

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**IOPACKAGER – DESENVOLVIMENTO DE UMA
FERRAMENTA AUTOMÁTICA CONVERSORA DE
OBJETOS EDUCACIONAIS EM PACOTES DE
CONTEÚDO SCORM® 2004 3RD EDITION**

TRABALHO DE GRADUAÇÃO

Ricardo Donato Iop

**Santa Maria, RS, Brasil
2007**

**IOPACKAGER – DESENVOLVIMENTO DE UMA
FERRAMENTA AUTOMÁTICA CONVERSORA DE OBJETOS
EDUCACIONAIS EM PACOTES DE CONTEÚDO SCORM®
2004 3RD EDITION**

por

Ricardo Donato Iop

Trabalho de Graduação apresentado ao curso de Ciência da Computação –
Bacharelado, da Universidade Federal de Santa Maria (UFSM, RS) como
requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

Orientadora: Prof. Dra. Roseclea Duarte Medina

Trabalho de Graduação N° 233
Santa Maria, RS, Brasil
2007

**Universidade Federal de Santa Maria
Centro de Tecnologia
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada, aprova o Trabalho de Graduação

**IOPACKAGER – DESENVOLVIMENTO DE UMA FERRAMENTA
AUTOMÁTICA CONVERSORA DE OBJETOS EDUCACIONAIS EM
PACOTES DE CONTEÚDO SCORM® 2004 3RD EDITION**

elaborado por
Ricardo Donato Iop

como requisito parcial para obtenção do grau de Bacharel em Ciência da
Computação

Comissão Examinadora

**Prof. Dra. Roseclea Duarte Medina
(Orientadora)**

Prof. Carlos Gustavo Martins Hoelzel

Prof. Oni Reasilvia de O. Sichonany

Santa Maria, 2 de março de 2007.

AGRADECIMENTOS

Agradecimentos sempre são complicados. Na intenção de expressar o quanto se é grato por quem lhe é importante, há sempre o risco de esquecer alguém e acabar passando por ingrato a estes. Mas, independente disso, o meu reconhecimento a várias pessoas precisa constar neste espaço.

Primeiramente, agradeço a Deus por ter me concedido capacidades intelectuais e físicas de empreender o desafio de ter realizado este trabalho;

À minha orientadora, Roseclea, por ser paciente em meus momentos de desânimo e demonstrar-se sempre confiante no meu êxito final;

Aos meus pais, Dionir e Moacir, sempre incondicionais no amor, no apoio e na certeza de que eu triunfaria, mostrando-me que as dificuldades ensinam mais do que castigam;

À minha namorada, Priscila, pela compreensão de minhas ausências e pelas palavras ternas de incentivo, acalmando-me quando eu achava difícil demais continuar e lembrando-me sempre que o objetivo final era maior do que todas as adversidades;

Aos meus amigos e colegas de trabalho da Decadium Studios, Dig, Jow, Fernando, Vicentini e Keller, pelo apoio e força transmitidos, e pela parte que cada um tomou de seu tempo compartilhando de seus conhecimentos e prestando-me “consultoria gratuita” para que eu alcançasse meus objetivos;

Enfim, a todos que de alguma forma contribuíram, e com certeza sabem disso, mas que não foram mencionados, recebam também minha gratidão.

Obrigado a todos por fazerem parte da minha vida e serem imprescindíveis ao meu sucesso.

RESUMO

Trabalho de Graduação
Ciência da Computação
Universidade Federal de Santa Maria

IOPACKAGER – DESENVOLVIMENTO DE UMA FERRAMENTA AUTOMÁTICA CONVERSORA DE OBJETOS EDUCACIONAIS EM PACOTES DE CONTEÚDO SCORM® 2004 3RD EDITION

Autor: Ricardo Donato Iop
Orientadora: Prof. Dra. Roseclea Duarte Medina
Local e Data de Defesa: Santa Maria, 2 de março de 2007

O presente trabalho propõe o desenvolvimento do IOPackager, uma ferramenta conversora de objetos educacionais em pacotes de conteúdo SCORM 2004 Terceira Edição. O IOPackager visa desvincular o conhecimento técnico relacionado ao padrão SCORM do processo de empacotamento de material instrucional, fornecendo um modelo genérico de adaptação de qualquer material educacional às normas conformantes com o SCORM. Dessa forma, busca-se a conformidade e a funcionalidade do pacote gerado de modo que ele possua a capacidade de ser armazenado, pesquisado, através de metadados descritivos que serão informados pelo usuário da ferramenta, e visualizado em ambientes de aprendizagem e sistemas de gerência de aprendizado (LMSs) adequados ao modelo de Referência SCORM 2004 3rd *Edition*.

