



**Campus de São Carlos**

**PROPOSTA DE UMA ARQUITETURA (FRAMEWORK)  
TECNOLÓGICA QUE APÓIA O DESENVOLVIMENTO  
DO APRENDIZADO INDIVIDUAL E ADAPTATIVA EM  
INSTITUIÇÕES DE ENSINO**

**Marco Antonio Aparecido Leite**

**Orientador: Prof. Dr. Edson Walmir Cazarini**

**UNIVERSIDADE DE SÃO PAULO**



**ESCOLA DE ENGENHARIA  
DE SÃO CARLOS**

Marco Antonio Aparecido Leite

**PROPOSTA DE UMA ARQUITETURA (FRAMEWORK)  
TECNOLÓGICA QUE APÓIA O DESENVOLVIMENTO DO  
APRENDIZADO INDIVIDUAL E ADAPTATIVA EM INSTITUIÇÕES  
DE ENSINO**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de mestre em Engenharia de Produção.

DEDALUS - Acervo - EESC



31100050042



Orientador: Prof. Dr. Edson Walmir Cazarini

SÃO CARLOS

2004

Serviço de Pós-Graduação EESC/USP

EXEMPLAR REVISADO

Data de entrada no Serviço.....12/11/04.....

Ass.....*[Signature]*.....

Class.	TESE EESC ✓
Codif.	3017
Tomo	1273/04
Sismo	1407531

Ficha catalográfica preparada pela Seção de Tratamento  
da Informação do Serviço de Biblioteca – EESC/USP

L533p Leite, Marco Antonio Aparecido  
Proposta de uma arquitetura (*Framework*) tecnológica  
que apóia o desenvolvimento do aprendizado individual e  
adaptativa em instituições de ensino / Marco Antonio  
Aparecido Leite. São Carlos, 2004.

Dissertação (Mestrado) — Escola de Engenharia de São  
Carlos-Universidade de São Paulo, 2004.

Área: Engenharia de Produção.  
Orientador: Prof. Dr. Edson Walmir Cazarini.

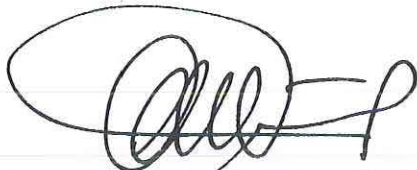
1. Agentes inteligentes. 2. EAD. 3. E-learning.
4. LOM. 5. Objetos de aprendizagem. 6. Sistemas tutores.
7. Serviços Web. 8. SCORM. Título.



**FOLHA DE JULGAMENTO**

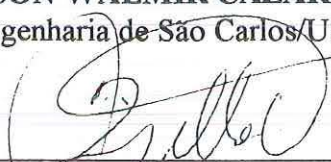
Candidato: Analista de Sistemas **MARCO ANTONIO APARECIDO LEITE**

Dissertação defendida e julgada em 27-09-2004 perante a Comissão Julgadora:



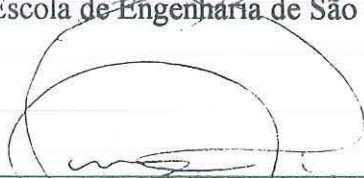
Prof. Dr. **EDSON WALMIR CAZARINI (Orientador)**  
(Escola de Engenharia de São Carlos/USP)

Aprovado



Prof. Associado **RENATO VAIRO BELHOT**  
(Escola de Engenharia de São Carlos/USP)

APROVADO



Prof. Dr. **ÂNGELO CÉSAR COLOMBINI**  
(Universidade de Franca/UNIFRAN)

Aprova.do



Prof. Doutor **FÁBIO MULLER GUERRINI**  
Vice-Coordenador do Programa de Pós-Graduação  
em Engenharia de Produção, em Exercício.



Prof. Associado **MARCIO ROBERTO SILVA CORRÊA**  
Vice-Presidente da Comissão de Pós-Graduação, em Exercício



## DEDICATÓRIA

À minha querida esposa Marcilene que, nos momentos difíceis, sempre me apoiou com seu amor e perseverança, a meus filhos Beatriz, Gabriel, Guilherme e Laura pela compreensão de minha ausência, ao meu orientador Dr. Cazarini que me apoiou e me orientou, a meus queridos pais pela minha formação e meu caráter, a meu irmão Paulo Henrique (In Memoriam) que me acompanha em todos os meus momentos, a minha avó Enedina minha segunda mãe a qual irei admirar sempre.

## AGRADECIMENTOS

A Deus pela sustentação que, quando cansado me carregou nos momentos difíceis.

Ao amigo e Prof<sup>o</sup> Dr. Edson Walmir Cazarini, pela sua simpatia, simplicidade, serenidade, seriedade e excelência na orientação da pesquisa e pela compreensão das minhas dificuldades e dedicação demonstrada durante esse período de convivência.

Aos Professores Dr. Renato Vairo Belhot que confiou em meu trabalho e ao Dr. Angelo César Colombini pelas considerações que contribuíram para o resultado final deste trabalho.

A todos os professores da pós-graduação do departamento de engenharia de produção em especial aos Professores Edmundo Escrivão Filho e Fernando César Almada Santos o qual me instruíram de maneira brilhante em suas aulas.

Aos amigos Alexandre Albuquerque, Mayb Fiats, Teobaldo Rivas que estiveram presentes com sua amizade.

Ao meu amigo Dr. Flávio Toledo e sua esposa Dr. Cláudia Queda Toledo pelo apoio em tantos momentos importantes de minha vida.

Ao Diretor de secretaria do departamento de engenharia de produção, José Luiz que, pelo seu excelente trabalho, me apoiou durante o período de mestrado.

A USP, que em sua missão de educar para o Brasil, me ofereceu todas as condições possíveis para a realização desta pesquisa.

## RESUMO

LEITE, M. A. A. (2004). *Proposta de uma arquitetura (framework<sup>1</sup>) tecnológica que apóia o desenvolvimento do aprendizado individual e adaptativa em instituições de ensino*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2004.

A maioria das instituições educacionais prioriza o desenvolvimento de sistemas informacionais com foco operacional ou administrativo em seu primeiro momento postergando o desenvolvimento de sistemas que controlam o treinamento e aprendizado. A arquitetura proposta visa adaptar-se aos sistemas existentes, suportando a utilização de tecnologias que ajudam a evolução do treinamento e aprendizado de forma individual e não apenas utilizar tecnologia como um veículo distribuidor de informação, característica presente na maioria das apresentações multimídia. Apresenta uma revisão bibliográfica sobre a importância do aprendizado individual, identificando como as avaliações empregadas de forma presencial podem ter um formato mais dinâmico no ambiente virtual. Também são avaliadas tecnologias que fornecem subsídios para a criação da arquitetura, suprimindo algumas deficiências de fragmentação e de isolamento do aprendizado do aluno, melhorando o desempenho do conhecimento individual. É apresentado no final um modelo de estrutura que irá permitir a integração das tecnologias abordadas que apóia o desenvolvimento do aprendizado individual e adaptativa a sistemas de informações existentes em instituições de ensino.

Palavras-chave: Agentes Inteligentes, EAD, E-Learning, LOM, Objetos de Aprendizagem, Sistemas Tutores, Serviços Web, SCORM.

---

<sup>1</sup> *Framework* – Estrutura que permite o desenvolvimento de ambientes customizáveis integrando ferramentas disponíveis.



## ABSTRACT

LEITE, M.A.A. (2004). *Proposition of a technological architecture (framework) that supports the development of individual learning and is adjustable in teaching institutions*. Dissertation (Ms) - Sao Carlos' School of Engineering, University of Sao Paulo, Sao Carlos, 2004.

The majority of educational institutions prioritize the development of informational systems with an operational or administrative focus at first, depreciating the development of systems that control training and learning. The proposed architecture aims to adapt itself to existing systems, supporting the application of technologies that help the evolution of training and learning in an individual way and not only using technology as a vehicle that distributes information, characteristic which is present in most of the multimedia presentations. The paper presents a review of the literature on the importance of individual learning, identifying how the evaluations employed in a present way may have a more dynamic format in the virtual environment. Technologies which supply subsidies for the creation of the architecture are also evaluated, suppressing some deficiencies of fragmentation and isolation of the student's learning, improving the performance of the individual learning. It is presented at the end a model of structure that will allow the integration of the approached technologies which support the development of the individual learning and which can be adapted to existing information systems effective in teaching institutions.

Keywords: E-learning, intelligent agents, LOM, Learning objects, SCORM, tutoring systems, web-services.

## LISTA DE FIGURAS

Figura 1 - Modelo didático do sistema de avaliação	23
Figura 2 - Modelo gráfico de sistema de avaliação virtual	24
Figura 3 - Desenvolvimento histórico dos paradigmas educacionais	39
Figura 4 - Estrutura de um sistema especialista	41
Figuras 5 - Componentes de um sistema tutor inteligente genérico	45
Figura 6 - Definição de agentes	49
Figura 7 - Arquitetura de ambiente ensino aprendizagem utilizando agentes e sistemas tutores inteligentes	51
Figura 8 - Funcionamento global de um objeto de aprendizagem	54
Figura 9 - Divisão de um objeto de aprendizagem	55
Figura 10 - Representação do esquema de uma estrutura funcional	58
Figura 11 - <i>PIF - Package Interchange File</i>	59
Figura 12 - Construção da árvore do conhecimento	60
Figura 13 - Codificação de um manifesto ( <i>manifest</i> )	61
Figura 14 - Visão hierárquica do metadado <i>LOM</i>	63
Figura 15 - Ambiente de execução ( <i>RunTime Environment</i> ) do <i>SCORM</i>	66
Figura 16 - Relação entre o <i>SCO</i> e a sessão aberta	68
Figura 17 - Tipos de chamadas de um <i>SCO</i>	69
Figura 18 - Dinâmica da execução das chamadas pela <i>API</i>	70
Figura 19 - Flash5 - Aplicando a chamada <i>LMSInitialize</i> na <i>API</i>	70
Figura 20 - Flash5 - aplicando a chamada a <i>LMSGetValue</i> na <i>API</i>	71
Figura 21 - Flash5 - aplicando a chamada a <i>LMSFinish</i> na <i>API</i>	71
Figura 22 - Diagrama da funcionalidade do <i>Web-Service</i>	75
Figura 23 - Estrutura de uma mensagem <i>SOAP</i>	76
Figura 24 - Componentes da arquitetura proposta	81
Figura 25 - Adaptação do modelo apresentado ao modelo de Prado (2002)	83
Figura 26 - Modelagem da estrutura informacional formal	85
Figura 27 - Sincronização da base de conhecimento de domínio e	

da estrutura informacional funcional	92
Figura 28 - Formação da base de objetivos	95
Figura 29 - Solicitação de treinamento para um ou mais domínio	97
Figura 30 - Os objetos são apresentados ou interagidos pelo aluno através das <i>API</i> 's	98
Figura 31 - Atualização da base de perfil individual	99
Figura 32 - Atualização da base de avaliação	102
Figura 33 - Envio de recurso (conteúdo) a base de conhecimento	103
Figura 34 - Funcionalidade global da interface	105
Figura 35 - Funcionalidade global da estrutura proposta	107



## LISTA DE QUADROS E TABELAS

Quadro 1 – Comparação análoga da evolução da tecnologia de ensino-aprendizagem e a interação com o aluno	38
Quadro 2 – Alguns exemplos dos elementos e sub-elementos do LOM	64
Quadro 3 – Identificação das entidades e atributos da estrutura informacional formal	89
Quadro 4 – Tipo de recurso	92
Quadro 5 – Outras características do objeto de aprendizagem com relação a funcionalidade e interação	93

## LISTA DE SIGLAS

- ADL - *Advanced Distributed Learning*
- AICC - *Aviation Industry CBT Committee*
- API - *Application Program Interface*
- ASP - *Active Server Page*
- CAI - *Computer Aided Instruction*
- CBT - *Computer Based Training*
- CDROM - *Compact Disk Read Only Memory*
- CSCCL - *Computer Supported Collaborative Learning*
- DoD – *Department of Defense*
- DOM – *Document Object Model*
- DOU – *Diário Oficial da União*
- EAD - *Educação à Distância*
- FK - *Foreign Key*
- HTML - *Hypertext Markup Language*
- IA - *Inteligência Artificial*
- ICAI - *Intelligent Computer Aided Instruction*
- IEEE - *Institute of Electrical and Electronics Engineers*
- ILE - *Intelligent Learning Environment*
- IMS - *Instructional Management System - Global Learning Consortium*
- ITS - *Intelligent Tutoring Systems*
- KQML - *Knowledge Query and Manipulation Language*
- LED - *Laboratório de Educação à Distância*
- LMS - *Learning Management Systems*
- LOM - *Learning Object Metadata*
- LTSC - *Learning Technology Standards Committee*
- MEAS - *Modelos de Ensino-Aprendizagem Semipresencial*
- MEC – *Ministério da Educação e Cultura*
- MER - *Modelo Entidade e Relacionamento*
- NASA - *National Aeronautics and Space Administration*

PC - *Personal Computer*

PIF - *Package Interchange File*

PK - *Primary Key*

RPC - *Remote Procedure Call*

RTS - *Run Time Service*

SCO - *Sharable Content Object*

SCORM - *Sharable Content Object Reference Model*

SE - *Sistemas Especialistas*

SOAP - *Simple Object Access Protocol*

STI - *Sistemas Tutores Inteligentes*

UDDI - *Universal Description Discovery and Integration*

UFSC - *Universidade Federal de Santa Catarina*

USP - *Universidade de São Paulo*

W3C - *World Wide Web Consortium*

WBT - *Web-based Training*

WSDL - *Web Services Description Language*

WS-I - *Web Services Interoperability Organization*

XML - *Extensible Markup Language*



## SUMÁRIO

<b>RESUMO</b>	<b>1</b>
<b>ABSTRACT</b>	<b>2</b>
<b>LISTA DE FIGURAS</b>	<b>3</b>
<b>LISTA DE SIGLAS</b>	<b>6</b>
<b>SUMÁRIO</b>	<b>8</b>
<b>1 INTRODUÇÃO</b>	<b>10</b>
1.1 Caracterização do tema da pesquisa	11
1.2 Formulação do problema da pesquisa	14
1.3 Declaração do objetivo da pesquisa	15
1.4 Justificativa da pesquisa	15
1.5 Motivação pela pesquisa	16
1.6 Estrutura dos capítulos	17
<b>2 A NECESSIDADE DA IDENTIFICAÇÃO DO PERFIL DO APRENDIZADO INDIVIDUAL</b>	<b>18</b>
2.1 Identificação do perfil do aluno através do treinamento e avaliação	19
2.2 Identificação do perfil do aluno através da tecnologia	26
<b>3 AMBIENTE VIRTUAL DE ENSINO</b>	<b>28</b>
3.1 A Internet no ambiente virtual de ensino	30
3.1.1 Ensino à distância	31
3.1.2 Treinamento à distância ( <i>E-Learning</i> )	33
3.2 Evolução das tecnologias virtuais de treinamento na educação	35
3.3 Tecnologias utilizadas na construção da arquitetura	40
3.3.1 Sistemas Especialistas	40
3.3.2 Sistemas tutores	43
3.3.3 Agentes Inteligentes	47

3.3.4	<b>Objetos de Aprendizagem (<i>Learning Objects</i>)</b>	52
3.3.5	<b>Serviço Web (<i>Web-Services</i>)</b>	72
<b>4</b>	<b>ARQUITETURA PROPOSTA</b>	<b>77</b>
4.1	<b>Componentes da arquitetura</b>	<b>80</b>
4.2	<b>Estrutura informacional formal</b>	<b>83</b>
4.3	<b>Base de conhecimento do domínio</b>	<b>89</b>
4.4	<b>Base de objetivos</b>	<b>94</b>
4.5	<b>Base de informações do Estudante</b>	<b>96</b>
4.6	<b>Base de perfil individual</b>	<b>98</b>
4.7	<b>Base de avaliação</b>	<b>100</b>
4.8	<b>Recursos enviados</b>	<b>102</b>
4.9	<b>Interface</b>	<b>104</b>
4.10	<b>Integração dos recursos na arquitetura proposta</b>	<b>107</b>
<b>5</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS</b>	<b>108</b>
	<b>REFERÊNCIA BIBLIOGRÁFICA</b>	<b>111</b>

## 1 INTRODUÇÃO

Novas tecnologias estabelecem novos formatos de comunicação e interação, e na sua utilização desafiam antigos e modernos educadores ao integrar em sua metodologia estes recursos, na busca de maior eficiência e eficácia do aprendizado. Mas não basta que as escolas sejam instrumentalizadas com computadores e equipamentos de última geração com o intuito de manter o mesmo procedimento aplicado em aulas presenciais, o movimento de atualização deve aplicar novas concepções e abordar as vantagens na utilização dessas tecnologias. Os novos recursos oferecidos pela tecnologia, aplicados na inovação dos processos de ensino-aprendizagem apóiam a questão da expansão de ensino, produzindo a disseminação e acessibilidade de conteúdo informacional de maneira mais dinâmica, interagindo com o aluno, possibilitando um caminho de comunicação bidirecional.

Andrade e Brasileiro (2003) reforçam que na dinâmica da avaliação do aprendizado a utilização de novos recursos tecnológicos permite que sejam eliminados elementos padronizados que podem empobrecer sua análise e a interação tende a suprimir as deficiências de fragmentação e de isolamento, maximizando a identificação do perfil individual, promovendo um melhor trabalho do professor em termos de grupo nas aulas presenciais. Esta possibilidade é apoiada através de sistemas computacionais ao trabalharem em rede com a tecnologia de telecomunicação (telemática), principalmente quando utiliza recursos tecnológicos integrados ao ambiente da internet, capacitando maior eficiência da interatividade, promovendo um ambiente integrado entre o ensino presencial e à distância. Segundo Oliveira (2001) a interatividade propiciada pelo uso da Internet permite maior individualização do processo educativo, propiciando um melhor acompanhamento do ensino à distância, ao permitir o estudo continuado em casa ou no

trabalho, oferecendo amplas soluções de ambiente colaborativo educacional das instituições de ensino.

A tecnologia amplia de modo exponencial a capacidade de atuação do professor e a acessibilidade do aluno a materiais que, antes, eram praticamente impossíveis de serem alcançados, mas essa capacidade de se atingir uma diversidade cada vez maior de alunos, pode, em certos momentos, criar um ambiente fraco na questão do quanto e como as pessoas estão compreendendo os termos de conteúdo da informação instrucional disponibilizada e se a mesma está sendo compreendida. Ocorre portanto uma necessidade de uma dinâmica maior na preparação e aplicação da avaliação, em que, após analisar seus dados obtidos, é possível construir elementos que possam identificar, observar e controlar indícios que indicam o sucesso do entendimento do material oferecido.

Através da tecnologia interativa é possível construir sistemas que permitem a monitoração e análise do comportamento das variáveis de desempenho do ambiente de aprendizagem presencial e virtual, identificando possíveis falhas e ameaças que podem trazer uma reflexão sobre a escolha de melhores caminhos a serem seguidos, promovendo um direcionamento objetivo para a utilização de seus recursos.

Moraes (1998) também reafirma o conceito de renovação de modelos de ensino-aprendizagem, onde as novas tecnologias e a forma de educar na era da informação e para a sociedade do conhecimento, exigem que sejam extrapoladas as questões didáticas, os métodos de ensino, os conteúdos curriculares, de maneira a encontrar caminhos mais adequados e congruentes com o momento histórico que estamos vivendo.

Todos esses aspectos de inovação e renovação tecnológica implicam o repensar da escola, dos processos de treinamento de ensino, da avaliação da aprendizagem, do redimensionamento do papel do professor, da instituição de ensino e seus métodos e da maneira como o próprio aluno deverá participar e interagir no processo de ensino-aprendizagem.

## **1.1 Caracterização do tema da pesquisa**

O acesso à informação e a liberdade total na exploração do material instrucional é uma idéia favorável ao processo de aprendizagem, uma vez que possibilita ao aluno ter



controle sobre o aprendizado e estudar se torna mais pessoal e interessante. No entanto, situações em que o aluno apresenta melhor rendimento são aquelas em que ele é guiado ou orientado de alguma forma; esta posição também é defendida por Freire (1998). Mas a orientação fica cada vez mais difícil quanto maior é o número de participantes, principalmente quando estes participantes são trabalhados na forma de ensino à distância.

As avaliações quando aplicadas de forma padronizada nas instituições de ensino são empobrecidas e reduzidas na identificação do aprendizado individual, quando, na maioria, a própria avaliação não consegue encampar todos os tópicos ministrados em sala de aula, dificultando a observância da evolução individual dos alunos e de reescrever ações corretivas na disponibilidade de material instrucional. Dias (2000) corrobora também nesse sentido, onde a aprendizagem é tomada como desempenho e que, na maior parte das vezes, medida de forma padronizada, não respeitando o perfil individual de cada aluno.

A partir de 1990, Matuzawa (2001), enfatiza a utilização de recursos tecnológicos na área educacional possibilitando a comunicação interativa, garantindo um *feedback* mais rápido e uma comunicação mais dinâmica entre os envolvidos, onde este ambiente pode sustentar o professor na oferta e preparo de avaliações de forma mais dinâmica, identificando necessidades individuais, proporcionando melhores condições para a orientação e disponibilização de material instrucional de forma mais objetiva às necessidades individuais.

Na ânsia da busca do perfil individual, os dados analisados podem apresentar-se bastante numeroso e diversificado, onde, métodos de forma convencional (manual) caracterizam-se com pouca eficiência, principalmente quando o número de alunos também é elevado. É necessário estar presente uma arquitetura tecnológica eficiente apoiada em sistemas de informações que facilite a aplicação de processos de treinamento e avalie o aprendizado (realizado ou não de forma presencial) de forma a identificar os caminhos de um melhor desempenho tanto do aluno quanto do professor.

É importante diferenciar treinamento de educação. Flippo (1973) afirma que o treinamento é um processo através do qual ocorre o incremento da perícia e conhecimento para a execução de um determinado trabalho.

O treinamento ocupa-se do ensino de habilidades para fins específicos (especialidade buscada na maior parte das vezes através das disciplinas oferecidas nas instituições de

ensino), enquanto a educação pressupõe o desenvolvimento global da pessoa (social, intelectual e físico), afirmação está também apoiada por Chiavenato (1981).

O treinamento personalizado favorece o aprendizado, vem propiciar foco objetivo para aumento do entendimento da disciplina, constituindo-se importante ferramenta de apoio à sedimentação do conhecimento. Esta individualização colabora com a identificação e monitoração de desempenho e da capacidade de resolução de problemas oferecidos; tais desempenhos também são melhorados quando o treinamento está relacionado ao encontro das necessidades pessoais.

Construir sistemas de informação que possibilitem acompanhar a evolução do aprendizado e promova a construção do conhecimento não ocorre de forma simplista, envolve um conjunto de informações de diversos grupos e suas combinações, Moresi (2001) define que para construir sistemas de informação que possam administrar bases de informações que, promovam a construção do conhecimento requer a parceria de três grupos de trabalhadores:

- peritos de domínio, que possuam profundo conhecimento sobre um determinado tema ou assunto;
- peritos em informação, que organizem a informação em recursos úteis e reutilizáveis;
- especialistas em tecnologia da informação, que construam a infraestrutura técnica para apoiar o gerenciamento e o compartilhamento da informação.

A constituição desses grupos requer em um alto grau de investimento tecnológico e humano, além da possível dificuldade de identificar um bom formato de conteúdo dos peritos de domínio, inviabilizando projetos desta complexidade. Prado (2002) afirma que esta base de informação deve promover o conhecimento, deve ser composta por conteúdos, exercícios e avaliações do domínio a ser ensinado. Portanto, seria interessante apresentar em sua construção uma estrutura organizada e padronizada e que possa ser intercambiável para que possam ter menor impacto de investimento nos grupos abordados.

Os fatos acima apresentados indicam a necessidade das instituições de ensino possuírem uma arquitetura, capaz de utilizar e reutilizar recursos tecnológicos, identificando a eficiência e eficácia das ações de aprendizagem (treinamento e avaliação)



de forma individual e que possa auxiliar a decidir questões como: O aproveitamento dos alunos em temas abordados pelo professor foi eficiente? Qual a deficiência da informação em um treinamento? O método abordado pela instituição ou o professor é eficaz? Como está a evolução deste aprendizado?

Para que essas informações possam ser colhidas, processadas e disponibilizadas de forma eficiente, no formato desejado e no momento oportuno, a arquitetura deve possibilitar a criação de um sistema de informações que possam ser reativos e que reúnam, processem e distribuam os dados e informações de maneira correta e ágil, colaborando na eficácia do aproveitamento da aprendizagem individual.

## 1.2 Formulação do problema da pesquisa

Não basta que as instituições de ensino trabalhem com equipamentos de alta tecnologia somente para movimentos administrativos, comportando-se ao utilizar a alta tecnologia para o aprendizado em ações instrucionais expositivas, tentando alcançar o máximo número de alunos possível, sem identificar se ocorreu o aprendizado.

Muitas escolas, quando adotam sistemas tecnológicos computacionais, implementam em sua primeira fase foco no operacional e administrativo, movimento este bem explicado na evolução de sistemas de informações apresentado por Laudon (2004), postergando movimentos tecnológicos na área de treinamento e aprendizado.

Sabe-se que em sistemas administrativos/financeiros educacionais quando é emitida a cobrança da mensalidade, a tecnologia identifica se a mesma foi paga ou não individualmente, ou seja, preocupa-se com o movimento de retorno, o que deveria acontecer com o aprendizado, ou seja, ter controles e *feedback's*, sustentando a identificação e a evolução do processo de ensino e aprendizagem também de forma individual.

Dessa forma, a essência da pesquisa, está no seguinte questionamento:

Como uma arquitetura tecnológica pode apoiar o controle do aprendizado individual nas instituições de ensino?

### 1.3 Declaração do objetivo da pesquisa

O objetivo geral desta pesquisa é propor uma arquitetura (*framework*<sup>2</sup>) tecnológica, que apóie o controle do aprendizado individual e que possa adaptar-se em sistemas operacionais acadêmicos existentes.

Como objetivos específicos:

- pesquisar literatura sobre o porquê da necessidade de identificar, monitorar e acompanhar o desempenho do aprendizado individual;
- investigar literatura das tecnologias emergentes e inteligentes que possam colaborar na monitoração e desempenho do aprendizado individual e que irão colaborar na construção da arquitetura proposta;
- propor um modelo de arquitetura tecnológica que irá contribuir para a monitoração e controle do aprendizado individual através das tecnologias investigadas.

Não faz parte do objetivo deste trabalho comentar e definir questões filosóficas de ordem educacional como o construtivismo, empirismo entre outros.

### 1.4 Justificativa da pesquisa

Ao utilizar a arquitetura proposta, construtores de sistemas de informações serão direcionados a como criar uma base de informações que, poderá fornecer informações para análise de desempenho de aprendizado para as esferas da coordenação, corpo docente e discente. Esta base de informações depois de processadas e analisadas por outros sistemas irão colaborar com:

- o discente que será orientado como acessar informações e treinamento de forma mais objetiva, através das interações realizadas, possibilitando retorno e *feedback* de desempenho individual.
- as interações realizadas alimentaram uma base de dados, que sustentará a identificação o perfil individual pelo docente, o que irá auxiliar no

---

<sup>2</sup> *Framework* – Estrutura que permite o desenvolvimento de ambientes customizáveis integrando ferramentas disponíveis.



direcionamento de forma mais eficaz na transmissão (ou reforço) dos temas abordados e informações a serem disponibilizadas;

- a coordenação das entidades de ensino-aprendizagem que poderão receber informações substanciais para a identificação da eficácia dos métodos abordados pelos professores em aulas presenciais quando confrontados com o treinamento virtual;

Para que as informações possam ser capturadas, manipuladas e distribuídas de forma coerente, é necessário desenvolver uma arquitetura que embase o gerenciamento do desempenho e controle da aprendizagem individual, mas, como foi mencionado, na formulação do problema da pesquisa, as entidades de ensino-aprendizagem investem em controles administrativos e operacionais e não dispõem de ferramenta computacional estruturada de uma ordem prática e padronizada, e que promova o controle e gestão do movimento de desempenho da aprendizagem dos alunos nela inseridos.

Dessa forma, este trabalho propõe a utilização de uma arquitetura que irá apoiar o controle do processo de treinamento individual e que possa adaptar-se em sistemas de informações existentes nas instituições de ensino.

## **1.5 Motivação pela pesquisa**

Programas computacionais de gestão em administração escolar, em sua maioria enfatizam a parte administrativa e o controle operacional. Sistemas que controlam o desempenho do ensino individual normalmente não se apresentam de forma constante e quando se apresentam não estão de forma estruturada e organizada. Esta inquietação torna-se mais contundente no momento em que novas tecnologias estão sendo utilizadas e propiciando um maior alcance de ensino das instituições de ensino-aprendizagem (de forma global) e por consequência aumentando a necessidade de se utilizar sistemas de acompanhamento da evolução do aprendizado dos alunos envolvidos. As tecnologias atuais permitem uma interação muito mais dinâmica; a possibilidade de acesso em tempo real se torna cada vez mais constante; novas tecnologias estão possibilitando a distribuição de informação em massa, criando salas virtuais com participação de até milhares de alunos; a possibilidade de desenvolver ferramentas de gestão adequada para o acompanhamento do

treinamento individual e a investigação de fatores que indicam o aprendizado contribui para a melhoria educacional e surge como um apoio possibilitando uma ação embasada em *feedback's*, ação esta que poderá ser executada de maneira mais rápida e objetiva.

## 1.6 Estrutura dos capítulos

O texto está dividido em cinco capítulos, que estão assim definidos:

O capítulo 1 apresenta o propósito da pesquisa e sua relevância, e as motivações do pesquisador para o desenvolvimento do tema de pesquisa.

O capítulo 2 apresenta a importância do aprendizado individual, mostrando seus benefícios quando controlado e administrado, identificando a importância da avaliação e treinamento, de forma a facilitar a identificação de perfil individual.

O capítulo 3 pesquisa a base estrutural de algumas das tecnologias que apoiam a identificação do perfil individual, ao qual irão subsidiar elementos para a construção da arquitetura proposta.

O capítulo 4 apresenta a arquitetura proposta, sua funcionalidade através de um modelo e as suas relações com os recursos tecnológicos abordados.

No capítulo 5 são apresentadas as considerações finais e sugestões para realizações de trabalhos futuros.

## **2 A NECESSIDADE DA IDENTIFICAÇÃO DO PERFIL DO APRENDIZADO INDIVIDUAL**

O aprendizado é um fenômeno que surge através do indivíduo em resposta aos seus esforços, cobrados pelo ambiente de aprendizado em que convive e isto ocorre em todos os indivíduos, o comportamento em contraste com o ambiente de ensino-aprendizagem pode ser diferente para cada indivíduo, quando as pessoas se comportam de maneira diferenciada, constroem seus conhecimentos de forma diversa e adversa, respondem de diferentes maneiras, reforçando a necessidade de se identificar o perfil do indivíduo, posição esta também defendida por Mattos (1998).

No modelo educacional tradicional Ribeiro (2000) enfatiza que a sua estrutura aborda a forma seqüencial (ano, série ou termo) e as ações instrucionais ocorrem, na maior parte das vezes, em sala de aula, no formato de aulas expositivas, complementadas com treinamento em laboratórios. O formato na exposição, transmissão de conteúdo e treinamento procuram maximizar a relação da quantidade de alunos por professor criando grupos (classes), expondo material de forma padronizada não levando em conta o perfil de cada aluno, executando avaliações em movimentos periódicos, tentando promover uma relação de ensino em massa. As instituições de ensino-aprendizagem defrontam-se com muitos problemas que não encontram soluções dentro do conceito da produção em alta escala (treinamento em massa e avaliação padronizada), pois as pessoas envolvidas possuem perfis diferentes e isto deve ser considerado. A importância de reconhecer a evolução do aprendizado, possibilita a identificação de comportamento deficitário em relação aos resultados desejados, que podem ser causados por outros fatores que não são originados dentro da instituição de ensino, porém sabe-se que vários problemas podem



originar-se dentro da própria sala de aula, problemas estes que podem ser causados pela baixa eficiência da metodologia de ensino do professor.

O professor necessita ter um *feedback* mais dinâmico de seu trabalho, e isto somente é possível se tiver controle de seus processos de ensino, uma abordagem mais estruturada para investigar porque alguns alunos não conseguem resolver exercícios propostos em treinamento. Alunos que respondem de forma deficitária em sua maior parte nas avaliações, geralmente mostram que o material instrucional ou treinamento, não foi apresentado com clareza ou o conceito não foi aprofundado suficientemente e denota necessidade de medidas corretivas. Uma das causas apontadas por Ribeiro (2000) é o fato dos métodos de ensino tradicional simplesmente jogarem quase sempre o conteúdo através de aulas expositivas, leitura de textos ou atividades em laboratórios, deixando ao aluno a tarefa de lembrar-se de tudo.

A dinâmica do professor de preparar um conjunto de avaliações de questões diversificadas em uma prova também não ocorre, pelo fato de que, o professor, na maior parte das vezes, não possui uma ferramenta que o auxilie na elaboração e preparação das avaliações. Sabe-se que quanto maior é a diversidade de itens pedido na prova, em formatos diferentes e aplicado de forma sincronizada com os temas abordados em sala de aula, maior será o mapeamento do aprendizado e identificação do perfil.

## **2.1 Identificação do perfil do aluno através do treinamento e avaliação**

A avaliação do conhecimento construído pelos alunos através de treinamento auxilia na identificação do perfil individual, principalmente quando esta avaliação é planejada de forma individualizada. O treinamento personalizado apresenta-se como importante ferramenta de apoio à formação do conhecimento, principalmente se o treinamento está relacionado com os objetivos de aprendizagem.

A abordagem da avaliação e treinamento justifica-se por fornecer subsídios necessários para a construção de objetos de aprendizagem (apresentados no tópico 3.3.4) que devem utilizar seus conceitos pedagógicos, e de como pode ser aplicada de forma correta e objetiva, tenta elucidar também a importância da análise dos resultados quando



executada na forma diagnóstica para identificar o perfil da classe. Em curto prazo, a avaliação ajuda na identificação de problemas e/ou deficiências, criando a oportunidade para revisões, pertinentes ao modelo de apresentação (ou explanação) de conteúdo abordado pelo professor, apóia no próprio processo de elaboração da metodologia de ensino, quanto mais rápido ocorrerem movimentos de avaliação, mais rápida poderá ser a tomada de ações corretivas caso haja necessidade.

Angelo (2001) afirma que a avaliação é entendida como um processo pedagógico que visa ao desenvolvimento da investigação da aprendizagem do aluno podendo ser tanto na forma individual como de forma coletiva. Ela tenta identificar o índice de aproveitamento dos alunos, ou seja, procura analisar de que maneira os alunos elaboram suas formas de pensar e relacionar os conteúdos que são objeto do processo de ensino-aprendizagem para que, de posse dessas informações, o professor continue a planejar, buscando as melhores maneiras de propor soluções para as dificuldades encontradas pelos alunos ou proponha novos desafios que respondam aos seus interesses e necessidades.

A questão da existência de avaliações procurando identificar o desempenho na forma quantitativa não desabona o método de se identificar o perfil individual de cada aluno, na verdade ela vem apoiar o levantamento de dados que venham a esclarecer a capacidade de resolução de problemas e, por conseqüência, identificar se ocorreu o aprendizado do tema avaliado. Se a avaliação tem como objetivo identificar o perfil individual, a mesma não pode ter um modelo padronizado em todos os tempos de sua aplicação, isto se torna mais complicado quando o período de intervalo entre as avaliações é demasiadamente grande o que dificulta o acompanhamento da evolução do aprendizado individual.

No processo de avaliação, normalmente as instituições de ensino-aprendizagem estabelecem regras de pontuação (média) mínima para serem considerados aptos para o próximo processo (próximo ano, termo ou série). Em uma escala de pontuação de 0 a 10, se a pontuação mínima é 7, o fato de que o aluno atingiu o mínimo, indica que ocorreu uma deficiência de 3 pontos, bastante próxima ao limite de rejeição. Quando tratamos individualmente o aluno, pode-se observar de maneira mais coerente as características da deficiência destes 3 pontos no conjunto, o que possibilita processos de correção, principalmente se esta deficiência for comum em uma grande parcela de alunos. Se a

estrutura da disciplina for bem decomposta (granulada) e a avaliação respeitar esta estrutura, pode-se identificar em qual parte ou partes resultaram maiores dificuldades do aprendizado e reforçá-los ou reformulá-los em ações ou planejamento corretivo.

