

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
PRODUÇÃO**

**Desenho de um Sistema Gerenciador Inteligente de Recursos em
um ambiente de Aprendizagem Cooperativa**

Ricardo José Rocha Amorim

Dissertação apresentada ao Programa
de Pós-Graduação em Engenharia de
Produção da Universidade Federal de
Santa Catarina como requisito parcial
para a obtenção do título de Mestre em
Engenharia de Produção

Florianópolis

2002

Ricardo José Rocha Amorim

**DESENHO DE UM SISTEMA GERENCIADOR INTELIGENTE DE
RECURSOS EM UM AMBIENTE DE APRENDIZAGEM
COOPERATIVA**

Esta dissertação foi julgada adequada para a obtenção do título de **Mestre em Engenharia de Produção no Programa de Pós -Graduação em Engenharia de Produção** da Universidade Federal de Santa Catarina.

Florianópolis, 11 de Setembro de 2002.

Prof. Edson Pacheco Paladini, Doutor
Coordenador do Curso

BANCA EXAMINADORA

Prof^a Édis Mafra Lapolli, Dr^a
Orientadora

Prof^a Ana Maria Benciveni Franzoni , Dr^a

Prof. Francisco Pereira Fialho, Dr.

Prof. Luiz Ricardo Uriarte, M. Eng.
Tutor de orientação

A minha família, em especial meus pais Juracy e Gicia,

A minha esposa Dinani, pelo amor, dedicação e
paciência,

Aos meus queridos filhos Natalia e Tiago como
estímulo aos novos desafios que virão e pela
alegria que sempre me proporcionam.

Agradecimentos

Aos meus pais, pela convicção de mostrar-me a importância da educação na vida das pessoas;

A minha querida esposa Dinani pelo amor, carinho, dedicação e, principalmente, compreensão nos momentos em que estive ausente;

A meus orientadores, Édis Mafra Lapolli e Ricardo Uriarte, que foram grandes amigos e souberam me levar pelos caminhos desta pesquisa;

Aos Professores Francisco Fialho e Ana Franzoni, por aceitarem participar da minha banca examinadora, além das contribuições e sugestões que enriqueceram minha dissertação;

A UFSC, em especial aos Professores e membros do LED e ao Instituto Anísio Teixeira que me proporcionaram a oportunidade de fazer este mestrado;

A UNEB, pela oportunidade de fazer este curso e de ter me proporcionado apoio financeiro com uma bolsa parcial de mestrado, e a todos os amigos do Campus VII, em especial à Professora Norma Leite que graças ao seu apoio foi possível a minha inscrição neste curso;

A todos os amigos da Universidade de Santiago de Compostela – Espanha especialmente aos Professores Dr. Senén Barro (Magnífico Reitor), Dr. Vila Sobriño, Dr. Eduardo Sánchez e o Dr. Manuel Penín pela atenção especial dispensada a este trabalho;

A Faculdade de Ciências da Administração de Petrolina, em especial ao Professor Valdenor Ramos, que sem o seu apoio não teria sido possível obter orientações por vídeo-conferência.

Sumário

Lista de Figuras.....	p. 07
Lista de Quadros.....	p. 08
Lista de Termos e Siglas.....	p. 09
1 INTRODUÇÃO.....	p. 12
1.1 Justificativa.....	p. 13
1.2 Perguntas da pesquisa.....	p. 15
1.3 Objetivos do trabalho.....	p. 15
1.4 Metodologia.....	p. 16
1.5 Estrutura da dissertação.....	p. 16
2 TEORIAS DE APRENDIZAGEM.....	p. 18
2.1 Construtivismo de Bruner.....	p. 18
2.2 Epistemologia Genética de Piaget.....	p. 19
2.3 Teoria do Desenvolvimento Sócio-Cultural de Vygotsky.....	p. 20
2.4 Teoria dos Operadores Condicionantes.....	p. 21
2.5 Teoria da Cognição Situada.....	p. 21
2.6 Teoria da Instrução Ancorada.....	p. 21
2.8 Formas de uso do computador na educação.....	p. 29
2.7 O que é Aprendizagem Cooperativa.....	p. 22
3 MODELOS DE CONHECIMENTO E METODOLOGIAS PARA O DESENVOLVIMENTO DE SISTEMAS BASEADOS EM CONHECIMENTO.....	p. 31
3.1 O que é Metodologia.....	p. 31
3.2 Aspectos Gerais das Metodologias Baseadas em Conhecimento.....	p. 34
3.3 CommonKADS.....	p. 36
4 ASPECTOS RELACIONADOS À APRENDIZAGEM COOPERATIVA APOIADA POR COMPUTADOR.....	p. 43
5 O MODELO PROPOSTO SEGUNDO A METODOLOGIA CommonKADS.....	p. 53
5.1 Modelo de Organização.....	p. 53
5.2 Modelo de Agentes.....	p. 60
5.3 Modelo de Comunicação.....	p. 65

5.4	Modelo de Conhecimento.....	p. 67
5.5	Modelo de Desenho.....	p. 74
6	CONCLUSÕES.....	p. 86
	Fontes Bibliográficas.....	p. 89

Lista de figuras

Figura 1:	A pirâmide metodológica.....	p.32
Figura 2:	Processo de desenvolvimento de um SBC segundo as metodologias baseadas em conhecimento.....	p. 36
Figura 3:	O conjunto de modelos do CommonKADS.....	p. 38
Figura 4:	Entidades de conhecimento identificadas no MoP.....	p. 41
Figura 5:	Estrutura e Pessoas na situação atual.....	p. 56
Figura 6:	Descrição do ambiente docente com equipamentos integrados a uma rede.....	p. 57
Figura 7:	Processos na situação atual.....	p. 59
Figura 8:	Modelo de tarefas correspondentes à tarefa de ensino realizada pelo professor.....	p. 69
Figura 9:	Modelo de tarefas do SGIR correspondentes a sua interação com o aluno ou grupo.....	p. 71
Figura 10:	Modelo de tarefas do SGIR correspondentes a sua interação com o professor.....	p. 74
Figura 11:	Visão de alto nível da arquitetura multi-camada.....	p. 77
Figura 12:	Arquitetura de software completa com todos os componentes.....	p. 80
Figura 13:	Arquitetura interna de um RMS genérico.....	p. 81
Figura 14:	Arquitetura distribuída do software em uma rede.....	p. 84

Lista de quadros

Quadro 1	Teorias de Aprendizagem.....	p. 22
Quadro 2	Framework para estudo de ambientes de aprendizagem cooperativa.....	p. 45
Quadro 3a	Classificação dos ambientes segundo framework proposto.....	p. 50
Quadro 3b	Classificação dos ambientes segundo framework proposto.....	p. 51
Quadro 3c	Classificação dos ambientes segundo framework proposto.....	p. 52
Quadro 4:	Contexto organizacional do sistema com prováveis soluções.....	p. 54
Quadro 5:	Descrição da área foco na organização.....	p. 55
Quadro 6:	O agente professor: sua função e a relação com outros agentes envolvidos.....	p. 62
Quadro 7:	O agente alunos: sua função e a relação com outros agentes envolvidos.....	p. 63
Quadro 8:	O agente SGIR: sua função e a relação com outros agentes envolvidos.....	p. 65

Lista de termos e siglas

CSCCL – Computer Supported Colaborative Learning.

CSCW – Computer Supported Colaborative Work.

C++ - Linguagem de programação orientada a objetos.

Freeware – Softwares gratuitos.

Java - Linguagem de programação orientada a objetos.

KQML – Knowledge Query and Manipulation Language. Linguagem formulada com tag's para consulta e manipulação de conhecimento.

Middleware – Refere-se a conectividade de software e consiste de um conjunto de serviços que permitem que múltiplos processos sejam executados em uma ou mais máquinas interagindo entre si.

MySQL – Banco de Dados de código aberto.

Red Hat Linux – Sistema operacional de código aberto produzido por Red Hat Inc. Red Hat é marca registrada de Red Hat Incorp. e Linux é marca registrada de Linus Torvalds.

XML – Extensible Markup Language. Linguagem formulada com tag's.

Windows CE – Sistema operacional para computadores de mão produzido por Microsoft Corporation. Windows é marca registrada de Microsoft Corp.

Resumo

AMORIM, Ricardo José Rocha. **Desenho de um Sistema Gerenciador Inteligente de Recursos em um ambiente de aprendizagem cooperativa** . 2002. 95f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, UFSC, Florianópolis.

Este trabalho descreve um modelo de um Sistema Gerenciador Inteligente de Recursos a ser utilizado em um ambiente de aprendizagem cooperativa presencial, desenvolvido a partir da metodologia CommonKADS, que foi aplicada para identificar os agentes presentes no ambiente, modelar as interações entre eles e obter o modelo de tarefas que fundamenta a arquitetura de software do sistema.

O ambiente foi pensado para servir de apoio às aulas presenciais e consiste de um sistema computacional que envolve um conjunto de computadores de mão – *Personal Digital Assistant* (PDA), interligados através de um sistema de rede sem fio, um equipamento para a digitalização do que é escrito no quadro branco e um projetor multimídia, apoiados por um sistema de banco de dados multimídia com ferramentas para armazenamento e recuperação de informações. O PDA, nesse ambiente, é para funcionar como um elemento facilitador na aplicação de técnicas didáticas que promovam a aprendizagem cooperativa e para auxiliar a interação entre professor e aluno.

No trabalho, os fundamentos teóricos em relação à metodologia CommonKADS e aos ambientes de aprendizagem cooperativa são consolidados, a infra-estrutura necessária para dar suporte aos ambientes educacionais é avaliada de forma que permita, em um trabalho futuro, a implantação de um ambiente capaz de suportar as atividades de aprendizagem com base na cooperação. Ao final é feita uma revisão na literatura sobre ambientes de aprendizagem cooperativa, alguns ambientes são avaliados e, por último, um modelo é proposto.

Palavras-chave: aprendizagem cooperativa, modelo de tarefas, arquitetura de *software*.

Abstract

AMORIM, Ricardo José Rocha. **Desenho de um Sistema Gerenciador Inteligente de Recursos em um ambiente de aprendizagem cooperativa** . 2002. 95f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, UFSC, Florianópolis.

This work describes a model of an Intelligent Resource Management System to fit on a presential cooperative learning environment using the CommonKADS methodology to identify the agents presents in it, and modeling the interactions that occurs with them and to obtain the tasks model that represents the software architecture fundamentals of this system.

The system was imagined to support learning activities in presential classes. It consists of a computing system with Personal Digital Assistant (PDA), connected by a wireless network, an whiteboard digitalyzer, which acquires what is written on the whiteboard and saves it as an image file, and a multimedia projector, supported by database multimedia systems for informations save and query requests.

PDA, in this one, is intended to facilitate the application of a cooperative learning techniques and to handle professor-student interaction. In this approach, pedagogical aspects and technical requirements needed to support cooperative learning in presential classes are discussed.

In this work, the theoretical fundamentals about CommonKADS methodology and cooperative learning systems are confirmed, the infra-structure required to support educational environments is evaluated to grants, in future work, the implementation of environments what handles learning activities based in cooperation. At the final of this work, a literature reviews about cooperative learning environments are made, some environments are evaluated and, at the end, an model is suggested.

Key-words: cooperative learning, task model, software architecture.

1 INTRODUÇÃO

1.1 Contextualização

Na sociedade globalizada, um mundo em constante mudança, as instituições sentem a necessidade de desenvolver a capacidade de migrar, de mudar, desenvolver novas competências, de adaptar-se às novas situações e de aprender .

Na sociedade do conhecimento, acredita-se que as relações com a educação tornam-se cada vez mais voltadas ao aprender contínuo, ao compartilhamento de saberes e a produção coletiva de conhecimentos. As competências adquiridas no início de um percurso profissional e acadêmico estarão obsoletas ao longo do tempo – surge uma nova relação do homem com o conhecimento: a navegação e o “*surf*”, onde a Internet representa a metáfora emergente nessa relação (LEVY, 1999).

Nesse sentido, pode-se fazer uma reflexão profunda deste tema tendo em vista a análise feita por Leslie Paas (PAAS, 1999), descrita a seguir:

- Atualmente, com a prevalência de computadores e redes, “a informação é transparente. Ninguém gera vantagens competitivas ao longo prazo só porque sabe algo que uma outra pessoa não sabe.” (RAMO, 1997) ;
- As mudanças na educação estão sendo alimentadas pelas novas necessidades e tendências no mercado de trabalho;
- A crescente demanda de educação vem acompanhada por críticas ao método pedagógico tradicional.

O sistema educacional nesse cenário precisa estar sintonizado com as tecnologias de informação: elas exercem a função de disseminadoras de conhecimento, liberando estudantes e professores das limitações de tempo e espaço, enriquecendo o ensino com recursos de multimídia/hipermídia, interação, simulações e permitindo o estudo em grupo, de forma cooperativa, ou individualizada.

Neste trabalho é proposto um modelo de um Sistema Gerenciador Inteligente de Recursos a ser utilizado em um ambiente de aprendizagem cooperativa presencial apoiado por computadores de mão. Para isto, aplica-se a metodologia CommonKADS a partir da qual obtém-se parte do modelo de conhecimento utilizado pelos agentes do ambiente e o modelo de tarefas que fundamentam a arquitetura software do sistema.

1.2 Justificativa

Atualmente, existem vários sistemas computacionais que se propõem a dar suporte ao processo educativo e que utilizam a internet como meio de interação entre professores e alunos (FERRAZ et Al, 2000 e FUKS, 2000); Existem diversos tipos de softwares de autoria para a produção de mídia de apoio aos conteúdos e vários são os softwares que empregam abordagens construtivistas. É fácil encontrar escolas bem aparelhadas com recursos como retro-projetor, *data show*, televisor, vídeo cassete e laboratórios de informática.

Uma parte importante da aprendizagem acontece quando todas as tecnologias são integradas: as telemáticas, as audiovisuais, as textuais, as orais, musicais, corporais. O professor tem um grande leque de opções metodológicas, de possibilidades de organizar sua comunicação com os alunos, de introduzir um tema, de trabalhar com os alunos de forma presencial e virtualmente, de avalia-los (MORAN, 1997).

O uso de tecnologias de comunicação e informação em sala de aula ainda é visto como algo excepcional: o aparato tecnológico mais utilizado hoje em dia, é restrito à mesa do professor, quadro negro, giz (ou quando muito, pincel) – os mesmos recursos utilizados há séculos atrás (RIERA, 2001). Esse conjunto de fatores sinaliza a necessidade de redefinir o ambiente onde a aprendizagem ocorre: é preciso adequar o espaço ao uso de novas tecnologias.

Nesse sentido, Moran (2002) propõe que:

Podemos modificar a forma de ensinar e de aprender. Um ensinar mais compartilhado. Orientado, coordenado pelo professor, mas com profunda participação dos alunos, individual e grupalmente, onde as tecnologias nos ajudarão muito, principalmente as telemáticas.

E sugere, ainda, que:

Muitas formas de ensinar hoje não se justificam mais. Perdemos tempo demais, aprendemos muito pouco, nos desmotivamos continuamente. Tanto professores como alunos temos a clara sensação de que em muitas aulas convencionais perdemos muito tempo (MORAN, 2002).

Assim, a partir dessas reflexões, surgiu a idéia de desenhar um Sistema de Gerenciamento de Recursos em um ambiente de aprendizagem cooperativa. Pensou-se em uma infra-estrutura que possa ser vista de forma transparente onde professores e alunos vêem esses recursos informáticos como um instrumento a mais na sala de aula, segundo o conceito de “invisibilidade” do computador sugerida por Gross(2000).

A decisão de utilizar PDA's em relação a computadores de mesa ou Notebooks, foi em função de ser a opção de menor custo, de ser portátil e o fato de que, em breve, será comum os telefones celulares virem com um PDA embutido, com tecnologia 2,5G ou 3G¹ e de ter sido inspirada em outros autores como Abowd(1999); Shotsberger e Vetter(2001).

A escolha do CommonKADS como ferramenta de modelagem foi em função de ser uma metodologia conhecida (SCHREIBER et al, 1999), da necessidade de estudar e compreender o domínio de pesquisa, neste caso, a docência presencial e porque o uso de uma metodologia facilita a identificação explícita do conhecimento usado pelos agentes no ambiente tecnológico, facilita a compreensão sobre que tipo de interação ocorre entre professor-aluno-tecnologia e ajuda a identificar os requisitos necessários à implantação deste ambiente: as necessidades de *hardware* e as características do software.

¹ Gerações de celulares

1.3 Perguntas da pesquisa

De que forma a utilização de recursos tecnológicos variados – *hardware, software*, redes, em sala de aula, podem contribuir no processo Ensino x Aprendizagem?

Como criar um ambiente, rico em recursos multimidiáticos e de simples utilização, que seja capaz de contemplar os pressupostos pedagógicos e requisitos metodológicos exigidos em cursos orientados a aprendizagem cooperativa?

Como, a partir de uma metodologia conhecida, especificar o desenho de um Sistema de Gerenciamento Inteligente de Recursos em um ambiente baseado em aprendizagem cooperativa?

1.4 Objetivos do trabalho

1.4.1 Objetivo Geral

Este trabalho propõe um modelo de um Sistema Gerenciador Inteligente de Recursos a ser utilizado em um ambiente que oportunize a aprendizagem cooperativa, em aulas presenciais.

1.4.2 Objetivos específicos

- Identificar e especificar, a partir da metodologia CommonKADS, o desenho de um Sistema Gerenciador Inteligente de Recursos a ser utilizado em um ambiente tecnológico de aprendizagem cooperativa apoiado por computadores de mão para ser aplicado em aulas presenciais.
- Consolidar os fundamentos teóricos relativos aos ambientes de aprendizagem cooperativa em rede e em aulas presenciais.
- Selecionar tecnologias que, implementadas em conjunto, sejam capazes de suportar atividades educacionais baseadas na cooperação.

1.5 Metodologia

Inicialmente é feita uma revisão bibliográfica para a fundamentação teórica em relação às teorias contemporâneas da aprendizagem, os pressupostos sobre aprendizagem cooperativa e sobre metodologias de desenvolvimento de sistemas baseados em conhecimento.

Essa revisão abrangeu também as formas que a tecnologia de informação e comunicação tem sido utilizadas na educação, os ambientes de aprendizagem cooperativa apoiados por computador e os seus principais aspectos.

Em seguida, a metodologia CommonKADS é utilizada para a definição do modelo de tarefas no qual fundamenta-se a arquitetura de software do sistema proposto e, após, procede-se a uma comparação com modelos conhecidos que implementam funções de cooperação em situações de aprendizagem, chegando-se, finalmente às conclusões e recomendações para trabalhos futuros.

1.6 Estrutura da Dissertação

O capítulo 1 apresenta, na introdução, uma motivação para o trabalho proposto, enfatizando aspectos sociais do cenário educacional atual com uma problematização, justificativa e os objetivos.

No capítulo 2, faz-se, inicialmente, um resumo sobre as teorias contemporâneas de aprendizagem e, em especial, a aprendizagem cooperativa, destacando-se as formas de uso do computador na educação.

No capítulo 3 apresenta-se os fundamentos das metodologias para o desenvolvimento de sistemas baseados em conhecimento e a estrutura da metodologia CommonKADS.

No capítulo 4 se faz uma descrição de alguns softwares educativos com ênfase na aprendizagem cooperativa e uma avaliação dos principais aspectos para a caracterização desses softwares.

O capítulo 5, apresenta o modelo proposto conforme a metodologia CommonKADS com aspectos técnicos e requisitos fundamentais à sua implantação e ao final, é feita uma análise dos resultados obtidos com este trabalho e sugestões para trabalhos futuros são apresentadas.

2 TEORIAS DE APRENDIZAGEM

Este capítulo apresenta um resumo sobre teorias da aprendizagem, destacando a aprendizagem cooperativa, uma reflexão sobre a forma de uso e aspectos sociais da aplicação de tecnologias de informação e comunicação na educação.

Na concepção de qualquer ambiente tecnológico de aprendizagem, é importante identificar a concepção teórica de aprendizagem que o orienta, pois deve ser pensado segundo uma teoria sobre como o sujeito aprende, como ele se apropria e constrói seu conhecimento. Várias teorias contribuem para o entendimento da aprendizagem cooperativa. Estas teorias têm em comum o fato de assumirem que indivíduos são agentes ativos na busca e construção de conhecimento, dentro de um contexto significativo (VIEIRA, 2002).

A seguir, um resumo sobre as principais teorias de aprendizagem centrando-se na colaboração/cooperação entre indivíduos, ou a interação social:

2.1 Construtivismo de Bruner

Considerado como uma metateoria que agrega um número de teorias cognitivas e da aprendizagem. Considera que a aprendizagem é um processo ativo no qual a construção de novas idéias ou conceitos é baseada em conhecimentos presentes e passados (BRUNER, 1960).

O aprendiz seleciona e transforma informação, hipóteses e toma decisões apoiadas em uma estrutura cognitiva, como, por exemplo, esquemas e modelos mentais. As estruturas cognitivas provêm significado e organização às experiências e permitem ir do específico ao além da informação dada. Assim que um conceito é aprendido, o professor deve encorajar o aluno a descobrir princípios por eles mesmos. Professores e alunos devem estar engajados em um diálogo ativo (aprendizagem socrática, por exemplo); A principal tarefa do professor é aproveitar o conhecimento prévio dos alunos com o que deve ser aprendido. O currículo deve ser organizado em uma forma espiral de modo que o aluno continuamente construa seu

conhecimento em cima do que já sabe. Bruner (1966) sustenta que a teoria da aprendizagem deve apoiar-se em quatro aspectos maiores:

- a predisposição para aprender dos alunos;
- o conhecimento deve ser estruturado de forma que possa ser melhor compreendido pelo aluno;
- na mais efetiva seqüência de apresentar os conteúdos e,
- na natureza e ritmo das punições e recompensas.