Palavras-chave: *e-learning*; objetos educacionais; empacotamento de conteúdo; SCORM

ABSTRACT

Graduation Work
Computer Science
Federal University of Santa Maria

IOPACKAGER – DEVELOPMENT OF AN AUTOMATIC TOOL FOR CONVERSION OF LEARNING OBJECTS INTO SCORM® 2004 3RD EDITION CONTENT PACKAGES

Author: Ricardo Donato Iop
Adviser: Prof. Dra. Roseclea Duarte Medina
Place and Date: Santa Maria, March 2nd, 2007

This work proposes the development of IOPackager, an automatic tool for conversion of learning objects into SCORM 2004 3rd Edition content packages. IOPackager aims to eliminate the need of technically knowing SCORM in the process of packaging instructional content, providing a generic model for conforming any learning content to SCORM specification. Therefore, the goal is to achieve a conformant and fully functional generated package so that it can be stored, searched through its metadata and launched by SCORM 2004 3rd Edition conformant Learning Management Systems (LMSs) and learning environments.

Keywords: e-learning; learning objects; content packaging; SCORM;

SUMÁRIO

1 INTRODUÇÃO.....	9
1.1 Objetivos deste trabalho.....	11
1.2 Organização do texto.....	11
2 EDUCAÇÃO A DISTÂNCIA.....	13
3 OBJETOS EDUCACIONAIS REUSÁVEIS.....	16
3.1 Modularidade e reusabilidade de objetos educacionais.....	17
3.2 Granularidade (nível de agregação) de objetos educacionais.....	18
3.3 Interoperabilidade de objetos educacionais.....	22
3.4 Modelos de representação de metadados de objetos educacionais.....	22
3.4.1 Dublin Core.....	23
3.4.2 IEEE Learning Object Metadata (IEEE-LOM).....	23
3.4.3 ARIADNE.....	24
4 PADRONIZAÇÃO DE OBJETOS EDUCACIONAIS E O MODELO DE REFERÊNCIA SCORM.....	27
4.1 Desenvolvimento de padrões para E-Learning.....	27
4.2 O modelo de referência SCORM.....	28
4.2.1 Modelo de Agregação de Conteúdo (Content Aggregation Model).....	31
Modelo de Conteúdo (Content Model).....	31
Empacotamento de Conteúdo (Content Packaging).....	31
Dicionário de Metadados (Metadata).....	32
4.2.2 Ambiente de Tempo de Execução (Run-time Environment).....	32
4.2.3 Seqüenciamento e Navegação (Sequencing and Navigation).....	33
4.3 Empacotamento de conteúdo (content packaging) no modelo SCORM.....	34
4.3.1 Estrutura do arquivo XML de manifesto do pacote SCORM.....	34
5 PROPOSTA DA FERRAMENTA IOPACKAGER.....	38
5.1 Características gerais e vantagens do empacotador.....	38
5.2 Desenvolvimento da interface de criação de metadados.....	39
5.3 Desenvolvimento do gerador de pacotes de agregação.....	39
5.4 Desenvolvimento do modelo de agregação e do mecanismo de lançamento do objeto educacional.....	41
5.5 Validação do IOPackager.....	42

5.6 Linguagem e ferramentas utilizadas no desenvolvimento do empacotador.....	43
6 IMPLEMENTAÇÃO DO IOPACKAGER.....	45
6.1 Modelagem da solução.....	45
6.2 Geração da seção de metadados do arquivo XML do manifesto.....	46
6.3 Implementação da classificação de arquivos em recursos de pacote.....	47
6.4 Geração do arquivo imsmanifest.xml e a finalização do pacote SCORM.....	49
7 RESULTADOS E VALIDAÇÃO DA FERRAMENTA IOPACKAGER.....	50
7.1 Finalidade das categorias de conformidade.....	50
7.2 Exemplo de empacotamento de objeto educacional e validação do pacote correspondente.....	51
8 CONCLUSÃO.....	56
9 REFERÊNCIAS BIBLIOGRÁFICAS.....	57

1 INTRODUÇÃO

O avanço das tecnologias de informação e comunicação tem estimulado o desenvolvimento das mais diversas e inovadoras formas de difundir conhecimento através da Internet, sendo a *World Wide Web* um dos mais destacados e reconhecidos recursos de veiculação de informações [25]. Como consequência dessa evolução, uma dentre as diversas áreas estabelecidas e que recebeu valiosos subsídios para seu desenvolvimento e expansão foi a educação a distância via *Web*, usualmente conhecida como *E-Learning*, que constitui-se de um conjunto abrangente de soluções e serviços que objetivam disponibilizar meios de aprendizagem, treinamento ou educação auxiliados por tecnologias (bem conhecidas) baseadas na Internet [15].