Segundo Bloom, Hastings e Madaus (1983) a avaliação possui grande importância, descrevendo alguns de seus objetivos:

- a avaliação é um método de coleta e de processamento dos dados necessários à melhoria da aprendizagem e do ensino;
- a avaliação inclui uma grande variedade de dados, superior ao rotineiro exame escrito final;
- a avaliação auxilia no esclarecimento das metas e dos objetivos educacionais importantes e consiste num processo de determinação da medida em que o desenvolvimento do aluno está se processando da maneira desejada;
- a avaliação é um sistema de apoio pelo qual se pode determinar, a cada passo o processo ensino-aprendizagem, se este está sendo eficaz ou não; e caso não esteja, indica que mudanças devem ser feitas a fim de assegurar sua eficácia antes que seja tarde demais;
- finalmente, a avaliação é um instrumento na prática educacional que permite verificar se os procedimentos alternativos são igualmente eficazes na consecução de uma série de objetivos educacionais.

Outros autores reforçam a importância da tarefa da avaliação:

logo que os alunos chegam à escola, o professor deve avaliá-los. Só assim poderá adquirir informações diretas, imprescindíveis e valiosas para planejar o seu trabalho. O trabalho do professor será tanto mais eficiente quanto mais estiver calcado em fatos reais, em informações acumuladas sobre o aluno. O professor deve procurar conseguir estas informações através de todos os meios que estejam ao seu alcance: entrevista com os alunos, observação do comportamento, entrevistas com pessoas que conheçam o aluno, leitura de ficha do aluno, etc. (NETTO, 2001).

Na questão organizacional a avaliação é considerada como base de desempenho:

a avaliação é um instrumento de análise comparativa entre os comportamentos das pessoas frente a certas situações. Um sistema de avaliação de desempenho deve seguir uma periodicidade freqüente. A grande vantagem desta ferramenta gerencial é a possibilidade de a organização conhecer melhor seus colaboradores e o resultado de suas ações gerenciais aos olhos de funcionários e clientes,



principalmente quando estes últimos participam do processo de avaliação. (FANTON, 2002).

Dentro do campo educacional, a avaliação assume diferentes papéis. A classificação a seguir é definida por Bloom, Hastings e Madaus (1983) como:

**Formativa:** ocorre durante o processo de instrução; inclui todos os conteúdos importantes de uma etapa da instrução; fornece *feedback* ao aluno do que aprendeu e do que precisa aprender; fornece *feedback* ao professor, identificando as falhas dos alunos e quais os aspectos da instrução que devem ser modificados; busca o atendimento às diferenças individuais dos alunos e a prescrição de medidas alternativas de recuperação das falhas de aprendizagem.

**Somativa:** ocorre ao final da instrução com a finalidade de verificar o que o aluno efetivamente aprendeu; inclui conteúdos mais relevantes e os objetivos mais amplos do período de instrução; visa à atribuição de notas; fornece *feedback* ao aluno (informa-o quanto ao nível de aprendizagem alcançado), se este for o objetivo central da avaliação formativa; presta-se à comparação de resultados obtidos com diferentes alunos, métodos e materiais de ensino.

**Diagnóstica:** ocorre em dois momentos diferentes: antes e durante o processo de instrução; no primeiro momento, tem por funções: verificar se o aluno possui determinadas habilidades básicas, determinar que objetivos de um curso já foram dominados pelo aluno, agrupar alunos conforme suas características, encaminhar alunos a estratégias e programas alternativos de ensino; no segundo momento, buscar a identificação das causas não pedagógicas dos repetidos fracassos de aprendizagem, promovendo, inclusive quando necessário, o encaminhamento do aluno a outros especialistas (psicólogos, orientadores educacionais, entre outros).

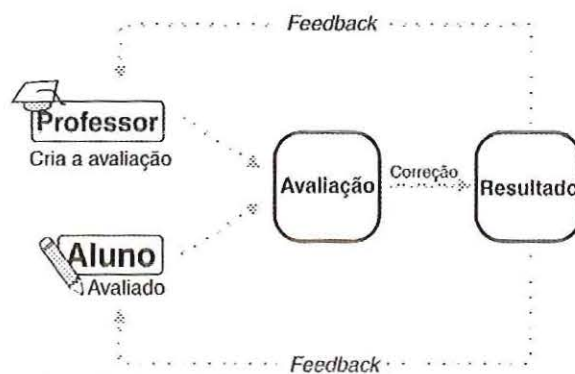
Em longo prazo, a avaliação formativa ajuda no fornecimento de dados para a verificação da resposta do aprendiz no ato da exposição instrucional, facilitando até decisões de ordem institucional, apresentando respostas da eficiência e eficácia do método aplicado pelo professor ou pelo sistema virtual quando aborda um tema ou assunto, ela fornece indícios se o conteúdo expositivo atingiu seus objetivos.

A avaliação somativa vem contribuir em dar retorno (*feedback*) ao aluno, identificando em que ponto ou pontos o mesmo deve melhorar seu desempenho após completar uma fase do período instrucional ou treinamento, seria interessante uma abordagem de conjunto com os temas abordados pelo professor.

A avaliação diagnóstica é bastante importante para identificar o nível de instrução do indivíduo ou do grupo para que o professor possa planejar um trabalho mais efetivo,

identificando suas características de aprendizagem, qual o nível da classe com relação ao conteúdo da matéria a ser abordada no ano ou semestre, onde, em um primeiro momento, o professor ao deparar com alunos em uma nova classe se ressentia da falta de informação diagnóstica, o que o leva a empregar seus próprios recursos.

No modelo tradicional, a avaliação quando é aplicada, necessita de um processo de preparo, aplicação e correção, dependendo da dinâmica do professor, este processo pode ser rápido ou lento, ou seja, o *feedback* dependerá do próprio tempo da correção executada pelo professor, e o aluno receberá somente após a divulgação de seu resultado, o que, a priori não pode acontecer ao mesmo tempo.



**Figura 1: Modelo didático do sistema de avaliação.**

**Fonte: Adaptado de Bettio (2003)**

Os modelos de avaliação aplicados a ambientes virtuais de aprendizagem possibilitam retorno instantâneo ao aluno, segundo Croft<sup>3</sup> (2001 apud BETTIO, 2003), a diversidade dos temas abordados pelo professor nas avaliações aplicadas em ambientes virtuais de aprendizagem tende a crescer não só pela necessidade dos professores, mas também pela exigência dos próprios alunos, que tendem a se sentir mais confortáveis e confiantes com os *feedback's* mais dinâmicos e individuais respeitando a identificação de seu perfil.

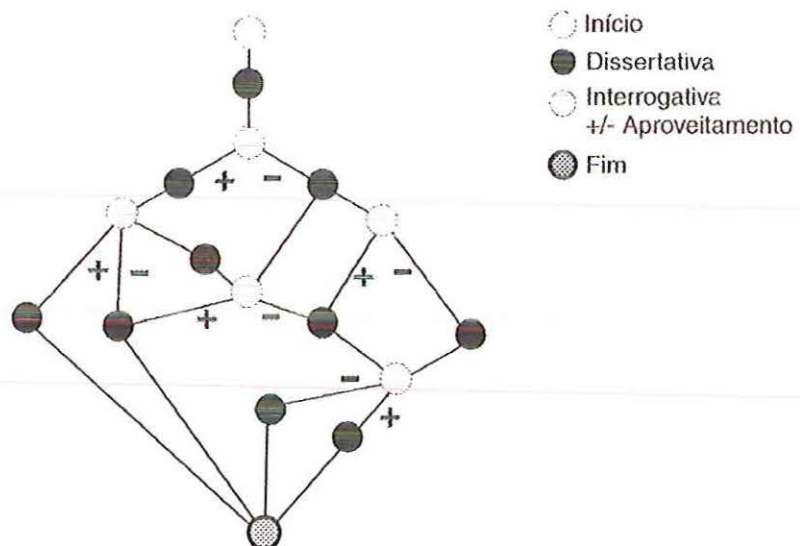
<sup>3</sup> CROFT, A.C. (2001). *Experiences of using computer assisted in engineering mathematics*. Loughborough University. Loughborough apud BETTIO, R.W. (2003). *Avaliações gráficas e dinâmicas aplicadas a ambientes virtuais de aprendizagem*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2003.



A avaliação se torna mais abrangente e importante através de novas tecnologias, pelo fato de que a avaliação poderá ser aplicada em mais vezes que os modelos tradicionais, possibilitando uma variação na composição dos dados (questões e alternativas) oferecidos para o aluno responder.

A estrutura abordada por esta pesquisa apresenta um modelo (Figura 24) que permite que as avaliações virtuais possam realizar interação com os temas abordados em sala de aula e, como consequência, a identificação do comportamento de um maior número de alunos com relação ao aprendizado e sucesso da aula presencial de forma expositiva ou exercícios de treinamento em laboratório e que oferece ao aluno uma avaliação de forma variada, o que pode se tornar uma prática significativa ao processo ensino-aprendizagem. Neste modelo será possível oferecer avaliações interativas, que se moldam ao perfil do aluno, conforme a capacidade de aprendizado individual. Esta personalização e adequação são obtidas através de um trabalho conjunto entre objetos de aprendizagem, agentes inteligentes e sistemas tutores apresentados no capítulo 3.

Um pequeno exemplo desta dinâmica de interação pode ser representado na figura a seguir:



**Figura 2: Modelo gráfico de sistema de avaliação virtual**

**Fonte: Adaptado de Bettio (2003)**

As ações dissertativas têm por intuito promover uma discussão que faz parte da avaliação, disponibilizando itens complementares como imagens ou vídeos (apresentados através do objeto de aprendizagem) e que ajudarão a promover o aprendizado. As ações

interrogativas têm como objetivo testar o aprendizado (formativa), podendo assim, direcionar o aluno para outros níveis da avaliação presentes no objeto de aprendizagem. Caso o aluno tenha atingido o objetivo desejado pelo objeto de aprendizagem, agentes de avaliação (figura 32) poderão aumentar o nível de dificuldade, e os agentes tutores (figura 29) poderão direcionar o aluno para o aprendizado de um novo conteúdo respeitando o perfil do mesmo (agentes tutores apresentam objetos de aprendizagem conforme o perfil do aluno). Caso o aluno tenha dificuldade, a estrutura apresentada procurará repassar os conhecimentos de uma maneira adequada, identificando quais são os objetivos de treinamento e o tipo de apresentação de conteúdo instrucional (objetos de aprendizagem) que obtiveram melhor desempenho. Assim, cada aluno terá a oportunidade de aprender a seu modo e a sua velocidade, identificando seu perfil individual nas interações realizadas.

O controle e análise da avaliação se torna mais necessário nas aulas não presenciais, principalmente quando o ambiente presencial se mistura ao não presencial, caracterizando modelos de ensino-aprendizagem semipresencial, que é composto de aulas presenciais e aulas à distância virtuais intercaladas (KEMCZINSKI, 2000).

O movimento do crescimento de intercalar aulas presenciais ao não presenciais, tende a ser crescente, o reconhecimento desta tendência deve-se ao fato de que a Portaria/MEC no. 2.253 de 18/10/2001 (DOU de 19/10/2001) autoriza a inclusão de disciplinas não presenciais em cursos superiores reconhecidos pelo Ministério da Educação, em seu artigo 1º enuncia que:

As instituições de ensino superior do sistema federal de ensino poderão introduzir, na organização curricular de seus cursos superiores reconhecidos, a oferta de disciplinas que, em todo ou em parte, utilizem método não-presencial, com base no artigo 81 da Lei 9394/96.

O artigo indica que “poderão introduzir”, o que desvincula uma obrigatoriedade, mas não se questiona que é um estímulo às tecnologias de ensino à distância.

O legislador no intuito de promover a utilização de tecnologia para o ensino pode ser mal interpretado com relação à utilização da mesma, onde uma vez que a escola poderá criar movimentos da substituição total de uma disciplina por um sistema computacional com mero objetivo econômico (na tentativa de reduzir o custo operacional) e não o de garantir a excelência do ensino. Fica por parte do MEC estabelecer, através de seu



programa de avaliação, identificar se a utilização de recursos tecnológicos possui somente objetivos financeiros e não o de aprendizagem.

A portaria afirma ainda que a carga horária a ser trabalhada deve se limitar em 20% da carga horária total do curso, e que, os exames finais serão sempre presenciais. Reforça que as disciplinas previstas para serem trabalhadas nos moldes propostos pelo Artigo 1º “deverão incluir métodos e práticas de aprendizagem que incorporem o uso integrado de tecnologias de informação e comunicação para a realização dos objetivos pedagógicos”.

Através da mescla de processos de ensino presencial e não presencial, gera maior necessidade de identificar o perfil do aluno que participa desta tecnologia, pois em aulas virtuais a relação face a face entre professor e aluno pode não ocorrer.

O modelo de arquitetura proposto por esta pesquisa tenta resgatar dados e informações através da exposição, treinamento e avaliações de aprendizado de forma individual e inteligente, e que possam levantar, também, indícios de falhas de processo das aulas presenciais, semipresenciais e virtuais, isto é possível através da criação de uma base de informações que poderá conter não só as informações de utilização dos recursos de ensino à distância, como também um conjunto de dados abordando as interações e suas respostas com as tecnologias acessadas, onde os dados poderão ser analisados e processados pela própria estrutura como por outros sistemas de informação presentes na instituição de ensino.

## **2.2 Identificação do perfil do aluno através da tecnologia**

Segundo Belloni (1999) o ensino à distância deve se preocupar com um elemento importante: o perfil do aluno à distância, uma vez que, a instituição de ensino deve possuir uma estrutura que promova o acompanhamento dos alunos, caso isto não ocorra provavelmente os mesmos se sintam desmotivados e abandonados, não executando o movimento de aprendizagem principalmente aquelas que serão obrigatórias quando estiverem inseridas no regimento oficial do curso.

A motivação de um ambiente educacional é fator determinante para o sucesso do aluno; ele deve perceber que suas necessidades devem ser atendidas, e nesse sentido, Belloni (1999) afirma que os sistemas educacionais terão que enfrentar as novas demandas



decorrentes, portanto será essencial conhecer os objetivos e necessidades dos alunos e conceber cursos, estratégias e metodologias que as integrem efetivamente. Isto requer que o sistema à distância reconheça seus objetivos formais e informais, presentes na estrutura proposta (Figura 24) e que ofereçam recursos para atingir estes objetivos.

Com o crescente uso da integração da tecnologia no processo educacional, as metodologias aplicadas em sala de aula podem ser complementadas com esse recurso, e sua negligência pode tornar-se um modelo pedagógico restrito, apesar de não anular a aula ou o professor. Os alunos precisam de contato humano para aprender. Segundo Ribeiro (2000), cabe usar o momento da aula para discussões e críticas, ao invés de exposições de conteúdos, que pode ser feito através de meios tecnológicos como vídeo-aula, internet, entre outros recursos tecnológicos de multimídia. A transmissão da informação como parte do ensino-aprendizagem não deve somente acontecer de forma rotineira e cansativa que privilegia a memorização e não desafia a criatividade do aluno.

Tecnologias de transmissão de conteúdo em massa como a vídeo-conferência e transmissão de aulas por satélite permitem uma abrangência cada vez maior de indivíduos, em que o aluno poderá ter uma interação empobrecida devido ao elevado número de alunos distribuídos em várias salas de aula concomitantemente com o professor.

Oferecendo um sistema de treinamento interativo mediado através do computador e relacionado com o material instrucional exposto pela tecnologia de ensino de massa, possibilita que dados sejam colhidos e processados por sistemas de informações, fornecendo dados e informações aos professores e coordenadores para a adaptação de metodologias ou a oferta de novos recursos de treinamento que levam em consideração a heterogeneidade dos alunos, que progridem em ritmos diferentes de acordo com suas habilidades naturais e, conseqüentemente, até entender problemas de formação em localidades regionais. As tecnologias abordadas e integradas à estrutura proposta permitem a recuperação de dados quando ocorre a interação, se comportando de forma mais dinâmica ao perfil individual fomentando subsídios para atender a esta demanda.

### 3 AMBIENTE VIRTUAL DE ENSINO

Teixeira (2000), enfatiza que as organizações do aprendizado são, por sua natureza, um eterno vir a ser, o processo é contínuo e nunca chegará a um estágio final. A utilização e revitalização de recursos tecnológicos deverão ser um processo constante.

A educação à distância deverá ser utilizada de forma mais comum, como uma ferramenta de apoio, apresentando técnicas inovadoras como autonomia e processos mediados pela comunicação e interação síncrona (como exemplos serviço de conversa eletrônica e sistema de vídeo-conferência, entre outros) e assíncrona (como exemplo correio eletrônico, grupos de notícia, sistema de apoio à decisão, páginas pessoais na internet, entre outros). A interação síncrona e assíncrona possui algumas características que devem ser investigadas que justifiquem componentes (agentes e serviços) presentes na estrutura do modelo proposto.

Com relação à comunicação e exposição síncrona, Ellis (2004) destaca as seguintes vantagens:

- motivação. A interação em tempo real foca na energia do grupo, provendo condições para que os alunos distantes estejam motivados para continuar seus estudos;
- telepresença. A interação em tempo real, com sua oportunidade de transmitir tons e nuances, ajuda a desenvolver a coesão no grupo e o senso de pertencer a uma comunidade de aprendizagem;
- bom *feedback*. Os sistemas síncronos provêm *feedback* rápido nas idéias e apóiam o consenso e a tomada de decisões nas atividades em grupo, que animam a educação à distância;
- velocidade compassada (*pacing*). Eventos síncronos encorajam alunos a se manterem atualizados com o curso e provêm um comportamento disciplinado para aprender, ajudando as pessoas a priorizar seus estudos.



Ellis (2004) lista também algumas vantagens do uso da distribuição assíncrona, no contexto educacional:

- flexibilidade. O acesso ao material de ensino pode ser acessado a qualquer hora e a partir de qualquer lugar;
- tempo para refletir. Permite ao aluno refletir sobre a questão, revisar as referências bibliográficas, referir-se a mensagens anteriores e elaborar um comentário;
- aprendizagem situada. Por poder acessar a partir do trabalho ou da casa, o aluno pode integrar facilmente as idéias discutidas pelo curso com o ambiente de trabalho, ou acessar recursos na Internet, conforme requerido no trabalho;
- tecnologia de custo efetivo. Os sistemas assíncronos, baseados em texto, requerem pouca largura de banda e computadores mais lentos para operar, tornando o acesso global mais equalizado.

As distribuições síncronas e assíncronas também se complementam em relação uma à outra, os alunos podem trocar informações entre si e com o professor (de forma síncrona) na sala de aula presencial ou virtual (vídeo conferência) e após o termino, a comunicação pode ser realizada de forma assíncrona (e-mail, troca de artigos, etc.), permitindo que os alunos e professores possam realizar reflexões e comentários e vice-versa.

Durante a interação com sistemas de computadores, as pessoas podem estar presentes (conectadas) através da Internet e em momento posterior podem não estar, o modelo apresentado (figura 24) é reativo e dependendo do movimento dos agentes, eles podem executar serviços de comunicação através de serviços de alertas, caso necessitem, mas os atores (coordenadores, professores e alunos) podem estar desconectados, nesta situação os serviços (*Services*) apresentados no modelo serão responsáveis e encaminham em momento oportuno quando os atores se conectarem novamente, isto é possível pelo fato de que serviços de comunicação e serviços de alerta podem permanecer em um estado de enfileiramento (*scheduling*) armazenado, aguardando o momento correto de entrega da mensagem ou informação, isto é o que acontece normalmente com os *email's* em sua forma convencional de operacionalidade.

Com a difusão dos computadores pessoais (*Personal Computers's* – PC's), duas novas tecnologias começaram a viabilizar produtos virtuais no ensino à distância, as redes de computadores globais e a multimídia (ANGELO, 2001).

Através da utilização de tecnologia no ambiente de aprendizagem, os computadores comparados aos demais recursos tecnológicos na área educacional (como vídeos e



transmissão por satélite), caracterizam-se pela interatividade e possibilidade de ser instrumentos que podem ser utilizados para facilitar a aprendizagem individualizada. Também o fato de o sistema computacional poder identificar o perfil individual, o torna uma ferramenta interessante para a motivação do ensino à distância.

### **3.1 A Internet no ambiente virtual de ensino**

A Internet, cuja interconectividade e abrangência são fatores contundentes, impulsionaram vários outros segmentos que auxiliam o ensino-aprendizagem. Freitas (1999) ressalta que a Internet, assim como a videoconferência, a teleconferência e outras mídias, como suporte para um ambiente de aprendizagem, não são um novo método de ensino, mas constitui-se num novo meio de construir soluções técnicas para muitas instituições de ensino. A possibilidade de o aluno interagir com sistemas através da internet, utilizando componentes virtuais de ensino (Agentes inteligentes, Sistemas Tutores Inteligentes e objetos de aprendizagem) amplia as dimensões do uso da tecnologia. Para entender a estrutura proposta é importante investigar os recursos usados, onde estes recursos possuem uma interação no ambiente de ensino-aprendizagem virtual.

A flexibilidade de acesso à informação através da Internet e a interação com sistemas das instituições de ensino-aprendizagem, propiciaram um ambiente bastante atrativo para que pudessem ser construídos novas tecnologias e modelos de interação de ensino-aprendizagem. Souza (2001) avalia em seu trabalho porque a Internet se tornou tão atrativa no processo educacional:

- a Internet é a maior e mais poderosa rede de computadores do mundo e vem se tornando, gradativamente, um meio comum de troca de informações, acesso a especialistas, formação de equipes de trabalho cooperativo e acesso a arquivos e repositórios remotos de Informação que ultrapassam as barreiras geográficas convencionais;
- a Internet rompe também barreiras de tempo, permitindo o compartilhamento de informações apoiando a cooperação e a comunicação em tempo real;
- o poder de sedução da Internet pode alterar a educação formal, uma vez que formas efetivamente inovadoras de educação utilizando os serviços da Internet, podem ser pensadas e postas em práticas, mudando as velhas concepções pedagógicas;

- a Internet é uma mídia que facilita a motivação dos alunos pelas possibilidades inesgotáveis de pesquisa e novidades que oferece. Para os alunos que estão motivados a adquirir conhecimento e qualificações, há abundância de materiais apropriados para aprender, o que amplia inclusive as chances de êxito na auto-aprendizagem.

Segundo Costa (1999), o desenvolvimento das novas tecnologias tem promovido no panorama mundial, um significativo incremento do ensino à distância, quer em termos do número de alunos envolvidos, quer em termos do número de universidades que passaram a incluir essa modalidade de ensino na estrutura curricular, no Brasil a regulamentação por lei foi abordada no capítulo anterior, como a estrutura proposta por esta pesquisa utiliza tecnologia aplicada no ensino à distância e treinamento à distância (*e-learning*), faz-se necessário entender como estas surgiram e se complementam.

### **3.1.1 Ensino à distância**

Geber (1991) afirma que o termo educação à distância, apareceu primeiramente impresso em 1892 no catálogo de cursos da Universidade de Wisconsin – Madson. A principal característica que distingue a ensino à distância do ensino presencial é a separação física entre o professor e o aluno. O contato entre esses dois personagens é mediado através de algum recurso pelo qual o professor, mesmo ausente, se faz presente através de um canal de comunicação.

O surgimento do ensino à distância no Brasil, não aconteceu apenas após o aparecimento da televisão, computador ou Internet, seus primeiros passos começaram logo no início do século passado, segundo Oliveira (2003) o ensino à distância iniciou seus primeiros pulsos em 1904 através de esforços de cursos realizados pelo serviço de correio postal (por correspondência). Em 1923, com o surgimento do rádio em ampla difusão, foi utilizado como o principal veículo de comunicação do primeiro programa educacional transmitido pela Rádio Roquete Pinto (o primeiro programa educacional de massa da América do Sul). Em 1936 surgiu o Instituto Rádio Técnico Monitor, com cursos focados para a área de eletrônica. Em 1941 surgiu o Instituto Universal Brasileiro que explorou os cursos por correspondência e até hoje promove cursos com modelos conjugados com outras tecnologias entre elas, fita e vídeo-cassete. Em 1980 a atenção focalizou-se no uso de redes de computador como ferramentas para incrementar o processo de ensino/aprendizagem, e



com a introdução da Internet uma década mais tarde muitas instituições encontraram a motivação que ainda estava faltando, a inexistência da distância do mundo digital e a integração e troca de informações através de recursos de multimídia.

A distância não pode refletir perda na eficiência do aprendizado, Casas (1999) enfatiza que novas soluções devem garantir esta necessidade.

O surgimento de novas tecnologias de treinamento e educação, somadas aos requisitos de aprendizagem exigidos pela dinâmica da sociedade atual, faz do ensino à distância uma real necessidade. O ponto principal é a possibilidade de expandir a sala de aula para um universo maior, permitindo educar pessoas em qualquer lugar e a qualquer hora. Assim sendo, é clara a necessidade de novas soluções que permitam realmente um ensino à distância de qualidade.

Santos<sup>4</sup> (1996 apud RIBEIRO 2000) em seu trabalho enfatiza que o ensino à distância possui qualidades que ampliam a possibilidade do acesso ao conhecimento, essa modalidade de ensino não tem o propósito de substituir o ensino presencial, mas sim, apoiá-la em vários momentos. Dentre as vantagens do ensino destaca:

- pode alcançar um grande número de pessoas;
- adapta-se ao ritmo de aprendizado de cada um;
- quando utilizada em grupo possibilita o exercício de aprender a ouvir, a discutir, a trabalhar em equipe;
- se no início o investimento é muito elevado, posteriormente conforme o aumento de pessoas participantes, o custo tende a baixar;
- favorece o uso de recursos de tecnologia como o rádio, televisão e computadores;
- desenvolve o comportamento autodidático, independência e autonomia.

Soares et al (2004) ainda destaca que, para promover o processo de ensino e aprendizagem à distância de maneira eficaz, obtendo bons resultados e vantagens significativas no uso de novas tecnologias, é necessário entender as características

---

<sup>4</sup> SANTOS, T.L. (1996). A reeducação e a escola do futuro. In: CONGRESSO BRASILEIRO DE ENSINO E ENGENHARIA, 24., 1996, Manaus. *Anais*. Manaus: UA/ABENGE. v.2, p.49-67 apud RIBEIRO, L.R.C. (2000). *Programas da qualidade total e educação: reflexões sobre a utilização de seus princípios no ensino de engenharia*. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2000.



intrínsecas às tecnologias disponíveis e por elas combinadas. O professor deve associar-se às novas tecnologias, empregando-as em seu auxílio, quebrando barreiras e ao extirpar as dificuldades em sua utilização, faz com que aumente as chances da melhoria de sua estratégia pedagógica, facilitando a exploração do cenário abordado pela interação e o entendimento da necessidade do aprendizado com foco no perfil individual.

Algumas instituições têm desenvolvido e estão promovendo iniciativas excelentes e merecem ser destacadas, pois suas experiências indicam a utilização de tecnologias virtuais e reforçam a utilidade das tecnologias na área de ensino, entre elas estão a Escola do Futuro da Universidade de São Paulo e o Laboratório de Educação à Distância (LED) da Universidade Federal de Santa Catarina (UFSC).

- Escola do Futuro da USP: desenvolve pesquisa sobre novas tecnologias de comunicação aplicadas à educação, usando e analisando ferramentas que utilizam recursos de Internet, buscando desenvolver novas metodologias e recursos pedagógicos atrelados à tecnologia. Tem como objetivo investigar de como as novas tecnologias de comunicação podem melhorar o aprendizado em todos os seus níveis. Tem como missão, melhorar a qualidade da educação elementar usando a tecnologia (USP, 2003).
- LED – Laboratório de Educação à Distância da Universidade Federal de Santa Catarina: O Programa de Pós-Graduação em Engenharia de Produção, ao qual o Laboratório de Ensino à Distância (LED) está vinculado, adota o sistema de ensino presencial e virtual. Este modelo se baseia na utilização de tecnologias da informação e comunicação, se tornando um referencial pedagógico e no desenvolvimento de pesquisas científicas aplicadas (LED, 2003). Ao mesmo tempo, cumpre aspectos formais, acadêmicos e regulamentares da UFSC e da Legislação Brasileira para cursos presenciais de mestrado, o que garante a certificação através da própria universidade.

### **3.1.2 Treinamento à distância (*E-Learning*)**

Com o surgimento dos computadores e a sua interligação através de uma rede de computadores surge o conceito de aprendizado através do computador, o *e-learning* (*Electronic Learning*). Rossett (2001) afirma que o *e-Learning* é um treinamento que

acontece através de uma rede de computadores, usualmente via Internet ou em uma Intranet<sup>5</sup> de uma instituição, surgiu em meados dos anos de 1980 com o surgimento das tecnologias de microcomputadores e leitores ópticos (CD-ROM) para ensinar os profissionais técnicos. Atualmente, esse tipo de ensino é adotado nas empresas e em instituições acadêmicas.

Segundo Lemos (2003) o *e-learning* é um complemento da educação à distância (EAD) apoiada na estrutura da internet, sua definição de *e-learning* é a seguinte:

O *e-learning* é uma instância da EAD apoiada nas tecnologias da internet. A grande questão é que sua expansão tem se dado não somente nos ambientes acadêmicos e nas estruturas tradicionais de ensino, mas também no âmbito das empresas, que passaram a vislumbrar mais uma possibilidade para capacitação e aperfeiçoamento de seu pessoal. De uma forma mais abrangente, dentro de um contexto de negócios de alta competitividade e em constante mudança, o *e-learning* representa, mais do que nunca, uma possibilidade de educação contínua para as pessoas individualmente e para a empresa como um todo.

Além de possuir uma relação de custo operacional muito menor ao presencial, o *e-learning*, possibilita a integração de vários conceitos abordados em várias disciplinas, promovendo uma integração maior entre elas. O conteúdo de um ensino embasado na tecnologia do *e-learning* deve ser muito bem elaborado e avaliado, pois a possibilidade de apenas oferecer material instrucional não significa a garantia do aprendizado. Andrade e Brasileiro (2003) reforçam que:

A elaboração e o desenvolvimento de um projeto de *e-learning* deve seguir estratégias e metodologias capazes de criar marcos passíveis de avaliação. Dessa forma, é possível ter controle dos resultados obtidos e assim tornar mais administráveis esses projetos. Apesar de fortemente ligados à área de tecnologia de informação, projetos de *e-learning* são multidisciplinares, pois abordam assuntos como: desenvolvimento de sistemas, redes de computadores, banco de dados, segurança de dados, padronização, recursos humanos, pedagogia, produção de conteúdo educacional, comércio eletrônico, multimídia, direitos autorais, administração de empresas, análise financeira etc.

Algumas ferramentas abordam os temas de ensino e treinamento à distância utilizando tecnologia da Internet, controlando também a evolução dos cursos neles implantados. Estes sistemas de gerenciamento de aprendizado os *LMS* (*Learning*

---

<sup>5</sup> Intranet: Rede interna de computadores de uma empresa ou escola que utiliza os recursos tecnológicos da Internet.



*Management Systems*) e treinamento *WBT (Web-based Training)* apresentam características interessantes de funcionamento que colaboraram na construção da arquitetura proposta, o que é investigado no próximo tópico.

### **3.2 Evolução das tecnologias virtuais de treinamento na educação**

Vários conceitos de tecnologias estão surgindo desde o século passado, cada uma procurando ser mais autônoma e sofisticada que outra, esta evolução acompanha os recursos computacionais e a capacidade de trabalhar cada vez mais com estruturas complexas. Segundo Prado (2002) os primeiros sistemas voltados para o controle do aprendizado através do computador, surgidos por volta de 1950, são os chamados programas lineares também conhecidos como Sistemas de Treinamento Baseado em Computador (*Computer-Based Training – CBT*) e Sistemas de Instrução Auxiliada por Computador (*Computer-Aided Instruction – CAI*), estes sistemas caracterizavam-se por mostrar o conhecimento de uma forma rígida e sentido único, onde nenhuma ação podia mudar a ordem de ensino estabelecida em sua programação. Ele não se adaptava às necessidades do aluno e apenas o pontuava ainda quando era avaliado. A proposta dos *CBT*'s e *CAI*'s é apresentar um problema ao aluno, registrar sua resposta e avaliar seu desempenho. Não havia praticamente uma relação bidirecional, a interação era transmitida de uma forma passiva, onde o sistema controla as ações de interação.

A partir dos anos 60 do século XX, os sistemas computacionais começaram a considerar que as respostas dos alunos podiam ser usadas para ajudar a controlar o material de estudo; no início da década de 1980, nascem os Sistemas Tutores Inteligentes (STI), como iniciativa no intento de tratar as falhas dos sistemas baseados em *CAI*. Esta iniciativa foi beneficiada pelo trabalho de pesquisas que utilizavam recursos de Inteligência Artificial (IA), que tinham uma permanente preocupação com o problema da melhor forma de como representar conhecimento dentro de um sistema inteligente. Nesta década, começaram a surgir pesquisas na área de IA através da criação dos *ICAI (Intelligent Computer Aided Instruction* ou Instruções Inteligentes Assistidas por Computador). Estes primeiros sistemas não realizavam interação com o modelo de conteúdo formal da instituição de ensino aprendizagem e também não identificava o perfil individual do aluno de forma integrada ao sistema da instituição de ensino.



Apenas o registro das interações não é suficiente para prover suporte efetivo à avaliação formativa, ocorre a necessidade do entendimento do desempenho destas interações para subsidiar uma avaliação focada ao perfil individual, mas a demanda pode consumir muito trabalho e tempo do professor, dependendo da quantidade de alunos a tarefa pode se tornar impossível de ser realizada de forma manual, o que consiste num dos principais problemas da avaliação aplicada no formato presencial ou à distância independentemente de seu objetivo (formativa, somativa ou diagnóstica).

Segundo Lachi (2002), novas tecnologias computacionais vêm sendo pesquisadas, a fim de explorar de maneira mais inteligente os registros das interações dos alunos em ambientes de ensino à distância e prover suporte para o professor na coleta, identificação, seleção e análise de dados relevantes às avaliações realizadas.

Beck, Stern e Haugsjaa (2003) enfatizam que os sistemas de instrução auxiliada por computador podem ser pouco eficazes no processo de ensino-aprendizagem, pois uma vez que eles não identificam um conhecimento individualizado de cada aluno, também não podem se comportar em uma interação individualizada como receberia de um tutor humano. Para um sistema de ensino-aprendizagem baseado em computador fornecer tal atenção, tem que, conhecer o aluno além de raciocinar sobre o domínio a ser ensinado.

Com o objetivo de criar um ambiente onde se possa maximizar a capacidade de aprendizado do aluno, aplicando conceitos de treinamento através do uso de componentes interativos foram criados sistemas conhecidos como *ILE (Intelligent Learning Environment* ou *Interactive Learning Environment*). Os princípios associados a este ambiente segundo Costa (1999) são:

- Construção e não instrução: o objetivo é explorar o fato de que alunos podem aprender mais efetivamente através de construção de seu próprio conhecimento;
- O controle é do aluno e não do tutor: a questão é dar mais liberdade para o aluno controlar suas interações no processo de aprendizado. O tutor deve atuar como guia, e não como único detentor do conhecimento;
- A individualização é determinada pelo aluno e não pelo tutor: a personalização da informação é o resultado da interação com o ambiente. Esta responsabilidade pode estar associada ao sistema, mas o aluno ainda terá uma boa parte do controle de sua individualização;

- O conhecimento adquirido pelo aluno é resultado de suas interações com o sistema e não com o tutor: a informação adquirida vem como uma função das escolhas e ações do aluno no ambiente de ensino e não como um discurso gerado pelo tutor.

Estes princípios apresentados permitem aos alunos investigar e aprender de forma livre, sem estar presente algum tipo de controle, Spector (2001) reforça a utilização de interatividade, que deve ser considerado um estímulo ao aprendizado, a interatividade apresenta segundo ele, algumas vantagens:

- Pessoas aprendem mais com o que fazem do que com o que visualizam;
- Ambientes de aprendizagem promovem muitas chances do aprendiz participar e evoluir;
- Oportunidades para reflexão geralmente promovem o aprendizado. Este tipo de reflexão é exigido dos aprendizes em ambientes que utilizam a interatividade;
- Retornos informativos são necessários. Aprendizes avançados são capazes de gerar seus próprios retornos, mas os aprendizes novatos ainda não possuem tal capacidade.