No seu trabalho mais recente, Bruner (1990) incorporou aspectos sociais e culturais da aprendizagem nesta metateoria que partem do princípio de que o processo de ensino/aprendizagem deve considerar a experiência prévia e habilidades dos alunos e que o ensino deve ser estruturado para facilitar a extrapolação – para ir além da informação recebida.

2.2 Epistemologia Genética de Piaget

Em um período de seis décadas, de 1920 a 1980, Jean Piaget conduziu um programa de pesquisa naturalística que afetou nosso entendimento sobre o desenvolvimento de crianças. Piaget denominou seu estudo teórico de “Epistemologia Genética” porque esteve inicialmente interessado em saber como ocorre o desenvolvimento do conhecimento em organismos humanos.

Piaget fundamentou seus conceitos a partir da influencia da biologia e filosofia no desenvolvimento de crianças. A sua teoria centra-se no conceito de estruturas cognitivas, que são padrões de ações físicas ou mentais que caracterizam atos de inteligência e correspondem a estágios de desenvolvimento da criança (PIAGET, 1969, 1970),(PIAGET e INHELDER, 1969, 1973). Divididas em quatro estruturas básicas (ou fases de desenvolvimento):

- sensório motor – durante a fase de 0 a 2 anos a inteligência manifesta-se na forma de ações motoras;
- pré-operatório – no período de 3 a 7 anos, o pensamento é de natureza intuitiva;

- operações concretas – entre 8 e 11 anos, raciocínio lógico que depende de referências concretas;
- operações formais – entre 12 e 15 anos, o raciocínio envolve abstrações.

De acordo com essa teoria, as estruturas mudam por meio de um processo de adaptação: a assimilação, envolve a interpretação de eventos em termos de estruturas de cognição existentes e a acomodação refere-se a mudanças na estrutura cognitiva de forma a dar sentido ao meio. O desenvolvimento cognitivo consiste de um constante esforço em termos de assimilação e acomodação (GALLAGHER e REID, 1981). Nesse sentido, a teoria de Piaget é similar, em sua natureza, a outras perspectivas de aprendizagem construtivistas como, por exemplo, de Bruner e Vygotsky.

2.3 Teoria do Desenvolvimento Socio -Cultural de Vygotsky

O estudo teórico de Vygotsky considera que o interacionismo social tem um papel fundamental no desenvolvimento da cognição. De acordo com Vygotsky (1978):

Cada função no desenvolvimento cultural da criança aparece duas vezes: primeiro, no nível social, e depois, no nível individual. Primeiro, entre pessoas (interpsicológico) e então dentro da criança (intrapsicológico). Isto se aplica igualmente à atenção voluntária, à memória lógica e à formação de conceitos.

Um segundo aspecto da teoria de Vygotsky é a idéia de que o potencial para o desenvolvimento cognitivo é limitado a um certo período de tempo que ele chama de “Zona de Desenvolvimento Proximal” (ZDP). Além do mais, um grande desenvolvimento durante a ZDP depende sobretudo de uma interação social ampla. Também, o tipo de habilidades que podem ser desenvolvidas para uma criança com a ajuda de adultos ou colaborando em pares, excede o que ela pode obter sozinha (Vygotsky, 1978).

A teoria de Vygotsky foi uma tentativa de explicar a consciência como produto final da socialização. É um elemento chave da “Teoria da Aprendizagem Situada” e “Instrução Ancorada”. Tendo em vista Vygotsky ter centrado sua teoria no

desenvolvimento cognitivo, torna-se interessante comparar essa teoria sob a perspectiva de Bruner e Piaget (WERTSCH, 1985).

Esta é uma teoria do desenvolvimento cognitivo. Muito do trabalho original foi feito no contexto da aprendizagem da linguagem em crianças (VYGOTSKY, 1962).

2.4 Teoria dos Operadores Condicionantes

A teoria de B.F. Skinner é baseada na idéia de que a aprendizagem é uma função de mudança de comportamento, como resultado de respostas individuais a eventos (estímulos) que ocorrem no meio. Uma resposta pode ser algo como lançar uma bola ou a resolução de um problema matemático. Quando um estímulo-resposta (E-R) particular é reforçado, o indivíduo é condicionado a responder. Essa teoria tenta dar explicações comportamentais a uma ampla quantidade de fenômenos cognitivos (SKINNER, 1953).

2.5 Teoria da Cognição Situada

J. Lave (1988) sustenta que a aprendizagem é uma função da atividade, contexto e cultura do meio em que ela ocorre. A interação social e colaboração são componentes críticos da cognição situada, envolve uma “*comunidade de prática*” que incorpora crenças e comportamentos a serem adquiridos. Os aprendizes movem-se da periferia para o centro desta comunidade, tornam-se mais ativos e engajados com a cultura - saem da situação de aprendizes para transformarem-se em expertos. Além disso, a cognição situada é mais desproposital (incidental) que deliberada: processo chamado de “Participação Periférica Legítima” (LAVE e WENGER, 1991).

2.6 Teoria da Instrução Ancorada

Tornou-se um importante paradigma da aprendizagem baseada em tecnologia, desenvolvida por the *Cognition & Technology Group at Vanderbilt (CTGV)*, tendo sido John Bransford o principal mentor dessa teoria. O foco inicial desse trabalho centrou-se no desenvolvimento de ferramentas de vídeo-disco interativo para motivar professores e alunos na proposição e resolução de problemas reais e

complexos. O Vídeo como material didático funciona como “âncora” para todo momento de aprendizagem subsequente (CTVG, 1993, p52).

O uso da tecnologia de vídeo-disco interativo possibilita o aluno explorar facilmente os conteúdos a serem aprendidos. A Instrução Ancorada é considerada como um estudo da “Teoria da Aprendizagem Situada” e é baseada em um modelo geral de resolução de problemas (CTGV, 1990, 1993), (BRANDSFORD e STEIN, 1993).

No quadro 1, encontram-se resumidas as principais características das teorias de aprendizagem que de alguma forma apontam a colaboração entre indivíduos, ou a interação social:

Quadro 1 - Teorias de Aprendizagem

Teorias de Aprendizagem	Características
Epistemologia Genética de Piaget	Ponto central: estrutura cognitiva do sujeito Níveis diferentes de desenvolvimento cognitivo. Desenvolvimento facilitado pela oferta de atividades e situações desafiadoras Interação social e troca entre indivíduos funcionam como estímulo ao processo de aquisição de conhecimento.
Teoria Construtivista de Bruner	Aprendiz é participante ativo no processo de aquisição de conhecimento. Instrução relacionada a contextos e experiências pessoais. Determinação de seqüências mais efetivas de apresentação de material. Teoria contemporânea: criar comunidades de aprendizagem mais próximas da prática cooperativa do mundo real.
Teoria Sócio-Cultural de Vygotsky	Desenvolvimento cognitivo é limitado a um determinado potencial para cada intervalo de idade (Zona Proximal de Desenvolvimento). Desenvolvimento cognitivo completo requer interação social.
Aprendizagem baseada em Problemas/ Instrução ancorada	Aprendizagem se inicia com um problema a ser resolvido (âncora ou foco). Centrada no aprendiz e contextualizada.
Cognição Situada	Aprendizagem ocorre em função da atividade, contexto e cultura e ambiente social na qual está inserida. Interação social e colaboração são componentes críticos para aprendizagem (comunidade de prática).

Fonte: Adaptado de Vieira (2002)

2.7 O que é aprendizagem cooperativa

A aprendizagem cooperativa é uma vertente das teorias sócio-cognitivas de aprendizagem cuja premissa fundamenta-se na interação social. É uma técnica onde os estudantes colaboram em grupos para adquirir novos conhecimentos, para desenvolver habilidades sociais e/ou para aprender a resolver problemas

específicos. Em um ambiente de cooperação os estudantes se ajudam no processo de aprendizagem, atuando como parceiros entre si e com o professor, e visando adquirir conhecimento sobre um dado objeto. A aquisição de conhecimentos e habilidades, nesse tipo de aprendizagem, não ocorre de forma inerentemente individual, mas como resultado da interação entre grupos e a cooperação apoiando-se (SANTORO et al,1998 apud SMYSER, 1993):

- (1) na responsabilidade individual pela informação reunida pelo esforço do grupo;
- (2) na interdependência positiva, de forma que os estudantes sintam que ninguém terá sucesso, a não ser que todos o tenham;
- (3) na melhor forma de entender um dado material, tendo que explicá-lo a outros membros de um grupo;
- (4) no desenvolvimento de habilidades interpessoais, que serão necessárias em outras situações na vida do sujeito;
- (5) no desenvolvimento da habilidade para analisar a dinâmica de um grupo e trabalhar com problemas;
- (6) na forma comprovada de aumentar as atividades e envolvimento dos estudantes; e
- (7) em um enfoque interessante e divertido.

De acordo com Santoro et al(1998):

A aplicação de técnicas de aprendizagem cooperativa na educação formal é importante não só para a obtenção de ganhos em relação ao próprio processo ensino-aprendizagem, mas também na preparação dos indivíduos para situações futuras no ambiente de trabalho, onde, cada vez mais, as atividades exigem pessoas aptas ao trabalho em equipe.

Aprender em grupo não só permite que os alunos aprendam por meio da cooperação, mas permite que eles aprendam a cooperar. A cooperação pode trazer benefícios ao processo Ensino x Aprendizagem: Slavin (1995), aponta quatro perspectivas teóricas para explicar os efeitos produzidos pela aprendizagem cooperativa:

- *motivacionais* – com foco no objetivo, cada membro de um grupo só realiza seus objetivos pessoais se o grupo como um todo for bem sucedido.
- *coesão social* – a aprendizagem ocorre em função da coesão do grupo.
- *cognitivas de desenvolvimento* – a interação entre aprendizes aumenta a maestria do grupo em conceitos críticos.
- *cognitivas de elaboração* - a informação é retida na memória e está relacionada à informação presente anteriormente nela, então o aprendiz deve estar engajado em algum tipo de reestruturação cognitiva, ou elaboração.

A responsabilidade do professor, em atividades de cooperação, centra-se na supervisão – acompanhamento da evolução da aprendizagem e possíveis mudanças em função dessa evolução. Assim, com esses objetivos, muitos ambientes virtuais tem sido desenvolvido (O´NEILL e GOMES, 1994), (SHABO et al, 1997).

2.7.1 Aprendizagem/Trabalho colaborativo assistido por computador - CSCL/CSCW²

A aprendizagem colaborativa assistida por computador (CSCL - *Computer Supported Collaborative Learning*) pode ser definida como uma estratégia educativa em que dois ou mais sujeitos (professor/aluno, aluno/aluno), para construir o seu conhecimento, interagem por meio de discussão, reflexão e tomada de decisões, tendo, como mediadores do processo de ensino-aprendizagem, determinados recursos informáticos.

A CSCL cresceu em torno de pesquisa sobre trabalho colaborativo assistido por computador (CSCW - *Computer Suported Collaborative Work*). CSCW é definido com um sistema de redes de computadores que suporta grupos de trabalho com tarefas comuns, fornecendo uma *interface* que possibilita a realização de trabalho em conjunto.

² <http://www.cica.indiana.edu/csc195/>

De acordo com Kumar(1996), tanto os ambientes CSCL quanto os CSCW trabalham na maior parte do tempo de uma forma denominada de passiva, pois o *software* de suporte gerencia os ambientes mas não exerce nenhum controle ou ação sobre as interações promovidas.

2.7.2 Colaboração/Cooperação

Percebe-se na literatura que alguns autores utilizam o termo “Colaborativo” quando fazem referencia à aprendizagem, cuja prática pedagógica, está baseada na abordagem sócio-cultural. Martins (2000 apud PANITZ, 1996) no seu trabalho, faz uma reflexão sobre o significado de cooperação e colaboração, estabelecendo uma diferença entre os termos - colaboração como uma filosofia de interação e um estilo de vida pessoal, e cooperação como uma estrutura de interação planejada para facilitar a elaboração de um produto ou realização de um objetivo. O autor afirma que:

A aprendizagem cooperativa está centrada na estruturação de ambientes de aprendizagem onde o enfoque é a oportunidade da cooperação” e, ainda, que “O suporte dado por computadores à aprendizagem cooperativa tem como objetivo dinamizar o processo, através de sistemas que implementem um ambiente de cooperação e possuam papel ativo na análise e controle desta

Tendo em vista o objeto de estudo deste trabalho estar centrado na estruturação de um ambiente tecnológico de aprendizagem que oportuniza a cooperação, preferiu-se adotar a designação “Aprendizagem Cooperativa”.

2.7.3 Formas organizativas de aula baseadas na cooperação

Agelet et al (1997) sugere como deveria funcionar o processo Ensino/Aprendizagem em aulas fundamentadas na cooperação. Propõe que sejam criados grupos heterogêneos de alunos com 4 ou 5 componentes e que o professor determine os conteúdos a serem trabalhados, assegurando-se de que existem objetivos

pertinentes para toda a diversidade de alunos. O processo, funciona da seguinte forma:

- os objetivos didáticos são apresentados aos alunos
- realiza-se uma avaliação inicial
- são priorizados os objetivos de cada aluno
- explicam-se detalhadamente os conteúdos e os critérios de avaliação
- os grupos realizam as atividades
- os alunos realizam uma auto-avaliação
- o professor realiza uma avaliação individual
- realiza-se a avaliação de cada grupo
- planeja-se a recuperação individual – plano de trabalho individual ou plano de trabalho para pequenos grupos homogêneos.

Esse modelo, segundo Agelet, pode ser aplicado para qualquer disciplina, utilizando diversos materiais didáticos e recursos tecnológicos.

2.7.4 Algumas técnicas didáticas para a aprendizagem cooperativa

De acordo com Domènech(1997), em uma estrutura de aprendizagem cooperativa, vê-se, de forma intensa, uma grande variedade de elementos: instrumentos, técnicas, estratégias, agrupamentos diversos de alunos, atividades mais abertas ou mais dirigidas, mecanismos de ajuda aluno/ aluno e professor/aluno, recompensas individuais e grupais. Ele sugere que, de forma mais específica, existem algumas técnicas que podem levar os alunos, com mais ou menos acerto, a estabelecer relações de cooperação:

1. O quebra-cabeças (Jigsaw)

É especialmente útil para as áreas de conhecimento nas quais os conteúdos são suscetíveis de ser fragmentados em diferentes partes (por exemplo, literatura, história, ciências experimentais).

Divide-se a classe em grupos heterogêneos de 4 ou 5 membros cada um. O material objeto de estudo é dividido em tantas partes quantos membros têm a equipe, de maneira que cada um dos seus membros receba um fragmento da informação do tema que, em seu conjunto, todas as equipes estão estudando, e não recebe a parte relativa a preparação de seu próprio sub-tema.

Cada membro da equipe prepara sua parte a partir da informação que o professor lhe facilita ou que ele pode consultar. Depois, com os integrantes das outras equipes que estudaram a mesma parte, formam um grupo de especialistas, em que trocam informação, aprofundam-se nos conceitos-chaves, constroem esquemas e mapas conceituais, esclarecem dúvidas. Pode-se dizer que chegam a ser os especialistas da sua parte. Em seguida, cada um deles retorna à sua equipe de origem e se responsabiliza de explicar ao grupo a parte que ele preparou. Assim, portanto, cada membro necessita dos outros e se vê obrigado a cooperar, porque cada um deles dispõe só de uma peça do quebra-cabeças e seus companheiros de equipe tem as outras, imprescindíveis para concluir com êxito a tarefa proposta.

II. Grupos de investigação

Tem afinidade com a anterior, mas é mais complexa. Passos:

- escolha e distribuição de sub-temas de acordo com aptidões ou interesses dos alunos, dentro de um problema geral, normalmente planejado pelo professor;
- formação dos grupos com 3 a 5 componentes, de livre escolha dos alunos;
- os alunos e professor planejam os objetivos concretos a que se propõem e os procedimentos que utilizarão para alcançá-los (encontrar a informação, sistematizá-la, resumí-la, esquematizá-la.);
- os alunos desenvolvem o plano descrito. O professor acompanha o progresso de cada grupo e oferece ajuda;
- os alunos apresentam o trabalho, delineando possíveis perguntas com as respostas e dúvidas que possam surgir;

- o professor e os alunos realizam conjuntamente a avaliação do trabalho em grupo e a apresentação. Pode-se complementar esse passo com uma avaliação individual.

III. Técnica STAD (Student Team-Achievement Divisions)

Ocorre uma cooperação intragrupal e uma competição intergrupala. São formados grupos de 4 ou 5 componentes. O professor apresenta um tema ao grupo com explicações e exemplos que acredita serem necessários. Logo após, os alunos trabalham em equipe durante várias sessões em que formulam perguntas, comparam respostas, discutem, ampliam a informação, elaboram esquemas e resumos, esclarecem conceitos, memorizam e se asseguram que todos os membros aprenderam o material proposto. No final, o professor avalia cada aluno individualmente. A pontuação obtida por cada aluno é comparada com as notas anteriores. Se forem iguais ou superiores, recebem pontos que, somados, geram a nota grupal. Os resultados obtidos por cada grupo são publicados.

IV. TGT (Teams-Games Tournaments)

Similar a anterior, ao invés de exames individuais, realiza-se um torneio em que os alunos dos grupos diferentes “batalham” entre si. Formam-se equipes de 2 ou 3 alunos de maneira eqüitativa: os 3 alunos que obtiveram a pontuação mais alta no último torneio, forma a equipe número 1, os 3 seguintes a equipe número 2 e assim por diante.

V. Tutoria por iguais

Sustenta-se na colaboração que um aluno oferece a um companheiro que, por sua vez, formulou uma demanda de ajuda. Neste caso, existe cooperação entre pares de alunos que são, normalmente, da mesma classe e idade, onde um faz o papel de tutor e outro de aluno. O tutor ensina e o aluno aprende sob a supervisão do professor (PARRILLA, 1992).

Conhecer algumas técnicas que tem sido empregadas para estabelecer relações de cooperação entre alunos, ajuda a compreender melhor como determinados recursos tecnológicos (*hardware/software*), multimidiáticos, podem contribuir no processo Ensino x Aprendizagem.

2.8 Formas de uso do computador na educação

A utilização de computadores diretamente no processo de ensino-aprendizado, de acordo com o livro *Projeto EDUCOM* (ANDRADE e ALBUQUERQUE, 1993), começou na década de 60 com a instrução programada, representada pela “*programação linear*” de Skinner (teoria dos operadores condicionantes). Na instrução programada eram aplicados princípios da psicologia behaviorista para desenvolver sequências de instruções que cuidadosamente controlavam o processo de aprendizado do aluno no sentido de aproximá-lo a um comportamento terminal desejável.

Para os behavioristas, o aprendizado deveria ser concebido em termos de uma mudança no comportamento observável do aluno, mudança que ocorria por meio de estímulos e respostas de acordo com princípios mecânicos. Assim, um programa de computador poderia fornecer um estímulo para o qual o aluno deveria fornecer uma resposta.

A transição para a instrução assistida por computador - Computer Assisted Instruction (CAI) não ficou muito óbvia, pois, muitos dos primeiros CAI's, ainda eram caracterizados como instrução programada. O CAI, a forma mais comum de uso dos computadores na educação, nos EUA, procurava refletir a experiência pedagógica dos professores diretamente no comportamento dos programas que possuíam unidades instrucionais pré-armazenadas as quais compunham a lição a ser ensinada ao aluno.

Há, segundo Valente (1993), dois tipos de abordagens de ensino que podem ser realizados por intermédio do computador:

- *máquina de ensinar* - O computador assume a característica de ser uma versão computadorizada dos métodos tradicionais de ensino. As categorias mais comuns desta modalidade são os programas tutoriais, os programas de exercício-e-prática, os jogos e a simulação.
- *Ferramenta* - O computador não é mais o instrumento que ensina o aprendiz, mas a ferramenta com a qual o aluno desenvolve algo. O aprendizado, nessa abordagem, ocorre pelo fato de estar executando uma tarefa por intermédio do computador. Nesse caso, podemos ter aplicativos para uso do aluno e do professor, resolução de problemas com auxílio do computador, produção de música, programas de controle de processo, computador como comunicador, ambientes de aprendizagem via WEB.

3 MODELOS DE CONHECIMENTO E METODOLOGIAS PARA DESENVOLVIMENTO DE SISTEMAS BASEADOS EM CONHECIMENTO

Neste capítulo, para o conceito de metodologia, preferiu-se fazer uma compilação de alguns artigos publicados em "The HandBook of Applied Expert Systems" - em especial os capítulos 1 , 2 e 3, em função de existir uma infindável lista de referências bibliográficas sobre este tema e os conteúdos presentes nesses artigos serem mais que suficientes para os objetivos propostos neste trabalho.

3.1 O que é Metodologia

No seu significado original, metodologia refere-se ao conhecimento sobre métodos. Isso implica que metodologia e método estão no mesmo domínio - mas não são a mesma coisa. Em um certo nível de meta linguagem, o método também contém conhecimento: o importante é saber a natureza deste (meta) conhecimento (HOOG, 1998) :

- O que é comumente entendido como conhecimento metodológico refere-se ao "Saber o que", "Saber como" e "Saber quando". Esses "Saber" são normativos ou prescritivos: eles não descrevem o estado do mundo, mas prescrevem como um simples agente pode atuar para encontrar um determinado objetivo.
- Declarações prescritivas podem diferir grandemente em precisão: por exemplo, a prescrição "trabalhar duro", é bastante imprecisa se o objetivo é tornar-se rico – é muito ampla. Por outro lado, a forma mais restritiva de prescrever algo que conhecemos, é com os algoritmos de computador, os quais não deixam espaço para interpretação pelo agente - o computador.