Mais recentemente explorados por desenvolvedores envolvidos com a área educacional, os Sistemas de Gerência de Aprendizado, ou LMSs (*Learning Management Systems*), são softwares que controlam o desenvolvimento, gerenciamento e acompanhamento de cursos de aprendizagem *online* [5], englobando funcionalidades dos sistemas tradicionais de *E-Learning*, porém ampliando suas capacidades no sentido da automatização completa dos eventos de treinamento via *Web*. Porém, com a expansão desordenada de diversas dessas plataformas desenvolvidas com finalidades e ferramentas próprias [15], surgiu a proposta de criação e utilização de padrões de desenvolvimento de soluções que se adaptassem a qualquer plataforma, ou seja, que promovessem interoperabilidade entre os sistemas de *E-Learning* existentes, o que resultou no direcionamento do projeto de LMSs com base no conceito de objetos educacionais (OEs).

Um objeto educacional, dentre vários conceitos usualmente aceitos, é qualquer recurso digital que possa ser reusado para dar suporte à aprendizagem. Esses recursos digitais surgiram com o intuito de resolver problemas relacionados à distribuição e armazenamento de informações [23], do ponto de vista dos sistemas de gerência de aprendizagem. Por sua vez, com relação ao uso pedagógico, os objetos educacionais passam a representar um novo paradigma de desenvolvimento de conteúdo instrucional, no qual um material qualquer com finalidade educativa deve ser quebrado em porções menores que possam ser reutilizadas em contextos diferentes, levando ao surgimento de blocos independentes de informação que possuem a capacidade de ser agrupados a fim de formar o todo, ou a lição que se pretende

ensinar [25][20]. Dessa forma, materiais instrucionais mantidos por LMSs seriam formados por conjuntos de diversos tamanhos de objetos educacionais agregados entre si, ou seja, uma composição de objetos que possuem granularidades variadas. Assim como a granularidade constitui uma importante propriedade, há algumas outras características essenciais aos OEs que promovem benefícios diretos à autoria de material educacional e aos ambientes de aprendizado, tais como flexibilidade, facilidade de atualização e customização, reusabilidade e compartilhamento, pesquisa e indexação, melhoramento contínuo e interoperabilidade, ou seja, a capacidade do OE de funcionar normalmente independentemente da plataforma ou LMS que o gereencie.

Com o intuito de tornar viável a implementação de todos os atributos essenciais aos objetos educacionais e aos LMSs, diversas organizações, grupos e comitês desenvolveram trabalhos de padronização para *E-Learning*, dentre os quais destacam-se *Aviation Industry CBT Committee* (AICC) [4], *IEEE's LTSC* [18], *IMS Global Learning Consortium* [19] e *ADL* [3]. O objetivo principal de tais organizações é estabelecer referências necessárias para interoperabilidade e conformidade entre diversos fornecedores [16], ou seja, garantir que produtos de *E-Learning* em conformidade com os padrões sejam compatíveis entre si e executem em quaisquer plataformas independentemente de fabricantes e características de implementação particulares das mesmas. Nesse contexto, englobando normas de padronização desenvolvidas por cada um dos grupos citados acima, surge o padrão SCORM, que consiste de um conjunto de especificações técnicas e de referência para a apresentação de conteúdo de ensino e aprendizagem em *E-learning* [20], definindo um modelo de como se fazer e como se executar cursos baseados na *Web*. A conformidade de um material educacional com o padrão SCORM permite efetivamente que ele seja reusável, durável, acessível e interoperável.

A adaptação do OE para a norma SCORM se dá por meio da criação de pacotes de conteúdo, que consistem dos arquivos físicos pertencentes ao objeto, um arquivo XML que descreve sua estrutura e os metadados utilizados para classificação, todos reunidos em um arquivo compactado, usualmente de extensão zip. Dessa forma o conteúdo empacotado é capaz de ser transferido com facilidade entre LMSs conformantes e armazenado para pesquisa e recuperação. A criação desses pacotes, embora imprescindível para a padronização do material instrucional, demanda considerável esforço do desenvolvedor [7], e usualmente emprega ferramentas apropriadas para sua realização, sem mencionar o fato de ser um processo excessivamente técnico para que seja difundido a um público desenvolvedor

preferivelmente amplo, incluindo profissionais da educação, como professores e educadores.