Sistemas tecnológicos contribuem também no ensino colaborativo, entre eles o Sistema de Ensino Colaborativo Apoiado por Computador (*Computer-Supported Collaborative Learning – CSCL*), que leva em consideração a possibilidade de vários alunos trabalhar de forma cooperativa em um mesmo ambiente, onde a mediação a partir de atividades interativas, o conhecimento do aluno também pode ser desenvolvido na relação de interação com outros alunos, em ambientes colaborativos os níveis de habilidades podem não ser iguais entre os alunos, onde os mesmos podem gerar novos conhecimentos entre si. Através da Internet os ambientes colaborativos promovem-se se tornando instrumentos poderosos de inteligência coletiva, segundo Lévy (2001) o espaço virtual não deve ser comparado diretamente ao presencial; o espaço virtual apresenta-se mais colaborativo. A aprendizagem colaborativa é superior à aprendizagem individual, a aprendizagem pode ser ampliada através de discussões com todos aqueles que têm diferentes opiniões, habilidades ou saibam mais sobre algum tópico do tema que está sendo investigado.

Wickert (2003) defende a importância da interatividade da aprendizagem colaborativa na educação à distância, reforçando que o futuro da educação à distância não se fundamentará no estudo solitário, em que o indivíduo conte somente com o material



educativo para desenvolver a sua aprendizagem. E, sim, em ambientes em que a autonomia na condução do seu processo educativo, conviva com a interatividade.

Com o objetivo de aproveitar a potencialidade da internet e o desenvolvimento do ensino à distancia foi desenvolvido um novo modelo de ferramenta de ensino-aprendizagem, conhecido como *Web-Tutores*. Giraffa<sup>6</sup> (2001 apud PRADO, 2002) descreve que o *Web-Tutores* permite ao aluno trocar experiências com seus pares e aceder diferentes fontes de informação em diferentes formatos, nas quais o aluno manipula o conhecimento no seu tempo e necessidade. Prado (2002) realizou um processo analógico identificando a evolução da tecnologia computacional e a evolução da pedagogia educacional ao usarem estes recursos, o que é apresentado no próximo quadro:

<b>Tecnologia</b>	<b>Técnicas Utilizadas</b>	<b>Características</b>	<b>Interação com o aluno</b>
<b>CBT CAI</b>	Programação Procedural	Perguntas e respostas dirigidas por acertos e erros	<ul style="list-style-type: none"> <li>• Aluno tem papel passivo diante do sistema</li> <li>• O Sistema não considera as características individuais de aprendizado de cada aluno</li> </ul>
<b>ICAI STI</b>	Técnicas de Inteligência Artificial	Utilização de módulos separados para representação do conhecimento, das estratégias de ensino e o modelo do aluno.	<ul style="list-style-type: none"> <li>• O Controle sobre o aprendizado pode ser alternado entre o aluno e o sistema</li> <li>• Considera as características de cada aluno para fornecer ensino individualizado</li> </ul>
<b>ILE</b>	Inteligência Artificial e Simulação	Uso de módulos e ênfase na interação com o aluno	<ul style="list-style-type: none"> <li>• O aluno tem papel ativo no processo de aprendizagem</li> </ul>
<b>CSCL</b>	Inteligência Artificial e Redes	Uso de módulos e ênfase na colaboração entre vários elementos participantes do processo de aprendizagem.	<ul style="list-style-type: none"> <li>• Existe a colaboração e interação aluno-aluno e aluno-tutor.</li> <li>• Considera o aspecto social da aprendizagem</li> </ul>
<b>Web-Tutor</b>	Inteligência Artificial, redes e Hipermídia	Possui características dos sistemas anteriores, aproveitando as vantagens oferecidos pela internet	<ul style="list-style-type: none"> <li>• Considera grande interação com o aluno</li> <li>• Aprendizagem proporcionada através do ensino à distância</li> </ul>

**Quadro 1 – Comparação análoga da evolução da tecnologia de ensino-aprendizagem e a interação com o aluno.**

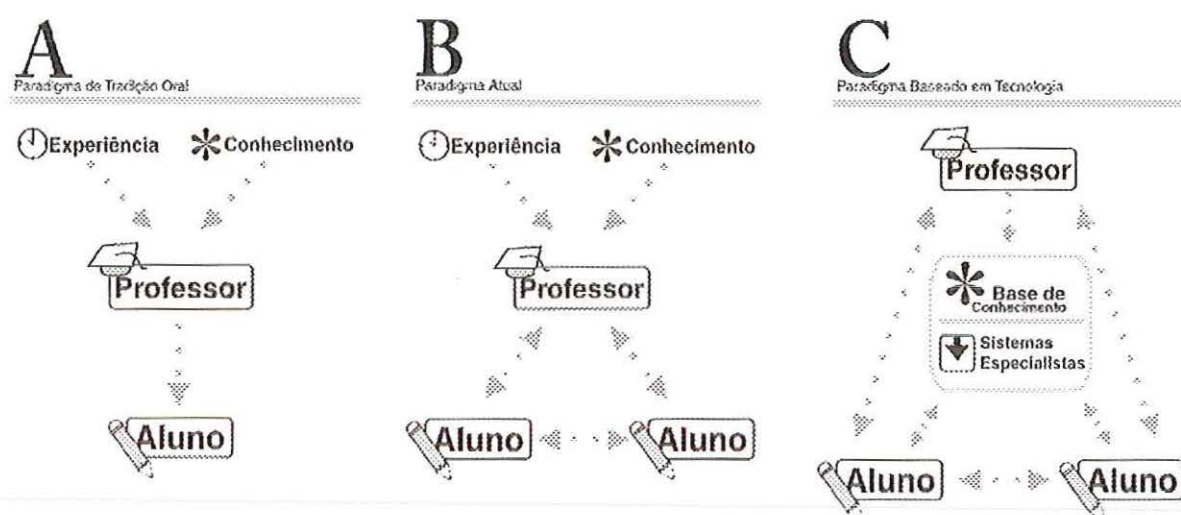
**Fonte: Prado (2002)**

<sup>6</sup> GIRAFFA, L. M. M. (2001). *A classificação dos softwares educacionais: Nós necessitamos mesmo de uma taxonomia?* Disponível em <<http://www.inf.purs.br/~giraffa>>. Acesso em 5 ago. 2001 apud PRADO, N.R.S.A. (2002). *Uma arquitetura para ambientes de ensino-aprendizagem utilizando o sistema de modelos tutores inteligentes e agentes*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.



Através da tecnologia hipermídia, desenvolvedores de ferramentas computacionais na área de ensino, estão oferecendo um conjunto de interface onde o aluno pode decidir o que e onde deve navegar conforme seus interesses pessoais, tornando o aluno um sujeito ativo e não passivo como nos primeiros sistemas. Estes desenvolvedores são conhecidos como designer do conhecimento, bastante utilizados na construção de objetos (programas interativos) que promovem a aprendizagem.

Acompanhando a evolução das relações entre professores, alunos e o conhecimento a ser ensinado através do *e-learning*, percebe-se uma tendência da alteração do modelo sistêmico tradicional de ensino, segundo Branson<sup>6</sup> (1990 apud CANDAU, 1999) a adoção de tecnologia propõe uma mudança de paradigma apresentado na figura a seguir:



**Figura 3: Desenvolvimento histórico dos paradigmas educacionais**

Fonte: Branson<sup>7</sup> (1990 apud CANDAU, 1999)

Analisando-se a figura, verifica-se que o primeiro paradigma (A) é o da instrução oral, no qual a aprendizagem está centrada no professor, que tem a posse do conhecimento e irá transmiti-lo ao aluno de forma direta e sem interferência. No segundo (B), o Paradigma Atual, o professor se torna mediador do processo. Apresenta uma bagagem maior de

<sup>7</sup> BRANSON, R. K. (1990) *Issues in The Design of Schooling: changing the paradigm, educational technology*. 30(4): p.7-10, 1990 apud CANDAU, V.M. (Org.) (1999). *Magistério: construção cotidiana*. Petrópolis. Editora: Vozes.

conhecimento sobre um determinado assunto, mas através do diálogo e interação entre o professor e aluno, assim como aluno com o aluno, acontecerá o aprendizado. Nesse processo de relação, a aprendizagem continua centrada no professor. O paradigma (C) baseado na tecnologia, cada aluno interage com o professor, com a base de conhecimento, com sistemas especialistas e formalmente com outros alunos. O aprendizado acontece de diversas formas e não apenas por intermédio do professor.

O modelo proposto nesta pesquisa enfatiza características deste ambiente de aprendizado, onde recursos enviados (figura 33) podem ser inseridos na base de conhecimento de domínio.

### **3.3 Tecnologias utilizadas na construção da arquitetura**

#### **3.3.1 Sistemas Especialistas**

O desenvolvimento de Sistemas Especialistas (SE) começou de maneira tímida, nos anos 50 do século XX, com programas de jogos de xadrez e programas de resolução automática de teoremas. Durante os anos 60 do mesmo século, teve um grande impulso através do projeto DENDRAL, desenvolvido pela Universidade de Stanford para a NASA, onde o seu objetivo era o de analisar a composição química do solo marciano, que utilizava pela primeira vez um banco de conhecimentos formado por conhecimentos heurísticos que eram manipulados por regras. Desde então, sistemas especialistas têm sido aplicados em diversas áreas, destacando-se sua utilização na área de medicina, engenharia, química e economia e nas mais diferentes tarefas, tais como: interpretação, planejamento e projeto, consultoria, diagnóstico, monitoramento, controle e simulação Durkin<sup>8</sup> (1996 apud CORDEIRO, 2001).

Um sistema especialista é composto de programas sofisticados que manipulam a base de conhecimento previamente adquirido de especialistas humanos, tem a capacidade de oferecer ao usuário conselhos inteligentes ou decidir inteligentemente sobre o

---

<sup>8</sup> DURKIN, J. (1994) *Expert Systems: Design and Development*. Prentice Hall, Englewood Cliffs New Jersey, USA, 1994 apud CORDEIRO, A.D. (2001). *Concepção e implementação de um sistema multi-agentes para gestão da comunicação de dados "ON-LINE" entre sistemas*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2001

processamento de uma função e também justificar sua própria linha de raciocínio de maneira direta quando requeridos.

Os problemas resolvidos por esses sistemas são delimitados em uma área específica do conhecimento humano, e necessariamente são problemas que podem ser simbolicamente representados. Feigenbaum<sup>9</sup> (1983, apud HARMON e KING, 1988) define um sistema especialista como:

É um programa inteligente de computador que usa conhecimentos e procedimentos de inferência, para resolver problemas que são bastante difíceis, de forma a requererem para sua solução, muita perícia humana. O conhecimento necessário para atuar a esse nível, mais os procedimentos de inferências empregados, pode considerar-se um modelo da perícia aos melhores profissionais do ramo. O nível de desempenho de um sistema especialista depende do tamanho e da qualidade do banco de conhecimento que possui.

Sistemas especialistas e bases de conhecimento sobre um domínio são usualmente construídos por programadores que em geral podem ter pouco conhecimento sobre este mesmo domínio. A arquitetura dos Sistemas Especialistas está baseada por três elementos básicos apresentados na figura a seguir:



**Figura 4: Estrutura de um Sistema Especialista**  
**Fonte: Cordeiro (2001)**

- Base de conhecimento: é a parte que mantém o conhecimento especialista. A base de conhecimento é, talvez, o mais importante componente do Sistema Especialista, pois a qualidade das respostas do SE depende diretamente de sua

<sup>9</sup> FEIGENBAUM, A.V. (1983). *Total quality control*, 3ª Ed., McGraw-Hill, New York, 1983 apud HARMON, P.; KING, D. (1988). *Sistemas especialistas*. Rio de Janeiro. Editora: Campus.



base de conhecimento. A base de conhecimento é, portanto, um agrupamento de conhecimentos representado mediante uma técnica adequada ao sistema em questão. As informações contidas na base de conhecimento são baseadas em fatos e incluem os relacionamentos entre os fatos (KELLER, 1991). Esse conhecimento é extraído por meio de entrevista com especialistas da área e de outras técnicas de aquisição de conhecimento. A aquisição do conhecimento representa a fase mais complexa do desenvolvimento do Sistema Especialista. Constitui-se no processo mediante o qual o engenheiro do conhecimento extrai os fatos e a heurística, associada à tarefa (CORDEIRO, 2001). Na estrutura proposta (capítulo 4) existe uma Base de Conhecimento de Domínio que não deve ser confundida com a Base de Conhecimento do sistema especialista, a Base de Conhecimento de Domínio será utilizada para o treinamento do aluno e a base de conhecimento do sistema especialista é um agrupamento de conhecimentos para ser utilizado por este sistema.

- Memória de trabalho: segundo Durkin<sup>10</sup> (1996 apud CORDEIRO, 2001), a memória de trabalho contém as informações iniciais e os fatos de um problema, que são descobertos durante uma consulta. Tais fatos podem ser informados pelo usuário do sistema ou fornecido pela avaliação de uma das regras da base de conhecimento. A memória de trabalho é dinâmica, possibilitando a inserção ou a alteração de um determinado fato durante a resolução de um mesmo problema.
- Máquina de inferência: contém o conhecimento geral que direciona e controla o raciocínio do sistema. O mecanismo de inferência examina os fatos e as regras existentes na base de dados, acrescentando novos fatos, sempre que possível, na medida das interpretações e conclusões que executa.

Segundo Cordeiro (2001) os SE apresentam as seguintes vantagens:

---

<sup>10</sup> DURKIN, J. (1994) *Expert Systems: Design and Development*. Prentice Hall, Englewood Cliffs New Jersey, USA, 1994 apud CORDEIRO, A.D. (2001). *Concepção e implementação de um sistema multi-agentes para gestão da comunicação de dados "ON-LINE" entre sistemas*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2001.

- Permanência do conhecimento: os SE não esquecem, mas humanos podem vir a se esquecer;
- Capacidade de reprodução do conhecimento: muitas cópias de um SE podem ser feitas, já o mesmo não é verdade para o caso de especialistas humanos;
- Eficiência: a velocidade de operação dos SE é muitas vezes maior que o especialista humano.
- Consistência e imparcialidade: um SE não é afetado por efeitos externos. Sempre terá o mesmo parecer, independentemente do dia, hora ou fatores externos à consulta.
- Documentação: os SE têm documentação de todo o seu conteúdo, já o ser humano, nem sempre.
- Completude: um SE pode rever todas as transações que realizou até inferir uma conclusão (rever todo o processo de encadeamento, ou seja, toda a memória de trabalho).

Por outro lado afirma Cordeiro (2001), os SE apresentam as seguintes desvantagens por não conseguirem lidar com:

- Criatividade: especialistas humanos podem responder criativamente para situações não usuais e inesperadas; já os SE não conseguem;
- Aprendizado: especialistas humanos podem automaticamente se adaptar às mudanças; já os SE devem sofrer alteração na base de conhecimento, o que nem sempre é tarefa simples e rápida;
- Degradação: normalmente, os SE não apresentam a capacidade de reconhecer quando não existe resposta ou quando o problema está além do seu domínio de conhecimento.

Sistemas especialistas serão utilizados na estrutura proposta (figura 30) para a identificação da evolução da aprendizagem, a possibilidade de trabalhar com amostragem, média, desvio padrão, comparação de amostras entre grupos de alunos, entre outras técnicas com objetivo de identificar o comportamento da curva de aprendizagem levam a considerar este tipo de tecnologia.

### 3.3.2 Sistemas tutores

Foi na década de 70 do século XX que os sistemas do tipo CAI sofreram uma reformulação, com o uso de técnicas de Inteligência Artificial, passando a ser chamados de

*ICAI (Intelligent Computer Assisted Instruction - Instrução Inteligente Auxiliada por Computador)* ou *ITS (Intelligent Tutoring Systems - Sistemas de Tutorização Inteligentes)* os quais permitiram uma maior flexibilidade e participação ativa do aluno e do sistema, o que resultou num ambiente cooperativo para o ensino e aprendizagem.

Thirry (1999) identifica sistemas tutores como:

Sistemas computacionais que se caracterizam por darem suporte ao aluno em um dado domínio de conhecimento, considerando as suas características individuais. Dessa forma, o aluno estabelece uma relação com o sistema tutor, na qual é considerado o seu conhecimento prévio e o seu ritmo de aprendizagem. Os STI oferecem considerável flexibilidade na apresentação do material e uma maior habilidade para responder às necessidades do usuário. Eles procuram não apenas ensinar, mas aprende informações relevantes sobre o aluno, proporcionando-lhe um aprendizado individualizado.

O principal objetivo dos Sistemas Tutores Inteligentes é proporcionar um ensino adaptado a cada aluno, tentando se aproximar do comportamento de um professor humano na sala de aula. Os Sistemas Tutores Inteligentes são caracterizados como ferramentas ou softwares educacionais aplicados nos ambientes de ensino-aprendizagem, objetivam a aquisição de conhecimento de maneira interativa. (COSTA, 1999).

Segundo Jonassen<sup>11</sup> et al (1993, apud ANDRADE e BRASILEIRO, 2003), um STI deve passar em três testes antes de ser considerado inteligente:

- Conteúdo do tema ou especialidade deve ser codificado de modo que o sistema possa acessar as informações, fazer inferências ou resolver problemas.
- Sistema deve ser capaz de avaliar a aquisição deste conhecimento pelo aluno.
- As estratégias tutoriais devem ser projetadas para reduzir a discrepância entre o conhecimento do especialista e o conhecimento do aluno.

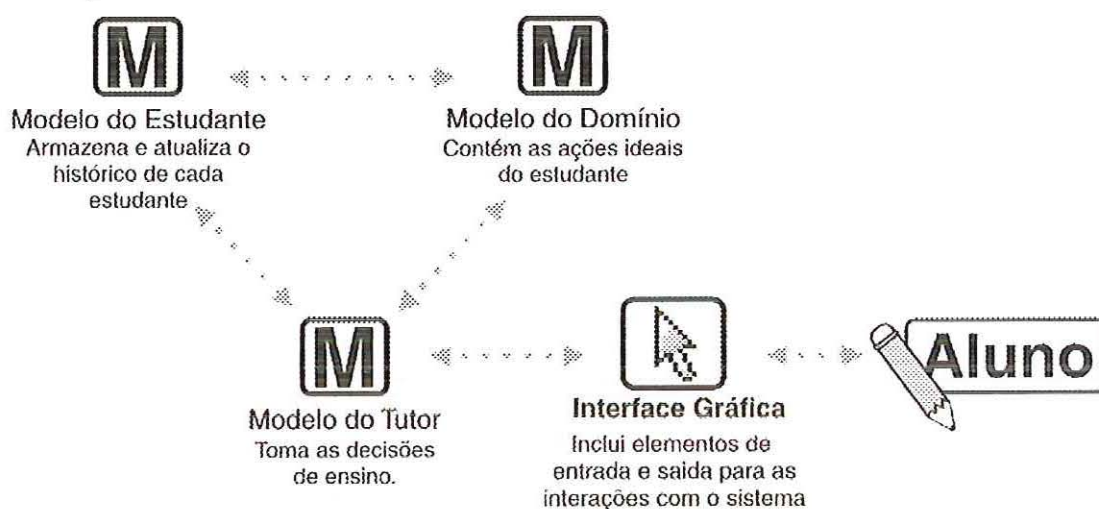
Urretavizcaya<sup>12</sup> (2001, apud ANDRADE e BRASILEIRO, 2003) afirma que as características mais importantes de um STI são:

<sup>11</sup> JONASSEN, D.H., Wilson, Brent G., Wang, Sherwood, & Grabinger, R, Scott. *Constructivist uses o expert systems to support learning*. Journal of Computer-Based Instruction, 29(3). pp. 86-94. 1993 apud ANDRADE, V.M.; BRASILEIRO, F.V. (2003). *Sistema de gerenciamento de aprendizagem, uma metodologia de avaliação*. 2003.



- O conhecimento do domínio está restrito e claramente articulado.
- Possuem conhecimento do aluno que lhes permite dirigir e adaptar o ensino.
- A seqüência do ensino não está predeterminada pelo designer instrucional.
- Realizam processos de diagnóstico mais adaptados ao aluno e mais detalhados.
- A comunicação Tutor-Aluno melhora, permitindo que o aluno realize perguntas ao tutor.

Segundo Faraco (2003), os Sistemas Tutores Inteligentes possuem uma estrutura modular onde cada módulo, além de desempenhar uma função específica dentro da arquitetura, está inter-relacionado e em sincronia com os demais.



**Figura 5: Componentes de um sistema tutor inteligente genérico**

Fonte: Mcarthur, Bishay e Lews (2004).

- **Modelo do Domínio:** contém o conhecimento armazenado do especialista na matéria a ser ensinada. Esse conhecimento é adquirido a partir do especialista, devendo, portanto, ser transferido para o aluno. Segundo Costa (1999) o Modelo de Domínio armazena a informação que o tutor está ensinando. Deve-se procurar uma representação do conhecimento que esteja preparada para o crescimento incremental do domínio.

<sup>12</sup> URRETAVIZCAYA L. M. (2001). *Sistemas Inteligentes em el ámbito de la educación*. Revista Iberoamericana de Inteligência Artificial. Número 12. p.5-12. 2001 apud ANDRADE, V.M.; BRASILEIRO, F.V. (2003). *Sistema de gerenciamento de aprendizagem, uma metodologia de avaliação*. 2003.

- **Modelo do Aluno:** contém as informações referentes a cada aluno que faz uso do sistema. Essas informações estão relacionadas com o nível de conhecimento do aluno sobre a matéria e o seu ritmo de aprendizagem. Costa (1999) afirma também que o módulo do aluno armazena informações específicas para cada aluno de forma individual. No mínimo, este módulo deve manter um histórico sobre como o aluno está trabalhando no material em questão. É interessante, também, manter registro sobre os erros do aluno. Na arquitetura proposta, os dados serão obtidos através do modelo de dados (*data model*) do projeto SCORM (anexo B e C), no modelo proposto sistemas especialistas que identificarão a evolução do aprendizado deverão ter acesso a todas as informações armazenadas na base de informações do aluno (figura 24).
- **Modelo do Tutor:** a partir das informações contidas no modelo do aluno e no modelo do domínio, faz-se a seleção e o seqüenciamento do assunto a ser apresentado e enviado à interface gráfica. Esse módulo contém as estratégias de ensino e decide quando e como apresentar o conteúdo para o aluno. Costa (1999) enfatiza que o módulo deve ter acesso a todas as informações do sistema, incluindo-se o conhecimento global (domínio) e o conhecimento individual de cada aluno. A preocupação deste módulo não é a representação do conhecimento global, mas como um indivíduo representa seu próprio conhecimento através do uso de suas habilidades dentro do domínio.
- **Interface Gráfica:** também chamado módulo de comunicação ou ambiente de ensino, é a interface entre o STI e o aluno. O objetivo é possibilitar a imersão total do aluno na utilização de recursos de aprendizagem.

Segundo Mesquita e Silva (2003), uma das maneiras que se usam para conhecer as preferências do aluno, é que quando o aluno ao errar um exercício este mesmo exercício é apresentado usando uma outra técnica de investigação pedagógica. No momento que o aluno acerta o exercício, é registrada no banco de dados do aluno a técnica de exposição que originou o acerto (tipo de mídia indicado no objeto de aprendizagem). Quando expuser um novo elemento de conhecimento, o sistema procura o objeto de mídia com maior taxa de acerto, antes de utilizar outras. Esta técnica poderá ser abordada pelos objetos de aprendizagem presente na arquitetura proposta.



Prado (2002) afirma ainda que as novas pesquisas em Sistemas Tutores Inteligentes vem ao encontro das modernas correntes da educação, que apóiam a geração do conhecimento através de troca de informações entre os alunos. A indicação de um material específico conforme a necessidade individual do aluno também é reforçada:

Uma vez que os alunos não são iguais, nem todos os alunos requerem a mesma quantidade de reforço para compreender um determinado tópico, em um Sistema Tutor Inteligente, o aluno pode progredir exatamente na medida do seu atendimento. Conceitos que o aluno não entende bem poderão ser repetidos, enquanto outros podem ser apresentados apenas uma única vez, conseqüentemente evitando a mentalidade de apostilas para classes padrões. (TOMPSOM, 1996)

A dinâmica da utilização de sistemas tutores será abordada pela estrutura proposta, já que sistemas tutores levam em consideração a individualidade do aluno, esta tecnologia vem a colaborar para a solução do problema proposto o que se complementa pela estrutura padronizada do projeto *SCORM* enfatizados no tópico 3.3.4 deste capítulo.

### 3.3.3 Agentes Inteligentes

Segundo Bogo (2003) o conceito de agentes é um resultado de mais de 40 anos de pesquisa, sua principal idéia é a de decompor sistemas complexos em diferentes entidades, na tentativa de obter maior eficiência nos resultados, não é uma abordagem recente na história da computação, mas pode ser uma maneira bastante útil quando um problema de alta complexidade pode ser decomposto em pequenas partes, e estas partes serem resolvidas por entidades especializadas.

Russel e Norvig<sup>13</sup> (1995, apud PRADO, 2002) afirmam ainda que existe um conjunto comum de características que são típicas de um agente inteligente:

- Autonomia - ser capaz de operar sozinho e disparar ações sem a intervenção do usuário;

---

<sup>13</sup> RUSSEL, S.; NORVIG, P. (1995). *Artificial intelligence – a Modern Approach*, p. 788-790. New Jersey, USA: Prentice Hall, 1995 apud PRADO, N.R.S.A. (2002). *Uma arquitetura para ambientes de ensino-aprendizagem utilizando o sistema de modelos tutores inteligentes e agentes*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.



- Comunicabilidade - comunica-se com o usuário, com outros agentes ou outros processos;
- Percepção - habilidade de perceber e responder às mudanças no seu ambiente de execução;
- Adaptabilidade - processar diferentes níveis de tarefas, de acordo com sua competência.

Softwares agentes, como pessoas, podem ser mais produtivos quando trabalham de forma cooperativa e colaborativa. Uma coleção de agentes que se comunicam uns com os outros podem trabalhar de forma mais complexa, esta comunicabilidade onde são realizadas trocas de informações e dados é abordada pela estrutura proposta (figura 24).

Na história da evolução dos agentes, os primeiros trabalhavam na resolução de problemas usando heurísticas ou métodos baseados em conhecimento, alguns desses agentes, segundo Martins (2002) tinham a capacidade de aprender e modificar suas percepções sobre o ambiente no qual estavam inseridos.

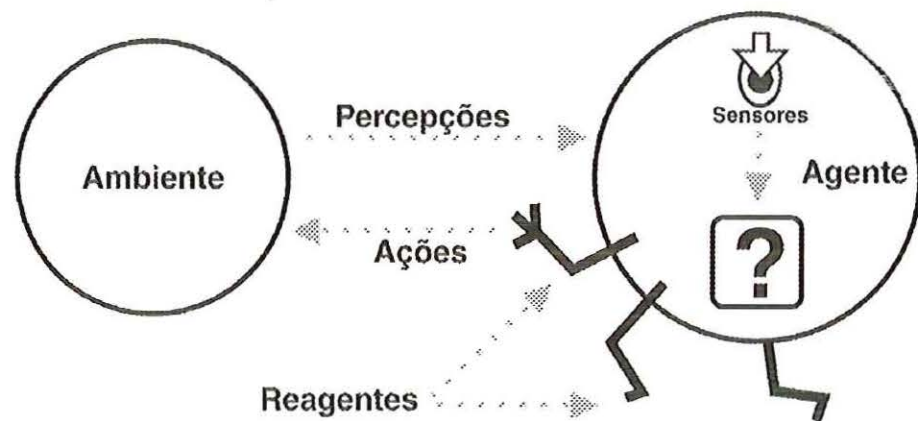
Essas características ainda estão presentes em um grande número de aplicações de inteligência artificial. A utilização de agentes foi durante muito tempo para automatizar aplicações e tarefas específicas, com o avanço da tecnologia, os agentes trazem uma abordagem mais genérica para a solução de problemas de treinamento e aprendizado, afirmação esta também apoiada por Coser (2000).

Segundo Martins (2002) os agentes inteligentes devem possuir autonomia, representando alguém ou algo interagindo de forma geral com o ambiente.

Pode-se, também, conforme ilustrado na Figura 6, a seguir, definir (genericamente) um agente como “qualquer coisa que possa ser vista, perceber seu ambiente de atuação, através de sensores, e agir sobre ele, através de atuadores” Russel e Norvig<sup>14</sup> (1995 apud PRADO, 2002).

---

<sup>14</sup> RUSSEL, S.; NORVIG, P. (1995). *Artificial intelligence – a Modern Approach*, p. 788-790. New Jersey, USA: Prentice Hall, 1995 apud PRADO, N.R.S.A. (2002). *Uma arquitetura para ambientes de ensino-aprendizagem utilizando o sistema de modelos tutores inteligentes e agentes*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.



**Figura 6: Definição de Agentes**

**Fonte: Russel e Norvig<sup>12</sup>(1995 apud PRADO, 2002).**

Agentes devem possuir propriedades sensoriais e meios para reagir ou interferir no ambiente computacional em que se encontram e comportamento baseado nas características implementadas internamente, na experiência adquirida e no conhecimento utilizado em sua construção (agente passivo). Porém, um agente é considerado autônomo apenas se seu comportamento for determinado por sua própria experiência, o que o torna capaz de interagir com seu ambiente (agente ativo) para concluir alguma tarefa, característica esta enfatizada por Giese (1998).

Prado (2002) em sua pesquisa destaca o conceito de agentes móveis que são softwares capazes de percorrer redes de computadores, como a internet interagindo com *host's*<sup>15</sup> externos (servidores de serviços), realizando chamadas nestes *hosts* e retornando ao seu ponto de origem, para que isto seja possível, deve haver uma estrutura de transporte que mova o código fonte do agente de uma localização para outra. Com o avanço da tecnologia de ambiente aberto, surgiram conceitos de linguagem que representam dados estruturados como o XML, protocolos de comunicação como o *SOAP* e novos conceitos de serviços como os de *Web Services* que permitem com que os agentes possam se comunicar de maneira mais segura e estruturada através da *web*. A linguagem de comunicação entre os agentes *KQML* (*Knowledge Query and Manipulation Language*) pode abordar a tecnologia

<sup>15</sup> Host - Computador da rede que desempenha funções centralizadas, colocando arquivos e programas à disposição das estações de trabalho; Servidor.

*XML* quando realiza troca de informações estrutura, permitindo a comunicação e execução de métodos dos agentes.

Prado (2002) propõe uma arquitetura composta de dois conjuntos de base de informações:

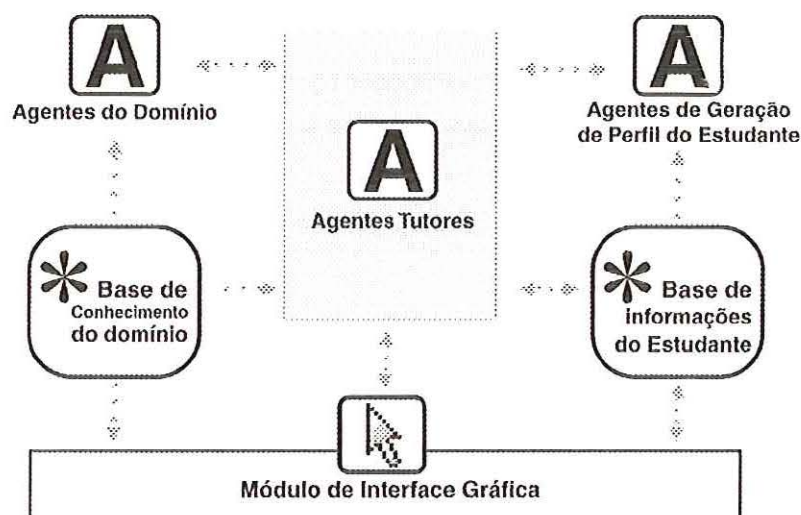
- Bases de Conhecimento de Domínio: que armazenam informações a respeito do domínio a ser ensinado.
- Bases de Informações do Aluno: que terá o repositório de dados que irá identificar do perfil do aluno de forma individual.

Ambas também utilizadas como base na construção da estrutura proposta, as bases de conhecimento de domínio segundo Prado (2002), são montadas na fase de construção do ambiente computacional de aprendizado, de acordo com um domínio específico a ser ensinado e a partir do conhecimento de especialistas ou fontes da informação. A base na estrutura proposta será composta por objetos de aprendizagem (*Learning Objects*) que poderão ser referenciados a outros objetos, permitindo assim a construção da árvore de conhecimento do domínio (demonstrado com mais detalhe no tópico 3.3.4 deste capítulo). À medida que se deseje que o ambiente trabalhe com outros domínios de conhecimento, novos conjuntos de objetos poderão ser inseridos (no caso do projeto *SCORM* empacotados).

Na arquitetura proposta os agentes são incumbidos de identificar alterações de informações que ocorrem nas bases de informações, disparando serviços (*services*) que atualizam outras bases desta arquitetura quando necessária estes serviços utilizam visões de dados repassados entre os agentes.

Prado (2002) enfatizou em sua pesquisa uma solução de treinamento através de agentes e sistemas tutores, apresentando um conjunto de procedimentos estruturados na construção dos sistemas tutores e agentes inteligentes na figura a seguir:





**Figura 7: Arquitetura de ambiente ensino aprendizagem utilizando agentes e sistemas tutores inteligentes**  
**Fonte: Prado (2002)**

A interação do aluno com o ambiente é realizada através do Módulo de Interface. Essa interação se dá através de um navegador internet que permite ao aluno realizar atividades, explorando páginas no formato HTML (*Hypertext Markup Language*) e recursos hipermídia.

A arquitetura de Prado (2002) possui, ainda, outros três módulos e estes são compostos por agentes inteligentes. As funções desses agentes são equivalentes às funções realizadas pelos componentes de um STI. Os módulos são:

- módulo de agentes do domínio: é composto por agentes que possuem acesso à base de conhecimento do domínio, inferem e buscam informações adequadas às situações de ensino.
- módulo de agentes para a geração de perfil do aluno: É composto por agentes que são responsáveis por traçar perfil do aluno, a partir de inferências feitas sobre as informações contidas nas Bases de Informações do Aluno.
- módulos de agentes tutores: é composto por agentes responsáveis pela elaboração de planos de ensino, pela definição das estratégias de ensino adequadas e o acompanhamento de desempenho do aluno.

O resultado desse desmembramento de informação é um conjunto de módulos independentes que poderão ser adaptados ao modelo apresentado (figura 24), a estrutura da base de conhecimento de domínio e a base de informações do estudante preferencialmente devem obedecer a um padrão para que possam adquirir a característica adaptativa entre as instituições de ensino, esta padronização é proposta através do projeto *SCORM* apresentada no próximo tópico.

### 3.3.4 Objetos de aprendizagem (*Learning objects*)

Os objetos de aprendizagem representam uma tecnologia alternativa para desenvolvimento da interação de conteúdos instrucionais, onde cada objeto é uma pequena parte de uma parte maior que pode ser remanejado da forma mais interessante, como se fosse um movimento de jogo do tipo Lego, a exigência é que cada pedaço deve ter a possibilidade de comunicar-se com os sistemas de aprendizagem usando um método padrão e que não dependa do sistema de treinamento e nem da interface do aluno, outra exigência é que o objeto tenha autonomia e trace sua melhor forma instrucional. Como o aluno pode mover-se entre esses objetos, leva-o a uma dinâmica bastante interessante já que o aluno pode ir diretamente ao conteúdo de seu interesse, não há nenhum padrão para o tamanho ou a granularidade de conteúdo de um objeto da aprendizagem, apenas que quanto maior o objeto ou maior a sua abrangência de conteúdo mais difícil será o seu reuso.

Segundo Bettio (2003) os objetos de aprendizagem podem ser qualquer tipo de mídia, que sirva como material pedagógico e que possa ser disponibilizada eletronicamente através da Internet.

Lucena (2004) também afirma que objetos de aprendizagem podem ser definidos como qualquer recurso digital que possa ser reutilizado para o suporte ao ensino, cujo principal objetivo é dividir o conteúdo educacional a ser ensinado em pequenos pedaços que possam ser reutilizados em diferentes domínios de aprendizagem, absorvendo as características de objetos temáticos. Portanto os profissionais que irão trabalhar no desenvolvimento de conteúdos de um curso *on-line* devem pensar que estão construindo pequenos pedaços que, quando utilizados, deverão formar um corpo maior independentemente do domínio de aprendizagem.