Enfim, pode-se dizer que esse tipo de conhecimento metodológico, encontra-se, claramente, entre dois extremos : se por um lado é muito geral, não serve como uma

grande ajuda. E, por outro lado, se é muito específico, a aplicabilidade torna-se muito limitada e vulnerável em outros contextos.

Com base nessas considerações, a pirâmide metodológica introduzida por Wielinga et al (1994) é uma boa forma de caracterizar o que está envolvido em uma metodologia (figura 1) :

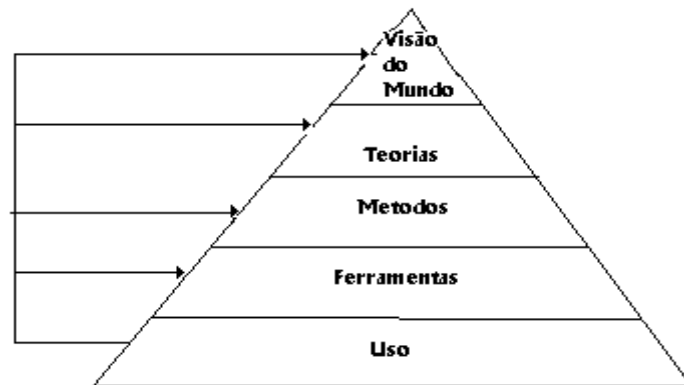


Figura 1: A pirâmide metodológica

Fonte: Adaptado de Wielinga et al (1994)

visão do Mundo - a camada do topo da pirâmide, refere se aos princípios e hipóteses que compõe a metodologia. Ela inclui os objetivos a que se propõe.

teoria – Essa camada elabora os princípios e hipóteses e forma a base do conhecimento no domínio da metodologia.

métodos e técnicas - Operacionalizam os conteúdos dessas teorias, a parte principal ou o "Como".

ferramentas - São instâncias computadorizadas dos métodos e técnicas da camada anterior. Sendo computadorizada sempre requer o uso de conhecimento adicional associado a ela.

uso - Representa o toque final da metodologia. Pode revelar detalhes que faltam na prescrição provida pela camada acima, o qual poderá induzir a revisões nos diferentes componentes da metodologia.

Metodologicamente, todas as camadas são interligadas - uma mudança de princípios poderá se propagar sobre a pirâmide promovendo maiores modificações.

A compreensão do que é e o que significa metodologia tem uma grande importância para a aquisição e modelagem de conhecimento, pois metodologias podem ser aplicadas no desenvolvimento de SBC. De acordo com Hoog (1998), as técnicas de construção dos SBC podem ser divididas em três classes gerais:

- *técnica de caixas e seta* - caracterizada pelo fato de que são principalmente limitadas a um nível de ciclo de vida consistente de caixas, que denotam atividades de desenvolvimento e setas unindo essas atividades, para gerar uma seqüência. O que essas técnicas necessitam é uma especificação detalhada do "Como" das caixas. Não existem métodos, técnicas ou ferramentas providas. Contudo, elas cobrem muitas das atividades normalmente vistas como importantes no desenvolvimento dos SBC. São amplas e rasas.
- *técnica do enfoque* - limita-se a um ou a pouco dos métodos e técnicas que cobrem parte do trabalho a ser feito. Em alguns aspectos pode ser visto como uma míni metodologia: no seu escopo limitado pode algumas vezes cobrir todos os níveis da pirâmide. Sendo o foco em um limitado número de atividades, ela é estreita e profunda.
- *bem-feito (Full-fledged)* - provê suporte para todas as camadas da pirâmide. Pode ser de duas formas:
 - Metodologia para desenvolvimento de sistemas convencionais com acréscimo de "orientado a SBC".
 - Metodologia especialmente desenhada para o desenvolvimento de SBC.

Hoog (1998) destaca, ainda, que deve ser enfatizado que a pirâmide metodológica e suas classificações derivadas não implicam em um julgamento de valor: não existe metodologia melhor ou pior - uma metodologia ruim para uns pode ser uma excelente metodologia para outros.

3.2 Aspectos gerais das Metodologias Baseadas em Conhecimento

A principal diferença dos Sistemas Baseados em Conhecimento – SBC, em relação aos sistemas convencionais, encontra-se na manipulação explícita do conhecimento em um domínio geralmente complexo. Inicialmente, nos SBC, tratava-se erroneamente conhecimento como parte do nível simbólico fazendo-se uso do formalismo das regras de produção (PENÍN, 2000).

Newell distinguiu claramente os conceitos de conhecimento e representação, forçando a modelagem do conhecimento fora do nível simbólico, postulando a existência de um novo nível computacional, situado acima do nível simbólico denominado nível de conhecimento (NEWELL, 1982, 1993). Para isso, utilizou entidades de conhecimento sem especificar em nenhum momento sobre qual suporte simbólico implementar o sistema. A partir dessa nova conceituação, iniciou-se a segunda geração de sistemas especialistas (STEELS, 1985), que fundamenta a construção de um SBC e o uso de uma metodologia de desenho baseada na modelagem do conhecimento.

Mesmo considerando-se que a hipótese do nível de conhecimento de Newell seja o ponto de partida em que devem repousar todas as metodologias, ou estudos, que tentem abordar o processo de desenvolvimento de um SBC, existem outros elementos fundamentais na maioria destas metodologias (PENIN, 2000 apud MUSEN e SCHREIBER, 1995a):

- *Princípio de limitação de papéis* - O conhecimento, para um sistema inteligente, está estruturado como um conjunto de elementos que distribuem papéis diferentes e especializados em todo o processo de resolução de problemas;
- *Princípio de tipificação do conhecimento* - Se for considerado que o conhecimento está estruturado em elementos que distribuem diferentes papéis no processo de raciocínio, é possível estabelecer uma classificação (ou tipificação) desses papéis em termos gerais, ainda que, em cada metodologia a terminologia usada pode variar:

- *Tarefas* - correspondem aos objetivos que se deve alcançar durante a resolução do problema;
 - *Método de resolução de problemas* - descreve uma forma de realizar uma tarefa. Tipicamente, um método define um conjunto de tarefas sobre as que operam os outros métodos e, especificam o controle imposto nessa decomposição de método-tarefa. Em algumas aproximações, no entanto, se estende o conceito de tarefas e o MRP se inclui em sua declaração (WIELINGA et al, 1994);
 - *Inferências* - São ações primitivas do processo de resolução do problema (não se podem decompor), que se aplicam em alguma parte do conhecimento do domínio para derivar nova informação;
 - *Ontologias* - descrevem a estrutura e vocabulário do conhecimento estático do domínio. Neste sentido, se podem considerar como estrutura de dados que define as entidades e relações em diferentes aplicações. Por exemplo, em um domínio médico uma ontologia poderia incluir termos como terapia, receita e diagnóstico;
 - *Conhecimento do domínio* - refere-se a um conjunto de declarações que fazem uso da ontologia do domínio.
- *Princípio de reutilização* - Uma das principais vantagens da modelagem no nível do conhecimento em relação ao nível simbólico é a possibilidade de reutilizar certos componentes do conhecimento em aplicações distintas. Para cada tipo de conhecimentos é possível identificar uma série de descrições genéricas cujo comportamento não muda ainda que sejam trasladados a diferentes domínios.

Adicionalmente a estes três princípios básicos que guiam a modelagem dos sistemas no nível do conhecimento, Van Heijst et al(1997) propõem um quarto princípio que está diretamente relacionado com a forma de fazer uso dos elementos reutilizados:

- *Uso de modelos esqueléticos* - nestes modelos se especificam uma parte do modelo de conhecimento como, por exemplo, um método de resolução de problemas. O engenheiro se limitará a preencher as outras partes para obter o modelo completo. Portanto, fazendo uso da tipificação do conhecimento, os elementos especificados em um modelo esquelético impõem a forma em que se deverão modelar os componentes restantes do conhecimento (figura 2).

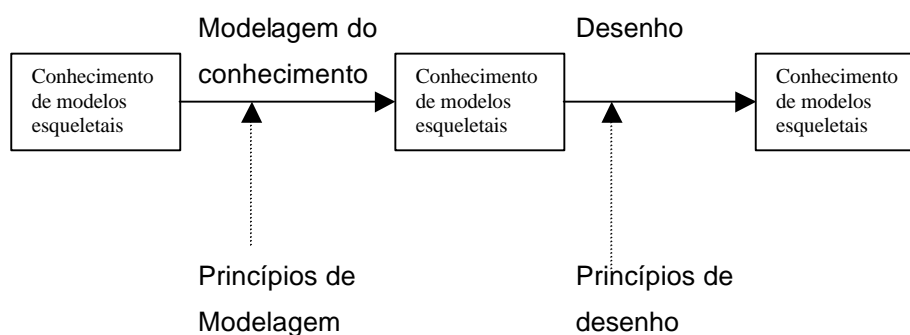


Figura 2: Processo de desenvolvimento de um SBC segundo as metodologias baseadas no conhecimento.

Fonte: adaptado de Penín (2000) apud VAN HEIJST et al (1997)

Uma segunda etapa da modelagem do SBC consiste em aplicar princípios de desenhos gerais e considerar as necessidades específicas do domínio para derivar um protótipo final (Figura 2). Para facilitar esse processo de desenvolvimento de um SBC, em todas as metodologias criaram-se bibliotecas de componentes do conhecimento, onde se pode selecionar ou adaptar a ontologia ou o método mais adequado para resolver uma tarefa em um domínio determinado.

3.3 CommonKADS

A partir de um projeto de pesquisa Europeu (ESPRIT P1098)³, surgiu KADS (WIELINGA et al, 1992; SCHREIBER et al, 1993) uma metodologia direcionada ao processo de aquisição de conhecimento, que é considerado como uma das atividades mais complexas no desenvolvimento dos SBC. Em seguida, com a

³ www.cordis.lu/esprit/home.html

intenção de resolver algumas deficiências na especificação do modelo de conhecimento em KADS, surgiu outro projeto (ESPRIT P5248) que resultou na metodologia CommonKADS (SCHREIBER et al, 1994a; WIELINGA et al, 1993; SCHREIBER et al, 1999). De acordo com Penín (2000), este projeto, estendeu a metodologia KADS, e tenta envolver todo o processo de desenvolvimento de um SBC: da descrição da organização em que se constrói o sistema, até o desenho final da aplicação.

Para alcançar esses objetivos, CommonKADS, com base em considerações sobre a modelagem de sistemas, apóia-se em três aspectos (PENÍN, 2000):

- o desenvolvimento de um SBC fundamenta-se na construção de um conjunto de modelos que, a partir destes, é possível modelar o comportamento do sistema sobre diversos pontos de vista no contexto de uma organização;
- a gestão do projeto de desenvolvimento do SBC se baseia no ciclo de vida espiral proposto por Boehm (1988), que propõe a revisão contínua da situação do projeto em cada momento - são identificados os riscos relativos a uma série de objetivos planejados no início do projeto e a revisão e ou modificação desses objetivos, depende de como, ou em que medida, são alcançados;
- a extrema necessidade do desenvolvimento de bibliotecas de componentes, definidas de forma genérica, que possam ser compartilhadas e que permitam sua re-utilização em diferentes aplicações ou domínios. E, além disso, do desenvolvimento de técnicas de modelagem que facilitem a re-utilização e a manutenção desses componentes. Pode-se encontrar um conjunto de bibliotecas de inferências básicas (ABEN, 1994, 1995) especificadas formalmente e um conjunto de métodos gerais (AAMODT et al, 1992; BREUKER E VAN DE VELDE, 1994) cuja função é a resolução de uma série de tarefas distintas para diferentes tipos de problemas.

3.4.1 O conjunto de modelos de CommonKADS

De acordo com Penín (2000), o desenvolvimento de um SBC utilizando-se CommonKads, consiste na descrição de um conjunto de modelos independentes entre si quanto a seu conteúdo (que aspectos modelam), ainda que apresentem certas dependências relativas as informações obtidas para a especificação de outros modelos (as entradas e saídas; figura 3).

Em CommonKADS são definidos seis modelos (formado por quadros) que permitem descrever a partir de diferentes pontos de vista, um ambiente de resolução de problemas. Estes modelos estão relacionados entre si em três grupos, conforme figura 3, abaixo (MENDEZ, 1999):

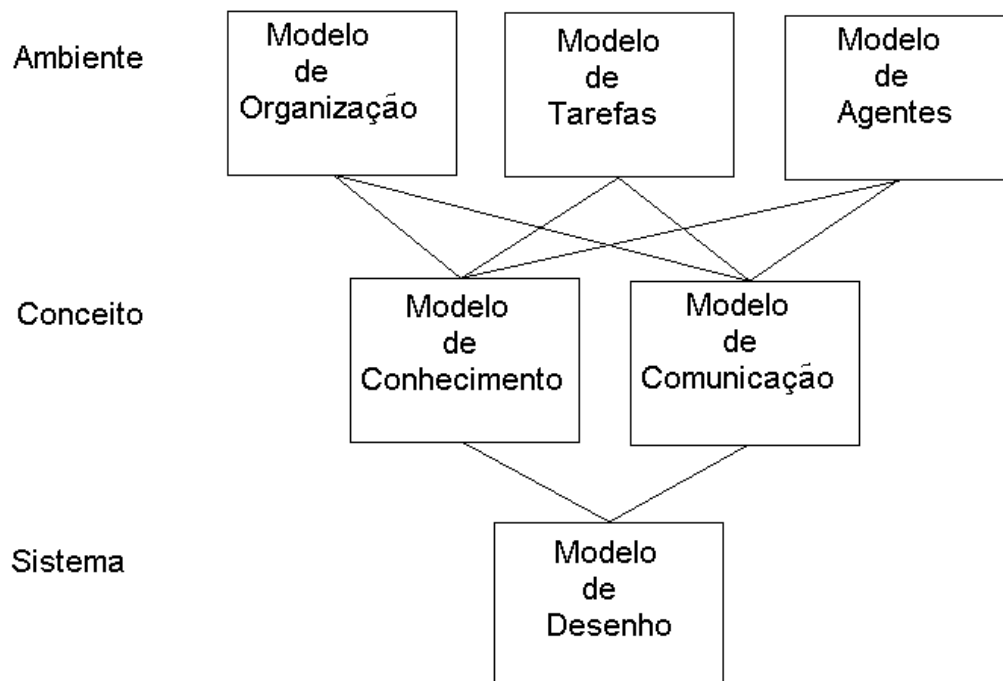


Figura 3: O conjunto de modelos em CommonKADS

Fonte: Adaptado de Mendez (1999) apud SCHREIBER et al (1999).

Estes modelos são agrupados tendo em vista responder perguntas relativas ao desenvolvimento do SBC (MENDEZ, 1999):

- *Por que?* – Quais os benefícios, custos e o impacto sobre a organização.
- *O que?* – Qual a natureza e estrutura do conhecimento e comunicações implicados.

- *Como?* – Como deve ser implementado e qual deve ser a arquitetura de *software*.

A seguir uma breve descrição desses modelos (PENÍN, 2000):

Modelo de Organização (MoO) - seu objetivo principal consiste em identificar e obter informações relativas a:

- características da organização que são relevantes na tomada de decisões;
- aos efeitos que o SBC possa ter após a sua introdução;
- áreas da organização onde é possível introduzir o SBC.

Alem disso, fornece informações às pessoas envolvidas no desenvolvimento do SBC sobre as potencialidades da organização.

Modelo de tarefas (MoT) - sua função é descrever as tarefas relacionadas com o SBC, que realizam uma função determinada dentro da organização (HOOG et al, 1994). Neste contexto, considera-se como tarefa um conjunto de atividades coerentes, definidas como ações desenvolvidas no mundo real, com objetivos bem definidos em um determinado domínio. Em CommonKADS, uma tarefa caracteriza-se como uma parte funcional da organização (modelo de organização) ou como um componente da estrutura do nível de conhecimento (modelo de perícia). O modelo de tarefas considera essas duas visões do conceito de tarefa e, em sua especificação, tenta facilitar a transição entre elas. Assim, pode-se dizer que uma tarefa está caracterizada por um conjunto de entradas/saídas, umas pré-condições em cuja satisfação se baseia sua execução e os agentes encarregados de executar essa tarefa.

Modelo de agentes (MoA) - descreve as propriedades daqueles agentes encarregados de realizar quaisquer tarefas descritas no MoT. Um agente, constitui um executor de tarefas que pode ser um usuário, um SBC ou qualquer outro sistema de *software*. Este modelo tem três utilidades principais:

- descreve os agentes que existem antes da introdução do SBC;
- medir os efeitos que provocam as mudanças na assinalação de tarefas entre os usuários do sistema devido à introdução do SBC;
- descrever funcionalmente os agentes do sistema que serão desenvolvidos como uma parte do projeto.

O principal inconveniente do MoA, segundo seu tratamento em CommonKADS, é o fato de que está orientado a descrever capacidades de interação pessoa-máquina, tornando claramente difícil a extensão do seu uso a agentes distribuídos, em uma rede, que cooperam para alcançar um objetivo. Assim, a especificação do MoA não facilita a descrição de quais são as capacidades de interação entre esses agentes, como se identificam ou como estão definidas na rede, por exemplo.

Modelo de Comunicação (MoC) - encarrega-se de descrever as transações que ocorrem entre os agentes do sistema. Essas transações são tarefas adicionais que é necessário executar em função da distribuição das tarefas entre os diferentes agentes da organização. Isso ocorre quando, por exemplo, assinala-se as sub-tarefas de uma tarefa a vários agentes; é preciso estabelecer um intercâmbio de informação entre elas para chegar a um resultado concreto. Tendo em conta esse objetivo, o modelo de comunicação servirá para um duplo propósito:

- especificar intercâmbio de informação (comunicações) entre os distintos agentes diferenciados na organização;
- validar a qualidade da interface pessoa/máquina desenvolvida para o SBC.

De novo a especificação do MoC em CommonKADS está fortemente centrada na interação pessoa-máquina, e com isso não permite descrever as comunicações entre agentes distribuídos em uma rede: não podemos caracterizar a interação entre agentes enquanto não indicar que protocolos utilizam ou em que condições e como estabelecem uma comunicação entre si. Isto sugere uma ampliação do modelo de CommonKADS que, principalmente, afetaria os modelos de agentes e de comunicação.

Modelo de Perícia (ou de conhecimento, MoP) - descreve o comportamento de um agente em termos do conhecimento aplicado para resolver uma dada tarefa no conjunto de tarefas, por meio da modelagem explícita do conhecimento atualizado pelos agentes em sua operação. Do ponto de vista epistemológico pode ser classificado em (Figura 4):

- conhecimento da aplicação – pode ser estruturado em: (1) conhecimento de domínio, contém o conhecimento específico e relevante do domínio; (2) conhecimento sobre tarefas; capta os objetivos de um agente e as atividades que devem ser executadas para esses objetivos serem alcançados; e (3) conhecimento sobre inferências, descreve como os elementos definidos no conhecimento do domínio são usados para realizar os processos de raciocínio básicos;
- conhecimento sobre a resolução de problemas - conhecimento sobre a organização e o entrelaçamento de tarefas, as inferências e o conhecimento do domínio. Pode ser subdividido em: 1) métodos de resolução de problemas, que capta uma possível forma de aplicar o conhecimento para resolver uma certa tarefa; e 2) conhecimento estratégico, onde é especificado como se selecionam e se aplicam os métodos para construir dinamicamente uma solução ao problema dado.

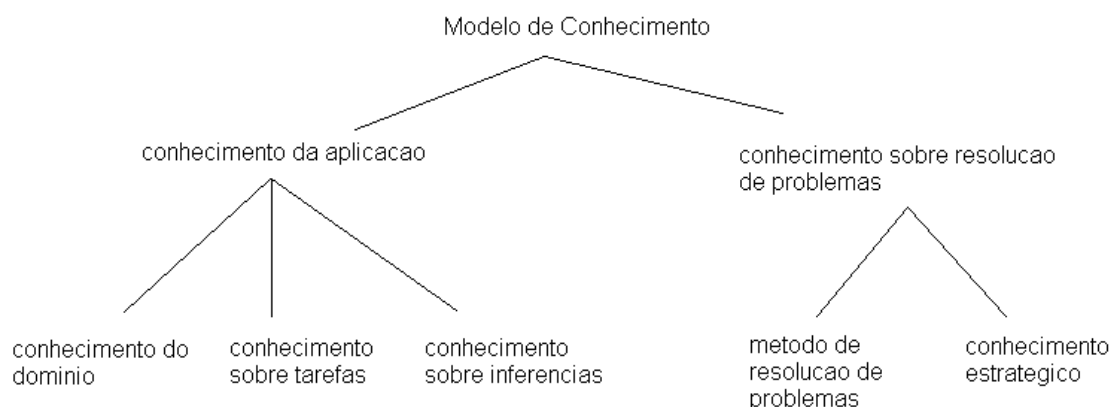


Figura 4: Entidades de conhecimento identificadas no MoP.

Fonte: Adaptado de Penín (2000) apud Wielinga et al (1994).

Modelo de Desenho (MoD) - descreve a realização computacional dos modelos de perícia e comunicação com as condições especificadas pelos modelos de organização, tarefas e agentes. A especificação desse modelo leva a três pontos de vista, que representam os níveis de abstração no desenho do sistema:

- camada de aplicação; orientadas a computar as entidades de conhecimento especificadas no modelo de perícia e além do mais, a estabelecer os critérios que permitem realizar a transição entre esses modelos;
- camada de arquitetura; descreve as estruturas computacionais abstratas nas quais se realizará o desenho do sistema a partir das entidades computáveis especificadas na camada de aplicação;
- camada de plataforma; especifica as características da plataforma de *hardware* e de *software* usadas para implementar o SBC.