1.1 Objetivos deste trabalho

Tendo em vista as tendências atuais de padronização de conteúdo através da utilização de modelos de referência abertos, como o SCORM, para o desenvolvimento e distribuição de material educacional via *Web*, e adicionalmente levando em conta o fato de esse processo geralmente envolver uma equipe técnica à parte no emprego de tecnologias para a construção de objetos educacionais [39], este trabalho propõe a desvinculação entre o projetista de conteúdo instrucional e o conhecimento técnico necessário para a adequação do material criado ao padrão SCORM. Esse objetivo será alcançado através de um procedimento automático e genérico de conversão de objetos educacionais ao citado modelo de referência.

A proposta do trabalho é desenvolver uma ferramenta que automatiza o processo de conversão de qualquer objeto educacional (OE) convencional ao modelo de referência SCORM 2004 Terceira Edição sem a necessidade de intervenção do usuário. Para isso, a funcionalidade básica da ferramenta consiste em gerar o pacote de conteúdo correspondente ao OE e prover uma interface para criação dos metadados descritivos do objeto, a serem incluídos no empacotamento. Além da capacidade de ser pesquisado e armazenado em repositórios conformantes com o SCORM, o pacote de conteúdo gerado será capaz de estabelecer comunicação com um sistema de gerência de aprendizado (LMS) para que possa ser lançado por este e posteriormente visualizado.

1.2 Organização do texto

Este trabalho inicia com uma conceituação breve no capítulo 2 a respeito da educação a distância (EAD), voltando o enfoque à EAD baseada na *Web*, ou *E-Learning*, que é o assunto principal ao redor do qual o projeto se estende. A seguir, no capítulo 3, a discussão é conduzida no sentido de esclarecer o papel dos objetos educacionais no *E-Learning* como nova abordagem educacional e também tecnológica. O capítulo 4 discute os motivos e benefícios da padronização de objetos educacionais reusáveis, passando o escopo ao modelo de referência SCORM, alvo específico deste trabalho, detalhando seus componentes, as outras especificações que o compõem, seus benefícios de utilização e a questão do empacotamento

de conteúdo. No capítulo 5, será apresentada a proposta da ferramenta IOPackager, bem como são descritos os meios de validação e as ferramentas e linguagens de desenvolvimento da mesma. As etapas do desenvolvimento entre outras questões de implementação do IOPackager serão discutidos no capítulo 6. O capítulo 7 analisa os resultados do IOPackager através de sua validação com o auxílio do utilitário validador de conformidade da ADL. As conclusões serão apresentadas no capítulo 8 e, por fim, no capítulo 9, se encontram as referências bibliográficas utilizadas no presente trabalho.

2 EDUCAÇÃO A DISTÂNCIA

O avanço das tecnologias de informação e comunicação têm estimulado o desenvolvimento das mais diversas e inovadoras formas de difundir conhecimento através da Internet. Nesse meio, a *World Wide Web* é destacada como o mais conhecido e mais amplamente utilizado recurso de comunicação para a veiculação de informações [25]. Como conseqüência dessa evolução, uma dentre as diversas áreas estabelecidas e que recebeu valiosos subsídios para seu desenvolvimento e expansão foi a educação a distância via *Web*. Concebida como uma forma adaptada das aulas presenciais ou uma alternativa a esse tipo de ensino, onde há comumente a separação física entre aprendiz e instrutor, a prática da educação a distância e a forma de interação entre os envolvidos no processo educacional vem sendo influenciadas e na maior parte das vezes moldadas ao longo dos anos pelas tecnologias de comunicação e de distribuição de mídia correntes, tendo alcançado um patamar notável com o emprego da Internet. Por meio desta, diversas características da EAD foram potencializadas e tiveram seu maior grau de desenvolvimento, como a ampla interatividade através de uma diversidade de hipermídias, que envolvem imagens, vídeos, animações e sons, e o surgimento de um ambiente de ensino novo, atraente e multissensorial [32][20].

Tarouco [37] relaciona diversos recursos comumente utilizados como ferramenta de apoio à educação a distância, que podem promover tanto a comunicação síncrona, que tem como característica principal ocorrer em tempo real, como a assíncrona. No caso assíncrono, destacam-se:

- **Email:** correspondência eletrônica via *Web*;
- **Grupos ou listas de discussão:** promovem troca de informações através de mensagens entre membros de uma comunidade virtual com interesses afins;
 - **World Wide Web (WWW):** rede mundial que permite acesso a e recuperação de grande quantidade de documentos e informações de hipermídia;
 - **FTP e download:** disponibilização de arquivos contendo áudio, textos, imagens ou vídeos;
 - **Vídeo e áudio sob demanda:** permite acesso e exibição assíncronos de vídeos e áudios previamente armazenados em um servidor;

Na comunicação síncrona, temos:

- **Salas de bate-papo:** comunicação em tempo real entre duas ou mais pessoas;
- **Videoconferência:** Comunicação bidirecional através de envio de áudio e vídeo em tempo real, via *Web*, por meio de câmeras acopladas ao computador;
- **Teleconferência:** Definida como todo o tipo de conferência a distância em tempo real, envolvendo transmissão e recepção de diversos tipos de mídia, assim como suas combinações;
- **Áudio-conferência:** Sistema de transmissão de áudio, recebido por um ou mais usuários simultaneamente.