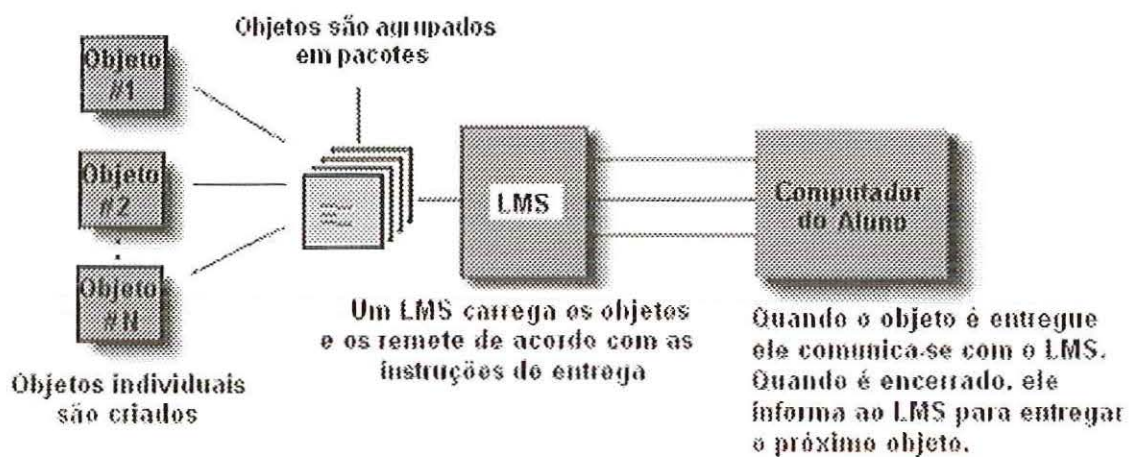
Para Longmire (2004) os Objetos de Aprendizagem possuem características que procuram resolver diversos problemas existentes atualmente, quanto ao armazenamento e distribuição de informação por meios digitais. As características enfocadas são:

- flexibilidade: Como os Objetos de Aprendizado são construídos de maneira que possuam início, meio e fim, eles já nascem flexíveis, podendo ser reutilizados sem necessitar de nenhum tipo de manutenção. Essa capacidade de reutilização possibilita a criação de novos cursos utilizando-se conhecimentos já armazenados e consolidados;
- facilidade para Atualização: Como os mesmos objetos são utilizados em diversos momentos, a atualização dos mesmos em tempo real é relativamente simples. Desde que todos os dados relativos a esse objeto estejam em um mesmo banco de informações, a necessidade de se atualizar este conhecimento em todos os ambientes que o utilizam é desnecessário. Desta forma, a pessoa que apenas utilizou o conhecimento de um autor poderá contar com correções e aperfeiçoamentos sem ter que se preocupar com isso;
- customização: A mesma característica que proporciona ao objeto flexibilidade, também proporciona a sua customização; como os objetos são independentes, a idéia de utilização dos mesmos em um curso, especialização ou qualquer outro tipo de qualificação, torna-se real, sendo que cada entidade educacional pode utilizar os objetos e arranjá-los da maneira que mais convier Também os indivíduos que necessitarem de aprendizado poderão montar seus próprios conteúdos programáticos;
- interoperabilidade: A criação de um padrão para armazenagem de Objetos de Aprendizado cria mais uma vantagem, a interoperabilidade, ou seja, a reutilização dos objetos não apenas em nível de plataforma de ensino, mas, também, em nível mundial. A idéia de um Objeto de Aprendizado ser criado e poder ser utilizado em qualquer plataforma de ensino do mundo aumenta ainda mais as vantagens destes objetos. Assim que a barreira lingüística for quebrada, a interoperabilidade entre bancos de objetos de todo o mundo será selada, trazendo vantagens jamais vistas na educação;
- aumento do valor do conhecimento: A partir do momento em que um objeto é reutilizado várias vezes e em diversas especializações, este vem, ao longo do tempo, sendo melhorado, assim, sua consolidação cresce de uma maneira espontânea. A melhoria significativa da qualidade do ensino é mais uma vantagem que pode ser considerada ao se pensar em Objetos de Aprendizado;



- indexação e procura: A padronização dos objetos virá também a facilitar a idéia de se procurar por um objeto necessário. Quando um conteduidista necessitar de determinado objeto para completar seu conteúdo programático, a padronização dos mesmos e a utilização de assinaturas digitais, tende a criar uma facilidade em encontrar objetos com as mesmas características, em qualquer banco de objetos que esteja disponível para eventuais consultas.

Todas estas características apontadas pela capacidade de reuso em sistemas de treinamento e avaliação, promovem uma atualização de conteúdo automática, a oferta do objeto possui uma relação de empacotamento e entrega conforme o interesse de disponibiliza-lo, seu funcionamento genérico é apresentado na figura a seguir:



**Figura 8: Funcionamento global de um objeto de aprendizagem**

Um *LMS* (*Learning Management System*) é um sistema que gerencia e administra eventos de treinamento de um *e-learning*. O *LMS* registra usuários, trilha cursos em um catálogo e grava dados de alunos. Usualmente não inclui capacidade própria de autoria; ao invés, foca compatibilidade com cursos criados por uma variedade de outras fontes.

O objeto de aprendizagem é concebido através de um modelo de armazenamento e distribuição de informação em sistemas de ensino à distância veiculados pela Internet de forma padronizada (definida pelo projeto *SCORM*). Segundo Singh<sup>16</sup> (2001, apud BETTIO, 2003), primariamente um objeto de aprendizagem é composto de três partes:

<sup>16</sup> SINGH, H. (2001) Introduction to Learning Objects. Fonte: [www.elearningforum.com/july2001/singh.ppt](http://www.elearningforum.com/july2001/singh.ppt). 2001 apud BETTIO, R.W. (2003). *Avaliações gráficas e dinâmicas aplicadas a ambientes virtuais de aprendizagem*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2003.



**Figura 9: Divisão de um objeto de aprendizagem**  
**Fonte: Bettio (2003)**

- **Objetivos:** Esta parte do objeto tem como intenção demonstrar ao aprendiz o que ele poderá aprender a partir do estudo desse objeto. Também poderá conter uma lista dos conhecimentos prévios necessários para um bom aproveitamento de todo o conteúdo disponível. Pode ser comparado, de uma forma grosseira, à ementa de uma disciplina.
- **Conteúdo instrucional:** Aqui deverá ser apresentado todo o material didático necessário para que no término, o aluno possa atingir os objetivos citados no item anterior.
- **Prática e feedback:** Uma das características importantes do paradigma Objetos de Aprendizado é que a cada final de aprendizado, julga-se necessário que o aprendiz verifique se o seu desempenho atingiu as expectativas. Caso não, o aprendiz deve ter liberdade para voltar a se utilizar do objeto quantas vezes julgar necessário, provendo caminhos alternativos.

#### 3.3.4.1 O projeto SCORM

Para desenvolver especificações não proprietária e padronizadas na utilização de recursos de aprendizagem através da Internet (e-learning), vários comitês se uniram com a tentativa de obter uma normalização destes recursos, entre esses comitês estão o *AICC (Aviation Industry CBT Committee)*, *IMS (Instructional Management System - Global Learning Consortium)* e o *IEEE LTSC (IEEE Learning Technology Standards Committee)*. Além desses comitês o governo americano para promover a unificação das especificações criou o *ADL (Advanced Distributed Learning)*, todos procurando definir uma compatibilidade e padronização para a transferência de conteúdos instrucionais. O objetivo dos comitês era de estabelecer um padrão estrutural e de funcionalidade dos objetos de aprendizagem que fossem compatíveis entre si, possibilitando a interoperabilidade entre as instituições de *e-learning*.



Em 1988, o *AICC* desenvolveu um conjunto de normas focando questões mais específicas para a implementação da interoperabilidade entre sistemas de *CBT* (*Computer Based Training*) e *WBT* (*Web Based Training*), além de produzir um conjunto de manuais técnicos para a orientação desta implementação, a origem destas normas foram originadas dos sistemas de simuladores de vôo (sistema de treinamento para pilotos).

O *IMS* surgiu com o intuito de desenvolver e promover especificações de uma estrutura de conteúdo e que pudesse direcionar a seqüência de utilização dos objetos de aprendizagem, além do que possibilitar uma representação da localização de conteúdos de forma abstrata na entrega de serviços de *e-learning* através da Internet. Entre outras contribuições, o *IMS* estabeleceu especificações e normas através de um metadado<sup>17</sup> (*metadata*), estas especificações tinham como objetivo de registrar e o progresso dos estudantes de forma individual, através da troca de dados dos alunos contidos nos sistemas das instituições de ensino.

O Departamento de defesa norte-americano (*DoD - Department of Defense*) lançou o projeto *SCORM* (*Sharable Content Object Reference Model*), com o objetivo de desenvolver um modelo de repositório que capacitasse a reutilização de componentes de conteúdos instrucionais para treinamento, independentemente da plataforma em que seja criado (aproveitando as especificações tanto do *AICC* e das normas estipuladas pelo *IMS*). O *ADL* não competiu com outros esforços de padronização; pelo contrário, este buscou incorporar o trabalho já realizado pelo *AICC*, *IMS* e *IEEE* num conjunto, e em janeiro de 2000 foi lançada a primeira versão do projeto *SCORM*.

Desde que o *SCORM* foi lançado sofreu algumas atualizações, o que foi atribuído pela prática de sua utilização através do *e-learning*. O projeto atualmente se encontra na versão 1.3.1 (publicado em 22 de julho de 2004), atualmente o projeto possui características bem definidas, representando um conjunto de especificações estruturais que são apresentadas a seguir:

---

<sup>17</sup> Metadado: conjunto de dados-atributos, devidamente estruturados e codificados, com base em padrões internacionais, para representar informações de um recurso informacional em meio digital ou não digital, contendo uma série de características e objetivos.



- um modelo de especificações de empacotamento de conteúdo em forma de metadado (*Content Packaging*), para a criação da estrutura dos cursos, seus conteúdos e os controles de seqüência de conteúdo;
- uma representação descritiva da estrutura baseada em *XML*<sup>18</sup> (*Extensible Markup Language*), de modo que esses conteúdos possam ser implementados nos mais diversos tipos de computadores e sistemas de aprendizagem (*LMS*);
- um conjunto de especificações para a definição do ambiente de execução *RTE* (*run-time environment*), presente não só do lado do cliente (*Client Side*) mas como do lado do servidor (*Server Side*) de sistema de treinamento (*LMS*);
- um modelo de dados (*Data Model*) para a comunicação entre os objetos de aprendizagem e os *LMS*'s.

### **Empacotamento de conteúdo (*Content Packaging*).**

O objetivo do modelo de empacotamento de conteúdo (*Content Packaging*) é o de agregar recursos (objetos de aprendizagem) em um processo de instrução (curso, disciplina, módulo, capítulo, etc.) criando assim um pacote de conteúdos. Este empacotamento de conteúdo é representado por uma estrutura identificada como “manifesto” que é constituído por:

- *assets* - é a representação digital de um arquivo de mídia, texto, imagens, sons, etc., que podem ser acessados por um navegador da internet (*Browser*). Os *assets* são representados com elementos identificados pela *tag XML* `<file>`;
- *resources* - É a menor unidade relacionada à estrutura de um curso (modelo de agregação de conteúdos). Pode ser de dois tipos, *Sharable Resource ou Sharable Content Objects (SCO)*.

---

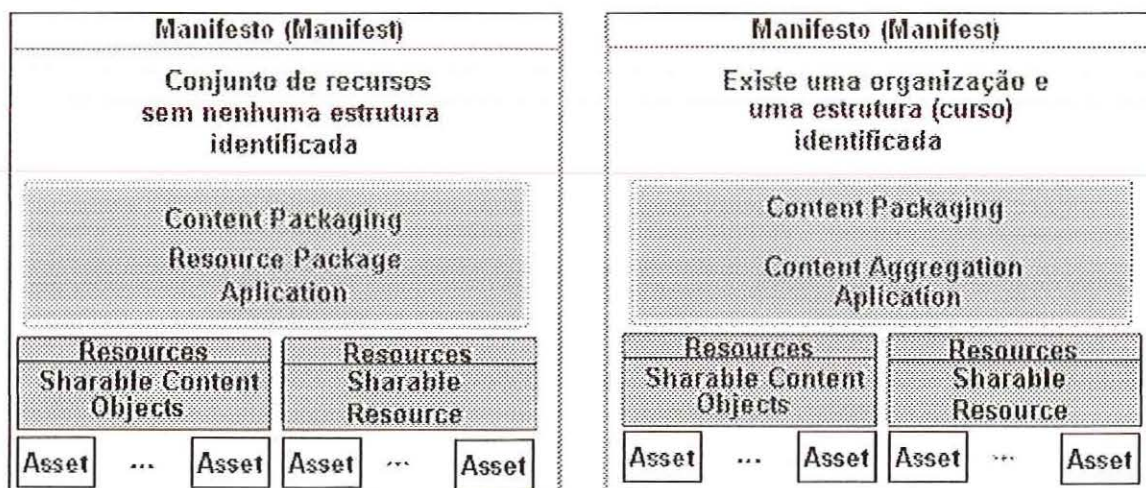
<sup>18</sup> XML – *Extensible Markup Language* - é uma linguagem flexível e poderosa padronizada pela indústria de tecnologia, usada para troca de informações em modelos de dados para a internet e aplicações distribuídas.

- *Sharable Resource* - Um *Sharable Resource* é um conjunto de *assets* que pode ser pesquisado em um repositório, possibilitando assim a sua reutilização.
- *Sharable Content Objects (SCO)* - Representa um conjunto de um ou mais *assets* ou *sharable resources*, mas com características de comunicação entre o aluno com o *LMS*, ou seja, no *SCO* ocorre comunicação bidirecional entre o aluno e o *LMS*, esta comunicação é propiciada pelo *RTE (Run Time Environment)*.

O *SCORM* utiliza o modelo de empacotamento de duas formas ou perfis distintos:

- *Resource Package Application* - Neste caso, o manifesto que acompanha os conteúdos apenas é constituído pela *tag* em XML `<resources>`, em que descreve um conjunto de recursos (*SCO's* ou *Sharable Resources*), sem nenhuma estrutura, apenas um aglomerado num pacote para intercâmbio.
- *Content Aggregation Application* - Neste caso, além da definição dos recursos, é descrito sua organização e estrutura, representado no manifesto pelas *tag's* XML `<organizations>` e `<resources>` (figura 11 e 13).

Para entender o conjunto: *manifest*, *assets*, *resources* e *content aggregation* observe a relação entre eles na figura a seguir:

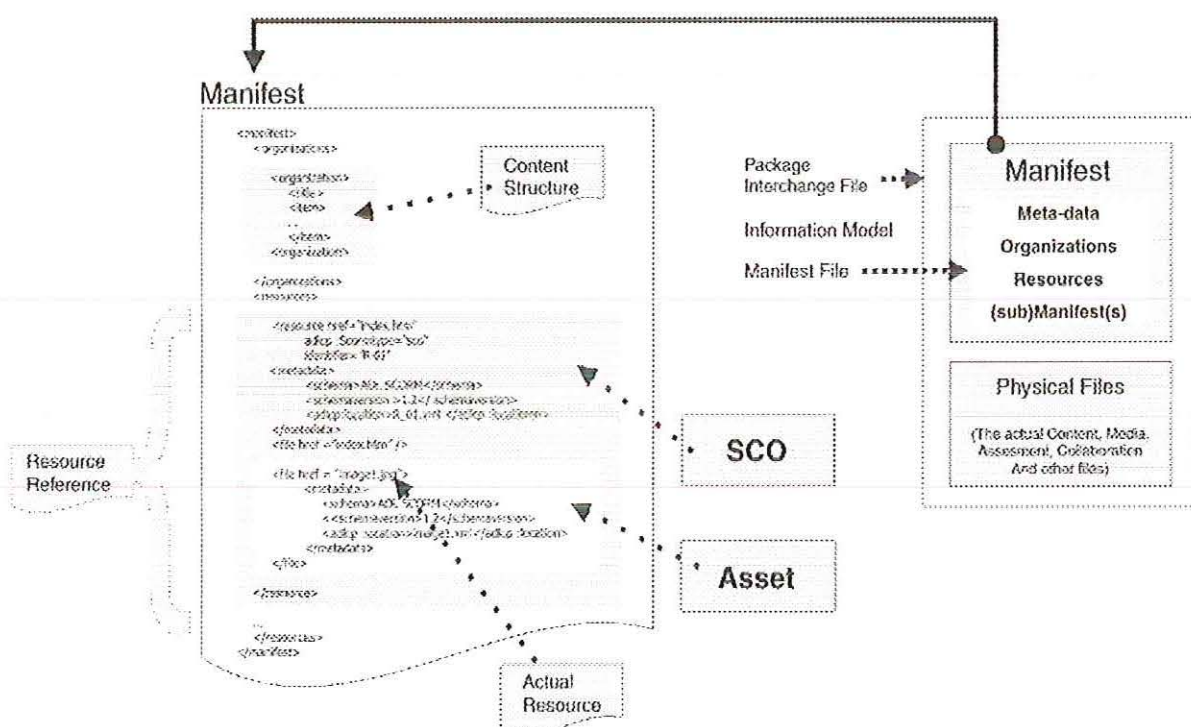


**Figura 10: Representação do esquema de uma estrutura funcional.**



Alguns elementos do manifesto são obrigatórios e outros são opcionais, estes elementos (e a sua descrição em *XML*) estão descritos no anexo A. Segundo a normalização do *IMS*, o modelo de agregação de conteúdo tem por objetivo empacotar os objetos de aprendizagem de uma maneira que possam ser, livremente compartilhados, possuindo assim um grande grau de reusabilidade. Este modelo de empacotamento é chamado de *PIF* (*Package Interchange File*) que é subdividido em duas áreas, o Manifesto (*Manifest*) e a área descritiva do arquivo (*Physical Files*). Observe a figura 11 ou a verifique ampliada no anexo F.

Na versão 1.3.1 o projeto *SCORM* incorporou em definitivo a estrutura do *IMS*, fazendo parte de seu modelo, o que provocou mudanças no conceito dos termos de agregação, mas a lógica da estrutura de funcionamento manteve-se, as alterações realizadas pela adoção da nova versão do *SCORM* publicada em 22 de julho de 2004 (versão 1.3.1) são encontradas no apêndice B ( identificação Url-3).



**Figura 11: PIF – Package Interchange File**  
**Fonte: Bettio (2003) adaptado IMS (2004)**

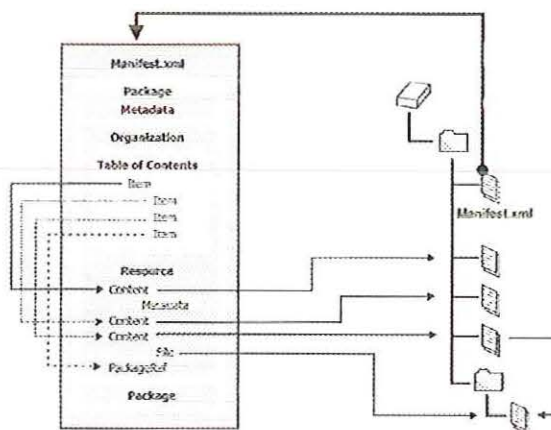


O manifesto (*manifest*) descreve a organização do conteúdo e os recursos do pacote (identificado com o nome *imsmanifest.xml* na versão 1.3.1) foi subdividido para uma melhor organização, em quatro sub-itens:

- *Meta-data*: São dados que referenciam dados, ou seja, é uma estrutura que descreve o pacote, explicitam que tipo de informação o pacote contém.
- *Organizations*: Descreve a organização do pacote, este campo contém a taxionomia (classificação) organizacional dos dados referenciados no pacote.
- *Resources*: Este item contém informações sobre os arquivos físicos englobados no item *Physical Files*.
- *(sub)Manifest*: Faz a ligação com outros manifestos, possibilitando assim a criação da árvore de conhecimento.

Os arquivos físicos (*Physical Files*) contém os arquivos físicos citados no item *resources* do manifesto. Esses arquivos podem ser do tipo documento, vídeo, áudio ou qualquer outro tipo de mídia, que possa ser usada como material de ensino.

Um manifesto pode apontar para outro sub-manifesto e que, pode apontar também para outro sub-manifesto, e assim por diante, desta forma com a utilização de diversos sub-manifestos é possível criar a árvore de conhecimento. Na próxima figura é demonstrada a criação desta árvore de conhecimento:



**Figura 12: Construção da árvore do conhecimento**

**Fonte: Downes (2003)**

Através de uma linguagem descritiva (característica do *XML*), é possível criar uma estrutura flexível e padronizada, esta flexibilidade permite a troca de recursos digitais de aprendizado entre diferentes sistemas computacionais, além de possibilitar a definição da estrutura de diferentes recursos de aprendizado. A figura a seguir exemplifica o empacotamento de conteúdo através do manifesto representado pelo *XML*:

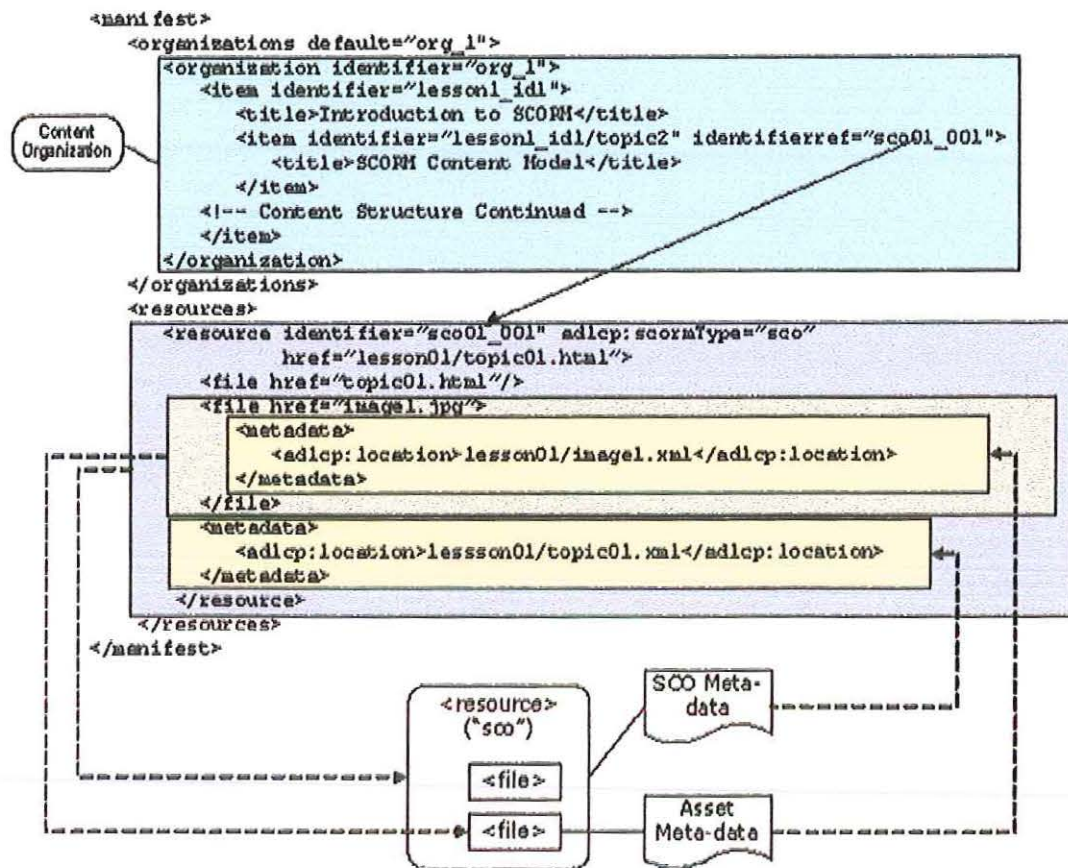


Figura 13 : Codificação de um Manifesto (*Manifest*)

Fonte: ADL(2004)

### O LOM (*Learning Object Model*)

Para que um objeto de aprendizagem possa ser utilizado (dependendo de seu conteúdo) pelas instituições de ensino ele primeiro necessita ser localizado, o que não é uma tarefa fácil ao pensar na vastidão e complexidade da *Web*, a solução proposta pelo projeto *SCORM* foi de não só guardar o objeto, mas também descrever o próprio objeto, ou seja, definir uma estrutura de dados que informasse os dados de um objeto. Objetos de aprendizagem potencialmente incluem informações sobre seu título (*title*), autor (*author*),



versão (*number version*), data de criação (*create date*), necessidades técnicas (*technical requirements*), um contexto educacional (*educational context*) e uma identidade (*ident*). Para definir todas essas informações, a comunidade de repositórios digitais adotou o modelo de referência de metadado do *IEEE* para objetos de aprendizagem conhecido como *LOM* (*Learning Object Model*), descrito com mais detalhes no anexo D e E.

Segundo Forth (2004), em 1995 na cidade Dublin, Estado de Ohio EUA, foi realizado o primeiro encontro para padronizar a estrutura de um repositório digital para o *e-learning*. Mas as necessidades da estrutura deste repositório eram diferentes entre as organizações de *e-learning*, surgindo então a necessidade de uma padronização. Para atender a esta demanda foi criada uma comissão chamada *IEEE LTSC LOM* (*Institute of Electrical and Electronics Engineers Learning Technology Standards Committee Learning Object Metadata*) com o propósito de construir uma especificação de um metadado padrão, que deveria ser utilizado por diversas instituições que praticam o *e-learning*. Contido neste metadado foi apresentado um *subset* específico para a área educacional (*Educational*).

Enquanto o *Content Package* é uma estrutura em *XML* que identifica a organização e a estrutura do pacote de objetos agregados, o *LOM* especifica um esquema conceitual padronizado de dados na qual define a estrutura do objeto de aprendizagem ou seja, uma coleção de detalhes do objeto de aprendizagem. Estes padrões facilitam a busca, avaliação e uso dos objetos de aprendizagem pelos alunos, instrutores e ferramentas de software que procuram conteúdos pela internet, isto é possível por que o *LOM* é caracterizado em possuir uma definição de blocos independentes que identificam e qualificam os conteúdos de aprendizagem (Anexo D e E).

Através do *LOM*, é possível descrever em detalhes os recursos de aprendizagem, onde através destas descrições, os recursos podem ser facilmente localizados e utilizados de uma forma apropriada no ambiente de *e-learning*, o *LOM* possui informações do tipo de mídia, tipo de sistema operacional, nível de interação, nível de dificuldade, entre outros que ajudam a identificar não só o perfil do ambiente em que o aluno atua (ou onde o objeto pode atuar).

Quando um objeto de aprendizagem é construído, vários dados sobre a sua estrutura devem ser inseridos (elementos e sub-elementos) para a construção do metadado *LOM*, permitindo assim, que qualquer plataforma de *e-learning* que siga o padrão poderá procurar



por objetos em qualquer local de armazenamento em que esteja disponibilizado objetos de aprendizagem, estes locais de armazenamento são conhecidos como repositórios de conteúdos, entre eles estão como exemplo o “*Wisc-OnLine*”<sup>19</sup>, “*VCILT Learning Objects Repository*”<sup>20</sup>, “*Merlot*”<sup>21</sup>, “*SMETE*”<sup>22</sup> entre outros.

O *LOM* (definido pelo *IEEE*) é hierárquico, e a base da hierarquia é o elemento *root* (raiz). O *root* contém muitos sub-elementos, e um sub-elemento podem conter sub-elementos adicionais, esse elementos adicionais são chamados de “*Branch*” (galhos). Se o sub-elemento não contém nenhum outro sub-elemento então ele é chamado de “*Leaves*” (folhas). O nome adotado para a hierarquia inteira de uma estrutura em *XML* que represente o *LOM* é chamado de “*tree structure*”. As relações entre o *root*, *branches*, e *leaves* é representado na próxima figura:

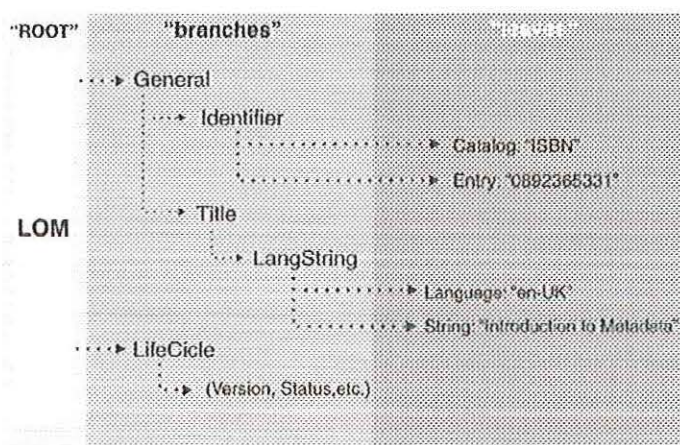


Figura 14: Visão hierárquica do metadado *LOM*

Fonte: *IMS-LOM (2004)*

Cada elemento na hierarquia do metadado possui uma definição, tipo de dado e valores específicos que podem ser inseridos na estrutura ou não, dependendo de sua necessidade, alguns elementos são obrigatórios definidos pela natureza do próprio objeto,

<sup>19</sup> Wisc-Online: disponível na internet em <<http://www.wisc-online.com/index.htm>>, acessado em 22 de jun. de 2004.

<sup>20</sup> VCILT Learning Objects Repository: Disponível em <<http://vcampus.uom.ac.mu/lor/index.php?menu=1>>. Acessado em 22 de jun. de 2004.

<sup>21</sup> Merlot: Disponível na internet em <<http://www.merlot.org/Home.po>>. Acessado em 22 de jun. de 2004.

<sup>22</sup> SMETE: Disponível na internet em <<http://www.smete.org/smete/>>. Acessado em 22 de jun. de 2004.

outros podem ser opcionais, no conjunto estes elementos definem as características da funcionalidade do objeto, como é o caso do elemento “*Educational*”, que possuem características educacionais. Um esquema mais detalhado do elemento “*Educational*” também pode ser visto no anexo D e E, alguns sub-elementos são apresentados no quadro a seguir:

Geral (General)	Ciclo de Vida (Life Cycle)	Técnicos (Technical)	Educacional (Educational)	Direitos (Rights)
<ul style="list-style-type: none"> <li>• Identificador</li> <li>• Título</li> <li>• Idioma</li> <li>• Descrição</li> <li>• Palavras-chave</li> </ul>	<ul style="list-style-type: none"> <li>• Versão</li> <li>• Status</li> <li>• Tipo de contribuição</li> <li>• Entidades que contribuíram</li> <li>• Data</li> </ul>	<ul style="list-style-type: none"> <li>• Formato</li> <li>• Tamanho</li> <li>• Localização</li> <li>• Tipo de Tecnologia</li> <li>• Nome da Tecnologia</li> <li>• Duração</li> </ul>	<ul style="list-style-type: none"> <li>• Tipo de interatividade</li> <li>• Recurso de aprendizagem</li> <li>• Nível de interatividade</li> <li>• Usuário final esperado</li> <li>• Ambiente de utilização</li> <li>• Faixa etária</li> <li>• Descrição</li> </ul>	<ul style="list-style-type: none"> <li>• Custo</li> <li>• Direito autoral</li> <li>• Condições de uso</li> </ul>

#### Quadro 2: Alguns exemplos dos elementos e sub-elementos do *LOM*

Os objetos são descritos segundo as especificações do *LOM*, a interação realizada entre o objeto de aprendizagem e o *LMS* é realizada através do ambiente de execução (*Run Time Environment*).

#### O ambiente de Execução – *RTE (RunTime Environment)*

O projeto *SCORM* define que os objetos de aprendizagem comunicam-se com o *LMS* através de *API's (Application Program Interface)*. Para que se entenda a forma de trabalho dos objetos de aprendizagem e a troca de mensagens através das *API's*, é necessário entender como o mesmo se relaciona em seu ambiente de execução.

O movimento de oferta e uso de um objeto de aprendizagem é conhecido como “*Launched*” e é oferecido ao usuário (segundo o padrão *SCORM*) através de três formas diferentes:

- *Content Package*
- *SCO*
- *Asset*

Na primeira o “*Content Package*”, é responsável pela seqüência de navegação que é pré-definida em sua estrutura, o que não acontecia nas versões anteriores ao *SCORM*, em



versões anteriores a estrutura de conteúdo e navegação não pertenciam ao “*Content Package*” e sim gerenciado pelo *LMS*.

Nas outras duas formas de oferta (*SCO* e *asset*), o *LMS* pode adaptar e determinar a seqüência conforme a necessidade do mesmo podendo ser uma navegação seqüencial, não seqüencial, especificado pelo usuário ou adaptativo dependendo da capacidade do *LMS*, na estrutura proposta (figura 24) será considerado e utilizado este tipo de oferta.

O *LMS* pode oferecer um menu que permite direcionar a navegação do usuário através de um “*Content Aggregation*” (figura 10). O menu pode aparecer como uma série de *hyperlinks*<sup>23</sup> que contém a localização dos recursos (objetos) de aprendizagem. Alternativamente o *LMS* pode conter um mecanismo de software que determina a adaptação da seqüência em que os recursos serão oferecidos dependendo da performance do aluno. Neste caso o “*Server-Side*”<sup>24</sup> se responsabiliza para a entrega do recurso. Quando o objeto é entregue, caso seja um *SCO* a comunicação é iniciada, se for apenas um *asset* o objeto será apenas oferecido. Na comunicação entre o *SCO* e o *LMS*, existem algumas regras que devem ser obedecidas pela normalização do projeto *SCORM*:

- Um *LMS* pode oferecer apenas um *SCO* para cada usuário ao mesmo tempo em uma *window* (janela do *Browser*);
- Só um *SCO* pode estar ativo ao mesmo tempo em uma *window*;
- Preferencialmente, um *SCO* não pode oferecer outros *SCO*;
- O *LMS* deve oferecer o *SCO* em uma janela dependente ou *frame* dependente;
- Apresentar o *SCO* como um modelo de objeto *DOM*<sup>25</sup>;
- É da responsabilidade do *SCO* procurar recursivamente e comunicar-se com o *LMS*.

<sup>23</sup> *Hiperlink* - A Web permitiu que cada documento da rede tenha um endereço único, que indica o nome do arquivo, diretório, nome do servidor e o método pelo qual ele deve ser requisitado.

<sup>24</sup> *Server-Side* – Do lado do servidor, ou seja são as *API*'s que ficam do lado do servidor e que oferecem os objetos de aprendizagem

<sup>25</sup> *DOM (Document Object Model)*: Trata-se do modelo de objetos (com suas propriedades e métodos) que são expostos ao programador de páginas dinâmicas em HTML. Através do envio de mensagens a estes objetos, o programador pode explorar a interatividade com o usuário.





Além de identificar também uma condição de erro de comunicação (caso ocorra) durante uma transação também as chamadas *API's* também recuperam dados. Pontuações, limites de tempo para a utilização do objeto, dados iniciais a serem apresentados, entre outros são exemplos de dados que podem ser trocados entre o *SCO* e o *LMS* através das *API's*. Estes dados são gravados e recuperados em uma estrutura de dados definida pelo projeto *SCORM* como *Data Model*.

O *Data Model* (modelo de dados) é utilizado para definir situações de execução de um *SCO* por um determinado aluno; pode recuperar dados para o *SCO* (caso necessite) no início de execução, onde o mesmo já possa ter sido utilizado pelo aluno (como, por exemplo, em uma avaliação interrompida).

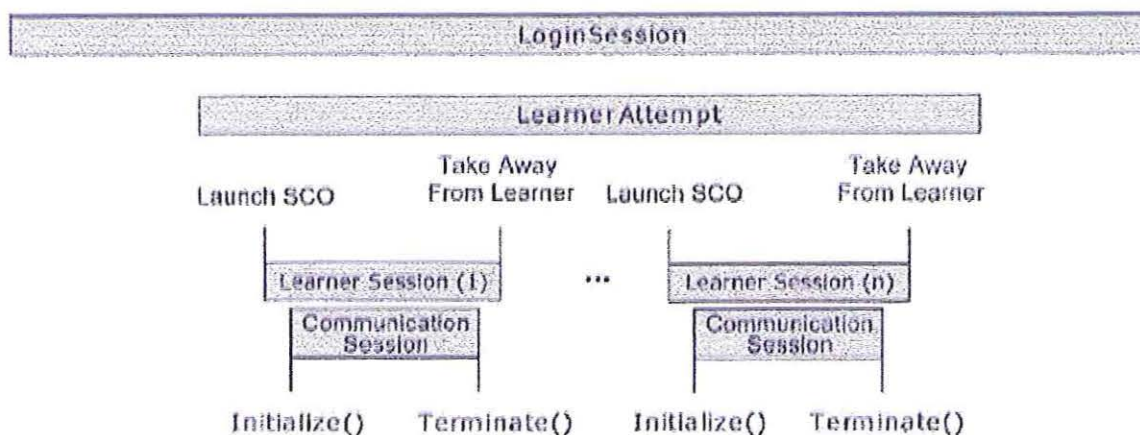
O modelo dos dados deve possuir elementos para assegurar a reutilização em diversas situações e através de múltiplos sistemas. A definição deste modelo de dados esta descrita nos anexos B e C.