Conforme visto, o conjunto de modelos de CommonKADS não se limita a cobrir a fase de aquisição e modelagem do conhecimento - abrange todos os aspectos de interesse no desenvolvimento de um SBC e permite uma especificação dos projetos ao nível de organização (gestão de pessoas e tarefas), do ambiente (agentes de comunicações), o modelo do produto final (conhecimento e desenho). Isso faz do CommonKADS uma metodologia flexível, que se está convertendo em um padrão de fato, ao ser adotada por um bom número de empresas e organizações (PENÍN,2000).

4 ASPECTOS RELACIONADOS À APRENDIZAGEM COOPERATIVA APOIADA POR COMPUTADOR

Nos últimos anos, o esforço despendido no desenvolvimento de ferramentas de aprendizagem pela web tem sido grande - verifica-se isso por meio do site do IMS-Global Learning Consortium (<http://www.imsglobal.org/>): O IMS é um consórcio global constituído por representantes de organizações educacionais, comerciais e governamentais. Está desenvolvendo e promovendo especificações abertas para facilitar atividades de aprendizagem distribuída on-line. O IMS tem como objetivos principais, a definição de especificações técnicas para interoperabilidade de aplicações e serviços em aprendizagem distribuída e oferecer suporte a incorporação das especificações do IMS em produtos e serviços ao redor do mundo. Como fonte de referencia dos principais *softwares* desenvolvidos e em utilização há o *site* (<http://www.c2t2.ca/landonline/>) desenvolvido por Bruce Landon (<http://bruce-landon.douglas.bc.ca/>). Nesse *site*, encontra-se o resultado de uma análise feita sobre uma grande variedade de *softwares* para cursos on-line. A análise centrou-se em especificações técnicas, desenho instrucional, ferramentas e características, facilidade de uso e acessibilidade, potencial para colaboração, nível de padronização de metadados conforme o IMS-Global Learning Consortium.

Em função da pesquisa em Aprendizagem Cooperativa apoiada por Computador ser recente, é importante avaliar se o presente trabalho apresenta alguma consistência em relação ao que se propõe. Para isso adotou-se o trabalho apresentado por Santoro et al(1998), descrito aqui de forma resumida, que pode servir como ponto de partida. Neste trabalho é proposto um *framework* com os principais aspectos relacionados a ambientes de aprendizagem cooperativa apoiados por computador, faz-se um estudo sobre os principais ambientes desenvolvidos com a intenção de facilitar a análise e poder apontar direções para propostas de novos ambientes. Assim, como aspectos mais importantes a serem observados, os autores apontam:

Teorias de Aprendizagem - Um dos fatores mais importantes que regulam a colaboração é a teoria de aprendizagem na qual a interação cooperativa será baseada.

Modelos de Cooperação ou Tipos de Tarefas para Aprendizagem Cooperativa - O suporte computacional a ambientes de aprendizagem pode ser empregado para tratar a cooperação em tipos diferentes de tarefas a serem realizadas. Estas tarefas determinarão o modelo de cooperação proposto pelo ambiente e podem ser enumeradas independentemente do domínio de assunto que está sendo ensinado. Kumar identifica três tipos de tarefas: tarefas cooperativas de aprendizagem de conceitos, tarefas cooperativas para solução de problemas e tarefas cooperativas de desenvolvimento de projetos (KUMAR, 1996).

Domínios - A aprendizagem cooperativa é geralmente mais efetiva em domínios onde as pessoas estão engajadas na aquisição de habilidades, categorização, planejamento conjunto e tarefas que requerem construção de memória coletiva (KUMAR, 1996).

Tecnologias - existem dois tipos de tecnologias que podem ser utilizadas em ambientes de aprendizagem cooperativos: comunicação assíncrona e comunicação síncrona. O uso de uma destas tecnologias ou da combinação delas irá determinar o grau de interação entre indivíduos permitido ou disponibilizado pelo sistema.

Atividades de Trabalho Cooperativo - Dependendo da proposta do ambiente de aprendizagem, pode ser necessário disponibilizar suporte a algumas atividades, tais como: coordenação de atividades, tomada de decisão, representação dos conhecimentos do grupo, compartilhamento de uma base de dados (memória de grupo, percepção da presença e das ações dos demais participantes).

Designação de Papéis - Alguns sistemas associam explicitamente diferentes papéis para usuários, os quais irão possuir direitos distintos, e poderão ser responsáveis por tarefas distintas dentro do processo de aprendizagem.

Relação com Outras Áreas de Pesquisa - Muitos ambientes e ferramentas CSCL combinam técnicas desenvolvidas em outras áreas de estudo, tais como Inteligência Artificial, Realidade Virtual e Banco de Dados, para promover a aprendizagem.

A partir desses aspectos, Santoro et al (1998) propõe um *framework* para o estudo e classificação de ambientes de aprendizagem cooperativa, conforme quadro 2:

Quadro 2 - *Framework* para estudo de Ambientes de Aprendizagem Cooperativa

Aspectos	Possibilidades
Teoria de Aprendizagem	Epistemologia Genética de Piaget, Teoria Construtivista de Bruner, Teoria Sócio-Cultural de Vygotsky, Aprendizagem baseada em Problemas/Instrução ancorada, Cognição Distribuída, Teoria da Flexibilidade Cognitiva, Cognição Situada, Aprendizagem Auto-regulada/ Metacognição, Aprendizagem por Observação
Modelo de Cooperação ou Tipo de Tarefa	aprendizagem de conceitos, solução de problemas, desenvolvimento de projetos, construção de conhecimento, fórum de discussões
Domínio	desenvolvimento do pensamento crítico científico, modelos ecológicos, textos científicos, outros, domínio não específico
Tipo de Interação	Assíncrona, síncrona
Qualidade ou Grau de Interação	Pequena, média, grande
Atividades de Trabalho Cooperativo	coordenação de atividades, tomada de decisão, representação dos conhecimentos memória de grupo, awareness
Implementação/Plataformas	UNIX, Windows NT, Macintosh, WWW
Designação de Papéis	Professor/instrutor, aluno/aprendiz
Relação com outras Áreas de Pesquisa	inteligência artificial, realidade virtual, banco de dados

Fonte: Adaptado de Santoro et al (1998).

Santoro et al (1998), a partir desse *framework*, realizou um estudo sobre os principais ambientes de aprendizagem cooperativa apoiados por computador, com vistas a classificá-los. A seguir, são descritos, de forma resumida, esses ambientes:

N.I.C.E. - Narrative, Immersive, Constructionist/Collaborative Environments - O projeto N.I.C.E. tem como objetivo a construção de ambientes de aprendizagem virtuais para crianças, baseados em teorias de narrativa, construcionismo e colaboração. O sistema foi projetado para ser executado no CAVE, que é um ambiente de realidade virtual do tamanho de um sala, onde várias pessoas podem se mover livremente, tanto física como virtualmente. Principais objetivos: aprendizagem a partir de múltiplas perspectivas, aprendizagem sobre como colaborar com outras pessoas, aprendizagem pelo controle e exploração ativa de

variáveis do ambiente, programação por demonstração, exploração de estruturas de estórias e criação de um produto final.

CSILE : Computer-Supported Intentional Learning Environments - O ambiente CSILE é uma base de dados coletiva, em rede, que contém idéias de estudantes, em formato textual ou gráfico, disponível para todos os participantes. Neste ambiente multimídia, os estudantes geram “nós”, contendo uma idéia, ou uma parte de informação relevante a um tópico em estudo. Os dados são indexados e organizados de tal forma que possam ser acessados por meio de uma série de canais, permitindo, então, que estudantes que estão estudando um tópico em um determinado domínio possam acessar informação relacionada em um outro domínio. A ênfase do CSILE é na aprendizagem cooperativa.

Collaboratory Notebook - é um ambiente multimídia em rede para construção de conhecimento, desenvolvido para ajudar estudantes, professores e cientistas a compartilhar questionamentos sobre os limites do tempo e do espaço. A estrutura de organização da base de dados do ambiente é construída conforme a metáfora da biblioteca, tendo como elementos primários de interface prateleiras de livros, *notebooks*, e páginas. A cada página escrita por um usuário, deve ser associado um ícone, que indica ou descreve aquilo que foi escrito (informação, comentário sobre o que outra pessoa escreveu, questão, conjectura, evidência a favor, evidência contra, plano para ação, ou passo dentro de um plano). As páginas que possuem relacionamento com outras são ligadas pelo sistema por meio de *links* hipermídia com os ícones correspondentes.

CLARE: Collaborative Learning And Research Environment - é um ambiente distribuído de aprendizagem apoiada por computador, cujo objetivo é facilitar a aprendizagem por intermédio da construção colaborativa de conhecimento. Para isto, CLARE provê uma linguagem de representação semi-formal chamada RESRA e um modelo de processo explícito chamado SECAI. RESRA (Representational Schema of Research Artifacts) é uma linguagem de representação de conhecimento semi-estruturada, projetada especificamente para facilitar aprendizagem colaborativa de textos científicos. A medida em que os aprendizes passam pelas atividades

propostas no modelo SECAI, o nível de colaboração cresce e, ao mesmo tempo, uma base de conhecimento é formada.

CaMILE : Collaborative and Multimedia Interactive Learning Environment - é um ambiente assíncrono de suporte à colaboração para Web que tem o objetivo de estimular a aprendizagem e se insere no contexto da abordagem a pesquisas em CSCL - análise em um nível alto de agregação: fóruns de discussões com grupos múltiplos ou uma classe inteira. Todos os acessos ao sistema são realizados a partir de um *browser* Web que acessa um servidor único. A interface do sistema é baseada em formulários e é igual para todos os usuários. As discussões no CaMILE são contextualizadas como em um *newsgroup*; Uma importante diferença entre *newsgroups* e CaMILE é que este ambiente apoia colaboração ancorada.

JavaCAP - é um programa para autoria de casos (Case Authoring Program), onde estudantes de escola média podem desenvolver casos colaborativamente e compartilhar suas experiências, enquanto aprendem ciência por meio da solução de problemas e projetos. Os professores podem colaborar personalizando o suporte e indexando as facilidades do JavaCAP para melhor atender às necessidades dos seus alunos. Os estudantes podem carregar objetos multimídia capturados durante seus estudos, e eventualmente publicar seus casos na Web. O ambiente JavaCAP foi desenvolvido a partir de bibliotecas de casos construídas no EduTech Institute para o curso de engenharia. O foco do ambiente está em prover facilidades para articulação dos estudantes e promover reflexão e habilidades meta-cognitivas. Para isto, foi desenvolvida uma *sessão de autoria*, onde os estudantes podem resumir sua experiência como uma equipe, compondo um caso que descreve várias *cenários* de acontecimentos dentro de um projeto.

PENCACOLAS: PEN Computer Aided Composing cOLIAborative System - foi desenvolvido para apoiar a aprendizagem da produção de documentos de forma colaborativa, suportando todas as fases que acontecem durante este processo: geração de idéias, planejamento, composição e revisão. A aprendizagem não pode ser efetiva se a interface computacional não permitir uma interação homem-computador fluente e uma efetiva transferência de toda a informação metatextual presente na escrita colaborativa; por isso, a técnica de *pen-computing* foi

considerada uma boa candidata a realizar esta tarefa, pois permite que os usuários continuem escrevendo da forma como estão acostumados. O objetivo principal deste projeto é a convergência entre escrita cooperativa e aprendizagem.

Dialogue Monitor - é um sistema inteligente cooperativo, baseado na análise de diálogos entre estudantes colaborando em tarefas de solução de problemas. Este sistema é a parte central de um programa educacional assistido por computador que atua como um estudante simulado, colaborando com um estudante real na solução de um problema. A análise de diálogos entre estudantes em tarefas cooperativas permitiu a construção do protótipo do Dialogue Monitor, cujo modelo de arquitetura servirá como base para o desenvolvimento de sistemas inteligentes cooperativos. O modelo do Dialogue Monitor é constituído por cinco módulos, que funcionam separadamente: solução de problemas, processador de diálogos, componente alteração, processador de foco central, interface baseada em menus.

ARCOO - Aprendizagem Remota Cooperativa Orientada a Objetivos - foi desenvolvido com o objetivo de apoiar aprendizagem cooperativa em ambientes distribuídos, onde ocorre interação entre pares na busca da solução de um problema, na realização de um projeto. ARCOO é um Ambiente para ser usado em atividades presenciais ou Remotas (usando redes de computadores), projetado para auxiliar atividades Cooperativas que são realizadas sob a Orientação de tutores visando alcançar certos Objetivos de aprendizagem. A metáfora adotada no ambiente é a sala de estudos, onde o aprendiz está envolvido com um projeto que irá desencadear a aprendizagem de novos conceitos ou o aprofundamento de outros já desenvolvidos em sala de aula. A sala de estudos possui os seguintes recursos: estante de livros, arquivos com informações, tela para assistir conferências, “auxiliares invisíveis”.

SAACI - Sistema de Apoio à Aprendizagem Colaborativa na Internet - O sistema SAACI tem como objetivo prover um conjunto de funcionalidades básicas que são capazes de suportar o desenvolvimento de atividades de aprendizagem colaborativa na Internet: (1) comunicação síncrona e assíncrona entre membros de um grupo; (2) suporte à tomada de decisões; (3) suporte à representação dos conhecimentos de um grupo; (4) suporte ao compartilhamento de informações entre o grupo; (5)

suporte à coordenação do grupo; e (6) suporte à percepção da presença e ações do grupo. O SAACI é constituído de dois blocos que agrupam as funcionalidades descritas: Administração e Colaboração (responsável pelo suporte às atividades colaborativas desenvolvidas pelos grupos. Composto pelos sub-blocos Co-autoria, Comunicação, Navegação e Percepção).

AlgoArena - é uma ferramenta que tem como objetivo transmitir noções de projeto de *software* para iniciantes, com o desenvolvimento da habilidade para o pensamento algorítmico e a visão sistemática das coisas. AlgoArena é um jogo de simulação de um tradicional esporte japonês - a luta de sumô. Os estudantes devem programar as ações de seu próprio lutador, utilizando uma linguagem de programação baseada em LOGO. Para tornar o lutador mais forte, os estudantes são encorajados a analisar sistematicamente a situação, desenvolver táticas melhores e incorporá-las ao programa. Estas atividades típicas de solução de problemas são comparáveis ao processo envolvido no desenvolvimento de *software*. O AlgoArena é baseado na teoria da cognição situada, onde a cultura em que está inserido o processo de aprendizagem é fundamental, e os sujeitos envolvidos, gradualmente, tornam-se parte de uma comunidade de prática.

PIE - Probability Inquiry Environment - foi desenvolvido com objetivo de observar como representações externas (textuais e icônicas) podem mediar conversas face-a-face entre estudantes, provendo suporte ao discurso matemático. Este ambiente ilustra um tipo de *software* educacional que reconhece práticas de discurso como um componente central na aprendizagem humana e utiliza a tecnologia para apoiar este tipo de classes e melhorar o aprendizado. Cada atividade PIE consiste de seis passos: regras, tentativa, prognóstico, jogo, conclusão, princípios.

Nos quadros a seguir, os ambientes analisados e classificados de acordo com o *framework* proposto:

Quadro 3a - Classificação dos Ambientes segundo o *Framework* proposto

Ambiente	NICE	CSILE	Collaboratory Notebook	CLARE	CaMILE
Aspectos					
Teoria de Aprendizagem	Construtivismo Teoria de Piaget	Construtivismo	Cognição situada	Construtivismo	Não é explícito nas referências
Modelo de Cooperação ou Tipo de Tarefas	Desenvolvimento de projeto Construção de conhecimento por meio de narrativa de estórias	Construção de conhecimento	Desenvolvimento de projeto Construção de conhecimento por meio do diálogo sobre um projeto	Construção de conhecimento por meio do modelo SECAI	Fórum de discussões
Domínio	Modelo Ecológico	Não específico	Prática da Ciência	Textos científicos	Não específico
Tipos de Interação	Síncrona e Assíncrona	Assíncrona	Assíncrona	Síncrona, Assíncrona	Assíncrona
Qualidade ou Grau de Interação	Grande	Média	Média	Grande	Pequena
Atividades de Trabalho Cooperativo	Tomada de decisões Representação de conhecimentos Memória de grupo Awareness	Representação de conhecimentos Memória de grupo	Representação de conhecimentos Memória de grupo	Representação de conhecimentos Memória de grupo	Memória de grupo Awareness
Plataformas	Ambiente de Realidade Virtual CAVE Outros ambientes de realidade virtual	Macintosh, UNIX	Aplicação Macintosh, cliente de uma base de dados Oracle, comunicando-se através de TCP/IP	UNIX/ X-Windows	WWW
Relação com outras Áreas de Pesquisa	Realidade Virtual Inteligência Artificial	Banco de Dados	Banco de Dados	Não possui	Não possui

Fonte: Adaptado de Santoro et al (1998).

Quadro 3b - Classificação dos Ambientes segundo o *Framework* proposto.

Ambiente	JavaCap	PENCACOLAS	Dialogue Monitor
Aspectos			
Teoria de Aprendizagem	Não é explícita nas referências, porém baseia-se em uma metodologia de argumentação baseada em casos	Não é explícito nas referências	Aprendizagem baseada em Problemas/Instrução ancorada Cognição situada
Modelo de Cooperação ou Tipo de Tarefas	Solução de problemas de Desenvolvimento de projetos Construção de conhecimento	Construção de conhecimento com a edição cooperativa de documentos	Solução de problemas
Domínio	Não possui	Não possui	Não específico
Tipos de Interação	Assíncrona	Síncrona e Assíncrona	Síncrona
Qualidade ou Grau de Interação	Pequena	Grande	Grande
Atividades de Trabalho Cooperativo	Representação de conhecimentos Memória de grupo	Awareness Memória de grupo Coordenação de Atividades	Memória de grupo Coordenação
Plataformas	WWW	UNIX (Solaris 2.4) para a Sparcstation Windows for Pen Computing sobre Windows for Group 3.11 para os PCs e o Notebook. Futuro: WWW	-
Designação de Papéis	Não possui	Não possui	Não possui
Relação com outras Áreas de Pesquisa	Não possui	Não possui	Inteligência Artificial

Fonte: Adaptado de Santoro et al (1998).

Quadro 3c - Classificação dos Ambientes segundo o *Framework* proposto.

Ambiente	ARCOO	SAACI	AlgoArena	PIE
Aspectos				
Teoria de Aprendizagem	Aprendizagem baseada em problemas	Não é explícito nas referências	Cognição situada	Cognição situada
Modelo de Cooperação ou Tipo de Tarefas	Solução de problemas Desenvolvimento de projetos	Desenvolvimento de Projetos	Solução de problemas	Solução de problemas
Domínio	Não específico	Não específico	Jogo de simulação de luta de sumô	Probabilidade
Tipos de Interação	Síncrona e Assíncrona	Síncrona e Assíncrona	Síncrona	Síncrona
Qualidade ou Grau de Interação	Grande	Grande	Grande	Grande
Atividades de Trabalho Cooperativo	Coordenação de atividades Tomada de decisão Representação dos conhecimentos Memória de grupo Awareness	Coordenação de atividades Tomada de decisão Representação dos conhecimentos Memória de grupo Awareness	Coordenação de atividades Awareness	Coordenação de atividades Awareness
Plataformas	Lotus Notes (protótipo ESTILINGUE) Internet (InterARCOO)	WWW	-	Macintosh
Designação de Papéis	Não possui	Não possui	Não possui	Não possui
Relação com outras Áreas de Pesquisa	Não possui	Não possui	Não possui	Não possui

Fonte: Adaptado de Santoro et al (1998).

A partir das tabelas acima, foram destacadas algumas observações:

- A maioria dos ambientes privilegia a construção colaborativa de algum tipo de conhecimento, fundamentando-se em teorias de aprendizagem construtivistas;
- Os ambientes que desenvolvem modelos mais elaborados disponibilizam interações de mais alto grau e combinam técnicas provenientes de outras áreas de pesquisa;
- A memória de grupo e a representação de conhecimento são características presentes na maioria dos sistemas.

5 MODELO PROPOSTO SEGUNDO A METODOLOGIA CommonKADS

5.1 Modelo de Organização

Tem como objetivo principal identificar e obter informações relativas às características da organização que são relevantes na tomada de decisões, aos efeitos que o sistema possa ter após a sua introdução e as áreas da organização onde é possível a sua introdução. O modelo de Organização será descrito a partir da especificação de um conjunto de planilhas (*worksheets*).

5.1.1 Problemas, Soluções e Contexto

O quadro 4 mostra o primeiro *worksheet*, o qual lista os problemas organizacionais percebidos, caracteriza-os no contexto organizacional (os quais, para o propósito desta análise são considerados invariáveis) e provê uma lista de prováveis soluções. Na primeira linha são listados dois problemas fundamentais:

- o fato de que tanto os professores, quanto os alunos demonstram-se insatisfeitos com os métodos tradicionais de ensino que ainda são bastante utilizados em sala de aula. Nesse sentido, os estudantes pedem por métodos de ensino que tornem a aprendizagem um processo mais agradável: eles querem mais participação/interação nas aulas, com mais motivação; E os professores, por sua vez, reclamam por alunos mais motivados e ferramentas que os auxiliem na preparação e durante as aulas;
- o fato de que avaliar é uma atividade complexa, sujeita à falhas, e não satisfazem totalmente os professores, que sentem necessidade de utilizar instrumental adequado para auxiliar neste processo de forma a garantir uma avaliação eficiente. E os alunos, por sua vez, sentem-se, muitas vezes, prejudicados pelas falhas do processo avaliativo.