O ensino a distância voltado para a *Web*, também conhecido como *E-Learning*, constitui-se de um conjunto abrangente de soluções e serviços que objetivam disponibilizar meios de aprendizagem, treinamento ou educação auxiliados por tecnologias (bem conhecidas) baseadas na Internet [15]. Diversas são as formas de implementar tais soluções, sendo o desenvolvimento de cursos uma prática tradicionalmente empregada na EAD [34]. Com vistas a oferecer suporte a essas abordagens e prover um ambiente de ensino estruturado e eficaz na difusão de informações e no cumprimento dos objetivos previstos para a atividade educacional, são empregados sistemas de *E-Learning* que facilitem a criação de cursos e permitam sua configuração de forma prática e adequada, e implementem ferramentas de apoio ao aluno e suporte à administração. Além dessas capacidades, esses ambientes devem ser capazes de disponibilizar acesso ao conteúdo instrucional, apoiar sua estruturação e desenvolvimento, armazená-los e viabilizar sua classificação visando recuperação ágil por meio de sistemas de busca.

Um conceito de ambiente de educação a distância que vem sendo mais amplamente explorado é o de Sistemas de Gerência de Aprendizado, ou LMS (*Learning Management System*). Os LMSs são softwares que controlam o desenvolvimento, gerenciamento e acompanhamento de cursos de aprendizagem *online* [5], englobando funcionalidades dos sistemas tradicionais de *E-Learning*, porém ampliando suas capacidades e permitindo a agregação e utilização de recursos inovadores até então não presentes naqueles, principalmente no que diz respeito à automatização completa dos eventos de treinamento via *Web*. A metodologia corrente de desenvolvimento de cursos online até o advento dos Sistemas de Gerência de Aprendizado modernos assemelhava-se à de criação de um *website*

institucional padrão, levando em conta apenas as restrições tecnológicas impostas pelas mídias utilizadas [23]. Por sua vez, a adoção dos novos sistemas de EAD implica em considerar também a arquitetura da informação e o aproveitamento correto do conteúdo que estará se relacionando diretamente com as ferramentas administrativas utilizadas especificamente para o *E-Learning*.

Os Sistemas de Gerência de Aprendizado (LMS) modernos apresentam uma série de características que promovem estruturação e gerenciamento mais eficazes ao ambiente de aprendizado. Podemos citá-los enfatizando o ponto de vista do curso, das aulas, do aprendiz, do instrutor (ou tutor) e do administrador [5]. Com relação ao curso, os LMSs permitem transferência de conteúdo (importação/exportação) em diversos formatos e produzidos independente da ferramenta de autoria; permitem personalização da apresentação dos materiais por estudante ou grupo, acompanham o progresso do aprendiz e selecionam o conteúdo instrucional adequado de acordo com sua evolução no aprendizado; suportam múltiplos instrutores vinculados a um curso e provêm ferramentas de criações de modelos no auxílio ao desenvolvimento de novos cursos. Do ponto de vista das aulas, os LMSs disponibilizam ferramentas de autoria para criação de conteúdo e permitem a apresentação de áudio, vídeo, texto e imagens. Com relação ao estudante, esses sistemas permitem ao aprendiz ter acesso aos conteúdos, acompanhar seu desempenho, e pesquisar informações por meio de ferramentas de busca. Para o instrutor, os LMSs oferecem ferramentas de análise e acompanhamento do aluno, ferramentas de desenvolvimento de conteúdo que utilizam as mais variadas mídias, capacidade de avaliações *online*, incluindo cronometradas. Com relação ao papel do administrador, os recursos disponibilizados pelos sistemas LMS dizem respeito ao registro, autenticação e definições de permissões dos usuários, bem como gerenciamento remoto via *Web* e organização de usuários em grupos.