O *IEEE* definiu um conjunto de serviços executados por uma rotina conhecida como *ECMAScript*, que contém serviços de comunicação no ambiente de execução (*RTS - RunTime Service*) entre o *SCO* e a *API*, O *RTS* é definido como um programa (normalmente codificado na linguagem *JavaScript*) que controla a execução de entrega de conteúdo e pode prover serviços como alocar recursos, enfileirar (*scheduling*), controle de entrada e saída de dados (*input-output*) enfim o gerenciamento entre o *SCO* e a *API*.

A comunicação é realizada através da sessão de *login*, a sessão de *login* é um período de tempo em que o aluno começa uma sessão com o sistema *LMS* até que ele termine esta sessão (*logout*).

O *Learner Attempt* começa quando ocorre uma atividade de acesso e entrega do objeto (*Launch*) que pode ser um *SCO* ou um simples *asset*, o objeto é acoplado ao *browser* e a sessão começa. Se for entregue um *SCO* o mesmo inicializa a sua comunicação com o *LMS* através da chamada da função *Initialize*. A sessão de comunicação termina quando o *SCO* executa a chamada da função *Terminate*. Se o aluno solicitou um *asset* simplesmente, o *Learner attempt* finaliza quando ele para de utilizar o objeto.





**Figura 16: Relação entre o SCO e a Sessão aberta**

**Fonte: ADL(2004)**

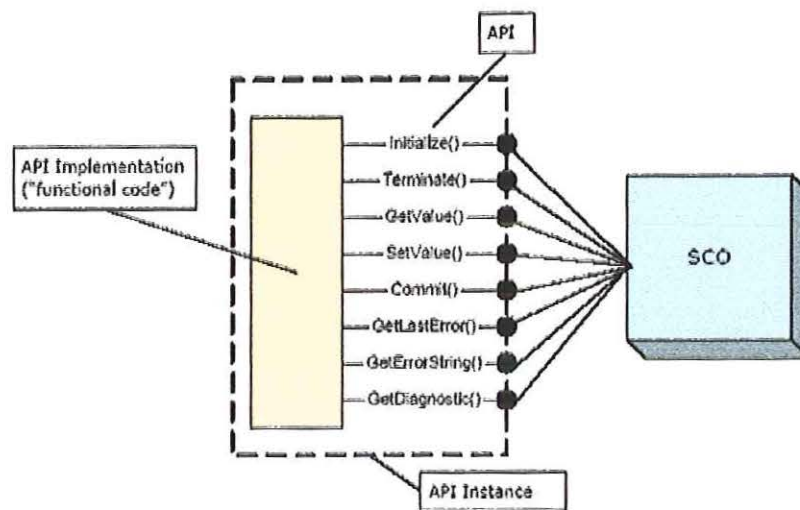
O objeto *SCO* comunica-se através de *API's* que podem buscar informações antes, durante e depois de apresentar o objeto de aprendizagem (estas informações estão presentes no modelo de dados). As categorias de dados (que já eram requisitadas na versão 1.2) do Data Model são:

- *Core*: Informações básicas para que o *SCO* poder interagir com o *LMS*
- *Launch Data*: Informações que serão usadas pelo *SCO* em inicialização
- *Objectives*: Identifica a performance de cada aluno relativamente aos objetivos de um determinado *SCO*.
- *Student Data*: Informação para suportar a customização de um *SCO* baseado na performance de um aluno ao utilizar este recurso.
- *Student Preference*: Preferência do aluno com relação à utilização do funcionamento do objeto de aprendizagem.
- *Interactions*: Uma interação é definida como um input ou grupo de inputs do aluno normalmente realiza em atividades de avaliação.

As categorias requisitadas já indicavam a preocupação do tratamento individualizado de cada aluno, o que facilitava a identificação do perfil individual, o que é possível através das chamadas das *API's*, estas eram divididas em três conjuntos que são:

- *Execution State (Initialize e Terminate)*
- *Data Transfer (GetValue, SetValue e Commit)*
- *State Management (GetLastError, GetErrorString e GetDiagnostic)*

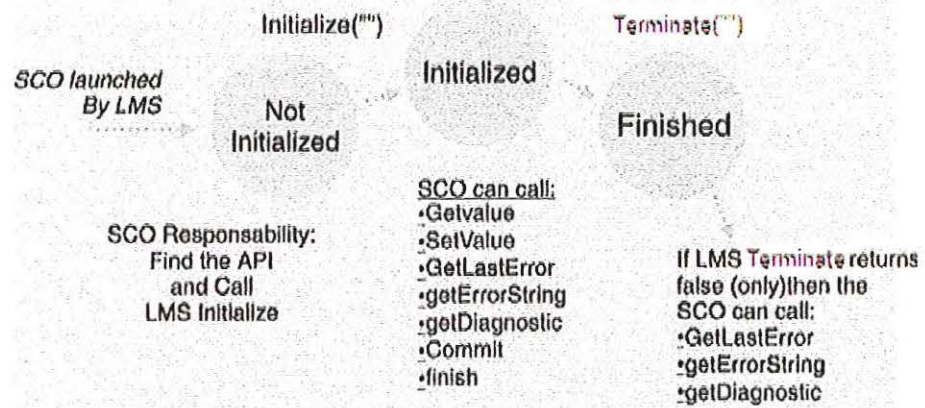




**Figura 17: Tipos de chamadas de um SCO**

**Fonte: ADL(2004)**

- *Execution State*: As funções *Initialize* e *Terminate* são obrigatórias em todos os *SCO*'s e indicam a que *SCO* pretende iniciar ou terminar, respectivamente, uma comunicação com o *LMS*. Para que um *SCO* seja considerado como tal, deve incluir pelo menos uma chamada a cada uma destas funções.
- *Data Transfer*: Para recuperar ou adicionar dados à base de dados do *LMS*, foram definidos três métodos, *GetValue*, *SetVaule* e *Commit* respectivamente. O *GetValue* têm como parâmetro qual o elemento do modelo de dados que se pretende acessar, o *SetValue*, quais os dados a inserir e seu conteúdo, o método *Commit* executa e finaliza a função.
- *State Management*: Este módulo de *API*'s contém as funcionalidades para confirmar o sucesso ou insucesso das chamadas aos métodos descritos anteriormente. São elas *GetLastError*, *GetErrorString* e *GetDiagnostic*.

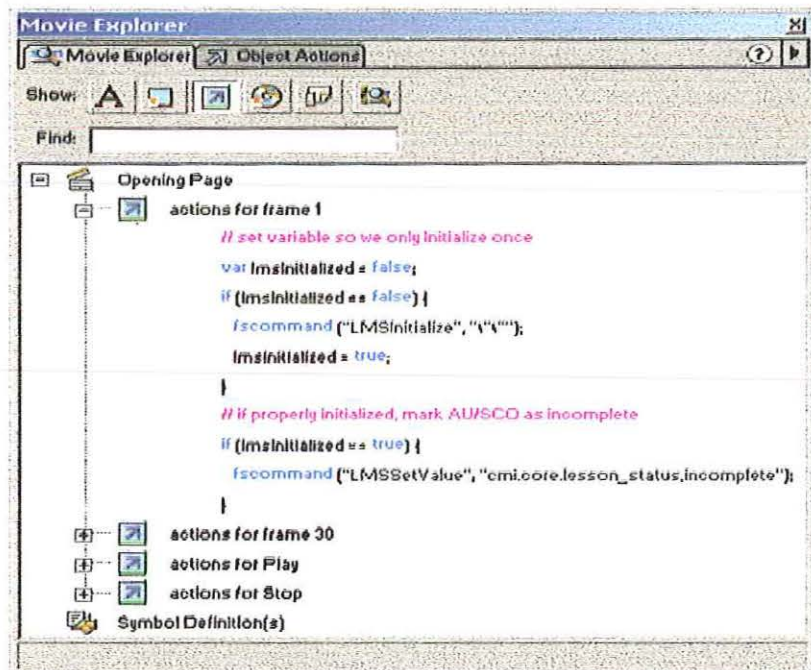


**Figura 18: Dinâmica da execução das chamadas pela API**

Fonte: ADL(2004)

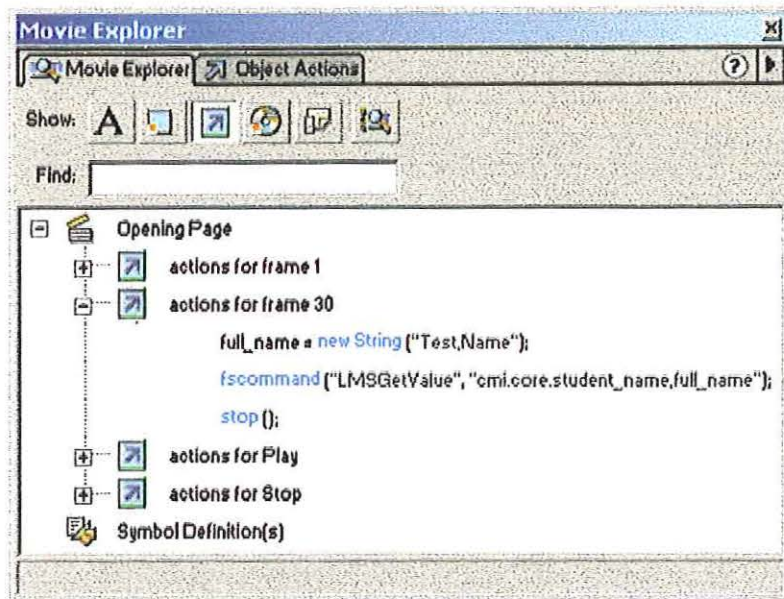
Já existem no mercado softwares que permitem a chamada as *API's* realizadas pelo *SCO* dentro do padrão *SCORM*, entre elas estão os produtos da Macromédia, maiores informações estão no apêndice B (identificado como Url-1).

Nas figuras 19,20 e 21 é exemplificada a programação das chamadas das rotinas através da ferramenta *Flash* versão 5 (que estão em conformidade com a versão 1.2 do projeto *SCORM*), a qual executa uma inicialização, consulta ao modelo de dados e a sua finalização respectivamente.

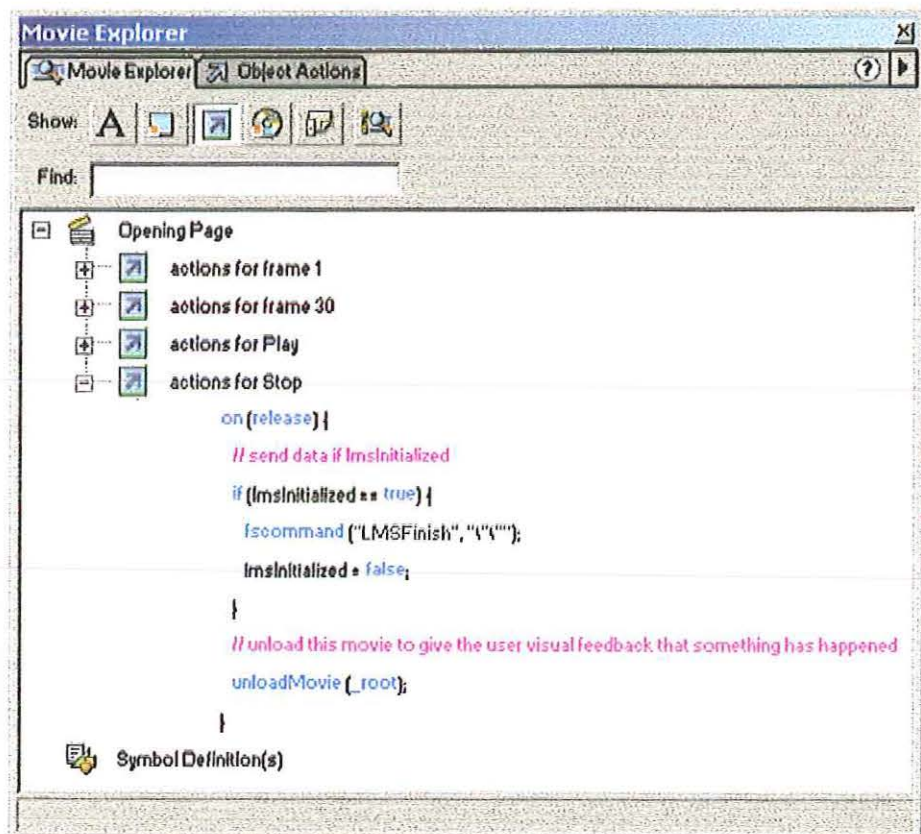


**Figura 19: Flash 5 – Aplicando a chamada *LMSInitialize* na API**

Fonte: Eduworks (2004)



**Figura 20: Flash5 – aplicando a chamada *LMSGeValue* na API**  
**Fonte: Eduworks (2004)**



**Figura 21: Flash5 – aplicando a chamada *LMSFinish* na API**  
**Fonte: Eduworks (2004)**



A nova versão de 2004 do SCORM implicou também algumas alterações no padrão das chamadas das API's da versão 1.2 para a versão 1.3.1, a seguir seguem estas mudanças:

- *LMSInitialize*("") trocou para *Initialize*("")
- *LMSFinish*("") trocou para *Terminate*("")
- *LMSGetValue*(parameter) trocou para *GetValue*(parameter)
- *LMSSetValue*(parameter\_1,parameter\_2) trocou para *SetValue*(parameter\_1,parameter\_2)
- *LMSCommit*("") trocou para *Commit*("")
- *LMSGetLastError*() trocou para *GetLastError*()
- *LMSGetErrorString*(parameter) trocou para *GetErrorString*(parameter)
- *LMSGetDiagnostic*(parameter) trocou para *GetDiagnostic*(parameter)

Através das interações realizadas entre o objeto, API's e o modelo de dados (*Model Data*), é possível acompanhar os objetos navegados e verificar o desempenho dos alunos nas avaliações, auto-avaliações, testes, exercícios, o que vai permitir construir a base de dados de informações dos estudantes proposta na arquitetura apresentada. Através da interação dos objetos de aprendizagem com agentes e sistemas tutores, será possível identificar a curva de comportamento dos alunos e identificação do perfil individual de cada um, oferecendo assim, material instrucional e treinamento (objetos de aprendizagem) adequado para a evolução de seu aprendizado, conforme os dados registrados pelas chamadas das rotinas API's. A importância do perfil já foi defendida nos capítulos dois. Os objetivos a serem alcançados pelo aluno e a seqüência dos objetos a serem acessados e utilizados embora sejam controlados agora pela versão 1.3.1 não será comentada, pelo fato da estrutura apresentada (figura 24) oferecer os objetos através de agentes inteligentes e sistemas tutores.

### 3.3.5 Serviço Web (*Web-Services*)

Quando a Internet surgiu as tecnologias presentes permitiam conectar-se e navegar em um site e receber arquivos baixar o conteúdo deste arquivo (*download*). A linguagem HTML (*Hiper Text Markup Language*) era a linguagem que permitia a apresentação da

informação presente na rede. Nos últimos anos, porém, novas tecnologias surgiram, permitindo maior integração entre os diversos aplicativos e serviços disponíveis na internet.

Um dos principais impulsionadores da interatividade das páginas web foi a possibilidade de interação entre estas e bases de dados, através de tecnologias de *Server Side Scripts*<sup>26</sup>, tais como o ASP, PHP, entre outras. Apesar destas linguagens provocarem o crescimento do mercado do *e-learning*, esta tecnologia levantou um problema de grandes dimensões para a interoperabilidade entre sistemas: os conteúdos que fazem uso das bases de dados de um determinado sistema só muito raramente podem ser reutilizados num sistema diferente. Mais uma vez, a normalização é uma questão fundamental para a interoperabilidade entre diferentes sistemas defendida pelo projeto *SCORM*.

A utilização de *Web-Services* pela estrutura proposta se dá pelo fato de que, a estrutura de comunicação dos *Web-Services* é baseada em *XML* o que vem de encontro com as características do projeto *SCORM* também possuir estruturas (*LOM* e do *Content Package*) em *XML*.

Segundo o *Web Services Interoperability (2004)*, consórcio mundial de padronização de *Web-Services*, os *Web-Services* são identificados por um único identificador de recurso conhecido como *URI (Unique Resource Identifier)*, e são descritos e definidos usando a estrutura da linguagem *XML*. Um dos motivos que tornam *Web Services* atrativos é o fato deste modelo padronizar serviços em sistemas que utilizam estruturas em *XML* e *http*, o que vem de encontro com o projeto *SCORM*, os *Web-Services* são usados para disponibilizar serviços interativos na web, podendo ser acessados por outras aplicações. A troca de mensagens entre os *Web-Services* é realizada através de um protocolo chamado *SOAP (Simple Object Access Protocol)* que é o protocolo padrão para a troca de mensagens entre aplicações e *Web-Services*, neste modelo, cada sistema que utiliza os *Web-Services* atua como um componente independente na arquitetura de integração. Todas as interfaces, transformações de dados e comunicações entre componentes são baseados em padrões abertos e vastamente adotados, independentes de fornecedores e plataformas.

---

<sup>26</sup> Serverside Scripts – Conjunto de linhas de programação que ficam contidas no servidor do sistema de informação



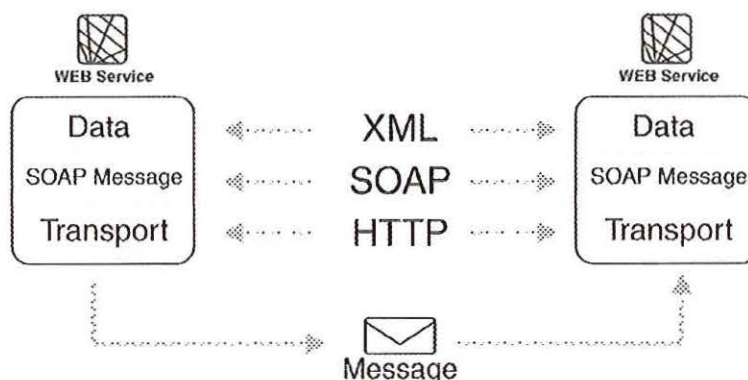
Segundo Alves (2004), para implementar *Web Service*, a W3C <sup>27</sup> (*World Wide Web Consortium*) definiu quatro componentes primários:

- *XML (eXtensible Markup Language)* - é uma linguagem padrão que especifica um formato de dados no qual todos os aplicativos que entendem *XML* devem comportar.
- *SOAP (Simple Object Access Protocol)* - é um protocolo utilizado para chamar procedimentos remotos pela internet. Ele é um documento *XML* que define, entre outras coisas, como uma aplicação pode dizer a um servidor que determinado objeto deve ser carregado, qual método deve ser executado, com quais parâmetros e qual é o valor de retorno.
- *WSDL (Web Services Description Language)* - é um documento *XML*. Serve de base para a descrição dos serviços. Proporcionam aos clientes de um *Web Service* todos os detalhes necessários que precisam para conectar e usar o *Web Service*.
- *UDDI (Universal Description, Discovery and Integration)* - *UDDI* é o diretório para *Web Services*. Para transformar a atual Web semi-estática numa Web dinâmica, dotada de serviços colaborativos, é necessário criar um diretório mundial para consultas dos serviços disponíveis, que as empresas divulgarão e procurarão serviços de forma dinâmica. O *UDDI* permite procurar os serviços em função do nome da empresa ou do tipo de atividade que exercem e indicam como negociar com as empresas que figuram no diretório, designando, por exemplo, os seus procedimentos comerciais ou descrevendo os serviços que disponibilizam. A estrutura do diretório *UDDI* possui um modelo de dados baseado em *XML*, e o seu acesso exige a utilização do *SOAP*, tanto para as preencher como para pesquisar os serviços que as contêm.

---

<sup>27</sup> W3C – Em 1994 foi criado um consórcio de empresas com o objetivo de desenvolver protocolos comuns para aumentar a interoperabilidade na internet.





**Figura 22: Diagrama da funcionalidade do Web-Service**  
**Fonte: (WEB SERVICES INTEROPERABILITY ORGANIZATION,2004)**

As mensagens dos *Web-Services* são transportadas através do protocolo *SOAP*, que é um protocolo que também foi projetado para chamar aplicações remotas através de *RPC* (*Remote Procedure Calls* ou Chamadas de Procedimento Remoto), em um ambiente independente de plataforma e linguagem de programação, desta forma, são garantidas a interoperabilidade e intercomunicação entre diferentes sistemas, através da utilização de uma linguagem (*XML*) e mecanismo de transporte (*HTTP*) padrões. O protocolo *SOAP* não foi aprofundado com maiores detalhes pelo fato de que a utilização de *Web-Services* no modelo da interface (figura 34) é apenas uma proposta, pois no projeto *SCORM* o meio de comunicação não é especificado. Uma mensagem *SOAP* consiste basicamente dos seguintes elementos:

- *Envelope*: Toda mensagem *SOAP* deve contê-lo. É o elemento raiz do documento *XML*. O *Envelope* pode conter declarações ou atributos adicionais que define o estilo de codificação que define como os dados são representados no documento *XML*.
- *Header*: É um cabeçalho opcional. Ele carrega informações adicionais, como por exemplo, se a mensagem deve ser processada por um determinado nó intermediário (é importante lembrar que, ao trafegar pela rede, a mensagem normalmente passa por diversos pontos intermediários, até alcançar o destino final). Quando utilizado, o *Header* deve ser o primeiro elemento do *Envelope*.

- *Body*: Este elemento é obrigatório e contém a informação a ser transportada para o seu destino final. O elemento *Body* pode conter um elemento opcional *Fault*, usado para carregar mensagens de status e erros retornadas pelos "nós" ao processarem a mensagem.



**Figura 23: Estrutura de uma mensagem SOAP.**

**Fonte: Cunha (2004)**

Entre outras utilizações, *SOAP* foi desenhado para encapsular e transportar chamadas de *RPC*, e para isto utiliza-se dos recursos e flexibilidade do *XML*, sob *HTTP*. *RPCs* ou *chamadas remotas de procedimento*, são chamadas locais a métodos de objetos (ou serviços) remotos. Portanto, pode-se acessar os serviços de um objeto localizado em um outro ponto da rede, através de uma chamada local a este objeto. Cada chamada ou requisição exige uma resposta.

Processo de uma chamada *RPC*, antes de ser enviada pela rede, é encapsulada (ou serializada) segundo o padrão *SOAP*. O serviço remoto, ao receber a mensagem faz o processo contrário, desencapsulando-a e extraindo as chamadas de método. A aplicação do servidor então processa essa chamada, e envia uma resposta ao cliente. O processo então se repete e a resposta é também encapsulada e enviada pela rede. Na máquina cliente, esta resposta é desencapsulada e é repassada para a aplicação cliente.

A especificação *SOAP* (definida pela *W3C*) define as seguintes informações, como necessárias em toda chamada de *RPC*:

- A *URI* do objeto alvo;
- O nome do método;
- Os parâmetros do método (requisição ou resposta);
- Uma assinatura do método opcional;
- Um cabeçalho (*header*) opcional.

## 4 ARQUITETURA PROPOSTA

Instituições de ensino, na maioria das vezes, administram cursos presenciais cuja estrutura do curso é composta por disciplinas (normalmente conhecida como grade curricular) e estas divididas em temas (conteúdo programático da disciplina). Esta estrutura representa uma hierarquia formal da evolução do aprendizado em que o curso deve, em prática, obedecer e desenvolver-se. Os alunos obedecem esta estrutura quando realizam matrículas nas disciplinas (respeitando os pré-requisitos caso existam). Neste ponto o sistema computacional operacional e administrativo, identifica as possíveis desconformidades das ações operadas no sistema pelos alunos e/ou funcionários.

Após o início da aula, o sistema computacional operacional e administrativo continua atuando, registrando na maioria das vezes, de forma ‘*on-line*’ a presença do aluno e do professor. Durante as aulas a abordagem dos temas (que normalmente obedecem ao conteúdo programático planejado) geralmente, é apenas anotado pelo professor em um diário de classe e em forma estática, e a partir deste ponto, a estrutura informacional dos sistemas operacionais administrativos se comportam de forma ‘*off-line*’ com relação à evolução do aprendizado, ficando esta tarefa de inteira responsabilidade do professor.

A possibilidade de a escola alcançar um número cada vez maior de alunos através da Internet trouxe um conjunto de modelos de sistemas computacionais que utilizam novas tecnologias que apóiam o treinamento e o aprendizado. Temas desenvolvidos de forma presencial ou virtual devem ser registrados, analisados e avaliados através das interações promovidas por estes sistemas, melhorando a identificação do perfil do aprendizado do aluno ou do grupo. Através destes dados coletados (que consiste a base de informações do estudante), poderão ser oferecidos *feedback's* em diversos momentos, possibilitando



identificar os resultados obtidos na exposição em sala de aula ou treinamento realizado, seja de forma presencial ou não.

O uso de dados levantados por sistemas de informações durante as interações, padronizam geograficamente e temporalmente a análise dos dados apoiando as decisões no processo educacional. Dados levantados durante estas interações ocorridas entre o aluno e o ambiente de aprendizagem virtual, facilitam a identificação do perfil individual e poderão demonstrar a conformidade das avaliações presenciais, desempenho do professor, da disciplina e do curso oferecido através do histórico de suas ações com o ambiente virtual proposto pela estrutura apresentada.

A arquitetura proposta promove a aderência do sistema acadêmico existente nas instituições de ensino com o propósito de utilizar os dispositivos tecnológicos apresentados no capítulo 3, estes recursos apóiam o desenvolvimento e aprendizado individual e a identificação de seu perfil individual, o que é possível através de *feedback's* da interação dos alunos com as tecnologias propostas, de maneira mais dinâmica e sustentável.

A inteligência do modelo é baseada na tecnologia de agentes inteligentes, os agentes presentes na estrutura são sugestivos e abstratos, o objetivo de apresentá-los dessa forma é demonstrar de como eles poderão trabalhar no conjunto. A proposta foi facilitar a criação de sistemas com habilidades de interação, onde estes agentes podem ser interpretados como componentes que se comunicam com seus pares por meio da troca de mensagens através de uma linguagem de comunicação de agentes, os sensores dos agentes podem ser ativados quando o tamanho da base de dados é alterada ou estímulos podem ser disparados na alteração de alguma linha ou registro das entidades apresentadas.

Os serviços (*Services*) presentes na estrutura têm como objetivo informar da necessidade de abstrair operações de dados (tarefa esta bastante lenta e custosa para a camada de agentes), propiciando uma independência dos agentes, com relação à base de dados a serem atualizados, além do que, os *Services* apóiam os agentes para se comunicarem de forma mais rápida e dinâmica.

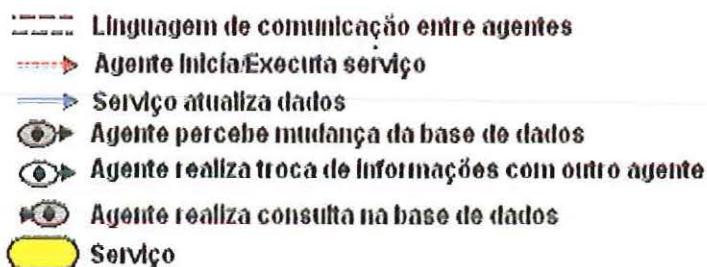
Optou-se em utilizar objetos de aprendizagem na construção da base de conhecimento, pelo fato de que já existe uma padronização reconhecida em termos de geração de conteúdo instrucional e sua capacidade de reutilização em âmbito global através da Internet, permitindo também uma uniformidade na construção da base do conhecimento

de domínio. A norma de utilização dessa tecnologia (projeto *SCORM*) já traz um conjunto de procedimentos padronizados que será considerada na funcionalidade do modelo proposto. Esta normalização do projeto *SCORM* que utiliza o metadado *LOM* vem colaborar para a interação com outras bases já desenvolvidas possibilitando um ambiente de crescimento rápido na coleção dos objetos de aprendizagem. Outra característica importante com relação à utilização de objetos de aprendizagem é a capacidade de os objetos se apresentarem de forma mais dinâmica, se aproximando do perfil individual do aluno, assim como registrar as interações realizadas, contribuindo para a identificação do perfil individual, onde a importância foi enfatizada no capítulo 2.

Como a exposição e utilização do material instrucional podem variar, o que possibilitará um ambiente moldável à velocidade e a evolução do aprendizado de cada aluno, a forma da estrutura dos objetos de aprendizagem respeitará o conceito de “*Resource Package Application*”, exposto na figura 10.

É apresentado, em cada módulo da estrutura proposta, seu objetivo, funcionalidade e necessidade dos componentes a serem desenvolvidos, isto foi realizado com a intenção de apresentar de forma mais clara o trabalho do conjunto.

Em cada módulo apresentado, é mostrada a relação entre as bases, serviços e agentes, que seguem a legenda a seguir:



No módulo da estrutura informacional formal é apresentada uma modelagem de dados para que possa ser contextualizada a aplicação da integração modelos da estrutura com os dados já existentes, além do que esclarece a relação entre elas.

Na questão em que a estrutura proposta é adaptativa, é reforçada em quatro situações possíveis:

- O modelo do projeto *SCORM* trabalha com o conceito de objetos de aprendizagem intercambiáveis, onde a estrutura de empacotamento e oferta



destes objetos são moldáveis conforme a necessidade da instituição de ensino.

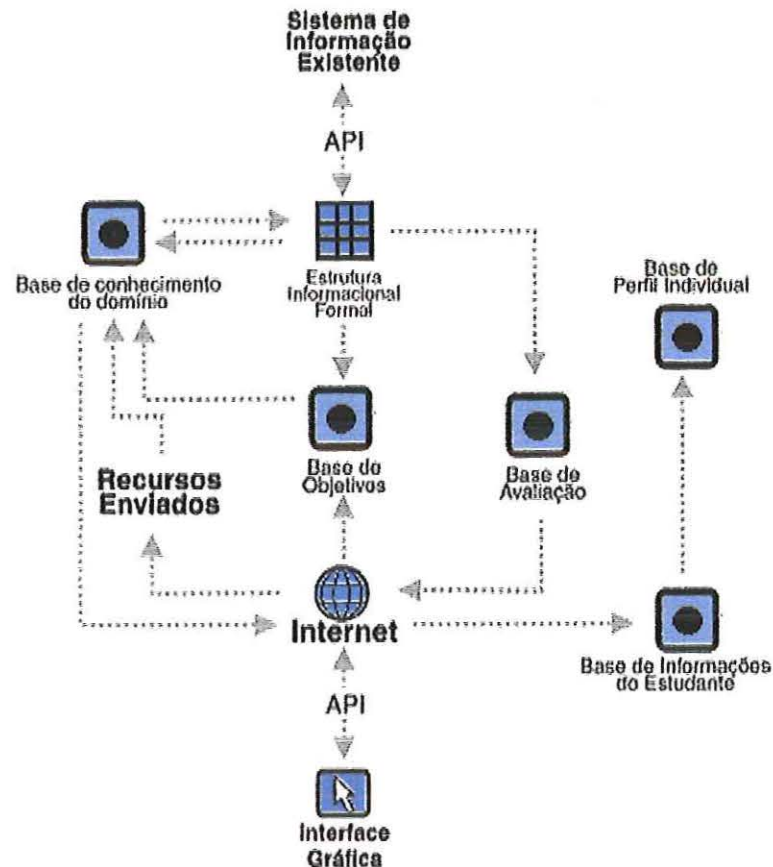
- Utilização de uma *API (Application Program Interface)* entre os sistemas de informações existentes que deverá importar os dados para uma estrutura modelada em banco de dados relacional proposta (Estrutura Informacional Formal). Caso a base de informações existentes esteja em um repositório de arquivos ou diretórios seqüenciais, deverá ser utilizada a opção de importação preferencialmente, pois o acesso a dados contidos em um banco de dados relacional é mais consistente, o que facilita a interação com o *Data Model* utilizado pelo *RTE (RunTime Environment)* do projeto *SCORM*.
- A modelagem da estrutura informacional formal pode se adaptar a entidades de um banco de dados relacional já existente, o que justifica a descrição do conteúdo das tabelas da modelagem da estrutura informacional formal, onde são identificados seus objetivos e funções.
- A *API* do *RTE* do projeto *SCORM* poderá acessar diretamente os sistemas de informações existentes sem utilizar a estrutura informacional formal, mas os sistemas de informações existentes devem obedecer de forma análoga a estrutura de dados das entidades expostas no modelo Informacional Formal, pois os agentes necessitam utilizar dados que obedecem as estruturas descritas na tabela, assim como será de responsabilidade da *API* buscar as informações no formato de arquivo dos sistemas de dados existentes da instituição de ensino.

A seguir é apresentada a proposta da integração das tecnologias apresentadas no capítulo 3 para a solução do problema apresentado no primeiro capítulo.

#### **4.1 Componentes da arquitetura**

Os componentes apresentados a seguir têm como objetivo elucidar o funcionamento global da arquitetura proposta e também para que seja entendida a relação do sistema de informação existente com o ambiente do sistema de gerenciamento do aprendizado (*LMS – Learning Management System*) utilizando as tecnologias apresentadas no capítulo 3.





**Figura 24: Componentes da arquitetura proposta**

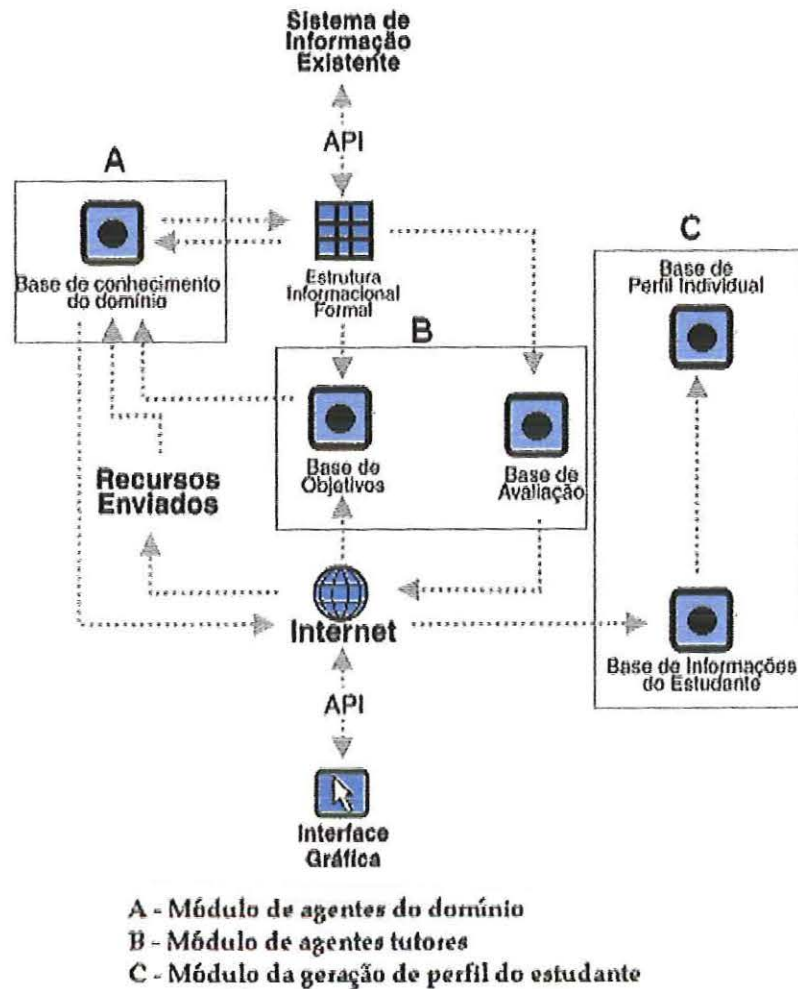
Para melhor entendimento da estrutura proposta, é identificado a seguir cada elemento que compõe a estrutura:

- Sistema de informação existente: sistema de informação operacional e/ou administrativo da instituição de ensino que irá interagir com a estrutura proposta.
- Estrutura informacional formal: é o *link* de aderência do sistema de informação existente com a estrutura proposta, será representado por uma modelagem para a contextualização da base de conhecimento do domínio.
- Base de conhecimento de domínio: será composta por objetos de aprendizagem (*Learning Objects*) que deverão explorar diversas mídias computacionais. Os objetos serão criados e utilizados tanto quanto maior for a decomposição da disciplina em temas.
- Recursos enviados: são materiais de conteúdo instrucionais enviados (pelo professor, aluno colaborador ou coordenador do programa instrucional) e agregados

à base de conhecimento, isto dependerá da permissão de um moderador de conteúdo instrucional.