Esses problemas podem ser solucionados com o uso das Tecnologias de Informação e Comunicação: elas funcionam como disseminadores de conhecimento, permitem liberar estudantes e professores das limitações de tempo/espço, permitem enriquecer o ensino com recursos de multimídia/hipermídia, interação,

simulações e o estudo em grupo, de forma cooperativa. A tecnologia serve como fator motivador e pode auxiliar o processo Ensino x Aprendizagem – desde a preparação de aulas até ao controle do que é ensinado/aprendido (MARTINS, 2000).

A segunda linha do quadro 4 descreve o contexto organizacional. Estes elementos são assumidos como fixos, isto é, admite-se que a missão e os objetivos da organização serão os mesmos durante o tempo de vida concebido para o projeto. Pode ocorrer que a conclusão do projeto determine mudanças nos objetivos da organização – nesse caso, as mudanças estarão fora do escopo atual. A missão neste trabalho reflete a idéia de que tanto os professores quanto os estudantes, buscam novas formas de ensinar/aprender que tenham como resultado um processo de ensino/aprendizagem mais eficiente e mais agradável.

Quadro 4: Contexto organizacional do sistema com prováveis soluções

Modelo de Organização	Problemas e Oportunidades WorkSheet MG1
Problemas e Oportunidades	Professores e alunos insatisfeitos comos métodos de ensino tradicionais empregados Avaliar é uma atividade complexa e causa insatisfação Possibilidade de promover maior interação entre professor x aluno O uso de tecnologia pode servir como fator motivador Auxilia o processo Ensino x Aprendizagem
Contexto Organizacional	Missão: Promover, através do uso de recursos tecnológicos e estratégias de aprendizagem cooperativa, um processo de Ensino x Aprendizagem de alta qualidade. Fatores Externos - Estrutura de suporte computacional Estratégias: Utilização de método didático de aprendizagem cooperativa Utilização de recursos tecnológicos inovadores e de baixo custo. Soluções: Especificação de modelo de conhecimento de aprendizagem cooperativa Desenvolvimento de Sistema Especialista de apoio ao ensino Utilização de recursos tecnológicos: PDA, Digitalizador de Quadro Negro, Projetor Multimídia, Rede sem fio. Programa de treinamento para professores.

5.1.2 Descrição da área foco na organização

Este *worksheet*, quadro 5, descreve a parte da organização sobre a qual o projeto está focado. Está dividido em cinco partes: as duas primeiras, organização e estrutura, são representadas por um gráfico. A figura 5 mostra a estrutura atual da organização. Esta figura combina a parte estrutura com a parte pessoas, indicando os papéis das pessoas na organização. Em muitas organizações, os papéis das pessoas estão estritamente relacionados com sua posição física: em tais casos este nível de combinação faz sentido. Vê-se aqui que a organização está estruturada em quatro setores: O “Departamento” que é o responsável pela definição de cursos, professores, disciplinas e por definir políticas relativas ao desempenho docente, a “Secretaria do Departamento” que se encarrega de atender aos serviços burocráticos necessários ao desempenho das aulas, a “Biblioteca” para apoio aos materiais didáticos necessários e os “Serviços Universitários” que tem como função informar os dados necessários dos alunos.

Quadro 5: Descrição da área foco na organização

Modelo de Organização	Aspectos Variantes WorkSheet MO2
Estrutura	Ver Figura 5
Processos	Ver Figura 7
Recursos	Banco de Dados de Conteúdos Software de Apoio a Aprendizagem Cooperativa
Conhecimento	Conhecimento sobre aprendizagem cooperativa, relações sociais
Cultura e Poder	Organização hierárquica Existe pouca participação dos alunos Processo voltado somente ao ensino

Com esta estrutura da organização, o ambiente docente descrito neste trabalho inclui a introdução de elementos de hardware e software no departamento e nas salas de aulas. Assim, pode-se dizer, como primeira aproximação, que os professores e alunos poderão ter acesso à informação necessária para o desenvolvimento de suas atividades dentro da sala de aula e também fora – no caso dos professores, por exemplo, a partir da sala do departamento. Como segunda aproximação, está previsto estender o ambiente docente para pode

solicitar/consultar, a partir de servidores, informação tais como disponibilidade de livros, informações sobre alunos, ocupação de salas e outras.

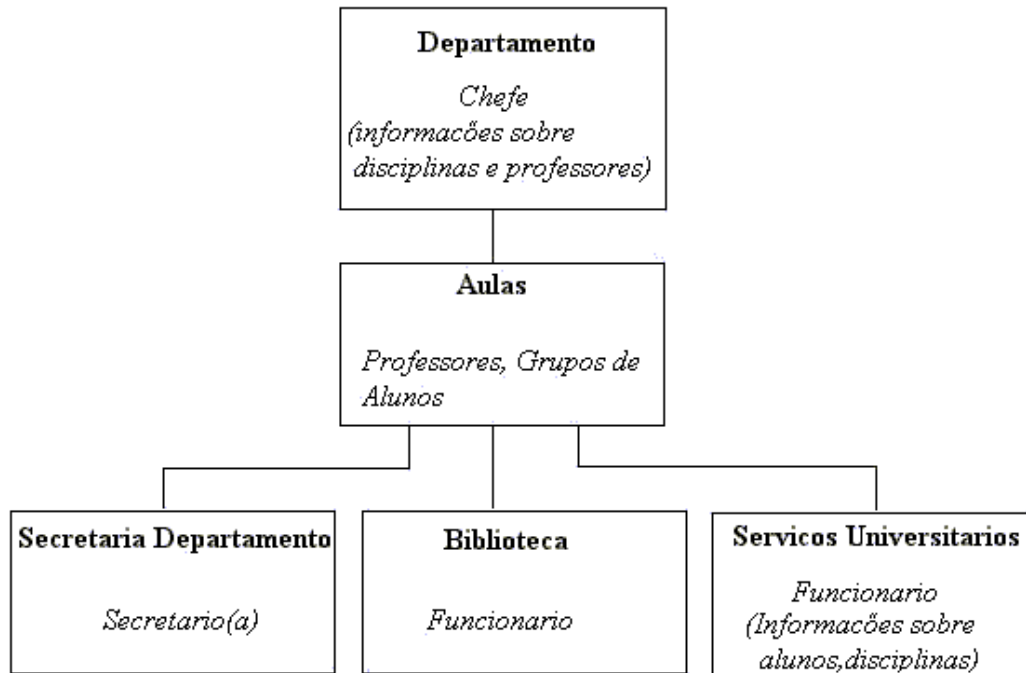


Figura 5: Estrutura e Pessoas na situação atual

5.1.3 Descrição do Ambiente Docente

O ambiente é constituído por um conjunto de dispositivos de hardware para dar a suporte às tarefas de aprendizagem cooperativa (SÁNCHEZ, 2002). De acordo com suas capacidades, estes dispositivos podem ser classificados como (figura 6):

- *recursos hardware de ensino*. Eles são usados pelo professor e, eventualmente pelos alunos, para facilitar o andamento de explicações:
 - *Mimio*. Digitaliza o que está escrito no quadro branco e salva como um arquivo imagem;
 - projetor multimídia. Para exibir as apresentações de alunos e professores.

- *dispositivos de usuários* (ou PDA's). Eles são usados pelos agentes humanos para gerenciar os recursos de hardware do ambiente, para cadastrar/consultar dados pessoais (agenda, materiais didáticos, etc.), ou para interagir com outros agentes para compartilhar informações em tempo real (tais como explicações de conteúdos ou recebimento de respostas provenientes de formulários de avaliações);
- *dispositivos de suporte*. Eles habilitam a comunicação entre os usuários e os dispositivos de ensino identificados no ambiente:
 - *servidor de aula*. Sua função é enviar/receber a informação do/para o *Mimio* e projetor multimídia. Ele executa componentes de software de serviços servidores que efetuam um controle de baixo nível dos recursos de hardware (RIERA et al, 2001);
 - *ponto de acesso*. Facilita a comunicação entre PDA's e os componentes do ambiente.

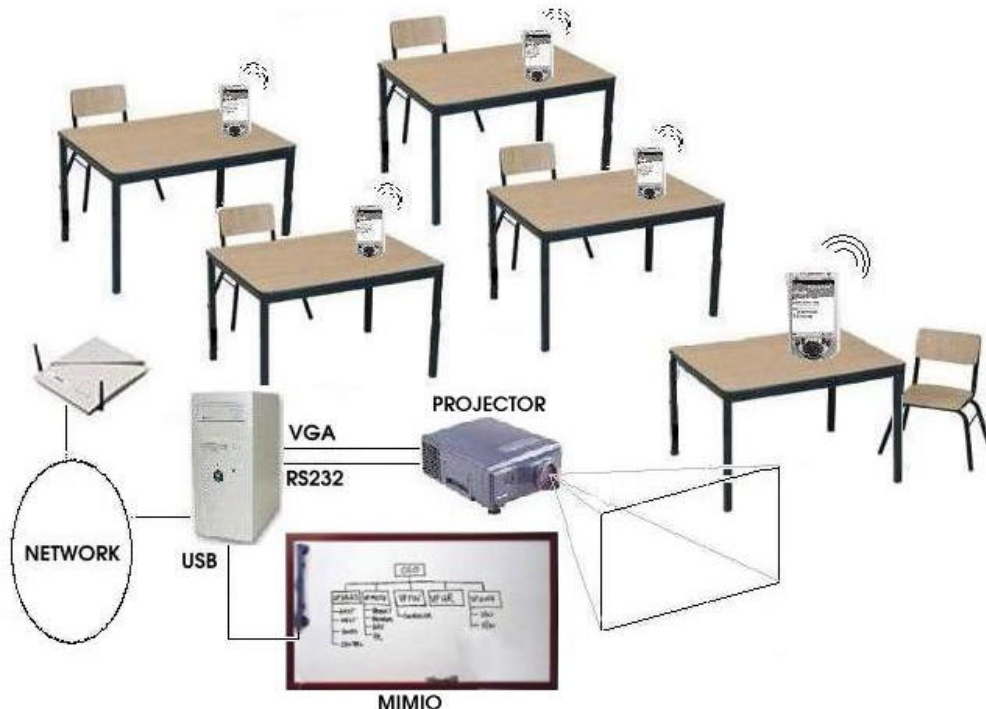


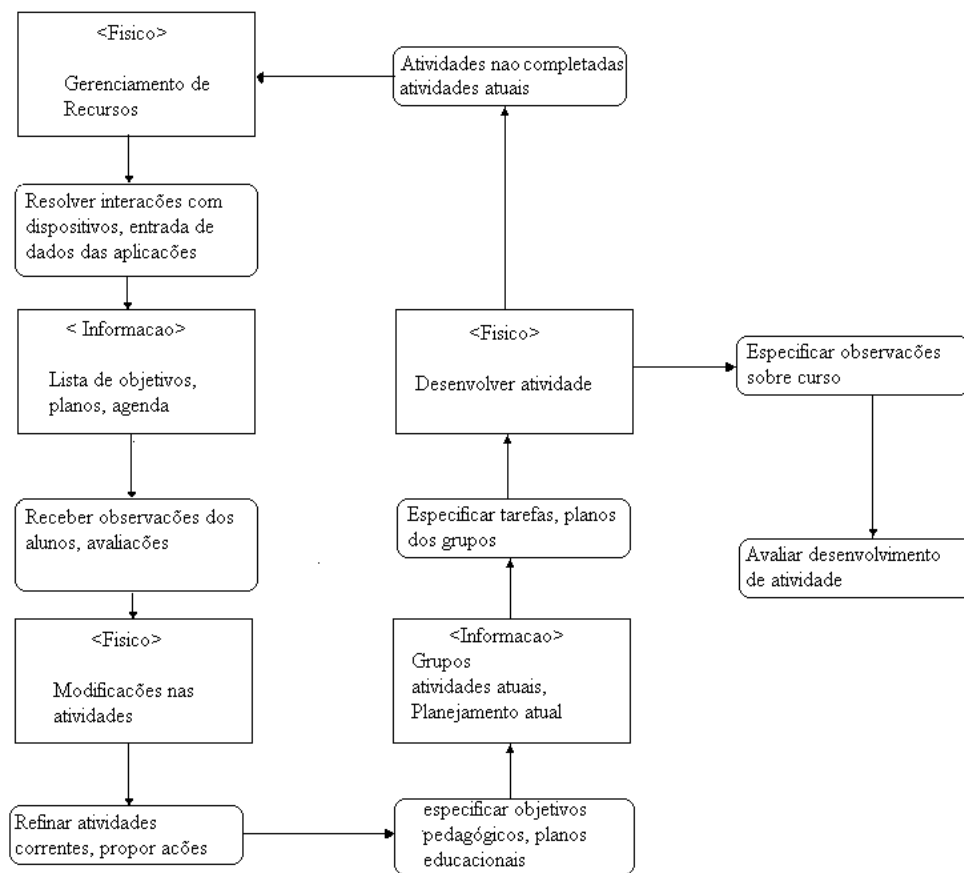
Figura 6: Descrição do ambiente docente com equipamentos integrados a uma rede
 Fonte: Lama et al (2002)

5.1.4 Descrição dos processos da organização

A figura 7, descreve os principais processos na organização. Esses processos são todas as etapas relativas ao trabalho docente:

- *especificar objetivos pedagógicos* - Inicialmente, são vistos os objetivos pedagógicos para uma determinada aula e os planos específicos são traçados, isto é, as técnicas e os recursos didáticos a serem empregados nessa aula são definidos. Em seguida, obtêm-se as informações relativas aos grupos de alunos:
 - Dados necessários à formação dos grupos, se for a primeira aula;
 - Informações relativas a atividades pendentes;
 - Informações relativas ao planejamento das atividades a serem desenvolvidas.
- *especificar tarefas* – as tarefas a serem desempenhadas pelos alunos, os conteúdos a serem estudados, exercícios e avaliações, são especificados e os planos de atividades dos grupos são determinados;
- *desenvolver atividades pendentes e atuais* - as atividades pendentes e as atuais são desenvolvidas com o suporte dos recursos tecnológicos do ambiente, que são o Mimio, o servidor de sala de aula, os PDAs e os softwares educacionais. Estas atividades são, por um lado, o trabalho a ser realizado na sala de aula pelos alunos e, por outro lado, as observações que o professor vai anotando sobre o comportamento dos alunos e a evolução do seu trabalho. Caso os objetivos não sejam atingidos, o professor poderá tomar as decisões mais adequadas para o seu êxito, pode “refinar” as atividades executadas pelos alunos e eventualmente fazendo uso do projetor multimídia e do Mímio para explicar determinados conceitos que não ficaram claros;
- *resolver interações com dispositivos* – os programas realizam uma verificação de baixo nível dos recursos *hardware* e, habilitam ou não, a comunicação entre os diferentes elementos do ambiente educativo;

- *receber informações dos alunos* - Após o desenvolvimento destas atividades, os alunos alimentam o sistema com as suas observações pertinentes à aula e com essas informações, é feito um “refinamento” das atividades. Esse “refinamento” seria algo como, por exemplo, um redimensionamento quantitativo das atividades baseado no desempenho da última aula. A partir disso, ações educacionais são propostas e servirão de base para os objetivos pedagógicos da próxima aula.



Legenda

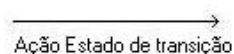
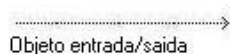
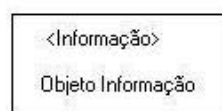
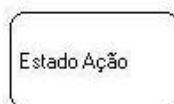


Figura 7: Processos na situação atual. É usada mesma notação aplicada em diagramas de atividades UML. (ver legenda).

5.2 Modelo de Agentes

Neste modelo, as informações são reorganizadas de forma que seja possível vê-las a partir da perspectiva dos agentes envolvidos. Os agentes poderão eventualmente ter (ter de fazer) seu próprio trabalho (sua própria função) na organização. O sucesso do sistema depende de sua disposição e habilidade de cooperar.

Os quadros 6, 7, 8 são *worksheets* para os agentes identificados na organização (ou domínio): Professor, Aluno e Recursos. Destacam-se as funções diretamente mais envolvidas com as soluções propostas. As informações acrescentadas nestes *worksheets* revelam muitas habilidades e competências requeridas pelos agentes. Considerando-se mudanças organizacionais propostas, as necessidades para estas habilidades poderão ser ampliadas depois.

5.2.1 Professor

O Professor, quadro 6, atua como um agente facilitador do processo Ensino/Aprendizagem segundo uma abordagem social do ensino: os alunos interagem cooperando entre si e de forma participativa nas aulas. Para isso o professor deve conhecer os critérios necessários à formação dos grupos de alunos, os requisitos necessários à condução das atividades pedagógicas segundo a abordagem cooperativa da aprendizagem e a forma de avaliá-los. No processo, o professor conta com um conjunto de recursos tecnológicos e um Sistema de Gerenciamento Inteligente de Recursos (SGIR) (LAMA, 2002; SÁNCHEZ, 2002) para apoiá-lo na condução das atividades. As atividades desempenhadas pelo professor são baseadas nas formas organizativas de aula e técnicas didáticas sugeridas por (AGELET et al., 1997) e (DOMÈNECH, 1997) citadas no capítulo 2 (itens 2.2.3 e 2.2.4):

- o professor apresenta os objetivos didáticos aos alunos;
- realiza uma avaliação inicial;
- prioriza os objetivos de cada aluno;
- explica detalhadamente os conteúdos e os critérios de avaliação.

- os grupos realizam as atividades (pendentes ou atuais), de acordo com uma das técnicas seguintes:
 - quebra-cabeças (Jigsaw);
 - grupos de investigação;
 - técnica STAD (Student Team-Achievement Divisions);
 - TGT (Teams-Games Tournaments);
 - tutoria por iguais.
- os alunos realizam uma auto-avaliação;
- o professor realiza uma avaliação individual;
- realiza-se a avaliação de cada grupo;
- o professor planeja a recuperação individual – plano de trabalho individual ou plano de trabalho para pequenos grupos homogêneos.

É imprescindível o trabalho prévio do professor de preparação do material e identificação dos conteúdos que irá utilizar com os alunos. É conveniente que o professor planeje tarefas em que sua presença seja mais necessária e outras em que os alunos não necessitem tanto de sua ajuda.

Assim, de posse desses recursos, pode-se imaginar, como exemplo, de que forma o professor pode organizar uma aula:

- (1) inicialmente, utiliza o projetor multimídia para apresentar os objetivos didáticos/pedagógico aos alunos;
- (2) após essa apresentação, para a avaliação inicial, envia aos PDA's um formulário em que os alunos indicarão os conhecimentos prévios em relação aos objetivos apresentados;
- (3) com essas informações o professor define prioridades de acordo com os objetivos dos grupos ou, individualmente, dos alunos. Neste caso, a avaliação inicial facilita a definição da parte da tarefa que cada grupo irá desenvolver;
- (4) em seguida, o professor pode utilizar o projetor multimídia e o Mímio para explicar conteúdos, técnicas didáticas a serem aplicadas e os critérios de avaliação;

- (5) logo após, os alunos começam a desenvolver as atividades de acordo com técnica didática escolhida pelo professor - essas atividades estão detalhadas no modelo de tarefas e definidas no modelo de conhecimento do agente, descrito a seguir no item 3;
- (6) após os alunos desenvolverem suas atividades, o professor procede às avaliações: a auto-avaliação dos alunos, avaliação individual e avaliação de cada grupo conforme critérios apresentados inicialmente. Como exemplo de forma de avaliar, o professor poderia enviar questionários aos PDA's;
- (7) ao final, o professor planeja a recuperação individual e;
- (8) elabora o plano de trabalho individual ou plano de trabalho para pequenos grupos homogêneos.

O quadro a seguir apresenta um resumo descritivo do agente professor:

Quadro 6: O agente professor: sua função e a relação com outros agentes envolvidos

Modelo de Agentes	WorkSheet Agentes MA-1
Nome	Professor
Organização	Desempenha o seu papel tanto na sala de aula como na sala do departamento.
Envolvido em	Promover o ensino em um ambiente de aprendizagem cooperativa.
Comunica-se com	Grupos de Alunos SGIR: Sistema de Gerenciamento Inteligente de Recursos
Conhecimento	Critérios para a formação de grupos. Regras para o desempenho, pelos alunos, de atividades de forma cooperativa. Critérios para avaliação da aprendizagem.
Outras competências	Habilidades para refinar atividades em andamento e propor ações pedagógicas
Direitos e Deveres	Garantir que os alunos tenham oportunidades iguais de aprender, que cooperem entre si e que a aprendizagem ocorra de fato.

5.2.2 Os alunos

Os alunos, quadro 7, participam em grupos e a cooperação ocorre com o apoio de computadores de mão (PDAs), por meio dos quais eles podem desenvolver atividades didáticas: compartilhar informações em tempo real com outros grupos, podem solicitar informações, material didático de apoio e responder a avaliações,

interagindo com um Sistema de Gerenciamento Inteligente de Recursos – SGIR. A existência deste sistema é imprescindível para a realização destas tarefas, uma vez que ele é quem facilita a interação direta entre alunos e professor: o SGIR recolherá a informação e/ou consultas enviadas pelos alunos durante o desenvolvimento da atividade e exibirá estes dados em tempo real no PDA do professor. Para realizar estas ações, os alunos precisam de um conhecimento mínimo de informática, necessário à utilização desses recursos tecnológicos.

Quadro 7: O agente alunos: sua função e a relação com outros agentes envolvidos

Modelo de Agentes	WorkSheet Agentes MA-2
Nome	Alunos
Organização	Desempenha o seu papel na sala de aula.
Envolvido em	Aprender em ambiente de aprendizagem cooperativa.
Comunica-se com	Professor SGIR : Sistema de Gerenciamento Inteligente de Recursos
Conhecimento	Conhecimento básico para o manuseio de recursos: Computadores de mesa e PDA's.
Outras competências	Conhecimento sobre Internet, eMail.
Direitos e Deveres	Efetuar tarefas planejadas, avaliações e fazer observações solicitadas.

