exercício com dica disponível.

7.3 Experimentação

7.3.1 Objetivos

Nesta seção relata-se um experimento com o protótipo de sistema tutor desenvolvido no âmbito da presente dissertação, em cumprimento de um dos objetivos propostos no capítulo 1. Nesse sentido, foram feitos experimentos com usuários, conforme descritos mais adiante, visando a validação do sistema tutor. Estes experimentos tiveram como objetivos: por um lado verificar a facilidade de utilização (usabilidade) do sistema pelos usuários; por outro lado, testar as diversas funcionalidades do sistema, tais como: estratégias pedagógicas, validação de respostas, diagnóstico cognitivo, modo inspecionado, exibição de dicas e exercícios resolvidos, e exibição de histórico das interações do usuário com o sistema.

7.3.2 Metodologia Adotada

A metodologia adotada na pesquisa incluiu dois grupos de usuários e duas fases, com um questionário a ser respondido antes e depois de cada usuário acessar o sistema. O teor do questionário é o seguinte:

- Antes de acessar o sistema:
 - 1) Você conhece música?
 - 2) Você já frequentou ou frequenta alguma escola de música?
 - 3) Qual o seu interesse em Harmonia tradicional?
 - 4) Você já usou algum sistema tutor anteriormente?
- Depois de acessar o sistema:
 - 1) Relate as dificuldades enfrentadas durante o teste.
 - 2) Qual a facilidade de uso do sistema?
 - 3) Como está a interatividade do sistema?
 - 4) Quais suas sugestões para melhorar o sistema?

As atividades desenvolvidas com tais grupos ocorreram do seguinte modo: os membros do primeiro grupo acessaram simultaneamente o sistema no mesmo local (servido por uma rede de computadores) e o testaram por cerca de duas horas, enquanto que os membros do segundo grupo acessaram o sistema individualmente em locais e horários diferentes sem um tempo delimitado.

7.3.3 Sujeitos da Pesquisa

Os sujeitos da pesquisa foram divididos em dois grupos: no primeiro grupo os sujeitos da pesquisa foram 5 alunos de uma Escola de Artes (Escola João Pernambuco) da rede municipal de ensino da Prefeitura da Cidade de Recife (PCR), todos com nível de conhecimento razoável em teoria musical e com pouca ou nenhuma experiência em usar o computador. Os dados coletados do primeiro grupo foram respondidos num questionário impresso e fizemos um acompanhamento de cada membro do grupo durante o teste para eventuais esclarecimentos. No segundo grupo, os sujeitos da pesquisa foram usuários (previamente escolhidos) da Internet acessando o sistema de suas residências ou locais de trabalho, sendo estes usuários já experientes no domínio da música e da Internet/Web. Os dados do segundo grupo foram coletados via *e-mail*.

7.3.4 Análise dos Resultados

A análise da pesquisa obtida com as atividades dos dois grupos estudados aponta essencialmente para os seguintes resultados:

- 1) Com respeito às condições de uso do sistema, verificou-se nos dois grupos estudados que o sistema correspondeu relativamente bem às expectativas, onde apenas 15% do total dos usuários apresentou alguma dificuldade no uso do sistema. Isso poderia ser superado se o sistema já contivesse um bom editor musical acoplado, assim como um sistema de *help* bem definido.
- 2) Com relação à análise dos serviços pedagógicos constatamos que:
 - O sistema apresentou de forma aceitável a seqüência do material didático que compõe uma estratégia pedagógica (definição, exemplo, exercício).
 - A validação de resposta (tanto do tipo que é buscada no banco de dados como do tipo que é feito inferência) pelo sistema obteve resultados convincentes, ou seja, o sistema detecta e exhibe na tela se a resposta está certa ou errada.
 - O diagnóstico cognitivo (que é dado sempre que o aluno responde a pelo menos um exercício) apresentou bem as pontuações de cada aluno (o saldo foi mais positivo no segundo grupo).

- O modo inspecionado (opção dada ao aluno quando o sistema apresenta outro exercício do mesmo tipo do que foi respondido com erro) funcionou também satisfatoriamente. É necessário também para se fazer este teste induzir o aluno a cometer erros para ser exibida esta opção.
- O suporte dado ao aluno (dicas e exercícios resolvidos) quando este está resolvendo um determinado exercício, bem como o histórico das interações do aluno com o sistema (revisão) foi exibido adequadamente.