- Base de objetivos: identifica os objetivos de aprendizagem do aluno, os objetivos podem ser formais, quando participam da estrutura do curso em que o aluno está matriculado ou informal, quando o objetivo não está relacionado com o curso freqüentado pelo aluno.
- Base de avaliação: contém objetos de aprendizagem de perfil avaliatório que sofrem influência dinâmica da estrutura informacional formal é constituída por objetos de aprendizagem do tipo avaliação.
- Base de perfil individual: têm como objetivo identificar o perfil do aluno. Isto é possível através do histórico das interações do aluno registrada na base de informações do aluno.
- Base de informações do aluno: contém o histórico das interações do aluno, possibilita a identificação do comportamento individual do aluno, os históricos são conseguidos através das funções executadas pela *API* do *RunTime Environment (SCORM)*. Conterá dados necessários para o levantamento das informações das interações do estudante, a estrutura dos dados das interações acompanhará quando necessário o modelo do *Data Model* (anexo B e C).
- Interface: O usuário interage através de um navegador internet obedecendo ao ambiente proposto do projeto *SCORM*.
- *API – Application Program Interface*, é um programa de interface que permite a comunicação entre dois ambientes, tem como objetivo executar funções necessárias de transposição de dados ou importação de dados de um ambiente para outro quando necessário.

Prado (2002) abordou o uso de agentes inteligentes utilizando os módulos do domínio, geração de perfil e agentes tutores, o que descreve de forma mais detalhada a construção de agentes inteligentes na abordagem estrutural (módulos), construção e funcionalidade. O modelo da estrutura apresentado por esta pesquisa também é compatível ao de Prado (2002), onde a adaptação é apresentada na figura seguinte:



**Figura 25: Adaptação do modelo apresentado ao modelo de Prado (2002).**

Os tópicos a seguir apresentam o objetivo de cada elemento da estrutura, assim como um ensaio de sua funcionalidade, no final de cada tópico é apresentada as camadas de softwares identificadas e necessárias.

## 4.2 Estrutura informacional formal

A estrutura informacional formal deve apoiar a funcionalidade do modelo proposto; deve ser constituída por componentes que possam sustentar a interação entre o sistema de informação existente na escola com a estrutura proposta.

As instituições de ensino possuem repositórios de dados em sua estrutura formal, que pode ser constituída de arquivos organizados e acessados em diversos formatos, entre eles a forma seqüencial, seqüencial e indexada, em banco de dados relacionais ou até mesmo banco de dados orientados a objetos. Para constituir uma flexibilidade de integração



(aderência) de menor impacto na utilização da estrutura formal, é apresentada uma *API* (*Application Program Interface*) que deverá ser composta por um conjunto de programas que irá importar e relacionar os dados da base de informação existente com os modelos das entidades relacionais apresentados na figura 26.

A funcionalidade da modelagem da figura 26 trabalha com uma estrutura de dados relacionados que no contexto deverá refletir o sistema de informação existente na instituição de ensino, esta estrutura de dados possui funcionalidades e relações entre si (e apóiam a arquitetura proposta), a descrição e objetivos de cada entidade está apresentada no quadro 3. O modelo de entidades e as relações do esquema funcional foram construídos pela metodologia de modelagem e entidade relacional (MER<sup>28</sup>), o que permite a visualização de forma geral o entendimento das relações entre as entidades. Isto não quer dizer que a proposta da estrutura informacional formal esteja rígida e “engessada”, ela poderá ser modificada dependendo da estrutura organizacional da escola para melhor representatividade do contexto.

A opção pelo MER parte do princípio de que a maioria dos sistemas de informações com foco operacional administrativo existentes nas instituições de ensino-aprendizagem utilizam base de dados (podendo ser relacionais ou não), a metodologia de entidade e relacionamento se apresenta pelo seu fácil entendimento e aderência na transposição dos dados existentes para a estrutura informacional formal da instituição de ensino. As características físicas dos atributos (tipo de dado) não estão sendo representadas no modelo pelo fato de que isto dependerá do sistema de informação já existente (que definirá o tipo e tamanho do dado).

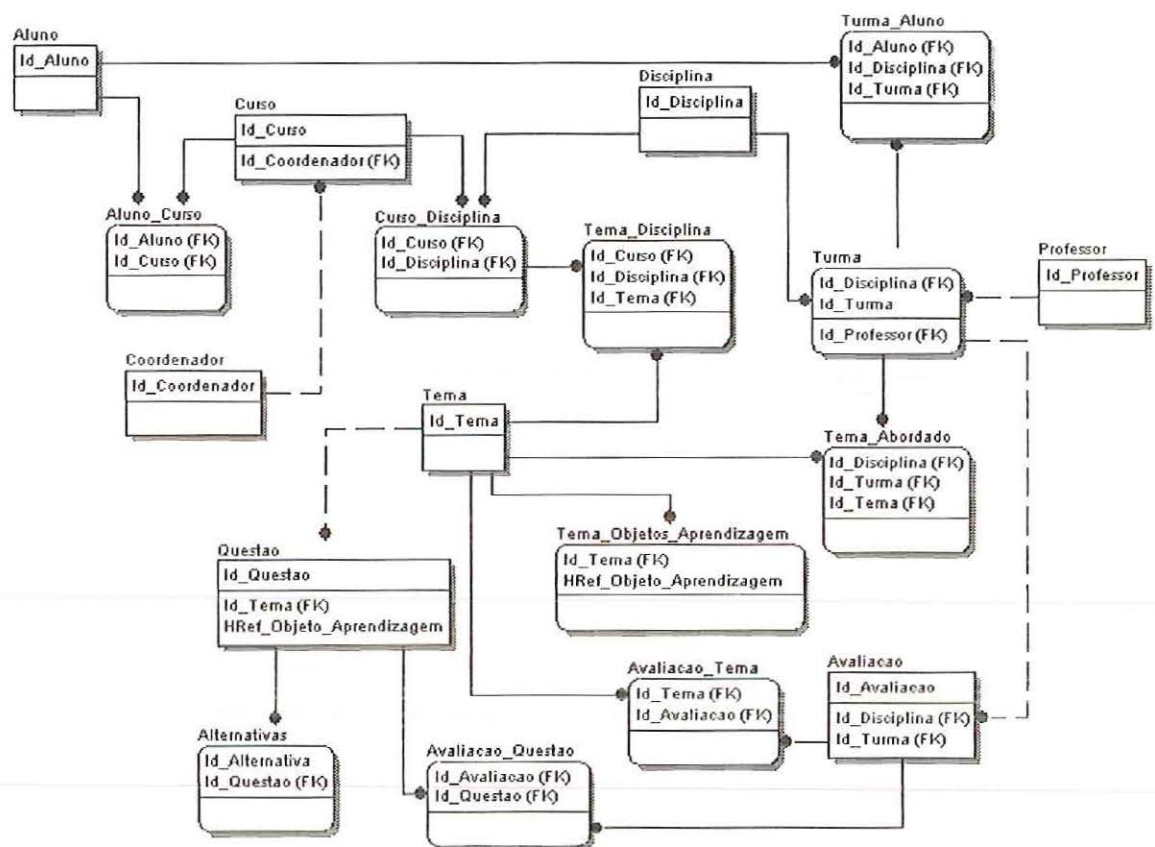
As entidades da modelagem foram organizadas para contextualizar a utilização de objetos de aprendizagem. Nesta modelagem, verifica-se a entidade “tema” é a que mais relacionamento possui, isto acontece por que os sistemas de informações com objetivos de

---

<sup>28</sup> MER- Modelo de Entidade e Relacionamento: muito utilizada atualmente para fins de representação de modelos de dados, que podem posteriormente ser implementados em Sistemas Gerenciadores de Banco de Dados de várias filosofias diferentes tais como relacionais, objeto, etc.

avaliação e treinamento deverão ter um tratamento granular do conteúdo das disciplinas para efetivo desempenho de treinamento, o que deverá vir de encontro com o programa proposto pela coordenação (ementa). Se as disciplinas apresentarem de forma granulada os seus conteúdos (temas) e as avaliações aplicadas com tecnologia também respeitarem esta abordagem, permitirá uma investigação refinada, delineando com mais riqueza nos detalhes o aproveitamento da disciplina.

A seguir é apresentada a figura do modelo da estrutura informacional formal.



**Figura 26: Modelagem da estrutura informacional formal**

As siglas *FK* (*Foreign Key* – chave estrangeira) presentes na modelagem identificam que o atributo da entidade (coluna) é herdado de entidade externa.

Os atributos (colunas) acima de cada linha dentro das entidades representam atributos de chave primária (*Primary Key*) e devem conter valores únicos obrigatoriamente, entidades com mais de um atributo acima da linha dentro de cada entidade são consideradas chaves primárias compostas.

As funções e características das entidades criadas na modelagem descrita acima estão apresentadas no quadro 3:

Entidade	Atributo	Objetivo	Observações
Coordenador	Id_Coordenador	Identifica o coordenador que receberá as informações ou alertas da estrutura proposta.	O serviço de autenticação será executado pelo <i>LMS</i> .
Curso	Id_Curso Id_Coordenador	Identificação do curso disponível para treinamento pela estrutura proposta.	Cada curso poderá ter apenas um coordenador, caso ocorra mais de um, deve-se criar uma nova entidade com relação de um (curso) para muitos (Coordenadores).
Aluno	Id_Aluno	Identificação do aluno que será treinado pela arquitetura proposta.	O serviço de autenticação será executado pelo <i>LMS</i> .
Aluno_Curso	Id_Aluno Id_Curso	Identifica o curso ou cursos que o aluno esta matriculado.	Através dessa informação poderá ser alimentado o módulo de objetivos formais, pois o curso possui uma grade curricular composta pelas disciplinas (Curso_Disciplina).
Disciplina	Id_Disciplina	Identificação da disciplina disponível para treinamento pela arquitetura proposta.	Disciplina de conteúdo programático diferente (domínio) deve ter Id_Disciplina diferente.



Professor	Id_Professor	Identificação do Professor que receberá o <i>feedback</i> da arquitetura proposta do desempenho dos alunos nos processos de treinamento e avaliação	O professor também poderá colaborar com novos conteúdos programáticos na arquitetura proposta, isto será possível através da utilização de <i>Web-Services</i> .
Tema	Id_Tema	Identificação dos temas que irão compor o programa temário das disciplinas oferecidas no curso.	Quanto mais temas existirem maior será a possibilidade da criação de objetos de aprendizagem, temas de conteúdo (domínio) diferente devem ter identificação diferentes
Tema_Objeto_Aprendizagem	Id_Tema HRef_Objeto_Aprendizagem	Irà conter o nome do <i>SCO (Sharable Content Object)</i> para cada tema.	O HRef_Objeto_Aprendizagem será alimentado através de serviços chamados por agentes inteligentes apresentado na figura 27.
Turma	Id_Disciplina Id_Turma Id_Professor	Identificação do Professor alocado em uma determinada turma de uma disciplina	A entidade turma não está relacionada com o curso pelo fato de que na contextualização abordada, as turmas podem ser compostas por alunos de diversos cursos.
Turma_Aluno	Id_Disciplina Id_Turma Id_Aluno	Identificação do Aluno alocado em uma determinada turma de uma disciplina	Informa para qual professor e coordenador serão encaminhados os resultados de desempenho da aprendizagem do aluno ou grupo de alunos .
Curso_disciplina	Id_Curso Id_Disciplina	Identifica qual a disciplina do curso.	Será possível identificar qual disciplina dentro de um determinado curso está disponível para treinamento pela arquitetura proposta

Tema_Disciplina	Id_Curso Id_Disciplina Id_Tema	Identifica o conteúdo temário da disciplina, identificando quais temas devem ser treinados e avaliados pelos alunos que frequentam a disciplina de um determinado curso.	Os temas a serem mais treinados pelo sistema tutor serão aqueles que apresentarem o menor desempenho na avaliação.
Tema_Abordado	Id_Disciplina Id_Turma Id_Tema	Identifica os temas abordados pelo professor na turma, o que irá fornecer informação para o treinamento pela arquitetura proposta.	Informa qual tema deve ser treinado pelo sistema tutor através de dados cruzados do aluno matriculado naquela turma e disciplina. A disciplina aqui faz parte dos objetivos formais.
Questão	Id_Questão	Identificação da questão disponível para treinamento pela estrutura proposta.	O formato da questão poderá apresentar um problema a ser resolvido por alternativas objetivas.
Alternativas	Id_Questão Id_Alternativa	Identificação da alternativa pertencente a uma questão disponível para treinamento.	A existência desta entidade é justificada para dar maior dinamismo do objeto de aprendizagem para a avaliação, sem a necessidade de reconstruí-lo quando alterar ou inserir algumas das alternativas
Avaliação	Id_Avaliação	Identificação da avaliação que será montada pelo professor ou coordenador, o aluno será convidado a participar da avaliação através de serviços de mensagem e alertas.	A avaliação será focada a objetivos formais. O objetivo é que as avaliações ocorram em um maior número de vezes possível, para aumentar a incidência de temas avaliados.
Avaliação_Questão	Id_Avaliação Id_Questão	Identificação da Avaliação e das questões que deverão ser aplicadas de forma estática e obrigatória, quando necessária.	A avaliação apresentará as questões que ainda não foram realizadas pelo aluno.

Avaliação_Tema	Id_Tema Id_Avaliação	Identificação da Avaliação e os temas que irão gerar as questões que irão compor a avaliação.	Seu objetivo é de dinamizar as avaliações abordando os temas que devem ser investigados, estes temas podem ser definidos pelo professor ou coordenador.
----------------	-------------------------	---	---

**Quadro 3: Identificação das entidades e atributos da estrutura informacional formal**

### 4.3 Base de conhecimento do domínio

#### Objetivo.

A base de conhecimento de domínio é constituída por objetos de aprendizagem (*Learning Objects*) obedecendo ao padrão do projeto *SCORM*, será composta por recursos (*SCO's ou asset's*) produzidos (ou utilizados de outras instituições) de acordo com as necessidades do domínio do conhecimento a ser ensinado e treinado, fazem parte desta necessidade informações de conteúdo, conjunto de recursos de treinamento, exercícios entre outros que apoiem a construção do conhecimento pelo aluno. Estes recursos são construídos a partir de conhecimentos de especialistas que geram fontes de informação e atividades (recursos) e que estimulam o aprendizado sobre o tema a ser abordado. O fato de se optar em organizar o conteúdo de treinamento utilizando a tecnologia de objetos de aprendizagem e a sua normalização obedecendo ao padrão do projeto *SCORM* é que ela traz vantagens apreciáveis:

- Os conteúdos criados podem informar ao aluno sobre o seu desempenho nos objetivos propostos por este mesmo conteúdo, através de questões formativas, minimizando o problema pedagógico gerado pela ausência do professor. Isto é possível por que os objetos de aprendizagem podem explicitar um tema, promover seu treinamento e verificar através da auto-avaliação se ocorreu seu entendimento, podendo registrar informações adversas no modelo de dados do aluno (*Data Model*) e na base de informações do estudante.



- Fácil reutilização de conteúdo criado, ao separar o que é conteúdo da estrutura formal, objetos de aprendizagem permitem maior flexibilidade na criação de conteúdos compostos pelo agrupamento destes objetos (através do *Content Packaging*) deste mesmo conteúdo, podendo-se criar blocos encaixados de objetos de aprendizagem (como em um jogo do tipo lego).
- Portabilidade de conteúdo entre diversas plataformas, o que possibilita uma abertura de utilização por outros sistemas de treinamento bem como o leque de escolha para a obtenção de conteúdo presentes em objetos localizados em outros repositórios que podem estar em outras instituições de ensino.
- Diminuem as dificuldades técnicas dos desenvolvedores de *e-learning*, uma vez que as especificações sobre a estrutura e os seus requisitos já estão estabelecidas pelo padrão do projeto *SCORM*.

A base de conhecimento proposta deverá conter elementos (objetos de aprendizagem) referentes a ementa da estrutura do curso representado pela entidade “tema\_objetos\_aprendizagem” (figura 26), ela será composta por objetos de aprendizagem que obedecem a relação estrutural do curso, pois a oferta de objetos aos alunos dependerá desta relação construída, constituindo assim um domínio de objetivos formais.

Disciplinas são constituídas de conteúdos programáticos, decompostos por temas de aprendizado e por conseqüência indicam a necessidade da criação de objetos de aprendizagem relacionados a seu conteúdo e treinamento, formando assim a base de conhecimento de domínio. Caso a disciplina não esteja decomposta por temas no sistema de informação existente será necessário contextualizá-la e inseri-la na entidade “tema\_disciplina” (figura 26).

Quanto maior a decomposição da disciplina em temas, maior será a possibilidade de criar e desenvolver objetos de aprendizagem; esta granularidade promove um maior refinamento da base de conhecimento. O trabalho é árduo, mas o conceito de objetos de aprendizagem (*Learning Objects*), permite que possam ser incorporados ou complementados com objetos presentes em repositórios de outras instituições de ensino, permitindo um trabalho colaborativo para a construção da base de conhecimento.

A base de conhecimento de domínio deve sustentar a estrutura formal da instituição de ensino e a base de objetivos pessoais do aluno, ou seja, deve haver objetos de

aprendizagem suficientes para sustentar os conteúdos programáticos das disciplinas oferecidas pelo curso (objetivo formal) e os objetivos pessoais (objetivos informais) que podem ser objetos formais que não pertencem a um critério de obrigatoriedade de aprendizagem.

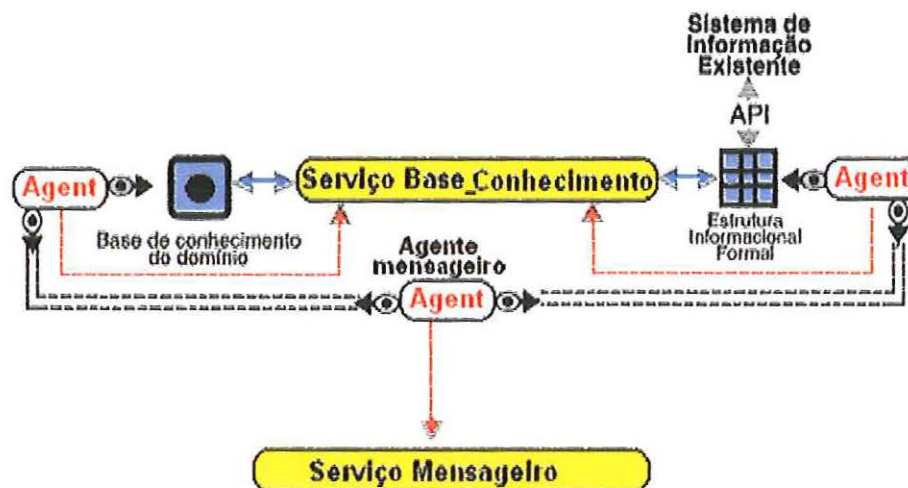
### **Funcionalidade.**

Quando ocorre a importação ou atualização dos dados do sistema de informações existente para o módulo da estrutura informacional formal (figura 26), é disparado um serviço chamado “Serviço Base\_Conhecimento”. Este serviço é disparado por um agente que percebe a manutenção (alteração, inclusão ou consulta) de um determinado tema que representa o conteúdo disciplinar. O objetivo deste serviço é de sincronizar os objetos de aprendizagem contidos na base de conhecimento de domínio com a estrutura informacional formal da instituição de ensino, assim como identificar para a estrutura informacional formal (tabela “tema\_objetos\_aprendizagem”) a referência dos recursos (objetos de aprendizagem) existentes para o treinamento.

A sincronização é conseguida quando se sabe quais são os objetos de aprendizagem existentes para um determinado tema, o serviço “Serviço Base\_Conhecimento” atualiza dados do atributo “Href\_Objeto\_Aprendizagem” da entidade “tema\_objetos\_aprendizagem”, que é a relação entre o tema e o objeto de aprendizagem. Os objetos de aprendizagem são pesquisados pelo sub-elemento *Keywords* do elemento *General* do *LOM* (presente na estrutura do objeto de aprendizagem apresentados no anexo D e E), o *LOM* possui uma estrutura descritiva em *XML* o que permite sua pesquisa, a quantidade de linhas a serem inseridas e ou atualizadas será a mesma quantidade de objetos que coincidirem com o conteúdo do *Keywords* e do atributo “Id\_Tema” da tabela “Tema” pesquisado. Da mesma maneira quando o agente da base de conhecimento se aperceber através de seus sensores a alguma manutenção (inclusão, exclusão ou alteração) dos objetos de aprendizagem da base de conhecimento de domínio, também iniciará o serviço, sincronizando todas as linhas da entidade “Tema\_Objeto\_Aprendizagem” que coincidirem pelo conteúdo da propriedade *Keywords* dos objetos de aprendizagem. Caso seja inserido um objeto de aprendizagem seja uma colaboração de um aluno, professor ou coordenador o agente mensageiro enviará uma mensagem ao moderador (a qual poderá ser assíncrona) pelo “Serviço Mensageiro”, da mesma maneira caso seja inserido um novo tema na



estrutura formal e não ocorra treinamento, será enviada uma mensagem para o coordenador do curso, o coordenador é identificado através do cruzamento de dados das entidades "Tema\_Disciplina", "Curso\_Disciplina" e "Curso".



**Figura 27: Sincronização da base de conhecimento de domínio e da estrutura informacional formal**

Após executar estas operações a Base de Conhecimento de domínio estará sempre sincronizada com a base da estrutura informacional formal da entidade de ensino.

Os tipos de recursos de aprendizagem estão divididos em diferentes grupos independentes, conforme o sub-elemento "*Learning Resource Type*" do elemento "*Educational*" do LOM, os tipos podem ser:

Learning Resource Type
<ul style="list-style-type: none"> <li>• exercise (Exercício)</li> <li>• simulation (Simulação)</li> <li>• diagram (Diagrama)</li> <li>• figure (Figura)</li> <li>• graph (Gráfico)</li> <li>• index (Índice)</li> <li>• slide</li> <li>• table (Tabela)</li> <li>• narrative text (Texto narrativo)</li> <li>• experiment (Experimento)</li> <li>• problem statement (declaração de um problema)</li> <li>• lecture (leitura)</li> </ul>

**Quadro 4: Tipo de Recurso**



Outras características informativas do objeto de aprendizagem com relação à funcionalidade e interação do mesmo são apresentadas abaixo (características também padronizadas pelo metadado *LOM*):

Descrição da característica	Valores atribuídos
<b>interactive type</b>	<ul style="list-style-type: none"> <li>• active</li> <li>• expositive</li> <li>• mixed</li> </ul>
<b>Interactivity Level</b>	<ul style="list-style-type: none"> <li>• very low</li> <li>• low</li> <li>• medium</li> <li>• high</li> <li>• very high</li> </ul>
<b>Semantic Density</b>	<ul style="list-style-type: none"> <li>• very low</li> <li>• low</li> <li>• medium</li> <li>• high</li> <li>• very high</li> </ul>
<b>Intended End User Role</b>	<ul style="list-style-type: none"> <li>• teacher</li> <li>• author</li> <li>• learner</li> <li>• manager</li> </ul>
<b>Difficulty</b>	<ul style="list-style-type: none"> <li>• very easy</li> <li>• easy</li> <li>• medium</li> <li>• difficult</li> <li>• very difficult</li> </ul>

**Quadro 5: Outras características do objeto de aprendizagem com relação à funcionalidade e interação.**

#### **Camada de Softwares abstratos a serem desenvolvidos:**

##### *Agentes (Agents)*

- Agente da base de conhecimento do domínio
- Agente da estrutura informacional formal
- Agente mensageiro

##### *Serviços (Services)*

- Serviço Base\_Conhecimento
- Serviço Mensageiro

#### 4.4 Base de objetivos

##### **Objetivo.**

A base de objetivos contém o plano de aprendizagem formal e informal a ser oferecido ao aluno, é composta por temas a serem treinados, os temas são identificados através da estrutura informacional formal, pois, esta inerente nesta estrutura a grade curricular decomposta em temas (figura 26), o que provém os objetivos de treinamento do aluno.

Através disso será possível dar orientação ao aluno de antecipar algum movimento de material instrucional para possibilitar em sala de aula momentos de reflexões proposta no capítulo dois, também a base de objetivos será alimentada pelo aluno através de oferta de temas que o aluno pode explorar, mas que não estejam em situação de obrigatoriedade da grade formal do curso em que o aluno está freqüentando.

##### **Funcionalidade.**

Os objetivos formais somente serão construídos pelo agente tutor inteligente caso existam objetos de treinamento referenciados a estes objetivos (treinamento para os temas formais), os temas são selecionados através do cruzamento de informações da tabela “Aluno”, “Aluno\_Curso”, “Curso\_Disciplina”, “Tema\_Disciplina” e “Tema\_Objetos\_Aprendizagem” (Figura 26).

O agente da estrutura informacional formal, através de seus sensores percebe quando ocorre alguma alteração na entidade “Tema\_Disciplina” e informa ao agente de planejamento de ensino (agente tutor inteligente) a necessidade de construir o planejamento de ensino ou alterá-lo (talvez reconstruí-lo) isto é possível, pois o agente da estrutura informacional formal irá cruzar informações identificando quais objetivos formais (Temas) de que alunos sofreram alteração e que devem ser alterados, o agente da estrutura formal informa ao agente planejamento de ensino o aluno (ou uma lista de alunos) e o tema (ou temas), que deverão ser trabalhados pelo agente.

O Agente tutor de planejamento de ensino encaminha dados (código do aluno e a identificação do Tema) ao agente da base de informações do estudante e cruza as informações da tabela “Tema\_Objetos\_Aprendizagem”, verificando se o aluno realizou interação de sucesso na base de informações do aluno, caso o aluno tenha baixo rendimento ou nunca interagiu com objetos deste tema, o agente passa para o serviço

“Base\_Objetivos” código do aluno e tema (“Id\_Aluno” e “Id\_Tema” ) a serem treinados e avaliados.

A existência dos objetivos informais ocorre, quando, o aluno seleciona treinamento em uma disciplina ou tema que não tenha relação formal com os cursos em que esteja matriculado, ou seja não há obrigatoriedade do treinamento. O acesso a esse treinamento dependerá da política da instituição de ensino de autorizá-lo ou não. O raciocínio de treinamento a ser aplicado pela estrutura é o mesmo raciocínio do treinamento formal.

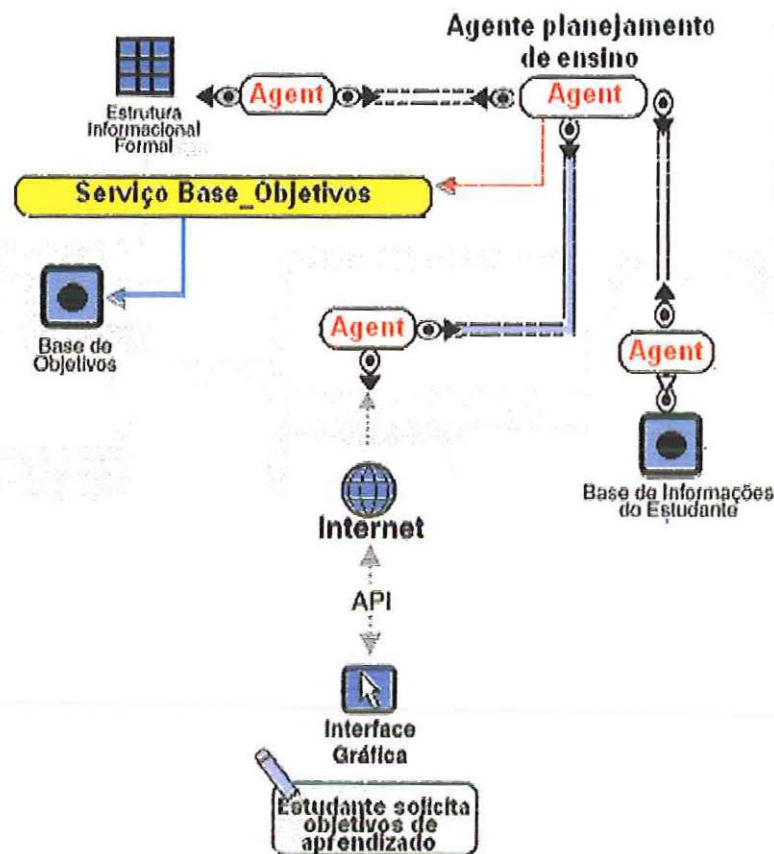


Figura 28: Formação da Base de Objetivos.

#### Camada de Softwares abstratos a serem desenvolvidos:

##### Agentes (*Agents*)

- Agente planejamento de ensino
- Agente Base de Informações do aluno
- Agente Internet

##### Serviços (*Services*)

- Serviço base de objetivos



## 4.5 Base de informações do Estudante

### **Objetivo.**

É a base que irá conter todas as informações do processo instrucional e evolução do treinamento e aprendizado constatado durante a interação do aluno com os objetos de aprendizagem. Quando o aluno interage com o objeto de aprendizagem, o objeto de aprendizagem realiza várias chamadas de *API's* para ler e gravar dados (*GetValue()* e *SetValue()*) no modelo de dados do aluno (*Data Model*), estas funções são executadas conforme o ambiente *RTE* descrito no tópico 3.3.4.1, onde através da estrutura do modelo do aluno (anexo C) irão contribuir para a formação da base de informações do estudante, a possibilidade de dados levantados estão descritos no anexo B, o que sustenta uma quantidade de elementos interessantes para o levantamento do perfil do aluno, necessidade proposta no capítulo 1 e 2. A estrutura dos dados gravados na base de informações do estudante dependerá do interesse do projetista em definir a granularidade do perfil que deseja identificar.

O modelo de dados do aluno pode conter informações referentes desde a performance do aluno ao utilizar o objeto como o modo em que o objeto deverá se apresentar ao aluno e em que condições de reinicialização caso tenha sido interrompido. A identificação do aluno e a identificação do objeto utilizado assim como outros dados de interesse devem ser registrados na base de informações do estudante

Em resumo a base de informações do estudante será o conjunto de dados necessários para a identificação do comportamento e perfil do aluno, o refinamento do perfil dependerá dos dados registrados nesta base.

### **Funcionalidade.**

A sessão é aberta quando ocorre o *login* do aluno e finaliza quando o mesmo sai do *LMS*. A base de informações do estudante é alimentada durante a interação do aluno com o objeto de aprendizagem na execução das *API's* que possuem nomes de chamadas padronizadas segundo o projeto *SCORM*, conforme ocorrem as chamadas *API's* que alteram o *Data Model*, serão gravados dados na base de informações do estudante (dados

necessários para identificação do perfil individual de treinamento e aprendizagem). Os objetos de aprendizagem apresentados serão decididos pelo *LMS* através do agente tutor, que cruza informações da base de objetivos, dados da base de perfil individual e dos objetos de aprendizagem, o agente seleciona um objeto ou conjunto de objetos de aprendizagem e oferece ao aluno os recursos (*SCO's* ou *asset's*) para serem utilizados pelo estudante, a lista de objetos oferecidos é realizada pelo Serviço Lista\_Sco, que tem como objetivo preparar um conjunto de códigos em linguagem *HTML* para oferecer ao estudante, o acesso é realizado através de *HyperLink's* presentes no *Browser* ao estudante. Os dados a serem gravados na base de informações do estudante devem ser executados pelo serviço Serviço Base\_Informação (figura 30), que deverá gravar os dados preferencialmente na formatação do *XML* devido a sua flexibilidade em separar estrutura de dados.

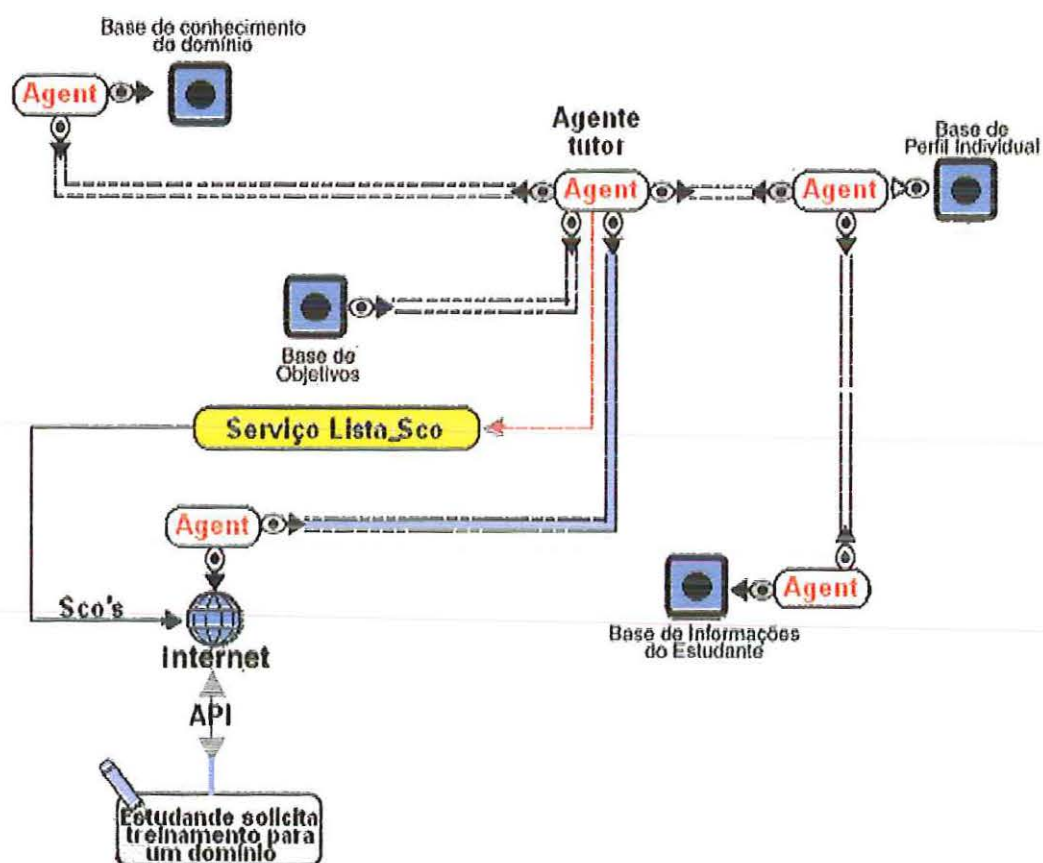


Figura 29: Solicitação de Treinamento para um ou mais Domínio

A evolução do aprendizado poderá ser também identificada através de sistemas especialistas que podem cruzar dados da base de informações do estudante (histórico) e os temas abordados da tabela “Turma”, “Disciplina\_Tema”, “Tema”, do modelo informacional formal. A opção em utilizar este tipo de tecnologia deve-se a necessidade de construção de uma métrica que permita identificar amostras de comportamento de aprendizagem, trabalhando-se com médias, desvios padrões, qualidade da amostra entre outras técnicas que poderão ajudar a identificar a evolução do aprendizado.