5.2.3 O Sistema Gerenciador Inteligente de Recursos

O Sistema Gerenciador Inteligente de Recursos – SGIR, quadro 8, é utilizado tanto pelo professor, como pelo aluno e tem como propósito gerenciar informações relativas ao curso, ao planejamento de aulas, à elaboração de atividades pedagógicas, às avaliações e ao desempenho de atividades educacionais durante as aulas. As principais funções do SGIR desde o ponto de vista das atividades requeridas pelo professor são as seguintes:

- possibilitar agendamentos, anotações, verificar calendário;
- controle sobre apresentação, em PowerPoint, por exemplo, de objetivos pedagógicos, de conteúdos, de conteúdos pré-gravados com o Mimio – funções consultar, iniciar, avançar, voltar;
- disponibilizar formulário (múltipla escolha, por exemplo) para avaliação inicial com opção de preenchimento, alteração e consulta por meio do PDA;

- fornecer estatística das respostas da avaliação inicial: essa estatística ajuda o professor na definição (proposição) de atividades conforme objetivos dos alunos;
- auxiliar na execução da atividade a ser aplicada segundo a técnica didática escolhida – para as técnicas i e ii, controlar o acesso do material didático (conteúdo) que vai estar (ou não) acessível para a consulta de cada grupo de alunos durante a resolução da tarefa solicitada (atividade de co-autoria com o PDA, por exemplo). Para as técnicas iii e v tornar disponíveis os roteiros individuais com questionários de avaliação (múltipla escolha, por exemplo) – roteiros de conteúdos ou casos, a serem seguidos por cada aluno (aprendizagem baseada em casos por meio do PDA);
- fornecer informações sobre como está o andamento das atividades (resolvidas e pendentes) e estatísticas sobre, por exemplo, quem participou na atividade de co-autoria, as mensagens trocadas entre os grupos ou individualmente (entre agentes) e sobre o tempo de cada grupo na execução das tarefas. Essas informações ajudam na avaliação de resultados e servem como suporte para a redefinição (ou “refinamento”) de atividades a serem desempenhadas posteriormente.

As funções do SGIR, tendo em vista as necessidades dos alunos:

- possibilitar fazer anotações, verificar calendário e agenda de atividades definidas pelo professor, informes e resultados de avaliações;
- em atividades de cooperação - Permitir troca de mensagens entre os grupos, a visualização do trabalho dos outros grupos e o trabalho final da turma;
- permitir acesso a software educacional (Sistema Tutor Inteligente, por exemplo);
- recuperar atividades pendentes para continuação;
- permitir o acesso a informações pessoais, internet;
- visualizar conteúdos disponíveis. (apostilas, apresentações em software como o *PowerPoint*, por exemplo, e aulas gravadas pelo Mimio).

No uso dos recursos por parte dos alunos e professores, o SGIR deverá controlar o seu acesso e resolver interações com os *drivers* dos respectivos recursos. Por exemplo, quando o usuário (professor ou aluno) realiza uma apresentação em PowerPoint a partir do seu PDA, manejará uma interface gráfica adequada para tal fim. Esta interface entra em contato com o SGIR para que este execute as ações necessárias para ativar o projetor multimídia com as preferências do respectivo usuário, acessar a base de dados com o objetivo de selecionar o arquivo da apresentação e executar o PowerPoint com este arquivo. É importante enfatizar que estas operações far-se-ão de forma transparente ao usuário, que deverá preocupar-se unicamente em ir avançando na apresentação fazendo uso dos elementos da interface gráfica reservados para isto.

Quadro 8: O agente SGIR: sua função e a relação com outros agentes envolvidos

Modelo de Agentes	WorkSheet Agentes MA-3
Nome	Sistema Gerenciador Inteligente de Recursos
Organização	Desempenha o seu papel dentro da sala de aula, mesmo que esteja relacionado com componentes que podem estar situados na sala de departamento (computador pessoal do professor).
Envolvido em	Fornecer informações sobre curso, dados de aluno, lista de objetivos pedagógicos, atividades educacionais prédesenhadas, serviços para promover interação aluno X professor, fornecer material adicional ou recursos aos alunos, facilitar o processo de avaliação pelo professor e gerenciar o acesso aos recursos de hardware.
Comunica-se com	Professor Alunos e Grupos de Alunos
Conhecimento	Funcionamento e controle dos dispositivos de <i>hardware</i> e da base de dados.
Outras competências	Facilitar ao professor e aos alunos o manuseio dos recursos do ambiente docente.
Direitos e Deveres	Gestão de recursos transparentes aos usuários do sistema.

5.3 Modelo de Comunicação

O modelo de comunicação descreve as interações que ocorrem entre os agentes descritos no modelo correspondente de CommonKADS (descrito no item 5.2). Estas interações denominam-se *transferências* entre os diferentes agentes do domínio. Uma transferência indica um requerimento de um agente dirigido a um segundo agente para que realize uma determinada ação. Neste sentido pode-se falar de:

- *transferência de informação.* Os agentes precisam comunicar determinada informação, entre si, necessária à execução de alguma tarefa. Por exemplo, o professor precisa dispor da informação sobre os objetivos pedagógicos da disciplina, os conteúdos e os dados pessoais dos alunos, para poder desenhar a atividade a realizar na sala de aula (tarefa identificada no seu modelo de conhecimento). Para isso o professor precisa realizar uma transferência de informação com o sistema de gestão de recursos, que é o encarregado de controlar o acesso à informação disponível na base de dados;
- *transferência de conhecimento.* Um agente necessita de outro agente para efetuar uma dada operação no ambiente (ou domínio). Por exemplo, quando o professor precisa enviar um formulário de avaliação aos alunos sobre certa atividade, o SGIR deve recolher essa petição e emití-la a cada um dos alunos para que estes preencham as respostas. Neste caso, o professor desconhece como esses formulários serão emitidos, mas precisa da intervenção do SGIR para poder fazê-lo.

A identificação das transferências entre os agentes do ambiente de aprendizagem cooperativa é um ponto central na modelagem do SGIR, tendo em vista que o objetivo deste sistema é facilitar o acesso aos recursos do ambiente docente de aprendizagem cooperativa por parte dos professores e dos alunos para, desta forma, facilitar a realização das suas tarefas. Em outras palavras, os usuários devem interagir com o SGIR para usar os recursos do domínio e, por isso, identificar estas interações (ou transferências) é fundamental na modelagem do sistema.

Na versão atual de CommonKADS, no entanto, as transferências entre agentes incorporam-se, também, ao modelo de conhecimento: incluem-se como tarefas denominadas funções de transferência, em cuja representação indica-se entre que agentes ocorre interação e, se necessário, indica-se, também, que informação espera-se receber de sua execução no outro agente. Tendo isto em conta, a identificação das transferências, ou dizendo de forma equivalente, das funções de transferência, far-se-á no modelo de conhecimento, para assim poder explicar

precisamente porque é necessário incluir uma transferência determinada (contextualizá-la entre a execução de duas tarefas do agente).

5.4 Modelo de Conhecimento

O SGIR tem como objetivo oferecer serviços de acesso ao conjunto de recursos de hardware e software presentes no ambiente de aprendizagem cooperativa descrito no item 5.1.3. Tais serviços (ou tarefas do SGIR) são operações de alto nível realizadas pelo SGIR por requisição do professor e/ou alunos: em um serviço, integra-se uma série de ações que é necessário executar para efetuar uma operação solicitada pelo professor e os alunos durante suas atividades. Em outras palavras, é necessário especificar os modelos de conhecimento do professor e alunos para poder identificar que serviços precisam ser executados pelos SGIR.

5.4.1 Método de resolução de problemas do professor

Em relação ao modelo de conhecimento do professor será apresentado unicamente o método de resolução de problemas, ficando a ontologia do domínio (ontologia de educação) fora do propósito deste trabalho. Justifica-se isso tendo em vista que para estabelecer que serviços, ou tarefas, que o SGIR deve possuir, não é necessário desenvolver a ontologia do domínio. Assim, pode-se considerar a tarefa ensinar como um problema de planejamento: descrever o conjunto de atividades a serem desenvolvidas pelos alunos em um determinado período de tempo. Este problema pode ser resolvido usando uma classe geral de métodos conhecido como *propon-verificar-criticar-modificar* (PVCM). Isto significa desempenhar sub-tarefas em um domínio:

propon – com base nas informações sobre o curso, o professor define os objetivos pedagógicos e propõe atividades a serem desenvolvidas pelos alunos com o intuito de atingir esses objetivos. Para cada atividade, são definidos os grupos de alunos e especificados o material didático e recursos tecnológicos a serem utilizados. Caso os objetivos não sejam atingidos, o professor pode propor mudanças no plano de atividades.

Para realizar esta tarefa o professor precisa interagir com o SGIR por meio da adequada função de transferência para solicitar a informação relativa ao curso (obtem) e para armazenar a atividade a ser desenvolvida na base de dados (apresenta). Neste sentido, tem-se que levar em conta que a tarefa propor, desenvolve-se na sala do professor (fora do horário de aula) e, portanto, é necessário armazenar o resultado da sua execução para poder recuperá-la durante o desenvolvimento da aula

verificar – O professor necessita verificar o rendimento dos alunos ou grupos durante a atividade educacional. Para isso, utiliza o PDA como meio de anotar as observações durante o desenvolvimento da atividade dos grupos de alunos. Para isto é necessário que o professor recupere da base de dados (obtem) a atividade desenhada previamente, a qual é vista no modelo como uma função de transferência.

criticar – a partir das observações feitas durante a aula sobre o comportamento e evolução dos alunos e das observações emitidas pelos próprios alunos por meio do seu PDA, o professor avalia o rendimento da atividade educacional de cada grupo ou aluno. O feedback do aluno pode ser obtido por meio de questionário disponibilizado pelo professor a partir do seu PDA.

Na execução desta tarefa, são identificadas as funções de transferência correspondentes às observações detectadas provenientes dos alunos (recebe) e à obtenção dos resultados das avaliações preenchidas pelos alunos em formulários (obtem). Tais tarefas, deverão ser realizadas pelos SGIR como serviços que o professor lhe apresenta.

modificar – após avaliar, o professor pode decidir “refinar” a atividade corrente. O professor pode optar por mudar o material didático utilizado pelos alunos e/ou propor uma nova forma de conduzir a atividade, com mais explicações e redistribuição dos conteúdos, por exemplo.

Para esta tarefa, as funções de transferência que se identificam são as correspondentes ao armazenamento da base de dados (apresenta) da modificação da atividade em curso - seja propondo uma nova ação pedagógica ou retocando tal atividade.

Estas tarefas de alto nível ajustam-se perfeitamente ao comportamento do agente professor descrito no modelo correspondente no CommonKADS. A figura 8 abaixo mostra o diagrama de tarefas gerais (propor, verificar, criticar e modificar) quebrado em um domínio de sub-tarefas (ovais) e funções de transferências (retângulos). É importante salientar que o professor interage, unicamente, de forma direta com o SGIR: na modelagem do comportamento do professor fazendo uso das possibilidades do ambiente docente, a interação com os seus alunos é considerada como um serviço a ser provido pelo SGIR. Por exemplo, quando emite os formulários de avaliação aos alunos, na realidade faz uma requisição ao SGIR para que este emita tais formulários aos PDA's dos alunos.

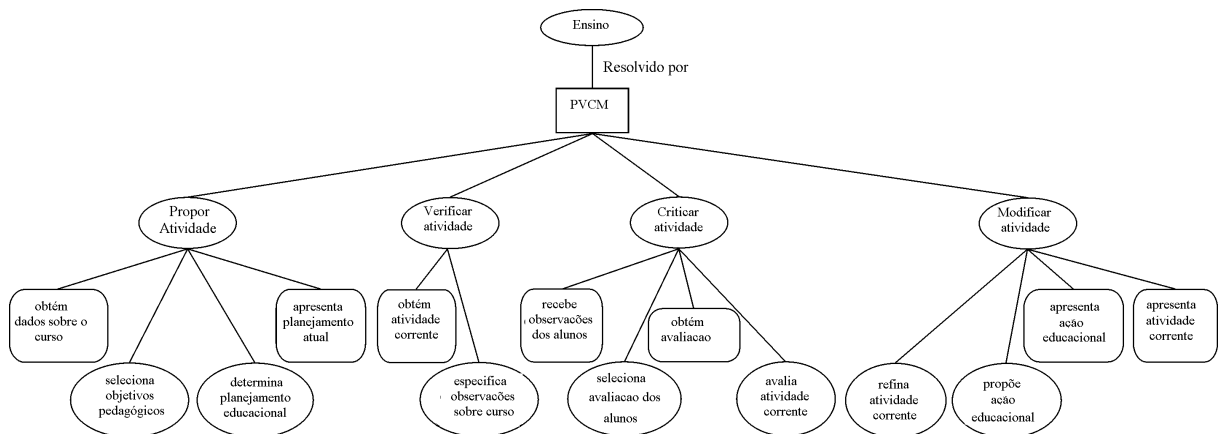


Figura 8: Modelo de tarefas correspondentes à tarefa de ensino realizada pelo professor

Fonte: Lama et al (2002)

5.4.2 Modelo de Tarefas dos alunos

Neste trabalho não houve a pretensão de obter-se um modelo de comportamento para os alunos, como foi feito para o modelo do professor. A razão disso é que o método de resolução de problemas do professor permitirá desenhar, num futuro próximo, um sistema de apoio à decisão para ser utilizado nos momentos de elaborar, avaliar e modificar uma atividade de aprendizagem cooperativa durante a aula. No caso dos alunos, portanto, identificou-se que tarefas são realizadas e que

informações são necessárias para realiza-las, prestando-se especial atenção às interações com o SGIR.

As tarefas dos alunos estão baseadas, também, nas funções de transferência identificadas no método de resolução de problemas do professor. Cada função de transferência pode ser associada a uma tarefa dos alunos cuja execução pode induzir a uma ação ou recurso do ambiente. O SGIR promove, com a ajuda de recursos tecnológicos variados, uma interação mais efetiva entre professor-aluno, professor-grupo de alunos e aluno-aluno. Essa interatividade representa, para o aluno, um leque de oportunidades para aprender:

resolver atividades - Para a resolução de uma dada tarefa, o aluno pode escolher os recursos conforme suas necessidades. O SGIR, por meio de interface adequada, permite ao aluno consultar materiais didáticos presentes em um banco de dados multimídia, verificar os recursos adicionais presentes, preparar slides e controlar equipamentos necessários a sua apresentação, por exemplo. Além disso, o aluno ou grupo de alunos pode gerenciar o andamento de suas atividades, verificar resultados parciais, armazenar informações sobre atividade a ser completada posteriormente e especificar observações sobre a atividade.

Durante a execução desta tarefa, a qual é vista no modelo como uma função de transferência, para o aluno verificar os recursos disponíveis é necessário que ele recupere da base de dados (obtem) a lista de recursos que foram assinalados previamente como disponíveis para a atividade.

responder avaliações – Essas avaliações são utilizadas pelo professor como feedback para acompanhamento do curso. Podem ser avaliações de aprendizagem ou avaliação das condições de oferecimento do curso, para verificar, por exemplo, se o material didático está adequado. Com os resultados dessas avaliações o professor pode redefinir as atividades futuras.

Na execução desta tarefa, são identificadas as funções de transferência correspondentes às avaliações emitidas pelo professor como formulários (recebe) e à obtenção, pelo professor, dos resultados das avaliações preenchidas pelos alunos em formulários (entrega). Tais tarefas, deverão ser

realizadas pelo SGIR como serviços que o professor e os alunos lhes apresentam.

avaliar atividades – Como verificação do andamento do curso, uma avaliação das atividades é feita tendo em vista questões como rendimento dos alunos, motivação e o estabelecimento do tempo para execução das tarefas. O SGIR permite um gerenciamento das atividades durante a sua execução e a obtenção do rendimento dos alunos após a execução das tarefas. Com o resultado dessas avaliações em mãos o professor pode, junto com os alunos, rediscutir o planejamento de atividades futuras do curso.

Na execução desta tarefa, identifica-se a função de transferência correspondente aos resultados obtidos no curso emitidos pelo professor como formulários (apresenta). Tal tarefa, deverá ser realizada pelo SGIR como um serviço que o professor lhe apresenta.

A figura 9 abaixo mostra o diagrama de tarefas gerais (resolve, responde e avalia) quebrado em um domínio de sub-tarefas (ovais) e funções de transferências (retângulos). O nível mais abaixo, representa as interações entre os alunos e os outros agentes no ambiente (professor e SGIR). Por exemplo, os alunos podem receber informações de outros agentes (apresenta) ou requisitar informações deles (obtem e recebe).

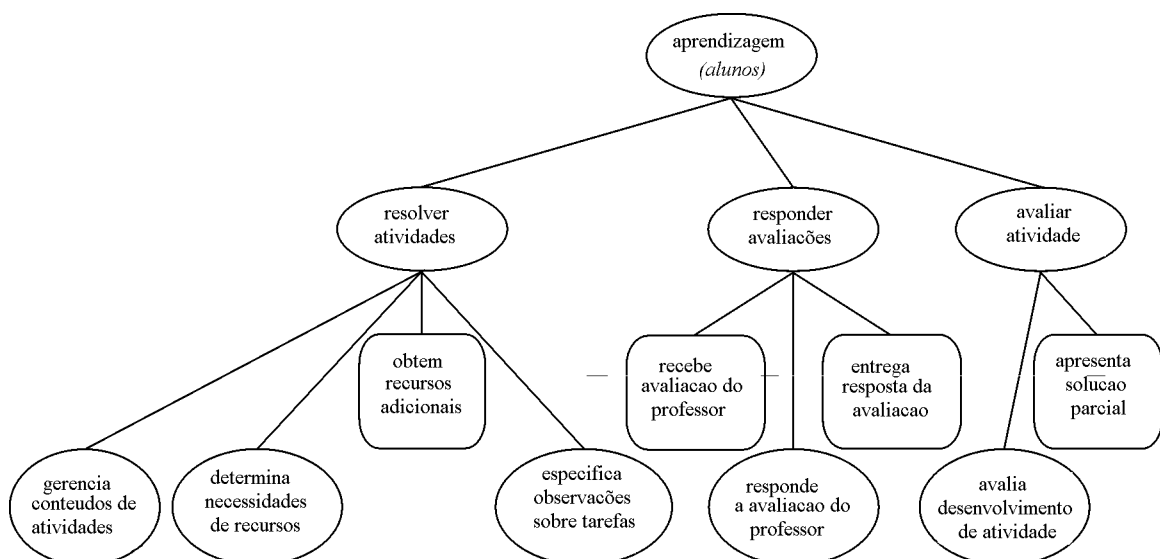


Figura 9: Modelo de tarefas do SGIR correspondentes a sua interação com o aluno ou grupo

Fonte: Lama et al (2002)

5.4.3 Tarefas do SGIR

As tarefas do SGIR estão baseadas nas funções de transferência identificadas nos correspondentes modelos de tarefas do professor e dos alunos. Como comentado anteriormente, cada função de transferência pode ser associada a uma tarefa do SGIR (ou serviço), cuja execução pode induzir a uma ação ou recurso do ambiente. As tarefas podem ser facilmente classificadas, considerando-se os recursos sob o seu controle:

gerenciamento de informações – apresenta ao professor informações necessárias à execução de suas tarefas, tais como dados de alunos, lista de objetivos pedagógicos definidos no início, atividades pré-definidas ou atividades educacionais em andamento. Este nível de tarefa também inclui interações entre professor-aluno, como por exemplo: requisição, pelos alunos, de material didático adicional ou recursos necessários à execução de uma atividade proposta ou requisição, pelo professor, de envio de questionários de avaliação durante o desenvolvimento de uma atividade e assim por diante;

acesso aos recursos Hardware – Isto significa interação com recursos com a intenção de enviar/receber informações durante explanações de conteúdos. Estas tarefas também podem redirecionar informações aos PDA's a partir de outros agentes para facilitar explicações.

As tarefas realizadas pelo SGIR, são as seguintes:

obter dados do curso – Por meio de solicitação do professor, o sistema faz uma busca, na base de dados, com a ajuda de mecanismos apropriados, das informações sobre os alunos, os objetivos pedagógicos do curso e as informações relativas ao planejamento de atividades. Esta tarefa é identificada no modelo como uma função de transferência (apresenta).

obter atividades correntes – O sistema fornece ao professor informações sobre as atividades que estão em andamento e pode, também, fornecer informações sobre as atividades que foram completadas. Para isso, o SGIR conta com um

conjunto de mecanismos específicos para armazenar informações detalhadas sobre o andamento das atividades, sendo esta tarefa identificada no modelo como uma função de transferência (apresenta).

obter avaliação dos alunos – Permite efetuar avaliações sobre o curso, sobre o andamento das atividades (aulas) e verificar a aprendizagem em sala de aula. O professor pode escolher questionários do banco de dados, enviar para os alunos (disponibiliza) e colher (recebe) as respostas. Com essas avaliações o professor pode redefinir as atividades planejadas anteriormente e, inclusive, rediscutindo com os alunos (apresenta). Essas tarefas executadas pelo SGIR, por requisição do professor estão representadas no modelo como as funções de transferência (disponibiliza, recebe, apresenta).

gerenciar agenda – Permite ao professor criar um calendário de atividades e fazer alterações posteriores. Aos alunos permite a consulta, a qualquer momento, das datas relativas as atividades definidas pelo professor, executada pelo SGIR como uma tarefa, requisitada tanto pelo professor quanto pelos alunos, tem sua representação no modelo como uma função de transferência (apresenta).

gerenciar dispositivos educacionais – Por meio de uma interface adequada, tanto o professor, quanto o aluno, podem controlar os dispositivos de apoio às aulas. Por exemplo, o professor ou aluno pode controlar a exibição de slides com o projetor multimídia utilizando o seu PDA, o aluno pode consultar material didático adicional armazenado em um banco de dados multimídia. Esse controle, na realidade, é uma tarefa executada pelo SGIR, por requisição dos agentes, e está representada no modelo como uma função de transferência (apresenta).

gerenciar mudanças nas atividades – O professor pode efetuar um “refinamento” das atividades a serem executadas pelos alunos: pode redefinir datas, objetivos pedagógicos, estratégias a serem utilizadas (plano de aulas) e os recursos didáticos necessários com a mudança. A visualização dessas

mudanças, por requisição dos alunos, é uma tarefa executada pelo SGIR e está representada como uma função de transferência (apresenta).