7.4 Conclusão

Neste capítulo foram vistos aspectos referentes à solução para implementação do protótipo de sistema, destacando-se as escolhas de ferramentas e os critérios para tais escolhas. Na outra parte do capítulo, foram discutidos o experimento, sua análise e suas condições de realização. As críticas e sugestões obtidas da pesquisa com os dois grupos de usuários, serviram para recomendar uma revisão nos recursos do sistema, notadamente, a necessidade de se investir mais no desenvolvimento de um editor musical que possa ser acoplado ao sistema para facilitar a manipulação de objetos gráficos musicais pelos usuários.

Capítulo 8

Conclusões e Trabalhos Futuros

8.1 Considerações Finais

Apresentamos neste trabalho, o SHART-Web: um Sistema Tutor Inteligente com hipermídia na Web, voltado para o ensino de harmonia tradicional, com base numa abordagem multiagentes.

Neste ponto, podemos nos referir a cada um dos objetivos específicos da nossa pesquisa, declarando que todos foram realizados. Modelamos e implementamos o Sistema Tutor Multiagentes com as características definidas na proposta. Dentro deste sistema, em particular, concretizamos cada um dos seus componentes: módulo do Especialista (que teve aqui a restrição de não ter sido programado para resolver problemas colocados pelo aprendiz), módulo do Tutor, módulo de modelagem do Aprendiz. Além disso, implementamos, ainda que com uma certa simplicidade, as interfaces entre aprendizes e sistema, professores e sistema. Fizemos adequadamente e de forma personalizada a integração do Sistema Tutor com o material didático disposto num documento hipermídia.

Vivenciamos com isto, uma experiência gratificante principalmente por termos chegado a uma implementação do sistema, e termos tido também a oportunidade de experimentar o sistema com turmas de alunos, recebendo retorno destes principalmente no que diz respeito à utilidade e à usabilidade do sistema. Isso nos permite afirmar ter cumprido todas as etapas previstas na nossa pesquisa, obtendo resultados bastante satisfatórios no que diz respeito aos objetivos propostos inicialmente.

8.2 Trabalhos Futuros

A partir da contribuição de nosso trabalho, podemos traçar novas metas que podem ser realizadas. Entre estas estão:

- 1) Ampliar a base de conhecimento do domínio da harmonia tradicional, acrescentando mais classes e regras (ampliando com isto o conhecimento musical do sistema), de modo que torne-se possível o aluno propor problemas para o sistema resolver.
- 2) Criar um editor musical para que o aprendiz-usuário possa ter mais recursos na manipulação objetos gráficos musicais.
- 3) Criar um editor para que cada professor possa montar seu curso de harmonia tradicional personalizado, com estratégias pedagógicas, duração e número de alunos definidos.
- 4) Agregar ao sistema, um conjunto de ferramentas de comunicação síncronas e assíncronas que permitam aos alunos interagirem entre si e com os professores. Tais ferramentas já podem ser obtidas, por exemplo, dos resultados gerados no projeto MathNet, já desenvolvidas e atualmente em fase de teste.

Referências Bibliográficas

[Abrams 98] Abrams, M. "World Wide Web - Beyond the Basics", Prentice Hall, 1998, <http://www.prenhall.com/abrams>

[Auberger 98] Auberger, M. "Student Modeling in Educational Multimedia Titles Using Agents", <http://aif.wu-wien.ac.at/usr/geyers/>, 1998.

[Auralia 2001] <http://www.risingsoftware.com/auralia21/networking.shtml>, 2001.

[Balaban 92] Balaban, M. "Music Structures: Interleaving the Temporal and Hierarchical Aspects in Music", em "Understanding Music with AI: Perspectives on Music Cognition", AAAI Press, 1992.

[Baker 92] Baker, M. J. "Design of an Intelligent Tutoring System for Musical Structure and Interpretation", em "Understanding Music with AI: Perspectives on Music Cognition", AAAI Press 1992.

[Band-in-a-Box 2001] <http://www.pgmusic.com/demodwn.htm>, 2001.

[Beck et alii 96] Beck, J. ; Stern, M. ; Haugsjaa, E. "Applications of AI in Education", ACM Crossroads Student Magazine, 1996.
<http://www.acm.org/crossroads/xrds3-1/aied.html>

[Benjamin et alii 98] Benjamin, T. ; Horvit, M. ; Nelson, R. "Tonal Music: Techniques et Materials of Tonal Music", Wadsworth Publishing Company, 1998.