Com a expansão desordenada de diversas plataformas de Sistemas de Gerência de Aprendizado desenvolvidas com suas próprias finalidades e ferramentas [15], e com a expansão do mercado de *E-Learning*, surgiu a proposta de criação e utilização de padrões de desenvolvimento de soluções que se adaptassem a qualquer plataforma, ou seja, que promovessem interoperabilidade entre os sistemas de *E-Learning* existentes e criação de conteúdo instrucional de alta qualidade e baixo custo. Dessa maneira, sendo abordados a seguir, entram em questão os conceitos de objetos educacionais reusáveis e da padronização do conteúdo manipulado e disponibilizado por essas plataformas de sistemas de aprendizado.

3 OBJETOS EDUCACIONAIS REUSÁVEIS

Com o surgimento da necessidade de normalização da apresentação de conteúdo *Web*, a pesquisa em *E-learning* começou a apontar para uma nova tecnologia: os Objetos Educacionais (OEs), ou *Learning Objects*. Esses recursos digitais surgiram com o intuito de resolver problemas relacionados à distribuição e armazenamento de informações [23], do ponto de vista dos sistemas de gerência de aprendizado. Por sua vez, com relação ao uso pedagógico, os objetos educacionais passam a representar um novo paradigma de desenvolvimento de conteúdo instrucional, no qual um material qualquer com finalidade educativa deve ser quebrado em porções menores que possam ser reutilizadas em contextos diferentes, levando ao surgimento de blocos independentes de informação que possuem a capacidade de ser agrupados a fim de formar o todo, ou a lição que se pretende ensinar [25][20].

A conceituação dos objetos educacionais é bastante variada. Segundo a IEEE LTSC [18], um OE é “qualquer entidade digital ou não digital que possa ser usada, reusada ou referenciada durante a aprendizagem baseada em Tecnologia da Informação”. O projeto ARIADNE [6] os define como “documentos pedagógicos”. Segundo David Wiley [38], OEs são entendidos como “qualquer recurso digital que possa ser reusado para dar suporte à aprendizagem” ou “qualquer recurso, grande ou pequeno que possa ser entregue sob demanda através da rede”. Para a Cisco [30], objetos educacionais são “uma combinação estrutural maior de objetos de informação reutilizáveis (RIOS), ou seja, de ‘pedaços’ de informações granulares, reutilizáveis, que são independentes de mídia”. Ou ainda, “qualquer recurso para apoiar a aprendizagem” [36]. Embora exista uma variedade de definições diferentes a respeito dos objetos de aprendizagem, há um consenso em torno da característica de que OEs são porções reutilizáveis de conteúdo instrucional [25][36].

A fim de complementar a discussão sobre a conceituação dos objetos educacionais, algumas propriedades essenciais que os mesmos devem apresentar são enumeradas a seguir.

- Flexibilidade: OEs podem ser reutilizados em contextos diferentes [22];
- Facilidade de atualização: A fragmentação do conteúdo em pequenas porções facilita sua localização e atualização [22];
- Customização: OEs são totalmente personalizáveis para cursos ou clientes

[22];

- Interoperabilidade e compartilhamento: Funcionam independentemente de plataforma, podendo ser reunidos em bibliotecas de objetos [22][25];
- Indexação e busca: Possibilidade de pesquisa e busca em bancos de objetos [22];
- Melhoramento contínuo: A reutilização de OEs promove o melhoramento de seu conteúdo ao longo do tempo [22];
- Granularidade: OEs devem ser projetados e construídos em pequenos conjuntos com vistas a maximizar as situações de aprendizagem onde o recurso pode ser utilizado [36];
- Composição: OEs podem ser combinados com outros, formando unidades de um curso, ou seja, são tratados como blocos básicos compondo um contexto de aprendizagem [36];

3.1 Modularidade e reusabilidade de objetos educacionais

Embora haja um grande número de abordagens e definições para o que representa ou do que é constituído um objeto educacional, de forma a simplificar a terminologia e delimitar o escopo do presente trabalho, consideraremos que um OE é qualquer recurso digital que pode ser reusado para apoiar a aprendizagem.

Uma das várias motivações de se tratar a criação de material instrucional de forma modular vem do paradigma de programação orientada a objetos. Segundo essa abordagem, o conteúdo instrucional é quebrado em porções menores e independentes que podem ser reaproveitadas em vários contextos diferentes, caracterizando portanto a principal propriedade de um objeto [38]. Complementar a isso, com relação ao aspecto pedagógico, o tratamento do conteúdo em forma de objetos permite flexibilidade e economia de esforços do instrutor na autoria de seu material educativo, visto que o processo se resume à combinação apropriada de objetos educacionais já existentes a fim de montar o contexto de ensino desejado [27]. No caso de algum objeto não atender os propósitos específicos do educador na criação do seu material, a organização em objetos permite readaptação dos mesmos ao propósito pretendido de forma fácil e prática [38]. A autoria de conteúdo educacional, portanto, com a utilização dos objetos educacionais implica em aumento da qualidade e na redução de custos e tempo de

produção tanto de objetos educacionais individuais quanto de cursos *online* inteiros, pois envolve compartilhamento de material existente [24][8].