A figura 10 seguinte mostra o modelo de tarefas do SGIR que dá suporte às operações realizadas pelo professor e suas interações com os alunos. Neste modelo, pode-se observar que a tarefa *gerenciamento de recursos* não foi associada a nenhum método geral para a sua resolução. Isto é consistente, considerando-se que a ativação do SGIR será em função da demanda dos agentes humanos.

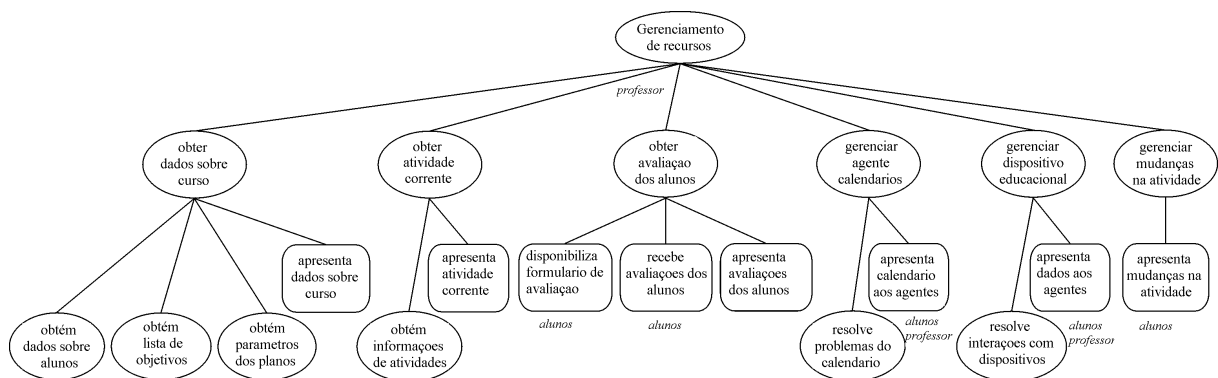


Figura 10: Modelo de tarefas do SGIR correspondentes a sua interação com o professor

Fonte: Lama et al (2002)

5.5 Modelo de desenho

O último modelo na metodologia CommonKADS é o modelo de desenho, onde se propõe uma arquitetura para o sistema, a partir da qual realiza-se a implementação em uma determinada linguagem de programação. Neste trabalho, apresenta-se uma arquitetura que é responsável pelos requerimentos funcionais – onde estão indicadas as funções (ou operações) que o sistema deve executar (SÁNCHEZ et al, 2002). Estes requerimentos estão descritos no modelo de tarefas do SGIR onde se especificam, precisamente, as operações a serem realizadas pelo SGIR (LAMA et al, 2002).

Antes de propor a arquitetura foram definidos os chamados requerimentos não funcionais, que influem enormemente na proposta de arquitetura e que estão relacionados com aspectos do desenho do sistema tais como: manutenção,

acessibilidade, confiabilidade e escalabilidade ou modularidade. Por exemplo, a arquitetura do SGIR tem que ser bem escalável, pois à medida que o professor e/ou os alunos vão explorando novas possibilidades de acesso aos recursos para o desenvolvimento de suas atividades, será necessário incorporar essa funcionalidade ao sistema. Isto implica em um crescimento do sistema e, portanto, a escalabilidade é um fator chave que se deve ter em conta no desenho da arquitetura. Caso contrario, pode ser que em um futuro próximo o sistema tenha que ser redesenhado.

5.5.1 Domínio, Modelo de Tarefas e Requerimentos do sistema

Este ambiente de aprendizagem é constituído por um conjunto de dispositivos de hardware, conforme descrito na seção 5.1.3, para dar suporte à aprendizagem cooperativa. O modelo de tarefas identifica os serviços que devem ser executados para efetuar as operações dos usuários. O modelo de tarefas, descrito na seção 5.4, foi definido a partir das necessidades dos professores e alunos para acessar os recursos presentes no ambiente. A seguir os principais requerimentos para o projeto:

- Funcionais (FR - *Functional Requirements*)
 - FR-1. Gerencia contas de professores, alunos, grupos e administradores;
 - FR-2. Disponibiliza uma interface de planos de curso (agenda) para a organização de atividades e o desenvolvimento de materiais apropriados;
 - FR-3. Armazena planos de cursos, programas, roteiros de atividades programadas e material didático;
 - FR-4. Permite controlar dispositivos a distancia: projetor multimídia, Mímio e outros;
 - FR-5. Permite acesso a materiais didáticos na sala de aula;
 - FR-6. Permite armazenar observações, anotações de leitura;
 - FR-7. Permite efetuar avaliações em sala de aula. O professor pode escolher questionários do banco de dados, enviar para os alunos e colher as respostas;
 - FR-8. Gerencia a comunicação professor-aluno e aluno-aluno. Cada usuário pode interagir com outros usuários e/ou grupos de usuários conectados ao sistema.

- Não funcionais (LOS - *Low-of-service requirements*)
 - LOS-1. Acessibilidade: permite o acesso a interface de planos de curso a qualquer hora e qualquer lugar;
 - LOS-2. Minimiza o tempo de resposta do sistema para as requisições dos usuários feitas na sala de aula;
 - LOS-3. Confiabilidade dos serviços fornecidos pelos dispositivos. Durante atividade de leitura, por exemplo;
 - LOS-4. Escalabilidade no número de alunos e grupos registrados para uma turma;
 - LOS-5. Extensibilidade no número de serviços acessíveis e dispositivos de hardware;
 - LOS-6. Padrão de segurança para cada conta de usuário.

5.5.2 Arquitetura de software do sistema

A seguir, uma descrição das principais características da arquitetura de software desenvolvidas para satisfazer aos requerimentos da seção acima:

i. Topologia

A arquitetura de software do sistema foi definida a partir de uma topologia multi-camadas com componentes encapsulados que apresentam características similares. Esta técnica busca atender aos requisitos do requerimento LOS-5, com o objetivo de reduzir o número de tipos de componentes diferentes com os quais cada componente deve interagir. A figura 11 fornece uma visão de alto nível desta arquitetura multi-camada: [1] Camada de Recursos (RT), [2] Camada de Gerenciamento de Recursos (RMST), [3] Camada de Interface e Camada de Serviços Orientados a Comunicação. O RT tem encapsulado todos os *drivers* de recursos disponíveis no ambiente; O RMST tem encapsulado todos os recursos de gerenciamento de serviços (RMSs), os quais oferecem ambos alto e baixo nível de serviços para manusear todos os recursos disponíveis; O IT representa o conjunto de todas as interfaces gráficas dos usuários e, finalmente, o ICST permite a interação interface-interface e serviço-interface.

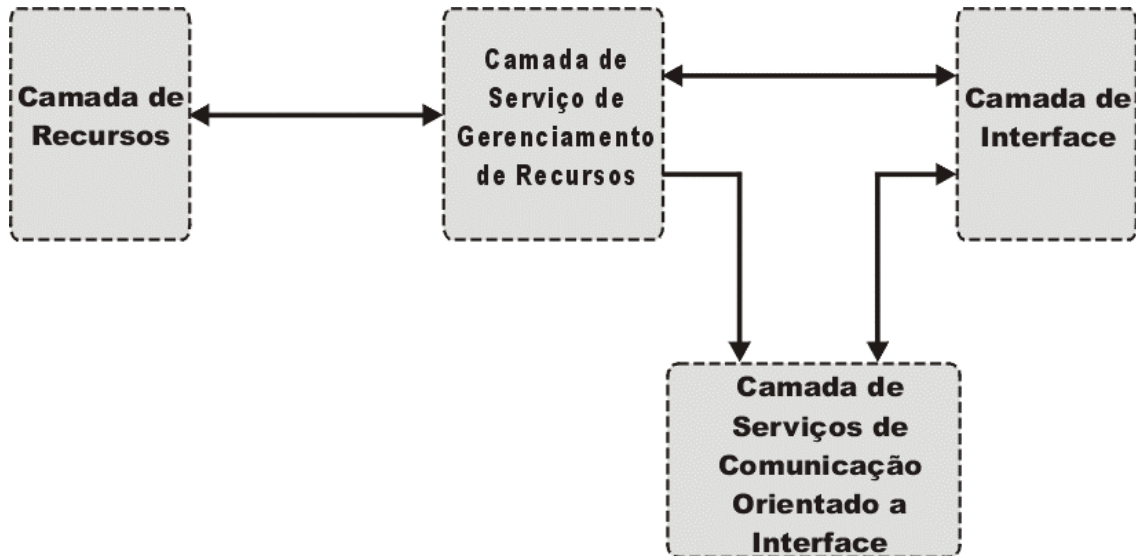


Figura 11: Visão de alto nível da arquitetura multi-camada

Fonte: Sánchez et al (2002)

É importante enfatizar que foram consideradas duas camadas diferentes de serviços. Isto é necessário para poder encapsular ambos os recursos de gerenciamento de serviços (os requerimentos FR-3, FR-4, FR-5, FR-6 e FR-7) e os serviços de comunicação orientados a interface (requerimento FR-8). A existência de diferentes tipos de serviços implica em diferentes fluxos de dados e controle nesta arquitetura (ver ítem iv, para mais detalhes).

ii. Componentes

A figura 12 mostra a arquitetura de software completa. Diversos tipos de componentes foram introduzidos:

- **DB/R (Data Base Resource)**. Gerenciador para controlar todas as operações relacionadas com banco de dados;
- **DE/R (Device Resource)**. Podem existir diversos tipos de dispositivos na sala de aula. Para cada um, existe um componente correspondente para o acesso e controle de sua operação;

- *A/RMS (Administrator Resource Management Services)*. O sistema necessita dar suporte à criação de contas (requerimento FR-1), para monitorar atividades de usuários e para checar o estado de cada atividade;
- *P/RMS (Professor Resource Management Services)*. Parte chave da arquitetura, contém todos os serviços disponíveis para o professor, dentro e fora da sala de aula. Facilita as tarefas de preparação de aulas tais como planejamento de curso e elaboração de atividades, tanto na fase pré-ativa quanto na fase pós-ativa. Além disso, disponibiliza serviços de acessibilidade e de comunicação alunos/grupos;
- *S/RMS (Student Resource Management Services)*. Controla os serviços disponíveis para os alunos. Estes serviços incluem acesso a internet, informações sobre curso e outros;
- *G/RMS (Group Resource Management Services)*. Controla os serviços disponíveis para os grupos: este projeto requer um sistema pronto para dar suporte a aprendizagem cooperativa;
- *RMS/LS (Resource Management Lookup Services)*. Antes da interface de componentes usar os serviços, tem que saber que serviços estão disponíveis. Cada componente é então, associado a um Lookup service (serviço de verificação) que permite duas operações principais: (1) registro do serviço e (2) busca e localização de serviços disponíveis;
- *A/I (Administrator Interface)*. Permite ao administrador utilizar o serviço ARM/S (requerimento FR-1);
- *POC/I (Professor Out-of-Class Interface)*. Para o professor planejar, preparar atividades e outras atividades típicas da etapa pré e pós-ativa (requerimento FR-2);
- *PIC/I (Professor In-Classroom Interface)*. Interface que o professor utiliza durante as aulas. Permite ao professor utilizar serviços típicos da fase ativa do processo de aprendizagem;
- *SIC/I (Student In-Classroom Interface)*. Interface pessoal dos alunos;
- *GIC/I (Group In-Classroom Interface)*. Interface dos grupos;
- *ICS/LS (Interface-oriented Communication Services Lookup Service)*. Permite a interface registrar e dar suporte aos serviços encontrados pela interface.

Este componente é essencial para as interações entre professor-aluno, aluno-aluno e serviço-aluno.

Existem outros componentes não mostrados com clareza na figura 11, os quais desempenham um papel importante. São componentes repositórios localizados na camada de gerenciamento de recursos. Estes repositórios contidos em cada RM/S, representam um mecanismo de memória cache criados para reduzir o número de acessos aos bancos de dados e, conseqüentemente, melhorar a performance dos serviços dos usuários (requerimento LOS-2).

A figura 13 mostra a arquitetura interna de um RMS genérico. Existe uma federação de agentes e componentes que estão prontos para executar os serviços solicitados pelos usuários. Os serviços baseados em componentes são serviços do tipo ativação de projetor multimídia ou do sistema de aquisição de quadro-branco (Mímio). Por meio disto, é feita uma validação do componente requisitado como medida de segurança (requerimento LOS-6). Os serviços baseados em agentes são, por outro lado, serviços de alto nível, obtidos do modelo de tarefas descrito na seção 5.4: obter informações sobre curso, obter atividades correntes, obter avaliação dos alunos e outros. Os agentes foram levados em conta tendo em vista que cada serviço de alto nível tem um objetivo associado e um plano para atingir esse objetivo. Além do mais, existe um agente mediador de serviços que controla cada requisição feita e a direciona ao serviço apropriado.

Deve ser enfatizado que um RMS é ajustado a cada tipo de usuário, tendo em vista a escalabilidade (requerimento LOS-4).

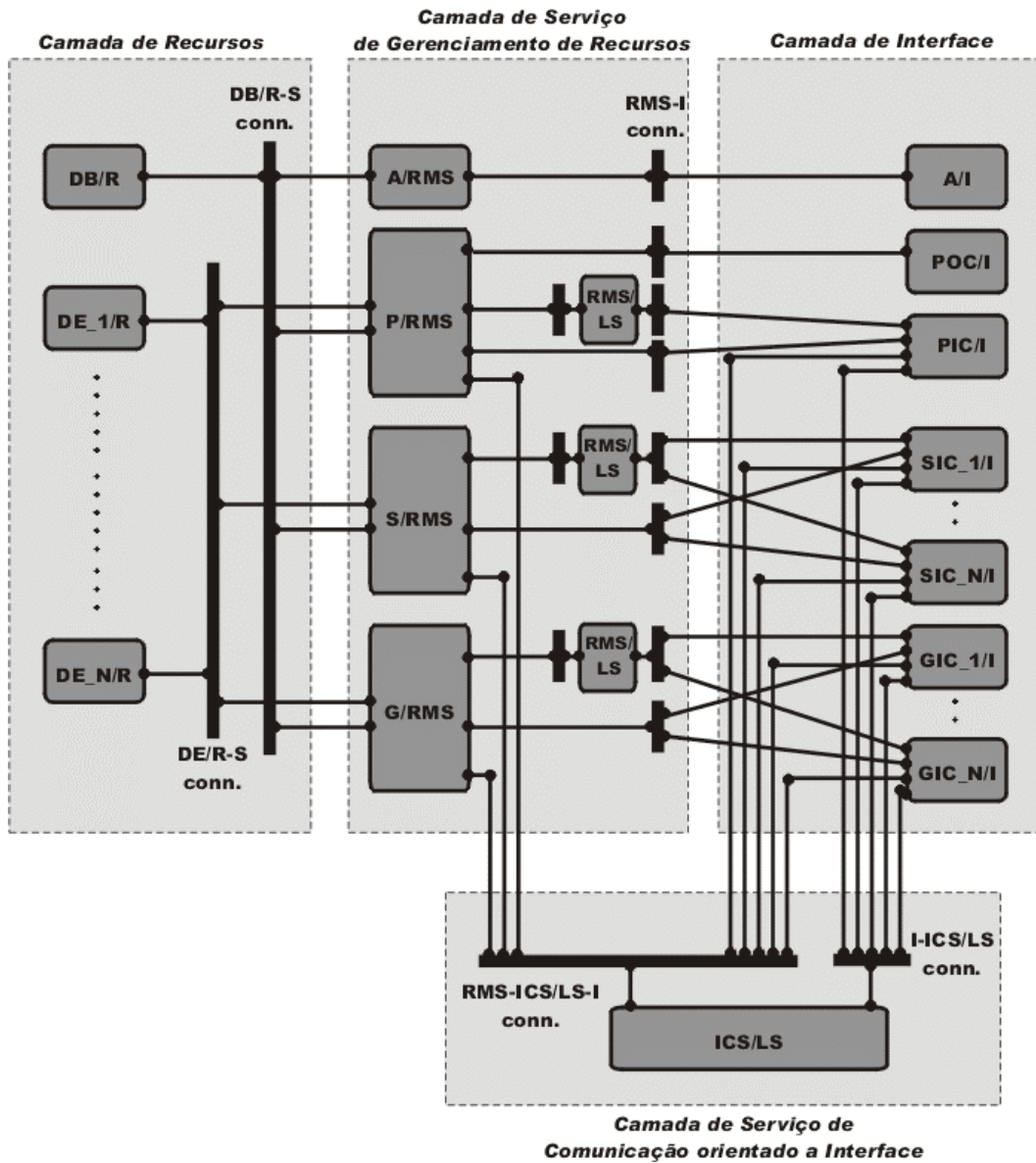


Figura 12: Arquitetura de software completa com todos os componentes

Fonte: Sánchez et al (2002)

iii. Conectores

A seguir os principais tipos de conectores arquiteturais conforme descrito por Shaw (1996), mostrados na figura 12:

- DE/R-S (*Device Resource-Service connector*). É um conector de invocação implícita (também chamado de evento). Os serviços enviam requisições aos componentes de recursos e esperam por notificações correspondentes. As

requisições podem ser consideradas como eventos que chegam ao conector, o qual direciona o evento para o componente de recurso apropriado;

- *DB/R-S (DataBase Resource-Service connector)*. É um conector de dados compartilhado (também chamado de acesso de dados). Ele permite acesso ao componente gerenciador de banco de dados em resposta a requisições de consulta;
- *RMS-I (Resource Management Service-Interface connector)*. É também um conector de evento. Uma interface requer certa ação que é traduzida em eventos, os quais disparam uma operação apropriada ao serviço correspondente;
- *I-ICS/LS (Interface-Interface-Oriented Communication Services Lookup connector)*. Outro conector de evento, controla as requisições de interfaces para o registro automático no componente I-ICS/LS;
- *RMS-ICS/LS-I (Remote Management Service-Interface-oriented Communication Services Lookup connector)*. Um conector distribuidor comum. Encarrega-se da comunicação de direcionamento entre interfaces, bem como de algum RMS para as interfaces disponíveis.

iv. Fluxo de Informações

Toda ação nesta arquitetura é iniciada por um usuário que seleciona a ação desejada por meio de uma interface de entrada. Para isso, existe uma interface *driver* de usuário. Uma discussão mais detalhada sobre os mecanismos de fluxo de informação é necessário em função dos três tipos de serviços descritos no item i, acima: serviços de gerenciamento puramente de recursos (ou tipo 1), serviços de gerenciamento de comunicações envolvidas (ou tipo 2) e serviços de comunicação orientada a interface (tipo 3). Quando um usuário requisita os serviços do tipo 1 ou do tipo 3, a ação executada é traduzida em eventos arquiteturais, os quais, apropriadamente gerenciados por conectores de invocação implícita, disparam a ativação dos componentes requisitados.

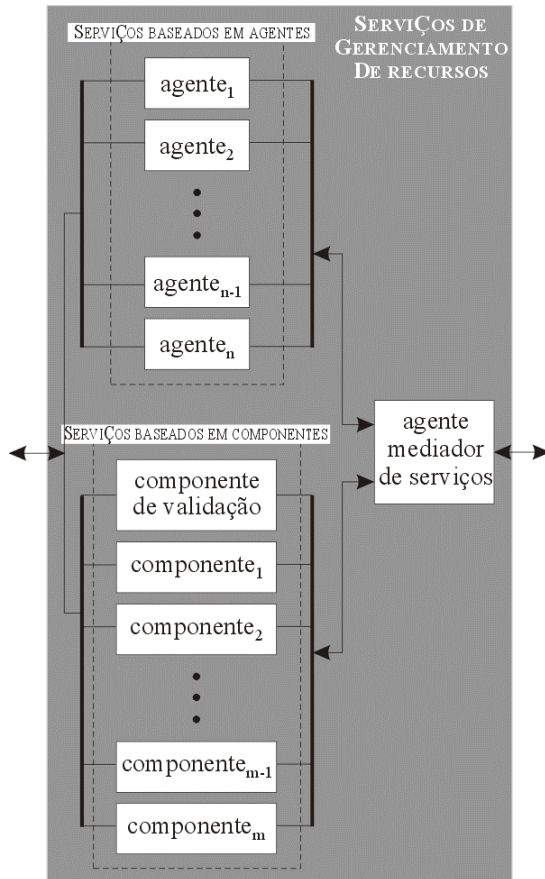


Figura 13: Arquitetura interna de um RMS genérico
 Fonte: Sánchez et al (2002)

Para isso, existe um mecanismo baseado em eventos que delinea essas operações. Por outro lado, quando um usuário requisita os serviços do tipo 2, dois mecanismos de fluxo de informações são envolvidos: (1) um mecanismo baseado em eventos para controlar requisições de usuários e, (2) um mecanismo *push-based* para enviar os dados obtidos das interfaces selecionadas. Sobre esta condição de operação, os serviços requisitados atuam como *broadcasters* e as interfaces selecionadas atuam como receptores.