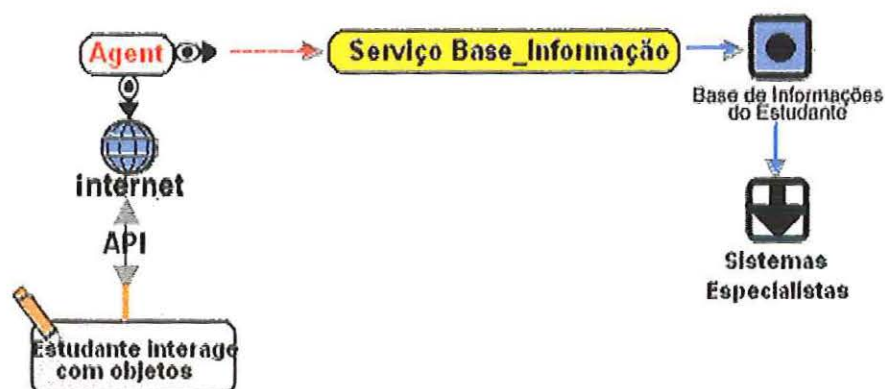


Figura 30: Os objetos são apresentados ou interagidos pelo aluno através das API 's

### Softwares abstratos a serem desenvolvidos

#### Agentes (Agents)

- Agente tutor
- Agente base de informações do estudante
- Agente internet

#### Serviços (Service)

- Serviço Lista\_Sco
- Serviço Base\_Informação

#### Sistemas

- Sistema especialista para métrica da evolução do treinamento e aprendizado

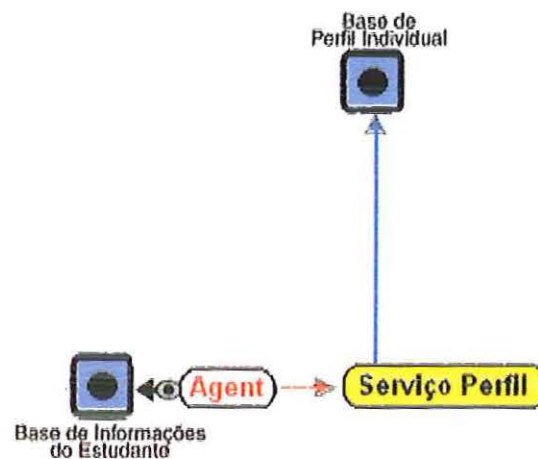
## 4.6 Base de perfil individual

### Objetivo.

Será construída através dos dados presentes na base de informações do estudante, os dados são armazenados conforme os tipos de exercício utilizado pelo aluno. O perfil do



aluno poderá se basear na escolha do objeto de aprendizagem de maior dificuldade ou de menor dificuldade, por exemplo, do tipo de formato da mídia, dos temas abordados com maior sucesso e menor sucesso características estas que podem ser identificadas através dos atributos do modelo *LOM*. Além do que, posições psicológicas e sociais que podem ser levantadas através de questionários aplicados e outras mais relacionadas à quantidade de interações realizadas. Outras informações poderão ser sintetizadas através do comportamento do histórico do aluno presente na base de informações do estudante. Conforme a necessidade de se identificar o perfil influenciará a estrutura da informação presente na base de informações do estudante.



**Figura 31: Atualização da base de perfil individual**

### **Funcionalidade.**

Dados gravados neste módulo são conteúdos resumidos da base de informação do estudante, seus dados são atualizados conforme ocorrem interações na base de informações do estudante. Quando ocorre a interação do estudante com os objetos de aprendizagem, as funções API's que ficam do lado do servidor ou gatilhos (*triggers*) programados presentes no modelo de dados do aluno (*Data Model*), poderão alterar ou incluir dados na base de informações do estudante. O agente da base de informações do estudante percebe a alteração ou inclusão dos dados e atualiza a base de perfil individual, conforme o interesse do perfil a ser detectado. Diversas informações da estrutura do *LOM* e do *Data Model*

poderão contribuir na identificação do perfil do estudante, como o “learner preference” , “Progress Measure” entre outros dados presentes no anexo B.

#### **Softwares abstratos a serem desenvolvidos.**

##### *Agentes (Agents)*

- Agente base de informações do aluno

##### *Serviços (Service)*

- Serviço Perfil

## **4.7 Base de avaliação**

### **Objetivo**

O modelo da avaliação formal tem como objetivo identificar a performance de aprendizado do aluno obtido nas aulas presenciais e/ou virtuais. O modelo de avaliação tem como objetivo identificar se o aluno adquiriu o conhecimento, ou seja, avaliar o conhecimento construído através das aulas interagidas com o professor, seja ela de forma virtual ou presencial.

Na estrutura proposta, o modelo de avaliação é composto também por objetos de aprendizagem. A base de avaliação é separada da base de conhecimento de domínio porque os objetos contidos têm objetivos avaliatórios o que devem ficar em um repositório privativo e não público. A avaliação deverá possuir uma dinâmica individual, ou seja, será diferente para cada aluno, esta dinâmica individual é possível, pois o *LMS* deverá respeitar os temas abordados em sala de aula e os objetos de avaliação já utilizados pelo aluno, constituindo uma variedade no formato em sua aplicação.

A interação poderá ser através de uma avaliação do tipo exercício que apóia a identificação do aproveitamento expositivo ou prático realizado pelo professor em sala de aula, permitindo assim um conjunto de informações que respaldam um *feedback* bastante detalhado dos temas abordados. O resultado das avaliações será registrado na base de informações do estudante, o fato de ser uma avaliação formal não quer dizer que ela substitua a avaliação presencial, observando que a portaria do MEC a proíbe de se comportar com esta intenção (o que já foi discutido no item 2.2 do capítulo 2), mas é uma ferramenta importante na identificação da eficácia da metodologia do professor já que ela pode ocorrer em muito maior número de vezes e em intervalos de menor tempo.



No modelo proposto os objetos de avaliação são construídos e oferecidos aos alunos através de agentes de avaliação. Como as questões e alternativas estão referenciadas aos temas abordados em sala de aula (entidades “Questão” e “Tema\_Abordado” presentes na figura 26) será possível estabelecer esta dinâmica, além do que, também será possível ao agente de avaliação construir um conjunto de avaliações que abordem temas específicos (entidade “Avaliação\_Tema”).

### **Funcionalidade.**

O agente observa a mudança da estrutura formal na entidade “Avaliação” (figura 26), que indica qual a turma e disciplina deverá ser avaliada, o agente reconhece quais os temas abordados na turma e quais alunos pertencem a esta turma e cruza informações na entidade “Questão” identificando quais são as questões disponíveis (estas informações estão presentes nas tabelas “Tema\_Abordado”, “Turma\_Aluno” e “Questão” respectivamente na figura 26).

Após selecionar as questões, o agente pesquisa na base de informações do aluno se o aluno errou anteriormente a questão e tenta encontrar uma questão relacionada ao tema para aplicá-la talvez de forma diferente. Caso o aluno tenha acertado a questão do tema abordado tenta encontrar a questão que tenha o maior nível de dificuldade para este tema. Após isto, o agente envia uma mensagem ao aluno convidando-o para a avaliação, quando ocorrer a interação, o agente irá armazenar os dados na base de informações do estudante, o que subsidiará dados para uma próxima avaliação.

Estes passos são seguidos para tentar identificar o aprendizado e perfil tanto do aluno quanto do professor, identificando a necessidade de cada indivíduo tanto do professor e sua metodologia quanto do aluno em seus esforços demonstrado no ambiente de aprendizagem, tentando identificar a resposta de entendimento através da avaliação, complementando uma avaliação somativa e diagnóstica do material oferecido e exposto em sala de aula.



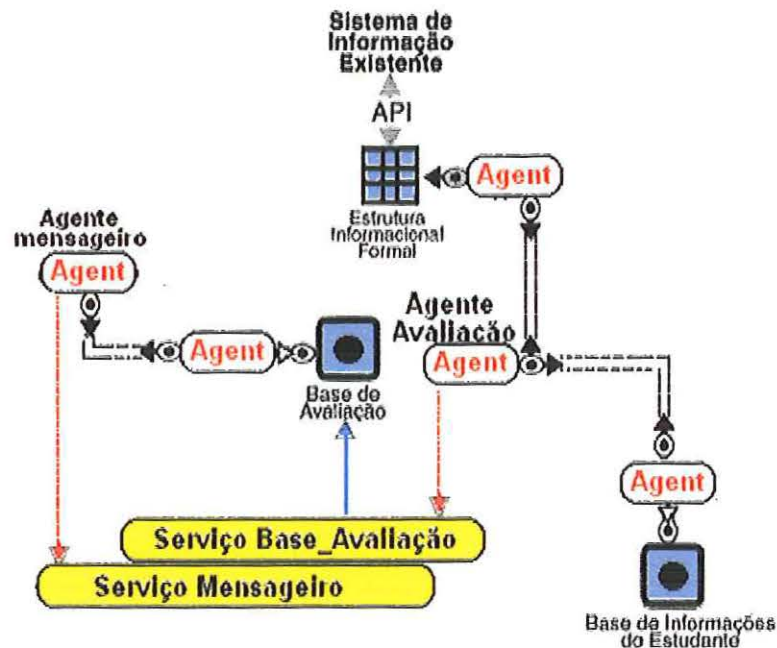


Figura 32: Atualização da base de avaliação

A interação da avaliação com o aluno será tratada da mesma forma quando ocorre uma interação de treinamento, ou seja, a performance da prova será gravada na base de informações do estudante (figura 30).

#### Camada de Softwares abstratos a serem desenvolvidos

##### Agentes (*Agents*)

- Agente base de avaliação

##### Serviços (*Service*)

- Serviço Base\_Avaliação
- Serviço Mensageiro

## 4.8 Recursos enviados

### Objetivo.

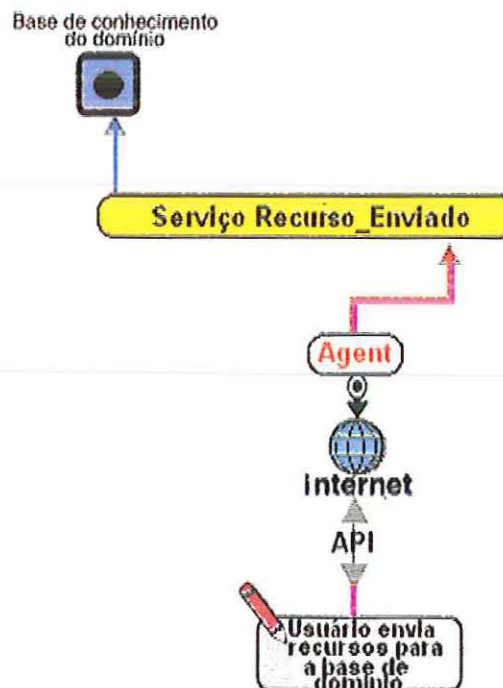
Seu objetivo é acrescentar na base de conhecimento recursos enviados pelo aluno ou professor que possam ajudar no desenvolvimento, contexto e elucidação do aprendizado. Professores podem adicionar novos conteúdos (novos objetos de aprendizagem) à base de conhecimento. Os alunos trocam informações entre si podendo gerar novo conhecimento,

assim como alguns já possuem um conhecimento mais aprofundado no tema e colaborarem com suas opiniões. Todo este contexto foi elucidado no paradigma baseado em tecnologia na figura 3, onde vem justificar o propósito da existência dos elementos e serviços apresentados a seguir.

### Funcionalidade.

Nem todo conteúdo pode ser aproveitado, por isso antes de ser utilizado por elementos da estrutura, há uma necessidade de passar por uma autorização de um moderador. Isto ocorre quando o conteúdo é inserido na base de conhecimento e é acionado o serviço mensageiro.

É importante evidenciar que as informações enviadas para participar da base de conhecimento preferencialmente devem utilizar um serviço de *Web-Service* (funcionalidade descrita no tópico interface), pois utiliza conceitos estruturais do *XML*. Quando o agente percebe que um conteúdo está sendo enviado dispara o serviço “Recurso\_Enviado” o que irá construir o objeto de aprendizagem na base de conhecimento em um repositório (se o conteúdo foi originado por um aluno deverá ser criado em um diretório de dados para ser liberado pelo moderador) em momento oportuno.



**Figura 33: Envio de recurso (conteúdo) a Base de Conhecimento**

**Camada de software a ser desenvolvida.**  
Serviço (*Services*).

- Serviço Recurso\_Enviado

## 4.9 Interface

### Objetivo

A interação com o modelo apresentado é realizada através de um *browser* (navegador da internet), o qual poderá chamar funções para a interação com os objetos de aprendizagem ou chamar *Web-Service* para inserção de material na base de conhecimento de domínio pelo aluno ou professor. A interface, portanto irá se relacionar com o modelo proposto através de duas maneiras:

- utilizando *API's* proposta pelo projeto *SCORM* ou
- Chamando um *Web-Service*.

A *API* é acionada quando a interface precisa interagir com objetos de aprendizagem através da estrutura proposta (durante uma interação de aprendizado, treinamento ou avaliação). A chamada as *API's* possui uma dinâmica de execução descrita pelo *RTE (RunTime Environment) Client Side*<sup>29</sup> atualiza os dados do *Data Model* e da base de informações do estudante; o movimento de interação (chamadas das *API's*) entre o objeto de aprendizagem esta descrito em um teste de ensaio apresentado no anexo G.

O agente da interface é acionado quando recebe uma mensagem ou alerta realizado através do serviço “Mensageiro” (Mensagem enviada pela arquitetura proposta), constituindo um serviço de alerta ao usuário final, estes serviços de alerta, por exemplo, informam ao usuário quando o mesmo é convidado para uma avaliação.

*Web-Services* são invocados quando o usuário quer contribuir com algum documento ou objeto que será enviado ao repositório da base de conhecimento de domínio, onde a estrutura do objeto a ser enviada deve ser em formato de manifesto (Figura 11) e estruturado em *XML*.

### Funcionalidade.

<sup>29</sup> *RTE (RunTime Environment) Client Side* – *API's* padronizadas pelo projeto *SCORM*, elas obedecem a funcionalidade descrita na figura 15.



Quando o objeto de aprendizagem necessita interagir com o sistema são executadas as chamadas as *API's* estabelecidas pelo padrão *SCORM*, que podem ser de três tipos:

- *Execution State*
- *Data Transfer*
- *State Management*

As características da funcionalidade desses tipos e do ambiente de execução dos objetos de aprendizagem já foram investigadas no tópico 3.3.4 capítulo 3. O padrão *SCORM* ao utilizar a *API*, promove uma ambiente estável, mas com pouca flexibilidade com relação a mudança de estrutura destas próprias chamadas, o que pode ocorrer em novas versões do projeto *SCORM*, já que trabalha com funções específicas e nome de dados padronizados pelo modelo de dados (*Data Model*). Seria interessante estabelecer uma estrutura mais aberta com relação à troca de dados e as chamadas das *API's* através de uma estrutura baseada em *XML*, uma vez que o *XML* promove uma separação de estrutura e dados.

As requisições das *API's* poderiam ser executadas de forma mais aberta e adaptativa através de *Web-Services*, onde estas chamadas são recebidas pelo ambiente de execução do servidor e transmitidas ao agente que recebe as solicitações (do lado do servidor). A figura a seguir demonstra a funcionalidade do modelo tradicional das chamadas *API's* e a utilização de *web services* (proposta pela arquitetura apresentada).

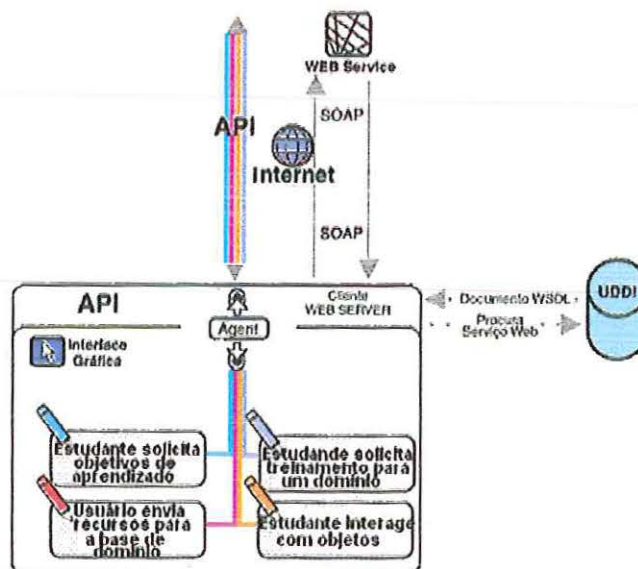


Figura 34. Funcionalidade Global da Interface

Quando ocorre uma chamada API's ao utilizar *Web-Service*, a interface realiza uma consulta ao diretório *UDDI* que contém informações do *Web-Service*, que recebe no caso o formato do documento a ser transferido pelo protocolo *SOAP*, assim como a estrutura das chamadas das API's. O *Web-Service* deve ser estruturado para que possibilitem inserir novos manifestos na base de conhecimento de domínios oriundos de um conjunto possível de instituições conveniadas para a alimentação de repositórios de objetos de aprendizagem. Uma outra razão em utilizar este tipo de tecnologia é reforçada pelo fato de que o manifesto está em formato *XML*, e a estrutura do documento fica extremamente maleável quando ela é fornecida por um serviço padronizado como é o caso do *Web Services*, além de que em processos transacionais é um modo mais eficiente e seguro através do protocolo *SOAP* (abordagem realizada no capítulo 3).

Outra proposta é a utilização de agentes de interface, onde as mensagens enviadas pelo serviço agente mensageiro (figura 27 e 32) são identificadas através de um agente inteligente que irá apresentar mensagens e alertas ao usuário; esta dinâmica pode ser utilizada em situações interessantes como:

- o modelo oferece uma mensagem do sistema oferecendo uma avaliação pelo agente de avaliação;
- o professor recebe uma informação de que algum aluno enviou um documento para ser incluído na base de conhecimento (professor é o agente moderador) para avaliá-lo e autoriza a publicação;
- outras que poderão ser implementadas através da estrutura proposta.

Essencialmente, agentes de interface suportam e providenciam assistência, tipicamente para o usuário aprender a usar uma aplicação em particular, como um sistema operacional. O agente observa o usuário e monitora suas atividades na interface, aprendendo maneiras novas de executar tarefas, e sugerindo maneiras melhores de executá-las. Desta maneira, o agente atua como um assistente pessoal autônomo que coopera com o usuário realizando algumas tarefas de auxílio aos usuários.

#### 4.10 Integração dos recursos na arquitetura proposta

Com o propósito de facilitar o entendimento da funcionalidade da interação dos módulos, agentes e serviços apresentados anteriormente, todas elas são integradas e apresentadas a seguir para que se possa ter uma visão global de conjunto.

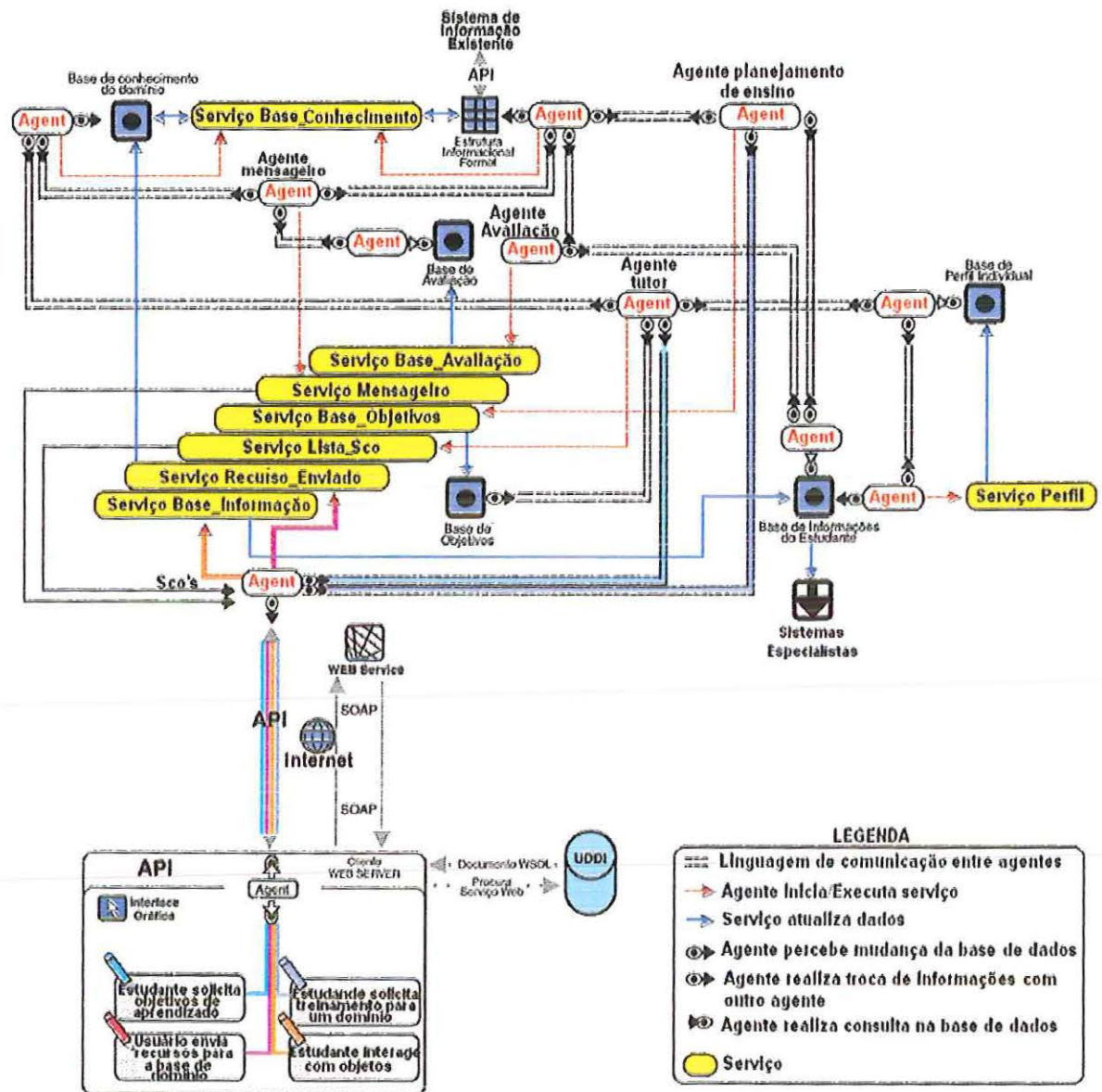


Figura 35. Funcionalidade global da estrutura proposta



## 5 Conclusões e trabalhos futuros

O baixo desempenho de um aluno reconhecido pela instituição de ensino deve ser considerado uma falha da qualidade dos processos de ensino da instituição, que pode não ter trabalhado de forma eficiente os problemas apresentados pelo aluno, e isto só pode ser identificado a partir do momento em que o aluno é tratado de forma individual.

Sistemas de informação que acompanham a evolução do treinamento e aprendizado individual não ocorrem de forma simplista. A utilização e integração de várias tecnologias são necessárias e fundamentais. O que se fez presente neste trabalho, foi propor a integração de recursos tecnológicos que apóiam o treinamento e aprendizado individual, entre elas os agentes inteligentes, sistemas tutores, sistemas especialistas, serviços específicos (*services*), serviços web (*web services*) e nesta integração levou-se em consideração a funcionalidade da padronização do projeto *SCORM*.

Para posicionar o projetista de sistemas de informações, o modelo proposto elucida o caminho a ser trilhado e aumenta as perspectivas na utilização das tecnologias envolvidas, mas o projeto tecnológico também deve ter fundamentos pedagógicos já que o objetivo é o aprendizado e treinamento individual, por isto a necessidade de elucidar conceitos de avaliação e suas características. A avaliação como ferramenta métrica de desempenho não deve ser utilizada de forma padronizada e sim se moldar às necessidades individuais de aprendizagem e treinamento, o modelo proposto se preocupa com isto, o que fundamenta a criação das bases de avaliação, objetivos, perfil individual e informações do aluno.

A característica de objetos de aprendizagem (*Learning Objects*) se moldar ao perfil individual integrada em uma inteligência tutorial através dos agentes inteligentes permite uma capacidade corretiva e reativa. Isto torna a estrutura apresentada um embasamento

para a construção de sistemas que apoiem o auxílio das tarefas a serem desempenhadas pelas instituições de ensino, quando buscam o comportamento do aprendizado individual.

A representação do aprendizado do aluno através do comportamento e utilização dos recursos tecnológicos, sincronizados com o conteúdo apresentado em sala de aula, permite identificar vários indícios da eficácia de conteúdo instrucional abordado pelo professor e/ou do próprio material disponibilizado na base de conhecimento de domínio, contribuindo para uma melhor atuação de ambos.

Os dados contidos na base de informações do estudante embasado pela estrutura do *Data Model* do projeto *SCORM* é um repositório importante e rico em detalhes que possibilita a análise de desempenho e identificação do perfil individual do aluno, propiciando possível adoção de medidas corretivas para melhoria contínua do conteúdo das disciplinas, das metodologias de ensino e, conseqüentemente reflete em um melhor desempenho na formação dos profissionais envolvidos.

A globalização da informação e a adoção de um padrão intercambiável permitem que instituições de ensino venham cada vez mais a participar de uma solução colaborativa e global, permitindo assim com a melhoria da informação e formação do corpo docente e discente.

A utilização de um modelo padronizado para a construção de repositórios e treinamento digitais apresentados pelo projeto *SCORM* vem propiciar um ambiente colaborativo entre as instituições de ensino, promovendo um trabalho em conjunto na construção desses mesmos repositórios, o que se espera, é que sejam derrubadas as barreiras feudais do conhecimento e que os centros de pesquisas sejam mais colaborativos entre si, buscando uma melhoria socializada dos centros de excelência de ensino, principalmente o centros educacionais de gestão pública.

O fato de identificar necessidades e reconhecer que existem falhas na exposição dos assuntos abordados em sala de aula não significa a extinção dos problemas educacionais, metodologias utilizadas com objetivo de melhoria contínua devem ser empregadas nos mais diversos aspectos, desde a exposição, treinamento, prática e avaliação, constituindo um movimento de inovação e renovação contínua.

Um mapeamento mais granular na interação de agentes inteligentes com a estrutura do modelo de objetos de aprendizagem com relação a navegação proposta na versão do

projeto *SCORM* de 22 de julho, seria bastante colaborativa no aprimoramento da solução proposta o que pode ser desenvolvido em trabalho futuro.

A padronização de estruturas da informação ocorre hoje pela necessidade de troca de informações através de um ambiente aberto e colaborativo, isto não está acontecendo somente com objetos de conteúdos, mas também com outros projetos para a padronização na troca de informações com foco em dados administrativos e operacionais como o caso do projeto *SIF (Schools Interoperability Framework)*, ele possui um conjunto de agentes que possibilita uma comunicação integrada entre as escolas através de um conjunto de regras, serviços e definições de estrutura de dados padronizados, possibilitando uma interação extremamente flexível na troca de informações, o que pode promover a construção de *data warehouse's* que quando cruzados com repositórios de conteúdos contribuiriam com dados de análise de comportamento educacional, possibilitando identificar rumos de investimento na área da educação. Vale a pena lembrar que isto depende de uma vontade de conjunto e regras a serem definidas pelos órgãos reguladores e entidades educacionais públicas e privadas.



## REFERÊNCIA BIBLIOGRÁFICA

ADL T. T. (2004). *Sharable Content Object Reference Model (SCORM) 2004 2nd Edition*. Disponível em : <<http://www.adlnet.org/index.cfm?fuseaction=DownFile&libid=648&bc=false>>. Acessado em 24 de jul.

ALVES, F.J.A. (2004). *Sistema de educação a distancia baseado em serviços Web*. Disponível em: <<http://www.abed.org.br/congresso2002/trabalhos/texto46.htm>>. Acesso em: 10 fev.

ANDRADE, V.M.; BRASILEIRO, F.V. (2003). *Sistema de gerenciamento de aprendizagem, uma metodologia de avaliação*. Disponível em: <<http://fad.uta.cl/dfad/docum/cedm/2-br-M%E1rio%20Vasconcelos%20Andrade.pdf>>. Acesso em: 18 ago. 2003.

ANGELO, M.F. (2001). *Uma ferramenta para auxílio ao treinamento em avaliação mamográfica via-internet*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2001.

BECK, J.; STERN, M.; HAUGSJAA, E. (2003). *Applications of IA in education*. Disponível em: <<http://www.acm.org/crossroads/xrds3-1/aied.html>>. Acesso em: 8 Ago.

BELLONI, M.L. (1999). *Educação à distância*. Campinas: Autores Associados.

BETTIO, R.W. (2003). *Avaliações gráficas e dinâmicas aplicadas a ambientes virtuais de aprendizagem*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2003. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 23 mar. 2004.

BLOOM, B.S.; HASTINGS, J.T.; MADDAUS, G.F. (1983). *Manual de avaliação formativa e somativa do aprendizado escolar*. São Paulo: Pioneira.

BOGO, H.L. (2003). *Criação de comunidades virtuais a partir de agentes inteligentes*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2003. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 17 fev.

CANDAU, V.M. (Org.) (1999). *Magistério: construção cotidiana*. Petrópolis. Editora: Vozes.

CASAS, L.A.A. (1999). *Contribuições para a modelagem de um ambiente inteligente de educação baseado em realidade virtual*. Tese (Doutorado) – Universidade Federal de Santa Catarina, Florianópolis, 1999. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 15 fev.

CHIAVENATO, I. (1981). *Administração de recursos humanos*. 2.ed. São Paulo: Editora Atlas.

CORDEIRO, A.D. (2001). *Concepção e implementação de um sistema multi-agentes para gestão da comunicação de dados "ON-LINE" entre sistemas*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2001. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 11 abr.

COSER, A. (2000). *Utilização de agentes inteligentes no trabalho colaborativo via internet*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2000. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 6 abr.

COSTA, M.T.C. (1999). *Uma arquitetura baseada em agentes para suporte à ensino a distância*. Tese (Doutorado) – Universidade Federal de Santa Catarina, Florianópolis, 1999. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 15 dez. 2003.

CUNHA, D. (2004). *Web services, SOAP e aplicações web*. Disponível em: <[http://devedge.netscape.com/viewsource/2002/soap-overview/index\\_pt\\_br.html](http://devedge.netscape.com/viewsource/2002/soap-overview/index_pt_br.html)>. Acesso em: 10 Jan.

DIAS, S.J. (2000). *Avaliação da educação superior*. Petrópolis. Editora Vozes.

DOWNES, S. (2003). *Learning objects: resources for distance education worldwide*. Disponível em: <<http://www.irrodl.org/content/v2.1/downes.pdf>>. Acesso em: 10 Dez. 2003.

EDUWORKS (2004). *FS Scorm publishing template for flash 5*. Disponível em: <[http://www.eduworks.com/LOTT/Mm-LOTT/fsScorm\\_help.htm](http://www.eduworks.com/LOTT/Mm-LOTT/fsScorm_help.htm)>. Acesso em: 10 Mai.

ELLIS, B. (2004). Virtual classroom technologies for distance education: the case for on-line synchronous delivery. In: NORTH AMERICAN WEB DEVELOPERS CONFERENCE, 1997. *Proceedings...* Disponível em: <<http://www.detac.com/solution/naweb97.htm>>. Acesso em: 14 Jun.

FANTON, M.G. (2002). *Comprometimento organizacional e qualidade de ensino: o caso da Faculdade de Ciências Sociais Aplicadas de Cascavel – PR*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2002. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 29 mar.

FARACO, R.Á. (2003). *Ferramenta de autoria de sistemas tutores inteligentes construindo o modelo do domínio do conhecimento com redes semânticas*. Disponível em: <<http://www.cbcomp.univali.br/pdf/2002/ine012.pdf>>. Acesso em: 15 mar.

FLIPPO, E.B. (1973). *Princípios da administração de pessoal*. Traduzido por Auriphebo Simões. 2.ed. São Paulo: Atlas.



FORTH, S. (2003). A meditation on metadata. *The Sophist*, n.7, Sept. Disponível em: <[http://www.isophinstitute.com/sophist\\_no7\\_metadata.aspx](http://www.isophinstitute.com/sophist_no7_metadata.aspx)>. Acesso em: 3 de Mai.

FREIRE, M.E.P. (1998). *O sistema tutor de um ambiente inteligente para treinamento e ensino*. Dissertação (Mestrado) – Instituto de Ciências Matemáticas de São Carlos, Universidade de São Paulo, São Carlos, 1998.

FREITAS, M.C.D. (1999). *Um ambiente de aprendizagem na internet aplicado na construção civil*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 1999. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 29 nov. 2003.

GEBER, B. (1991). Distance education. *Training*, New York, v.28, n.11, p.57-58, Nov.

GIESE, L.F. (1998). *Estrutura de agentes para os processos de compra e venda utilizando tomada de decisão difusa*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 1998. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 11 nov. 2003.

HARMON, P.; KING, D. (1988). *Sistemas especialistas*. Rio de Janeiro. Editora: Campus.

IMS. (2004). *Instructional Management System - Global Learning Consortium*. Disponível em: <<http://www.imsproject.org>>. Acessado em 15 de abr.

IMS-LOM. (2004). *Instructional Management System - Global Learning Consortium, Learning Object Model*. Disponível em: <<http://www.imsproject.org/metadata/mdbestv1p1.html#IEEE>>. Acessado em 15 de abr.

KELLER, R. (1991). *Tecnologia de sistemas especialistas: desenvolvimento e aplicação*. São Paulo: Makron Books do Brasil.

KEMCZINSKI, A. (2000). *Ensino de graduação pela internet: um modelo de ensino-aprendizagem semipresencial*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2000. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 29 jun.

LACHI, R.L. (2002). *Uso de agentes de interface no gerenciamento de informações em ambientes de ensino à distância*. Dissertação (Mestrado) - Instituto de Computação, Universidade de Campinas, Campinas, 2002. Disponível em: <<http://www.dcc.unicamp.br/~rlachi/Mestrado.html>>. Acesso em: 5 jan.

LAUDON, K.C. (2004). *Sistemas de informações gerenciais*. São Paulo: Prentice Hall.

LED (2003). Universidade Federal de Santa Catarina. Laboratório de Ensino a Distância (2003). Disponível em: <<http://www.led.br/>>. Acesso em: 8 ago. 2003.

LEMOS, D.C. (2003). *Educação corporativa: pesquisa de soluções e-learning e modelos de universidades corporativas*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2003. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 4 jun.

LÉVY, P. (2001). *Cibercultura*. Rio de Janeiro: Ed. 34.

LONGMIRE, W. (2004). *A primer on learning objects*. Disponível em: <<http://www.learningcircuits.org/2000/mar2000/Longmire.htm>>. Acesso em: 26 Jun.

LUCENA, B. (2004). *Novas tecnologias no e-learning: desafios e oportunidades para o design*. Disponível em: <<http://www.abed.org.br/publique/cgi/cgilua.exe/sys/start.htm?UserActiveTemplate=2ing&inford=883&sid=135>>. Acesso em: 10 mar.

MARTINS, G.G. (2002). *Uma arquitetura baseada em agentes inteligentes para ambientes computacionais voltados à educação a distância*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2002. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 23 mar.

MATTOS, K.M.C. (1998). *Um estudo da relação qualidade e recursos humanos em empresas de grande porte*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 1998.

MCARTHUR, D.; BISHAY, M.; LEWS, M. (2004). *The roles of artificial intelligence in education: current progress and future prospects*. Disponível em: <<http://www.rand.org/education/mcarthur/Papers/role.html#intro>>. Acesso em: 10 Abr.

MATUZAWA, F.N. (2001). *O conceito de comunidade virtual auxiliando o desenvolvimento na educação à distância*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2001. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 3 abr.

MESQUITA, L.J.; SILVA, W.T. (2003). Ferramenta de autoria para o domínio de conhecimento de STI baseado em Frames – FASTI. In: SIMPOSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 9. Disponível em: <<http://www.lia.ufc.br/sbie98>>. Acesso em: 12 jun.

MORAES, M.C. (1998). *Novas tendências para o uso das tecnologias da informação na educação*. Brasília: Papirus.

MORESI, E.A.D. (2001). Inteligência organizacional, um referencial integrado. *Ciência da Informação*, Rio de Janeiro, v.30, n.2, p.35-46, maio/ago.

NETTO, A.A.O. (2001). *Softwares de entretenimento e suas contribuições no processo de aprendizagem: o erro como tomada de decisão*. Dissertação (Mestrado) - Universidade



Federal de Santa Catarina, Florianópolis, 2001. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 12 fev.

OLIVEIRA, S.M. (2003). *Avaliação do programa de educação à distância oferecido pelo curso de mestrado em gerência de sistemas de informação da PUC - Pontifícia Universidade Católica de Campinas, Campinas, Brasil*. Disponível em: <<http://fad.uta.cl/dfad/docum/cedm/2-br-SILAS%20MARQUES%20DE%20OLIVEIRA.pdf>>. Acesso em: 10 ago.

OLIVEIRA, T.M.P. (2001). *Interatividade na educação a distância*. Dissertação (Mestrado) - Universidade Federal de Santa Catarina, Florianópolis, 2001. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 10 fev. 2004

PRADO, N.R.S.A. (2002). *Uma arquitetura para ambientes de ensino-aprendizagem utilizando o sistema de modelos tutores inteligentes e agentes*. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2002.

RIBEIRO, L.R.C. (2000). *Programas da qualidade total e educação: reflexões sobre a utilização de seus princípios no ensino de engenharia*. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2000.