3.2 Granularidade (nível de agregação) de objetos educacionais

Não há um consenso amplamente estabelecido a respeito da terminologia que envolve a classificação e o relacionamento hierárquico entre os componentes que formam a estrutura de um recurso digital abrangendo desde um arquivo isolado até um curso completo. No entanto, para os propósitos deste trabalho, é suficiente empregarmos a classificação definida em [31], descrita no próximo parágrafo, pois em termos técnicos é a que está estruturalmente mais associada à implementação do modelo de agregação do padrão SCORM, cujos detalhes serão vistos no capítulo 4.

Segundo *ReusableLearning.org* [31], a granularidade ou nível de agregação de um recurso digital se refere ao tamanho deste e indica o nível até o qual um recurso pode ser decomposto, podendo ser chamado de grau de divisibilidade, e até que ponto pretende-se que o recurso seja usado como parte de um recurso maior. Granularidades maiores implicam que o recurso torna-se cada vez menor em seu tamanho e mais difícil de se decompor em partes menores, sendo sucessivamente fragmentado até atingir uma unidade de informação indivisível. Por outro lado, granularidades menores indicam que o recurso torna-se maior e cada vez mais formado por componentes menores.

A granularidade representa um fator importante na definição e na determinação da reusabilidade de um recurso digital. Como exemplo, para uma imagem reusabilidade significa a capacidade de usá-la em várias situações ou contextos diferentes. Por sua vez, a granularidade de um curso determina que sua reusabilidade diz respeito à possibilidade de usar partes desse curso [31]. É importante destacar que a granularidade interfere diretamente na reusabilidade de um recurso digital. Para que haja maior variedade de situações em que um recurso possa ser utilizado, sua granularidade, ou o quão fracionado está o recurso, deve ser alta, indicando generalidade do mesmo. Uma utilização mais restritiva é obtida à medida que a granularidade do recurso diminui, o que implica em dizer que ocorre a especialização do recurso, limitando-o em sua reusabilidade.

Um modelo de agregação foi estabelecido por *Learnativity Foundation* [21], o qual descreve a estrutura hierárquica de classificação dos níveis de agregação de um recurso

digital, que pode ser de um simples arquivo tomado isoladamente a um curso *online* completo com propósitos claramente específicos.

Abaixo são descritos os níveis hierárquicos que formam o modelo de agregação [21][31]:

- **Recurso digital mínimo (*Content Asset*):** equivale a um bem ou recurso digital mínimo; a mídia em sua forma básica ou crua (*raw media*), como imagens, fragmentos de texto, arquivos de áudio etc.
- **Objeto de informação (*Information Object*):** passagens de texto, páginas *Web* que focam em um pedaço único de informação. Ilustrado do ponto de vista pedagógico, objetos de informação concentram-se na explicação de um conceito, princípio ou descrevem um processo. Exercícios individuais são considerados objetos de informação.
- **Objeto de aprendizagem (*Learning Object*):** Coleção de objetos de informação agregados com o intuito de ensinar um objetivo de aprendizagem único.
- **Componente de aprendizagem (*Learning Component*):** Termo genérico que define algo como lições ou cursos que tipicamente possuem múltiplos objetivos de aprendizado e são compostos por múltiplos objetos educacionais.
- **Ambiente de aprendizagem (*Learning Environment*):** equivale à fase final de agregação onde há combinação entre conteúdo e tecnologia, com a qual o aprendiz interage. Um ambiente de aprendizagem corresponde, por exemplo, a um LMS [5] implantado em uma determinada instituição juntamente com todas as suas ferramentas administrativas e de gerenciamento próprias.

A figura 1 (página 20) ilustra graficamente o modelo de agregação *Learnativity* [21]. A figura destaca também a relação de proporcionalidade inversa entre contexto e reusabilidade, ou seja, à medida que o contexto torna-se mais específico (aumenta), a capacidade de reusabilidade é restringida (diminui) e vice-versa.

Há também a tabela 1 (página 21), que mostra, para cada nível hierárquico do modelo de agregação, as respectivas descrições do ponto de vista das propriedades de divisibilidade e reusabilidade de um recurso educacional. A propriedade de divisibilidade diz respeito à capacidade de o recurso pertencente a um determinado nível de agregação ser decomposto em um ou mais recursos de granularidade menor.