[Booch 2000] Booch, G. , Rumbaugh, J. , Jacobson, I. "UML - Guia do Usuário", Editora Campus, 2000.

[Bracewell et alii 98] Bracewell, R. ; Breuleux, A. ; Laferrière, T. ; Benoit, J. ; Abdous M. "The Emerging Contribution of Online Resources and Tools to Classroom Learning and Teaching", 1998.
<http://www.tact.fse.ulaval.ca/ang/html/review98.html>

[Brisolla 79] Brisolla, C. M. “Princípios de Harmonia Funcional”, Editora Novas Metas, São Paulo –SP, 1979.

[Caminha 2000] Caminha, A. O. “MHITS – Um Sistema Tutor Inteligente em Harmonia Musical”, Dissertação de Mestrado”, Universidade Federal da Paraíba, Campina Grande, 2000.

[Caribou 2000] “Java Servlets and Applets”, Caribou Lake Software, Inc., http://www.cariboulake.com/techinfo/servlet_applet.html

[Caron 99] Caron, D. “Cours d’harmonie tonale”, Université de Montréal, <http://www.musique.umontreal.ca/technique/harmonie/index.html>, 1999.

[Costa 97] Costa, E. de B. “ Um Modelo de Ambiente Interativo de Aprendizagem Baseado numa Arquitetura Multi-Agentes” , Tese de Doutorado, Universidade Federal da Paraíba, Campina Grande, 1997.

[Costa 2000] Costa, E. de B. “ MathNet: Uma Abordagem via Sistemas Multiagentes para Concepção e Realização de Ambientes Interativos de Aprendizagem Cooperativa Assistidos por Computador. In Revista da Sociedade Brasileira de Informática na Educação, abril de 2000 (publicado como resumo).

[Dannenberg 87] Dannenberg, R. “Music Understanding”, Computer Science Research Review, Carnegie Mellon School of Computer Science, pp 19-28, 1987.

[Dannenberg 93] Dannenberg, R. “Music Representation Issues, Techniques, and Systems” , Computer Music Journal, Vol. 17 n° 3, MIT Press, 1993.

[Dannenberg et alii 97] Dannenberg, R. ; Desain, P. ; Honing, H. “Programming Language Design for Music”, <http://www.nici.kun.nl/mmm/papers/ddh-97-a.html>

[Davies 97] Davies, J. E. “Multimedia and Hipermedia as Mindtools”. <http://www.quasar.ualberta.ca/edpy485/mmedia/hypmind.htm> 1997.

[Durfee & Rosenschein 94] Durfee, E. ; Rosenschein, J. “Distributed Problem Solving and Multi-Agent Systems: Comparisons and Examples”. 13th International

Workshop on Distributed Artificial Intelligence, Washington, July 17-19, 1994.

[Ear Master 2001]

<http://www.hitsquad.com/smm/programs/EarMaster/download.shtml>, 2001.

[Ebcioglu 92] Ebcioglu, K. “An Expert System for Harmonizing Chorales in the Style of J. S. Bach”, capítulo 12 de “Understanding Music with AI: Perspectives on Music Cognition, AAAI Press, 1992.

[Ferber 99] Ferber, Jacques “Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence”. Addison-Wesley, 1999.

[Flores-Mendez 2000] Florez-Mendez, R. A. “Towards a Standardization of Multi-Agent System Frameworks”.

<http://www.acm.org/crossroads/xrds5-4/multiagent.html>, 2000.

[Fritsch & Viccari 95] Fritsch, E. ;Viccari, R. M. “SETMUS: uma ferramenta computacional para o ensino da Música”, II Simpósio Brasileiro de Computação e Música”, Canela – RS, 1995.

[Gang et alii 97] Gang,D. ; Lehmann,D. ; Wagner,N. “Harmonizing Melodies in Real-Time: the Connectionist Approach”, Proceedings of the International Computer Music Conference, Thessaloniki, Setembro, 1997.

[Genesereth & Fikes 94] Genesereth, M. ;Fikes, R. “Knowledge Interchange Format Version 3.0 ,1994.

<http://logic.stanford.edu/kif/Hypertext/kif-manual.html>

[Genesereth & Ketchpel 94] Genesereth, M. R. ; Ketchpel, S. P. “Softwares Agents” Communications of ACM, pp 48-53, 1994.