5.5.3 Design do Sistema Distribuído

A figura 14 mostra a arquitetura distribuída de *software* do sistema em uma rede de computadores. Três elementos estão relacionados: os componentes de *software*, computadores hospedeiros (*hosts*) e conexões de rede. É importante destacar três principais espaços envolvidos neste projeto: (1) A sala de aula, onde a atividade de aprendizagem ocorre; (2) o departamento/organização, onde os servidores e

computadores *desktop* estão localizados e (3) locais remotos (requerimento LOS-1), como uma localização divisão de serviços de informação, para permitir a administração remota de tarefas e a casa do professor, para suporte ao trabalho remoto.

i. Suporte computacional

Os tipos de computadores utilizados neste sistema encontram-se bem detalhados na seção 5.1.3.

ii. Suporte de redes

Existe uma estrutura de rede Ethernet para garantir a conectividade entre o servidor de sala de aula, entre os *hosts* do departamento/organização e uma estrutura de rede sem fio na sala de aula para o acesso dos PDA's.

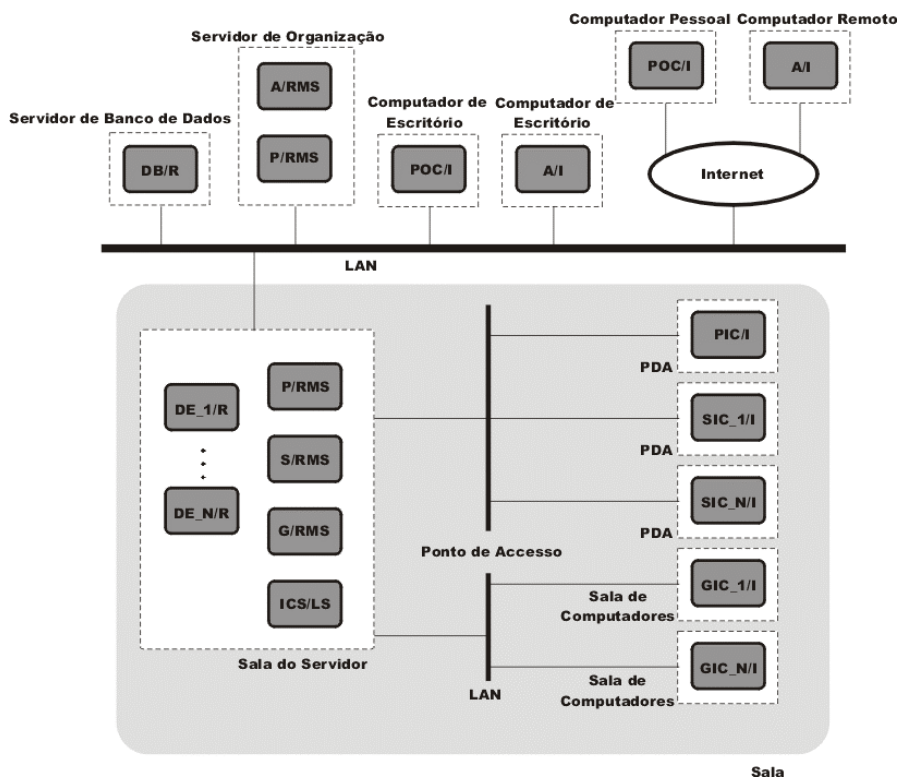


Figura 14: Arquitetura distribuída do software em uma rede

Fonte: Sánchez et al (2002)

Existe, também, uma estrutura de acesso remoto por meio de um *gateway/router* que não está na figura 13, para garantir acesso a internet.

iii. Sistemas operacionais

Existe uma grande variedade de sistemas operacionais neste sistema: (1) Os PDA's rodam com Windows CE, (2) os computadores de sala de aula e servidores utilizam Windows 2000, (3) o servidor de banco de dados e organização roda com Red Hat Linux e, (4) os PC's *desktop* rodam com ambos Windows e Linux.

iv. Banco de dados

Para reduzir custos foi escolhido o MySQL, *freeware* popular que oferece uma performance razoável e gerenciamento de transações, recurso acrescentado recentemente.

v. Middleware

Um apropriado *middleware* é requerido para o componente de acessibilidade remota. A seguir as principais tecnologias envolvidas:

- conectividade do Banco de Dados. O *driver* escolhido para desenvolver o conector para a acessibilidade remota no banco de dados MySQL foi JDBC;
- *Remote Method Invocation* (RMI). Os conectores DE/R-S, RMS-ICS/LS-I e I-ICS/LS serão desenvolvidos com Java RMI;
- serviços de Busca. Conforme item 3 na seção 5.5.2, existem dois serviços de busca: (1) para a interface de clientes saber quais são todos os serviços de gerenciamento de recursos disponíveis na sala de aula e (2) para cada interface de cliente e alguns serviços de gerenciamento de recursos saberem quais são todas as interfaces de clientes correntemente conectadas. Foi escolhido o JINI, *Java Discovery service technology*, para implementar esses serviços. O JINI oferece características adicionais, como um serviço de redundância, que agrega confiabilidade, dando suporte ao requerimento LOS-3.

vi. Linguagem de Programação

A linguagem JAVA satisfaz a todas as necessidades para praticamente todos os componentes. Sua capacidade em atender às necessidades relativas a rede e como *middleware*, representa uma facilidade que terá um grande impacto na redução do tempo de desenvolvimento. Para alguns componentes DE/R, o sistema de aquisição de quadro-branco, por exemplo, o código proprietário ou código-fonte em C++ é a única opção para controlar dispositivos específicos. Nestes casos, um adaptador/acoplador será desenvolvido para cada componente.

vii. Comunicação entre agentes

O protocolo escolhido foi o KQML para a comunicação entre agentes. Será utilizado um conjunto reduzido de comandos porque as interações entre agentes são relativamente simples. Com esta técnica, o sistema oferece uma comunicação padrão que facilita a integração de novos agentes, nesta federação, que forem acrescentados no futuro.

viii. Descrição de dados e mensagens

XML foi a linguagem escolhida para a descrição de dados e mensagens. XML é um formato padrão de dados e isso permite que outros componentes possam “conversar” facilmente com este sistema.

6. Conclusões

Durante o desenvolvimento do trabalho, a metodologia CommonKADS mostrou-se muito eficiente para a concepção do desenho do sistema, durante a fase de modelagem do ambiente em que se integrará o sistema de gestão de recursos, para expressar de forma explícita as necessidades da organização, os agentes envolvidos na operação a ser realizada pelo sistema e, principalmente, para resolver o ponto chave: a descrição das interações entre os agentes e a identificação das tarefas realizadas por cada um deles (que formam parte do seu modelo de conhecimento). A metodologia CommonKADS ajustou-se muito bem para abordar tais atividades, tendo em vista que define perfeitamente como expressar as interações entre agentes e como integra-las dentro de uma árvore de tarefas (que forma parte do método de resolução de problemas dos agentes).

O modelo proposto integrado ao ambiente docente simplifica o acesso aos recursos usados pelos agentes humanos no processo de aprendizagem, dinamizando as explicações do professor, facilitando o acompanhamento das aulas por parte dos alunos (a partir do seu PDA) e favorecendo a avaliação contínua e em tempo real do rendimento dos alunos. Isso permite a detecção imediata, para uma posterior correção, de possíveis deficiências no desenvolvimento das atividades por parte dos alunos. Este modelo, no entanto, apesar de ter sido imaginado inicialmente para ser utilizado apenas em atividades de aprendizagem cooperativa, pode ser aplicado a atividades de aprendizagem individualizada, pois nada impede que as atividades planejadas para um grupo não possam ser redefinidas para alunos individualmente ou que um grupo de alunos não possa ser constituído por um único aluno.

A completude do modelo, no entanto, não está assegurada tendo em vista que podem aparecer novas tarefas do domínio como consequência de explorar-se mais as possibilidades de interação entre os elementos do ambiente. O enriquecimento deste modelo far-se-á de acordo com o resultado de sucessivas avaliações feitas pelos usuários do ambiente (tanto professores como alunos), o qual permitirá descobrir ou não, a necessidade de planejar o acréscimo de novas tarefas ou, inclusive, de eliminar tarefas existentes, consideradas desnecessárias. Neste sentido, o modelo proposto originou-se a partir de reuniões de aquisição de

conhecimento mantidas com especialistas em ciências da educação e de uma primeira avaliação realizada por três professores no uso do ambiente docente.

6.2. Considerações finais

Esta dissertação apresentou um ambiente que oportuniza uma forma inovadora de construção do conhecimento em uma sala de aula diferente e, durante todo o processo de pesquisa, observou-se que a proposta da arquitetura concebida é inovadora e consistente, de acordo com os seguintes aspectos:

- Existem ambientes semelhantes ao descrito neste trabalho, mas não foi encontrado algum que utilize efetivamente o PDA para auxiliar na interação entre professor/alunos/recursos durante as aulas presenciais.
- Não foi encontrado algum ambiente tecnológico de aprendizagem cooperativa que tenha utilizado CommonKADS como metodologia para o desenvolvimento de sistemas.
- O modelo proposto, a exemplo dos softwares discutidos no cap. 4, combina técnicas provenientes de diversas áreas de conhecimento.
- Existem softwares que empregam técnica semelhante ao descrito neste trabalho, a exemplo do projeto Pebbles que oferece acesso a programas rodando em um PC por meio do PDA (MYERS, 2001). No entanto, a técnica empregada aqui é claramente mais ambiciosa, considerando-se que o sistema está estruturado na forma de camadas de serviços específicos, os quais oferecem serviços de alto-nível, dinamiza a performance computacional e resolve problemas de escalabilidade, extensibilidade e confiabilidade.

6.3. Sugestões e perspectivas futuras

Como trabalho futuro propõe-se:

- Desenvolver uma ontologia de educação em que se descreva os elementos do domínio assim como as relações entre eles. Esta ontologia é imprescindível para facilitar a comunicação entre os agentes de software do

ambiente; a ontologia constitui a linguagem comum que os agentes utilizarão para manejar as mensagens enviadas entre eles. A ontologia, além do mais, servirá como modelo conceitual no qual se baseará a base de dados que armazenará toda a informação compartilhada pelos agentes do ambiente ou necessária para realizar as suas operações.

- A implementação da arquitetura do SGIR constitui o trabalho mais importante a ser realizado em curto prazo. Atualmente estão implementados os processos que controlam diretamente o acesso aos recursos, ou dito de outra forma, os *drivers* dos recursos presentes no ambiente (projetor, mídio e base de dados). O primeiro passo, portanto, consistirá em implementar as partes do SGIR que se comunicam diretamente com estes processos para realizar as operações básicas sobre o *hardware* (ligar/desligar de dispositivos) e *software* (leitura de dados básicos do usuário).

O SGIR constitui o passo inicial, e necessário, para o desenho de um sistema baseado em conhecimento que aconselha o professor na planificação de atividades, realize um acompanhamento do comportamento dos alunos e sugira possíveis modificações em tais atividades em função das observações registradas pelos agentes humanos. A implantação do SGIR é necessária na medida que permite validar o modelo de conhecimento do professor e a viabilidade do gerenciamento dos recursos do ambiente educativo.

Fontes Bibliográficas

AAMODT, A. et al. **The CommonKADS library** , University of Amsterdam, Netherlands Energy Research Foundation, 1992.

ABEN, M. **A library of inference schemata** . Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam, 1994.

ABEN, M. **Formal methods in knowledge engineering** . 1995. Tese (Ph.D. Thesis) – University of Amsterdam, Netherlands.

ABOWD, G. D. **Classroom 2000: An experiment with instrumentation of living educational environment** , IBM System Journal, vol 38, N. 4, 1999.

AGELET, Joan; BASSEDAS, Eulália; COMADEVALL, Magdalena. **Algunos modelos organizativos, facilitadores del tratamiento de la diversidad, y alternativos a los agrupamientos flexibles** . Revista Aula de Inovación Educativa, maio/1997, Barcelona.

ANDRADE, P. F.& ALBUQUERQUE Lima, M.C.M. **Projeto EDUCOM** . Brasília: MEC/OEA, 1993.

BRANSFORD, J.D.; Stein, B.S. **The Ideal Problem Solver (2nd Ed)** . New York: Freeman, 1993.

BREUKER, J.; VAN DE VELDE, W. **CommonKADS library for expertise modelling**. Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam, 1994.

BRUNER, J. **The Process of Education** . Cambridge, MA. Harvard University Press. 1960.

BRUNER, J. **Toward a Theory of Instruction** . Cambridge, MA. Harvard University Press. 1966.

BRUNER, J. **Actual Minds, Possible Worlds** . Cambridge, MA. Harvard University Press. 1986.

BRUNER, J. . **Acts of Meaning** . Cambridge, MA. Harvard University Press. 1990

BOEHM, B. **A spiral model of software development and enhancement** . IEEE Computer, 1988

BUCHANAN, B.et al. **Constructing an expert system** , Adison-Wesley, 1983.

CTGV. **Anchored instruction and its relationship to situated cognition.** *Educational Researcher*, 19 (6), 2-10, 1990.

DOMÈNECH, Joaquim. **Algunas Técnicas para el aprendizaje cooperativo.** Revista Aula de Inovación Educativa, No. 59, Fevereiro 1997, Barcelona.

FERRAZ, C. A. Guimarães; TRINTA, F. A. Mota; MACEDO, R. C.; ARAÚJO, R. Cysne, **Co-Autoria Distribuída de Cursos na Internet.** Revista Brasileira de Informática na Educação, No 6, Abril de 2000, Florianópolis.

FUKS, Hugo. **Aprendizagem e Trabalho Cooperativo no Ambiente AulaNet** , Revista Brasileira de Informática na Educação, No 6, Abril de 2000, Florianópolis.

GALLAGHER, J.M.; REID, D.K. **The Learning Theory of Piaget and Inhelder** . Monterey, CA: Brooks/Cole, 1981.

GROSS, Salvat. **El ordenador invisible** . Ed. Gedisa, colección Biblioteca de la Educación, N. 1, Barcelona, 2000.

HOOG, R. Et al. **The CommonKADS model set** . University of Amsterdam, Lloyd's Register, Brussels, 1994.

HOOG, Robert de. **Methodologies for Building Knowledge Based System: Achievements and Prospects** . The Handbook of Applied Expert Systems, cap. 1, págs 1-14. CRC Press LLC, London, 1998.

HOOG, Robert de et al. **The Handbook of Applied Expert Systems** . CRC Press LLC, London, 1998.

KUMAR, V. **Computer-Supported Collaborative Learning: issues for research** . Department of Computer Science, University of Saskatchewan, Canada. 1996 .

LAMA, M. et al. **A task model for a Management Resource System integrated in a cooperative learning environment** , KES 2002, Setembro de 2002.

LAVE, J. **Cognition in Practice: Mind, mathematics, and culture in everyday life**. Cambridge, UK: Cambridge University Press. 1988.

LAVE, J.,; WENGER, E. **Situated Learning: Legitimate Peripheral Participation**. Cambridge, UK: Cambridge University Press. 1990.

LÈVY, Pierre. **Cibercultura** . São Paulo: Editora 34, 1999.

MARTINS, Ronei X. **Aprendizagem Cooperativa via Inter net – A implantação de dispositivos computacionais para a viabilidade técnica de cursos on -line**. 2000. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, UFSC, Florianópolis.

MENDEZ, José Tomas Palma. **Ingenieria Del Conocimiento para Sistemas en Tiempo Real Basados en Conocimiento: Una Extension a CommonKADS**. 1999. Tese (Doutorado em Informática) - Programa de Doutorado do Departamento de Informática, Inteligência Artificial e Eletrônica, Universidade de Murcia, Espanha.

MORAES, Maria Candida. **Informática Educativa no Brasil: um pouco de história**. Brasília, revista Em Aberto, ano 12, n. 57, jan/mar. 1993.

MORAN, José Manuel. **Como Utilizar a Internet na Educação**. Artigo publicado na Revista Ciência da Informação, Vol 26, n.2, maio-agosto 1997, pág. 146-153

MORAN, José Manuel. **Mudar a forma de ensinar e de aprender com tecnologias**: Transformar as aulas em pesquisa e comunicação presencial-virtual. Disponível em: <<http://www.eca.usp.br/prof/moran/uber.htm>>. Acesso em: Maio 2002.

NEWELL, A. **The knowledge level**. Artificial Intelligence. 1982.

NEWELL, A. **Reflections on the Knowledge Level** . Artificial Intelligence, 1993.

PANITZ, Ted. **A definition of Collaborative vs Cooperative learning** . The DeliBerations Website. London Guildhall University, 1996.

PARK, O., PEREZ, R.S. and SEIDEL, F.J., **“Intelligent CAI: Old Wine in New Bottles or a New Vintage ?”** , em Kearsley, G., “Artificial Intelligence and Instruction - Applications and Methods” ; pp. 11-45, 1993.

PARRILLA, A. **El professor ante la integración escolar: investigación y formación**. Cincel, Buenos Aires, 1992.

PASS, Leslie Christine. **A Integração da Abordagem Colaborativa à Tecnologia Internet para Aprendizagem Individual e Organizacional no PPGE** . 1999. Dissertação (Mestrado em Engenharia de Produção) – Programa de Pós-Graduação em Engenharia de Produção, UFSC, Florianópolis.

PENÍN, M. Lama. **Modelo del conocimiento y arquitetura para la síntesis de un especialista terapéutico en Unidades de Cuidados Intensivos y Coronarios**. 2000. Tese (Doutorado em Eletrônica e Informática) – Programa de Doutorado em Eletrônica e Informática do Departamento de Eletrônica e Computação, Universidade de Santiago de Compostela, USC, Espanha.

PIAGET, J. **The Science of Education and the Psychology of the Child** . NY. Grossman. 1970.

PIAGET, J.; Inhelder, B. . **The Psychology of the Child** . NY: Basic Books. 1969.

PIAGET, J.; Inhelder, B. . **Memory and intelligence** . NY: Basic Books. 1973.

RAMO, J. **Welcome to the Wired World** . Time Magazine Special Report "The Networked Society" Canadian Edition, v. 149, n. 5, 1997.

RIERA, A.; BARRO, S.; VILA, J. **A PDA classroom computer system** , Proceedings of the World Conference on Educational Multimedia, Hipermedia; Telecommunications (ED-MEDIA'01), 2001.

SÁNCHEZ, E. et al. **A multi-tiered agent-based architecture for a cooperative learning environment**. 11-th Euromicro Conference on Parallel Distributed and Network based Processing, Genova, Itália. Fevereiro de 2003. Submetido em junho de 2002. no prelo.

SANTORO, F. M.; BORGES, M. R. S.; SANTOS, N. **Um framework para estudos de ambientes de suporte à aprendizagem cooperativa**. Revista Brasileira de Informática na Educação. 4, p. 51-68. 1998.

SCHREIBER, A.; WIELINGA, B.; BREUKER, J. (eds.). **KADS: A principled approach to knowledge -based system development** . Knowledge-Based Systems, Academic Press, London, 1993.

SCHREIBER, A. et al. **CommonKADS: A comprehensive methodology for KBS development** . IEEE Expert Systems, 1994.

SCHREIBER, G. et al. **Knowledge Engineering and Management: The CommonKADS Methodology** . MIT Press, Boston, 1999. ISBN 0-262-19300-0.

SHABO, A.; NAGEL, K.; GUZDIAL, M. **JavaCAP: A Collaborative Case Authoring Program on the WWW**. Proceedings of Computer Support for Collaborative Learning'97.1997.

SHOTSBERGER; VETTER, IEEE Computing, Março 2001.

SKINNER, B.F. **Science and Human Behavior** . New York: Macmillan. 1953.

SLAVIN, R.E. **Research on Cooperative Learning and Achievement: What We Know, What We Need to Know**. Artigo adaptado de Slavin, 1992 Center for Research on the Education of Students Placed at Risk Johns Hopkins University. 1995.

SMYSER, B.M. **Active and Cooperative Learning** . 1993.

STEELS, L. **Second Generation Expert Systems** . Future Generation Computer Systems, 1985.

SUTHERS, D. **Combining Pedagogical and Technological Paradigms for Educational Software**. *Position Paper CHI'96 Research Symposium*. 1996.

VALENTE, José Armando. **Diferentes usos do computador na educação** . Brasília, revista Em Aberto, ano 12, n. 57, jan/mar. 1993.

VAN HEIJST , G.; SCHREIBER, A.; WIELINGA, B. **Using explicit ontologies for KBS development** . International Journal of Human-Computer Studies, 1997.

VERMESAN, A.; BENCH-CAPON, T. **Techniques for the verification and validation of knowledge-based systems: A survey based on the symbol/knowledge level distinction**. Software Testing, Verification, Reliability, 1995.

VIEIRA, F. **Avaliação de Software Educativo: Reflexões para uma Análise Criteriosa**. Disponível em: <<http://www.edutecnet.com.br/Textos/Alia/MISC/edmagali2.htm>>. Acesso em: 06 fev. 2002.

VYGOTSKY, L.S. . **Thought and Language** . Cambridge, MA: MIT Press. 1962.

VYGOTSKY, L.S. . **Mind in Society** . Cambridge, MA. Harvard University Press. 1978.

WERTSCH, J.V. **Cultural, Communication, and Cognition: Vygotskian Perspectives** . Cambridge University Press. 1985.

WIELINGA, B.; SCHREIBER, A.; BREUKER, J. **KADS: A modelling approach to knowledge engineering** . Knowledge Acquisition, 1992.

WIELINGA, B. et al. **Towards a unification of knowledge modelling approaches** . Second Generation Expert Systems, págs 299-335. Springer-Verlag, 1993.

WIELINGA, B. et al. **Expertise Model Definition Document** . Inf. Téc. ESPRIT Project P5248 KADS-II/M2/UVA/026/5.0, University of Amsterdam, 1994.