ROSSETT, A. (2001). *The ASTD e-learning handbook: best practices, strategies, and case studies for a emerging field*. New York: McGraw-Hill. Disponível em <<http://books.mcgraw-hill.com/authors/rossett/km.htm>>. Acessado em 10 de jan.

SOARES, J.M. et al. (2004). *Integração e gestão de aplicações multimídia para educação à distância com garantia de qualidade de serviço em redes de alta velocidade*. Disponível em: <[http://www.rnp.br/wrnp2/2001/palestras\\_aplicacao/res\\_aplic\\_01.pdf](http://www.rnp.br/wrnp2/2001/palestras_aplicacao/res_aplic_01.pdf)>. Acessado em: 10 abr.

SOUZA, L.S.H. (2001). *O uso da internet como ferramenta de apoio ao processo de ensino-aprendizagem da engenharia de transportes*. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2001.

SPECTOR, J.M. (2001). *Tools and principles for the design of collaborative learning environments for complex domains*. Disponível em <<http://soeweb.syr.edu/faculty/spector/publications/tools-principles-collab-design.pdf>> Acessado em 10 de jan.

TEIXEIRA FILHO, J. (2000). *Gerenciando conhecimento: como a empresa pode usar a memória organizacional e a inteligência competitiva no desenvolvimento dos negócios*. Rio de Janeiro: SENAC.

THIRRY, M.C.C. (1999). *Uma arquitetura baseada em agentes para suporte ao ensino a distância*. Tese (Doutorado) - Universidade Federal de Santa Catarina, Florianópolis, 1999. Disponível em: <<http://teses.eps.ufsc.br/tese.asp>>. Acesso em: 13 jun.

THOMPSON, J.E. (1996). *Student modeling in an intelligent tutoring system*. Thesis (Master) - Graduate School of Engineering of the Air Force Institute of Technology Air University. Disponível em: <<http://www.stormingmedia.us/06/0682/A068223.html>>. Acesso em: 16 jun.

USP (2003). Universidade de São Paulo. A Escola do futuro. Disponível em: <<http://www.futuro.usp.br/>>. Acesso em: 17 ago.

WICKERT, M.L.S. (2003). *O futuro da educação a distância no Brasil*. Disponível em: <[http://www.intelecto.net/ead\\_textos/lucia1.htm](http://www.intelecto.net/ead_textos/lucia1.htm)>. Acesso em: 11 dez. de 2003.

WEB SERVICES INTEROPERABILITY (2004). Usage Cenários. Disponível em: <<http://www.ws-i.org/Documents.aspx#document5>>. Acesso em: 25 Abr .

## APENDICE - A

## Url's para acesso a materiais do padrão IMS

As url's abaixo possuem o intuito de apresentar material dinâmico do projeto IMS, seu objetivo é de facilitar a localização de conteúdo do material pesquisado.

<b>Identificação</b>	<b>URL a ser pesquisada na Internet</b>
URL-1	<a href="http://www.msglobal.org/question/qtinfo03.html#1.1.1">http://www.msglobal.org/question/qtinfo03.html#1.1.1</a>



## APENDICE - B

### Url's para acesso a materiais do padrão SCORM

As url's abaixo possuem o intuito de apresentar material dinâmico do projeto SCORM, seu objetivo é de facilitar a localização de conteúdo do material pesquisado.

<b>Identificação</b>	<b>URL a ser pesquisada na Internet</b>
URL-1	<a href="http://www.macromedia.com/resources/elearning/objects/">http://www.macromedia.com/resources/elearning/objects/</a>
URL-2	<a href="http://www.msglobal.org/ssp/imssspv1p0pd/imsssp_infov1p0pd.html">http://www.msglobal.org/ssp/imssspv1p0pd/imsssp_infov1p0pd.html</a>
URL-3	<a href="http://www.adlnet.org/index.cfm?fuseaction=DownFile&amp;libid=648&amp;bc=false">http://www.adlnet.org/index.cfm?fuseaction=DownFile&amp;libid=648&amp;bc=false</a>

## ANEXO A

## SCORM - Perfil de exigências do Content Package Application

No.	Elements	Resource Package	Content Aggregation Package
1	<manifest>	M	M
1.1	identifier	M	M
1.2	version	O	O
1.3	xml:base	O	O
1.4	<metadata>	O	O
1.4.1	<schema>	O	O
1.4.2	<schemaversion>	O	O
1.4.3	{Meta-data}	O	O
1.5	<organizations>	M	M
1.5.1	default	NP	M
1.5.2	<organization>	NP	M
1.5.2.1	identifier	NP	M
1.5.2.2	structure	NP	O
1.5.2.3	adlseq:objectivesGlobalToSystem	NP	O
1.5.2.4	<title>	NP	M
1.5.2.5	<item>	NP	M
1.5.2.5.1	identifier	NP	M
1.5.2.5.2	identifierref	NP	O
1.5.2.5.3	<title>	NP	M
1.5.2.5.4	isvisible	NP	O
1.5.2.5.5	parameters	NP	O
1.5.2.5.6	<item>	NP	O
1.5.2.5.7	<metadata>	NP	O
1.5.2.5.7.1	{Meta-data}	NP	O
1.5.2.5.8	<adlcp:timeLimitAction>	NP	O
1.5.2.5.9	<adlcp:dataFromLMS>	NP	O
1.5.2.5.10	<adlcp:completionThreshold>	NP	O
1.5.2.5.11	<imsss:sequencing>	NP	O
1.5.2.5.12	<adlnav:presentation>	NP	O
1.5.2.6	<metadata>	NP	O
1.5.2.6.1	{Meta-data}	NP	O
1.5.2.7	<imsss:sequencing>	NP	O
1.6	<resources>	M	M
1.6.1	xml:base	O	O
1.6.2	<resource>	O	O
1.6.2.1	identifier	M	M
1.6.2.2	type	M	M
1.6.2.3	href	O	O
1.6.2.4	adlcp:scormType	M	M
1.6.2.5	adlcp:persistState	O	O
1.6.2.6	xml:base	O	O
1.6.2.7	<metadata>	O	O
1.6.2.7.1	{Meta-data}	O	O
1.6.2.8	<file>	O	O
1.6.2.8.1	href	M	M
1.6.2.8.2	<metadata>	O	O
1.6.2.8.2.1	{Meta-data}	O	O
1.6.2.9	<dependency>	O	O
1.6.2.9.1	identifierref	M	M
1.7	<manifest>	O	O
1.8	<imsss:sequencingCollection>	NP	O

A tabela define a exigência para cada um dos itens mencionado do *Content Package Application Profile*. Cada perfil é listado com sua correspondente exigência do elemento/atributo para cada manifesto o empacotamento de conteúdo:

- “M” indica que o elemento/atributo é Obrigatório (*Mandatory*).
- “O” indica que o elemento/atributo é Opicional(*Optional*).
- “NP” indica que o elemento/atributo não é permitido (*Not Permitted*).

Os elementos são indicados notação em que são escritos em XML `<element>`. Os atributos são indicados sem nenhuma notação (indica que é um atributo do elemento de um manifesto `<manifest>`) . A numeração é baseada na especificação do *IMS Content Packaging*.



## ANEXO B

### Classes dos Elementos do Data Model – SCORM

Data Model Element	Description
Comments From Learner	Contains text from the learner.
Comments From LMS	Contains comments and annotations intended to be made available to the learner.
Completion Status	Indicates whether the learner has completed the SCO.
Completion Threshold	Identifies a value against which the measure of the progress the learner has made toward completing the SCO can be compared to determine whether the SCO should be considered completed.
Credit	Indicates whether the learner will be credited for performance in this SCO.
Entry	Contains information that asserts whether the learner has previously accessed the SCO.
Exit	Indicates how or why the learner left the SCO.
Interactions	Defines information pertaining to an interaction for the purpose of measurement or assessment.
Launch Data	Provides data specific to a SCO that the SCO can use for initialization.
Learner Id	Identifies the learner on behalf of whom the SCO instance was launched.
Learner Name	Represents the name of the learner.
Learner Preference	Specifies learner preferences associated with the learner's use of the SCO.
Location	Represents a location in the SCO.
Maximum Time Allowed	Indicates the amount of accumulated time the learner is allowed to use a SCO in the learner attempt.
Mode	Identifies the modes in which the SCO may be presented to the learner.
Objectives	Specifies learning or performance objectives associated with a SCO.
Progress Measure	Identifies a measure of the progress the learner has made toward completing the SCO.
Scaled Passing Score	Identifies the scaled passing score for a SCO.
Score	Identifies the learner's score for the SCO.
Session Time	Identifies the amount of time that the learner has spent in the current learner session for the SCO.
Success Status	Indicates whether the learner has mastered the SCO.
Suspend Data	Provides information that may be created by a SCO as a result of a learner accessing or interacting with the SCO.
Time Limit Action	Indicates what the SCO should do when the maximum time allowed is exceeded.
Total Time	Identifies the sum of all of the learner's learner session times accumulated in the current learner attempt prior to the current learner session.

## ANEXO C

### Elementos do Data Model – SCORM

Req. No.	Conformance Requirement
1	<p>If the SCO, through it's correct usage of the Run-Time Environment API, has achieved "SCORM Version 1.2 Run-Time Environment Conformant – Minimum"(SCO-RTE1) status, then in order to also be considered "SCORM Version 1.2 Run-Time Environment Conformant – Minimum with Mandatory" (SCO-RTE1+Mandatory), the SCO shall correctly set or get one or more of the following LMS Mandatory data model elements:</p> <ul style="list-style-type: none"> <li>• cmi.core._children</li> <li>• cmi.core.student_id</li> <li>• cmi.core.student_name</li> <li>• cmi.core.lesson_location</li> <li>• cmi.core.credit</li> <li>• cmi.core.lesson_status</li> <li>• cmi.core.entry</li> <li>• cmi.core.score._children</li> <li>• cmi.core.score.raw</li> <li>• cmi.core.total_time</li> <li>• cmi.core.exit</li> <li>• cmi.core.session_time</li> <li>• cmi.suspend_data</li> <li>• cmi.launch_data</li> </ul>
2	<p>If the SCO, through it's correct usage of the Run-Time Environment API, has achieved "SCORM Version 1.2 Run-Time Environment Conformant – Minimum"(SCO-RTE1) status, then in order to also be considered "SCORM Version 1.2 Run-Time Environment Conformant – Minimum with Optional" (SCO-RTE1+Optional), the SCO shall correctly set or get one or more of the following LMS Optional data model elements:</p> <ul style="list-style-type: none"> <li>• cmi.core.score.max</li> <li>• cmi.core.score.min</li> <li>• cmi.core.lesson_mode</li> <li>• cmi.comments</li> <li>• cmi.comments_from_lms</li> <li>• cmi.objectives._children</li> <li>• cmi.objectives._count</li> <li>• cmi.objectives.n.id</li> <li>• cmi.objectives.n.score._children</li> <li>• cmi.objectives.n.score.raw</li> <li>• cmi.objectives.n.score.max</li> <li>• cmi.objectives.n.score.min</li> <li>• cmi.objectives.n.status</li> <li>• cmi.student_data._children</li> <li>• cmi.student_data.mastery_score</li> <li>• cmi.student_data.max_time_allowed</li> <li>• cmi.student_data.time_limit_action</li> <li>• cmi.student_preference._children</li> <li>• cmi.student_preference.audio</li> <li>• cmi.student_preference.language</li> <li>• cmi.student_preference.speed</li> </ul>



Req. No.	Conformance Requirement
	<ul style="list-style-type: none"> <li>• cmi.student_preference.text</li> <li>• cmi.interactions._children</li> <li>• cmi.interactions._count</li> <li>• cmi.interactions.n.id</li> <li>• cmi.interactions.n.objectives._count</li> <li>• cmi.interactions.n.objectives.n.id</li> <li>• cmi.interactions.n.time</li> <li>• cmi.interactions.n.type</li> <li>• cmi.interactions.n.correct_responses._count</li> <li>• cmi.interactions.n.correct_responses.n.pattern</li> <li>• cmi.interactions.n.weighting</li> <li>• cmi.interactions.n.student_response</li> <li>• cmi.interactions.n.result</li> <li>• cmi.interactions.n.latency</li> </ul>
3	<p>The SCO shall only attempt to set or get CMI data model elements that are supported by the SCORM Run-Time Environment Data Model.</p>
	<p>The SCO shall only attempt to get (read) data model elements that are defined as “read-write” or “read-only”. These elements are:</p> <ul style="list-style-type: none"> <li>• cmi.core._children (read-only)</li> <li>• cmi.core.student_id (read-only)</li> <li>• cmi.core.student_name (read-only)</li> <li>• cmi.core.lesson_location (read/write)</li> <li>• cmi.core.credit (read-only)</li> <li>• cmi.core.lesson_status (read/write)</li> <li>• cmi.core.entry (read-only)</li> <li>• cmi.core.score._children (read-only)</li> <li>• cmi.core.score.raw (read/write)</li> <li>• cmi.core.score.max (read/write)</li> <li>• cmi.core.score.min (read/write)</li> <li>• cmi.core.total_time (read-only)</li> <li>• cmi.core.lesson_mode (read-only)</li> <li>• cmi.suspend_data (read/write)</li> <li>• cmi.launch_data (read-only)</li> <li>• cmi.comments (read/write)</li> <li>• cmi.comments_from_lms (read-only)</li> <li>• cmi.objectives._children (read-only)</li> <li>• cmi.objectives._count (read-only)</li> <li>• cmi.objectives.n.id (read/write)</li> <li>• cmi.objectives.n.score._children (read-only)</li> <li>• cmi.objectives.n.score.raw (read/write)</li> <li>• cmi.objectives.n.score.max (read/write)</li> <li>• cmi.objectives.n.score.min (read/write)</li> <li>• cmi.objectives.n.status (read/write)</li> <li>• cmi.student_data._children (read-only)</li> <li>• cmi.student_data.mastery_score (read-only)</li> <li>• cmi.student_data.max_time_allowed (read-only)</li> <li>• cmi.student_data.time_limit_action (read-only)</li> <li>• cmi.student_preference._children (read-only)</li> </ul>



Req. No.	Conformance Requirement
	<ul style="list-style-type: none"> <li>• cmi.student_preference.audio (read/write)</li> <li>• cmi.student_preference.language (read/write)</li> <li>• cmi.student_preference.speed (read/write)</li> <li>• cmi.student_preference.text (read/write)</li> <li>• cmi.interactions._children (read-only)</li> <li>• cmi.interactions._count (read-only)</li> <li>• cmi.interactions.n.objectives._count (read-only)</li> <li>• cmi.interactions.n.correct_responses._count (read-only)</li> </ul>
	<p>The SCO shall only attempt to set (write) CMI data model elements that are defined as “read-write” or “write-only”. These elements are:</p> <ul style="list-style-type: none"> <li>• cmi.core.lesson_location (read/write)</li> <li>• cmi.core.lesson_status (read/write)</li> <li>• cmi.core.score.raw (read/write)</li> <li>• cmi.core.score.max (read/write)</li> <li>• cmi.core.score.min (read/write)</li> <li>• cmi.core.exit (write-only)</li> <li>• cmi.core.session_time (write-only)</li> <li>• cmi.suspend_data (read/write)</li> <li>• cmi.comments (read/write)</li> <li>• cmi.objectives.n.id (read/write)</li> <li>• cmi.objectives.n.score.raw (read/write)</li> <li>• cmi.objectives.n.score.max (read/write)</li> <li>• cmi.objectives.n.score.min (read/write)</li> <li>• cmi.objectives.n.status (read/write)</li> <li>• cmi.student_preference.audio (read/write)</li> <li>• cmi.student_preference.language (read/write)</li> <li>• cmi.student_preference.speed (read/write)</li> <li>• cmi.student_preference.text (read/write)</li> <li>• cmi.interactions.n.id (write-only)</li> <li>• cmi.interactions.n.objectives.n.id (write-only)</li> <li>• cmi.interactions.n.time (write-only)</li> <li>• cmi.interactions.n.type (write-only)</li> <li>• cmi.interactions.n.correct_responses.n.pattern (write-only)</li> <li>• cmi.interactions.n.weighting (write-only)</li> <li>• cmi.interactions.n.student_response (write-only)</li> <li>• cmi.interactions.n.result (write-only)</li> <li>• cmi.interactions.n.latency (write-only)</li> </ul>
4	<p>When setting (writing) data model element values, the SCO shall provide the value for the element according to the data model data type and vocabulary definition (if any) for the corresponding data model element as follows:</p> <ul style="list-style-type: none"> <li>• cmi.core.lesson_location - CMISString255</li> <li>• cmi.core.lesson_status - CMIVocabulary (Status)</li> <li>• cmi.core.score.raw - CMIDecimal or CMIBlank</li> <li>• cmi.core.score.max - CMIDecimal or CMIBlank</li> <li>• cmi.core.score.min - CMIDecimal or CMIBlank</li> <li>• cmi.core.exit - CMIVocabulary (Exit)</li> <li>• cmi.core.session_time - CMITimespan</li> <li>• cmi.suspend_data - CMISString4096</li> </ul>

Req. No.	Conformance Requirement
	<ul style="list-style-type: none"> <li>• cmi.comments - CMISString4096</li> <li>• cmi.objectives.n.id - CMIIentifier</li> <li>• cmi.objectives.n.score.raw - CMIDecimal or CMIBlank</li> <li>• cmi.objectives.n.score.max - CMIDecimal or CMIBlank</li> <li>• cmi.objectives.n.score.min - CMIDecimal or CMIBlank</li> <li>• cmi.objectives.n.status - CMIVocabulary (Status)</li> <li>• cmi.student_preference.audio - CMISInteger</li> <li>• cmi.student_preference.language - CMISString255</li> <li>• cmi.student_preference.speed - CMISInteger</li> <li>• cmi.student_preference.text - CMISInteger</li> <li>• cmi.interactions.n.id - CMIIentifier</li> <li>• cmi.interactions.n.objectives.n.id - CMIIentifier</li> <li>• cmi.interactions.n.time - CMITime</li> <li>• cmi.interactions.n.type - CMIVocabulary (Interaction)</li> <li>• cmi.interactions.n.correct_responses.n.pattern - CMIFeedback</li> <li>• cmi.interactions.n.weighting - CMIDecimal</li> <li>• cmi.interactions.n.student_response - CMIFeedback</li> <li>• cmi.interactions.n.result - Vocabulary (Result)</li> <li>• cmi.interactions.n.latency - CMITimespan</li> </ul>
5	<p><i>(Not tested)</i></p> <p>When setting (writing) the cmi.core.exit element, the SCO shall set the value according to the following:</p> <ul style="list-style-type: none"> <li>• "time-out" - This indicates that the SCO ended because the SCO has determined an excessive amount of time has elapsed, or the cmi.student_data.max_time_allowed has been exceeded. The max_time_allowed can be found in the Content Packaging Manifest (adlcp:maxtimeallowed).</li> <li>• "suspend" - This indicates that the student left the SCO with the intent of returning to it later at the point where he/she left.</li> <li>• "logout" - This indicates that the student logged out from within the SCO instead of returning to the LMS system to log out. This implies that the SCO passed control to the LMS system, and the LMS system automatically logged the student out of the course -- after updating the appropriate data model elements.</li> <li>• "" - The empty string shall be used to represent a normal exit state.</li> </ul>
6	<p><i>(Not tested)</i></p> <p>When setting (writing) the cmi.core.lesson_status element, the SCO shall set the value according to the following:</p> <ul style="list-style-type: none"> <li>• "passed" – This value is used when the SCO is taken for credit (i.e. cmi.core.credit is "credit").</li> <li>• "failed" – This value is used when the SCO is taken for credit (i.e. cmi.core.credit is "credit").</li> <li>• "completed" – This value is used when the SCO is taken for no-credit (i.e. cmi.core.credit is "no-credit").</li> <li>• "incomplete" – This value is used when the SCO is taken for no-credit or credit, when the SCO is exited prematurely (before a passed/failed/completed status can be determined).</li> </ul>

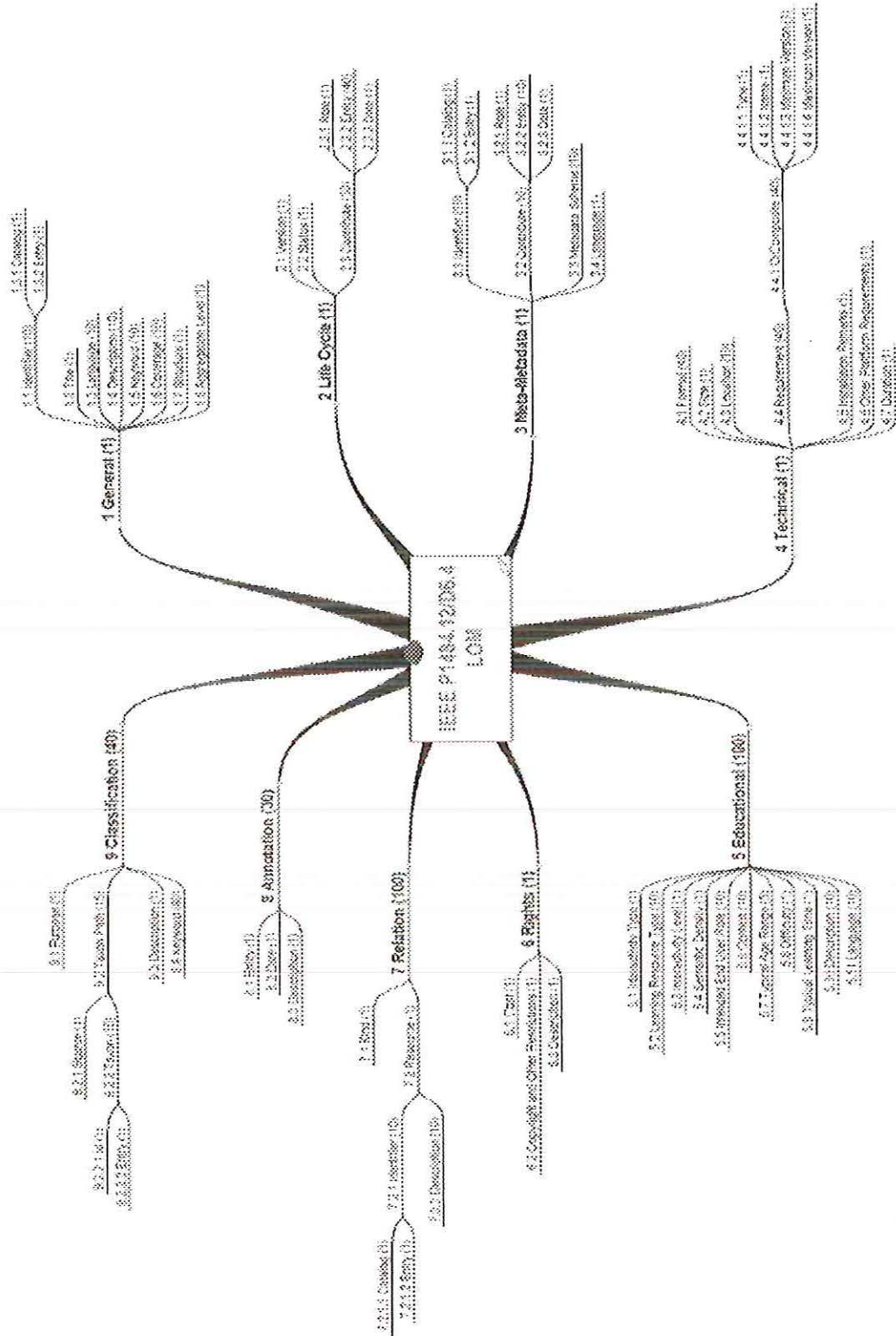
Req. No.	Conformance Requirement
	<ul style="list-style-type: none"><li data-bbox="544 322 1417 443">• “browsed” – This value is used when the <code>cmi.core.lesson_mode</code> is “browse”. Note: If the SCO attempts to get the value of <code>cmi.core.lesson_mode</code>, and the LMS returns an empty string and sets the API Error Code to “401” (not implemented), the SCO shall assume a mode of “normal”.</li></ul>



ANEXO – D

LOM (Learning Object Model Overview)

Os elementos estão descritos no próximo anexo (Anexo E)



Overview of LOM draft 6.4  
 The numbers in parentheses denote the multiplicity of the element.  
 Numbers greater than 1 indicate the number of parallel instances of  
 elements and implementation must allow. The word role was preferred by  
 "Subject metadata" Working Group, Germany. Please refer any  
 questions to [lo@adobe.com](mailto:lo@adobe.com).

## ANEXO E

## Descrição dos elementos do LOM (Learning Object Model)

IEEE LOM v6.4 April					
Categoria	Conjunto de elementos	Qualifiers	Metadata Authority	Vocabulary (Encoding S	
1. General			DC.Identifier		
	1.1 Identifier		DC.Identifier		
	1.2 Title		DC.Title		
	1.3 Catalog Entry				
		1.3.1 Catalogue	DC.Identifier	ISO/IEC 10646-1	
		1.3.2 Entry	DC.Identifier		
	1.4 Language		DC.Language	ISO 639 RFC 1766	
	1.5 Description		DC.Description		
	1.6 Keywords		DC.Subject		
	1.7 Coverage		DC.Coverage		
	1.8 Structure			Collection Mixed Linear Hierarchical Networked Branched Parceled Atomic	
	1.9 Aggregation Level			1 = the smallest level of aggregation, e.g. raw media data or fragments.  2 = a collection of atoms, e.g. an HTML document with some embedded pictures or a lesson.  3 = a collection of level 2 learning objects, e.g. a 'web' of HTML documents, with an index page that links the pages together or a course.  4 = the largest level of granularity, e.g. a set of courses that lead to a certificate.	
	2. Life Cycle				
		2.1 Version			
2.2 Status				Draft Final Revised Unavailable	
2.3 Contribute					

		2.3.1 Role		Author Publisher Unknown Initiator Terminator Validator Editor Graphical Designer Technical Implementer Content Provider Technical Validator Educational Validator Script Writer Instructional Designer
		2.3.2 Entity	DC.Creator	
		2.3.3 Date	DC.Date	
3. Meta-metadata				
	3.1 Identifier			
	3.2 Catalog			
		3.2.1 Catalog		ISO/IEC 10646-1
		3.2.2 Entry		
	3.3 Contribute			
		3.3.1 Role		See IEE LOM v6.1 3.3.1
		3.3.2 Entity		Vcard
		3.3.3 Date		
	3.4 Metadata Scheme			ISO/IEC10646-1
	3.5 Language			ISO 639 RFC 1766
4. Technical				
	4.1 Format		DC.Format	MIME see RFC2048
	4.2 Size			ISO 646 (0 .. 9)
	4.3 Location			ISO/IEC 10646-1
	4.4 Requirements			
		4.4.1 Type		Operating System Browser
		4.4.2 Name		if Type='Operating System', then: PC_DOS, MS-Windows, MacOS, Unix, Multi-OS, None  if Type='Browser' then : Any, Netscape Communicator, Microsoft Internet Explorer, Opera, Amaya
		4.4.3 Min version		ISO/IEC 10646-1
		4.4.4 Max version		ISO/IEC 10646-1
	4.5 Installation Remarks			

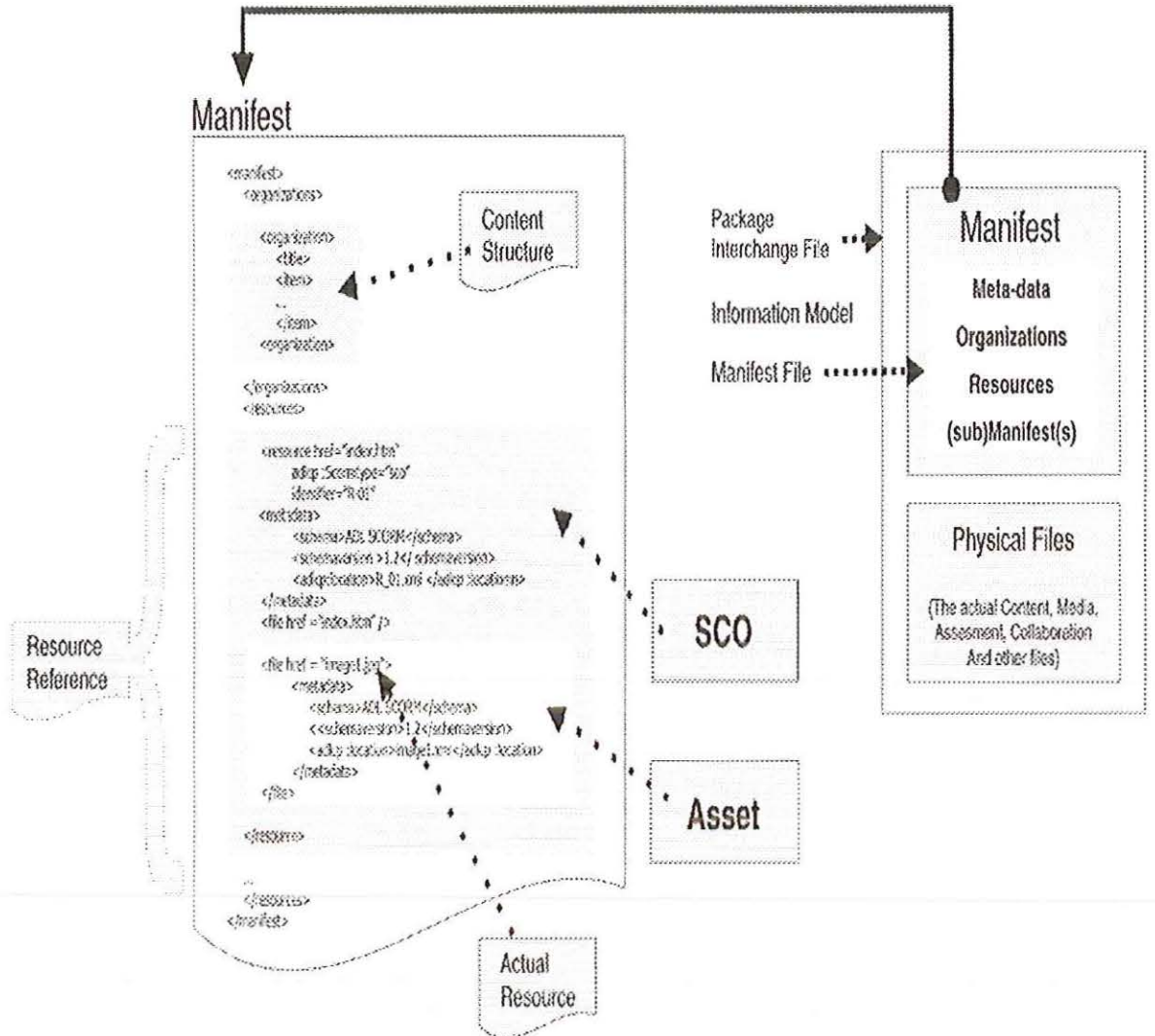


	4.6 Other platform requirements			
	4.7 Duration			
5. Educational				
	5.1 Interactivity Type			Active Expositive Mixed Undefined
	5.2 Learning Resource Type		DC.Type	Exercise Simulation Questionnaire Diagram Figure Graph Index Slide Table Narrative Text Exam Experiment Problem Statement Self Assessment
	5.3 Interactivity Level			very low low medium high very high
	5.4 Semantic Density			very low low medium high very high
	5.5 Intended end user role			Teacher Author Learner Manager
	5.6 Context			Primary Education Secondary Education Higher Education University First Cycle University Second Cycle University Postgrade Technical School First Cycle Technical School Second Cycle Professional Formation Continuous Formation

				Vocational Training
	5.7 Typical Age Range			
	5.8 Difficulty			very easy easy medium difficult very difficult
	5.9 Typical learning Time			
	5.10 Description			
	5.11 Language			ISO 639 RFC 1766
6. Rights			DC.Rights	
	6.1 Cost			yes/no
	6.2 Copyright & other restrictions			yes/no
	6.3 Description			
7. Relation			DC.Relation	
	7.1 Kind			DC.vocab
	7.2 Resource	7.2.1 Identifier	DC.Source	
		7.2.2 Description		
		7.2.3 Catalog entry		
8. Annotation				
	8.1 Person			Vcard
	8.2 Date			
	8.3 Description			
9. Classification			DC.Subject	
	9.1 Purpose		DC.Subject	Discipline Idea Prerequisite Educational Objective Accessibility Restrictions Educational Level Skill Level Security Level
	9.2 Taxon path			
		9.2.1 Source		ISO/IEC 10646-1
		9.2.2 Taxon		
		9.2.2.1 ID		ISO/IEC 10646-1
		9.2.2.2 entry		
	9.3 Description			
	9.4 Keywords			

# ANEXO F

## Masifesto padrão IMS ampliado





## ANEXO G

Teste de ensaio da funcionalidade das chamadas API's entre o objeto de aprendizagem e o LM's, onde são recuperados e inseridos dados no modelo de dados (Data Model), a versão abordada é a 1.2 do RTE do projeto SCORM, mas a lógica de sua funcionalidade continua a mesma.

Validating SCO launch sequence.

- ✓ The SCO has been launched in the appropriate sequence.
- ☒ Attempting to call LMSInitialize function
- ✓ LMSInitialize completed successfully
- ✓ LMSGetValue(cmi.core.\_children) returned a value that is of the correct datatype: student\_id,student\_name,lesson\_location,credit,lesson\_status,entry,score,total\_time,lesson\_mode,exit,session\_time
- ✓ LMSGetValue(cmi.core.student\_id) returned a value that is of the correct datatype: **100000000111**
- ✓ LMSGetValue(cmi.core.student\_name) returned a value that is of the correct datatype: **Student, Joe**
- ✓ LMSGetValue(cmi.core.lesson\_location) returned a value that is of the correct datatype: **<http://scu5.www.cedu.com:8080/CourseImports/SC0001/Lesson01/Resources/sco09.htm>**
- ✓ LMSGetValue(cmi.core.credit) returned a value that is of the correct datatype: **credit**
- ✓ LMSGetValue(cmi.core.lesson\_status) returned a value that is of the correct datatype: **not attempted**
- ✓ LMSGetValue(cmi.core.entry) returned a value that is of the correct datatype: **ab-initio**
- ✓ LMSGetValue(cmi.core.score.\_children) returned a value that is of the correct datatype: raw,min,max
- ✓ LMSGetValue(cmi.core.score.raw) returned a value that is of the correct datatype: **0**
- ✓ LMSGetValue(cmi.core.score.max) returned a value that is of the correct datatype: **0**
- ✓ LMSGetValue(cmi.core.score.min) returned a value that is of the correct datatype: **0**
- ✓ LMSGetValue(cmi.core.total\_time) returned a value that is of the correct datatype: **00:00:00.0**
- ✓ LMSGetValue(cmi.suspend\_data) returned a value that is of the correct datatype:
- ✓ LMSGetValue(cmi.launch\_data) returned a value that is of the correct datatype: **SCORMTROOPERS**
- ✓ LMSGetValue(cmi.comments) returned a value that is of the correct datatype:
- ✓ LMSGetValue(cmi.comments\_from\_lms) returned a value that is of the correct datatype:
- ✓ LMSGetValue(cmi.objectives.\_count) returned a value that is of the correct datatype: **0**
- ✓ LMSGetValue(cmi.objectives.\_children) returned a value that is of the correct datatype: id,score,status
- ✓ LMSGetValue(cmi.student\_data.\_children) returned a value that is of the correct

datatype: mastery\_score,max\_time\_allowed,time\_limit\_action

- ✓ LMSGetValue(cmi.student\_data.max\_time\_allowed) returned a value that is of the correct datatype: **02:00:00.00**
- ✓ LMSGetValue(cmi.student\_preference.audio) returned a value that is of the correct datatype: **3**
- ✓ LMSGetValue(cmi.student\_preference.speed) returned a value that is of the correct datatype: **4**
- ✓ LMSGetValue(cmi.interactions.\_children) returned a value that is of the correct datatype:

id,objective\_ids,time,type,correct\_responses,weighting,student\_response,result,latency

- ✓ LMSGetLastError returned: **0**
- ✓ LMSGetErrorString returned: *No Error*
- ✓ LMSGetDiagnostic returned: *The previous LMS API Function call completed successfully.*
- ☒ Attempting to call LMSCommit function
- ✓ LMSCommit completed successfully
- ✓ LMSSetValue(cmi.core.lesson\_status, "completed") succeeded
- ✓ LMSSetValue(cmi.core.exit, "") succeeded
- ✓ LMSSetValue(cmi.core.session\_time, "00:00:30") succeeded
- ☒ Attempting to call LMSFinish function
- ✓ LMSFinish completed successfully