[Giarratano & Riley 89] Giarratano,J. ; Riley,G. “Expert Systems Principles and Programming” PWS-KENT Publishing Company Boston, 1989.

[Giardina 92] Giardina, M. “Interactive Learning Environments”. Springer-Verlag, 1992.

[Glanzmann 95] Glanzmann, J. H. “Expert Piano: Um Ambiente de Auxílio à Aprendizagem Musical”. Dissertação de Mestrado, UFRJ Rio de Janeiro, 1995.

[Greer & McCalla 91a] Greer, J. E. ; McCalla, G. I. “Student Modelling: The Key to Individualized Knowledge-Based Instruction”, Capítulo 1, NATO ASI Series, vol. 125, 1991.

[Greer & McCalla 91b] Greer, J. E. ; McCalla, G. I. “Student Modelling: The Key to Individualized Knowledge-Based Instruction”, Capítulo 2, NATO ASI Series, vol. 125, 1991.

[Halff 88] Halff, H. M. “Curriculum and Instruction in Automated Tutors” , em “Foundations of Intelligent Tutoring Systems” Eds. Polson, M. & Richardson, J. J. Lawrence Erlbaum Associates Publishers, New Jersey, 1988.

[Holland 94] Holland, S. “Learning about Harmony with Harmony Space: An Overview”, in “Music Education: An Artificial Intelligence Approach” , Springer-Verlag, London, 1994.

[Holland 2000] Holland, S. “Artificial Intelligence in Music Education: A Critical Review” , Capítulo 13 de Readings in Music and Artificial Intelligence, Harwood Academic Publishers, 2000.

[Honing 93] Honing, H. “Issues in the representation of time and structure in music”, Contemporary Music Review, 9(221-239), 1993.

[Howell et alii 91] Howell, P. ; West, R. ; Cross, I. “Representing Musical Structure”, Academic Press, 1991.

[JDBC 2001] <http://java.sun.com/products/jdbc/>

[KBS 99] “KBS Publications Sorted by Area”,
<http://www-kbs.ai.uiuc.edu/kbs/publications-area-main.htm> 1999.

[Kearsley 87] Kearsley, G. “Artificial Intelligence and Instruction: Applications and Methods”. Addison-Wesley Publishing Company, Inc., 1987.

[Knapik & Johnson 98] Knapik, M. ; Johnson, J. “Developing Intelligent Agents for Distributed Systems”, McGraw-Hill, 1998.

[Kommers et alii 96] Kommers, P. ; Grabinger,S. ; Dunlap, J. “Hipermedia Learning Environments – Instructional Design and Integration”, Lawrence Erlbaum Associates, Inc., 1996.

[Maes 91] Maes, P. “The Agent Network Architecture” , *Proceedings of th 1991 AAAI Spring Symposium on Integrated Intelligent Architectures*, AAAI Press, Stanford, 1991.

[McLaughlin & Oliver 95] Mclaughlin,C. ; Oliver,R. “Who Is In Control? Defining Interactive Learning Environments” Edith Cowan University, 1995.

<http://www.ascilite.org.au/conferences/melbourne95/smtu/papers/mcloughlin.pdf>

[Miranda 90] Miranda, E. R. “Música e Inteligência Artificial (Paradigmas e Aplicações)”, CPGCC-UFRGS, Porto Alegre, 1990.

[Miranda 98] Miranda, E. R. “An Introduction to Music and Artificial Intelligence”, <http://website.lineone.net/~edandalex/ai-essay.htm> ,1998.

[Moore 90] Moore, F. R. “Elements of Computer Music”, University of California, Prentice Hall, 1990.

[Murphy 97] Murphy, E. , “Computer-based Learning Environment”, <http://www.stemnet.nf.ca/~elmurphy/emurphy/computers.html> , 1997.

[Murray 99] Murray, T. “Authoring Intelligent Tutoring Systems : An Analysis of the State of the Art”. *Journal of Artificial Intelligence and Education*, vol. 10, 1999.

[MySQL 2001] What is MySQL ?
<http://www.mysql.com/information/index.html>

[Netobjects 97] NetObjectsFusion 2.0.2 1995-1997
<http://www.netobjects.com>

[Nwana 96] Nwana, H. "Software Agents: An Overview", Knowledge Engineering Review, Vol. 11, N. 3 1996.

[Omnicores 2001] Omnicores Software <http://www.omnicore.com>

[Padgett 97] Padgett, D. "Hipermedia Definitions and Components", <http://www.auburn.edu/academic/education/eflt/hm.html> 1997.

[Paiva 95] Paiva, A. M. "About User and Learner Modeling: An Overview", 1995, <http://www.cbl.leeds.ac.uk./amp/personal.html>

[Pérez et alii 95] Pérez, M. J. ; Fornari, J. E. "Reconhecimento de timbres musicais através da rede neural auto-organizável de Kohonen", II Simpósio Brasileiro de Computação e Música, Canela - R.S. ,1995.

[Piston 83] Piston, W. "Harmony" , Norton & Company, Inc. , 1983.

[Repsold 93] Repsold, M. "O Computador e a Educação Musical : Transformação ou Conservadorismo", Anais do Encontro Brasil-França, Informática na Educação, Rio de Janeiro, RJ , 1993.

[Rich 91] Rich, Elaine "Artificial Intelligence", McGraw-Hill, Inc., 1991.

[Roads 85] Roads, C. "Research in Music and Artificial Intelligence Computing Surveys", vol. 17, n°2, June 1985.

[Roads 96] Roads, C. "The Computer Music Tutorial", MIT Press, 1996.

[Russel & Norvig 95] Russel, S. ; Norvig, P. "Artificial Intelligence: A Modern Approach". Prentice Hall, 1995.

[Scherer & Schlageter 95] Scherer, R. ; Schlageter, G. ; "Multi-Agent Approach for Integration of Neural Networks and Expert Systems", John Wiley & Sons, 1995.

[Schwier & Misanchuk 93] Schwier, M. ; Misanchuk, E. "Interactive Multimedia Instruction", Educational Technology Publications, N.J. , 1993.

[Selfridge-Field 97] Selfridge-Field, E. "Beyond MIDI The Handbook of Musical Code", MIT Press, 1997.

[Smith 2000] Smith, B. "Artificial Intelligence and Music Education", Capítulo 12 de "Readings in Music And Artificial Intelligence", Harwood Academic Publishers, 2000.

[SparxSystems 2001] <http://www.sparxsystems.com.au>

[Stefik 95] Stefik, M. "Introduction to Knowledge Systems", Morgan Kaufman, Inc. 1995.

[Steiner 95] Steiner, V. "What is Distance Education ? "
<http://www.dln.org/library/dl/whatis.html>, 1995.

[Stone & Veloso 97] Stone, P. ; Veloso, M. "Multiagent Systems: A Survey from a Machine Learning Perspective", <http://www.cs.cmu.edu/~pstone/pstone-papers.html> 1997.

[Sun 2000] <http://www.sun.com>

[Swanwick 79] Swanwick, K. "A Basis for Music Education", London: Routledge, 1979.

[Teixeira 97] Teixeira, L. M. "Da representação do conhecimento musical ao esboço conceitual de uma sociedade de agentes em Harmonia", Dissertação de Mestrado, Universidade Federal da Paraíba, Campina Grande, 1997.

[Urban-Lurain 96] Urban-Lurain, Mark - "Intelligent Tutoring Systems: An Historic Review in the Context of the Development of Artificial Intelligence and Educational Psychology", <http://aral.cse.msu.edu/Publications/ITS/its.htm>, 1996.

[Weiss 99] Weiss, Gerhard "Multiagent Systems: Modern Approach to Distributed Artificial Intelligence", MIT Press, 1999.

[Wenger 87] Wenger, E. "Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approachs to the Communication of Knowledge", Morgan Kaufmann Publishers, Inc.,1987.

[Wiggins et alii 93] Wiggins, G. ; Miranda, E. ; Smaill, A. ; Harris, M. "A Framework for the Evaluation of Music Representation Systems" , Computer Music Journal, Vol. 17 n° 3, MIT Press, 1993.

[Wilson 96] Wilson, B. G. "Construtivist Learning Environment: Case studies in instructional design". Educational Technology Publications, 1996.

[Wooldridge & Jennings 95] Wooldridge, M. ; Jennings, N. "Intelligent Agents: Theory and Practice",
<http://www.ai.univie.ac.at/~paolo/lva/vu-sa98/html/ker95-html.html> ,1995.

[Woolf 88] Woolf, B. "Intelligent Tutoring Systems: A Survey", Morgan Kaufmann Publishers, Inc., 1988.