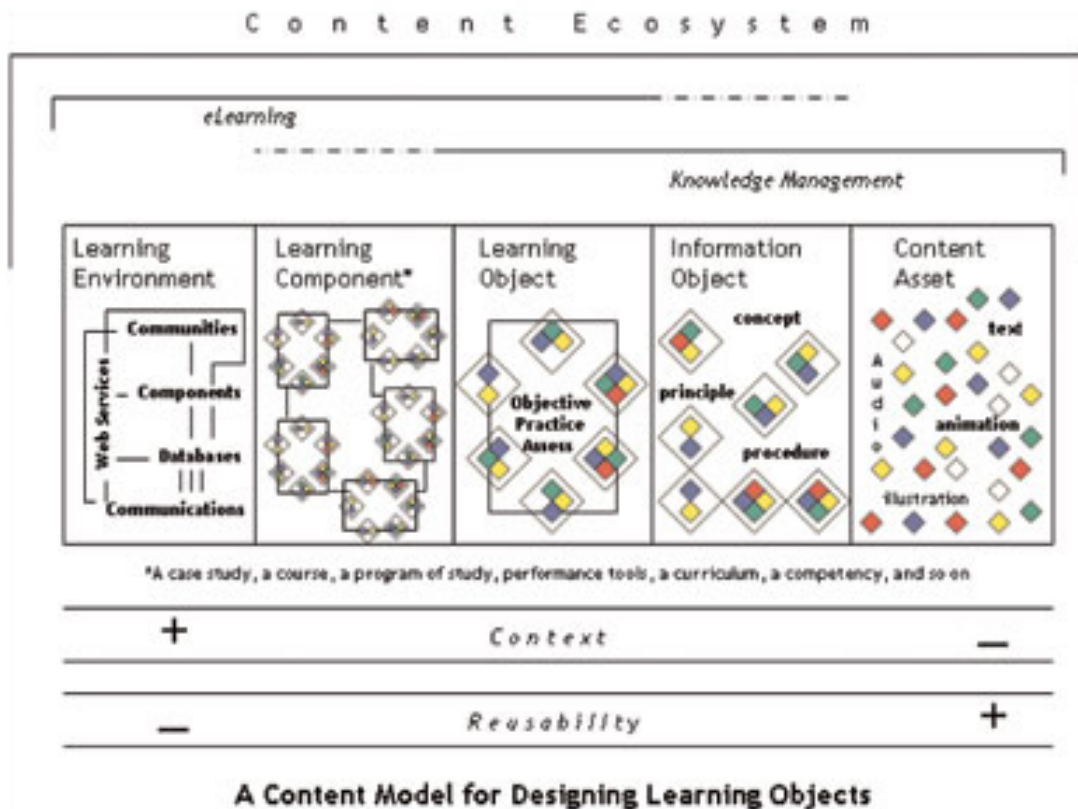


Figura 1: Modelo de agregação *Learnativity* [21] mostrando os níveis de granularidade hierárquicos na composição de um recurso educacional

Tabela 1 - Decomposição e reuso em função da granularidade (adaptado de [17])

Granularidade	Divisibilidade	Reuso
Recurso digital mínimo (<i>content asset</i>)	Indivisível	São usados sem a necessidade de modificações em apresentações e estilos
Objeto de informação (<i>information object</i>)	Decomposto em recursos digitais mínimos	Normalmente reusados em unidades auto-contidas. Também é possível extrair e reutilizar os <i>content assets</i> que o compõem
Objeto educacional (<i>learning object</i>)	Decomposto em recursos digitais mínimos (<i>content assets</i>) e objetos de informação (<i>information objects</i>)	São projetados a fim de serem utilizados como unidades auto-contidas. Eventualmente recursos digitais mínimos e objetos de informação são extraídos a partir de um OE e reutilizados
Componente de aprendizado (<i>learning component</i>)	Decomposto em objetos educacionais	Podem ser usados em sua totalidade, mas estima-se que grande parte do reuso se dá pela extração e utilização de partes menores, sendo estas usualmente os objetos educacionais
Ambiente de aprendizado (<i>learning environment</i>)	Decomposto em conteúdo, tecnologia e processos de suporte ao aprendizado	Os componentes dos ambientes de aprendizado podem ser reusados, porém estes não são considerados objetos reusáveis dentro do contexto abordado nesta classificação

3.3 Interoperabilidade de objetos educacionais

Uma propriedade essencial dos objetos educacionais diz respeito à interoperabilidade. Um recurso digital é interoperável quando possui a capacidade de funcionar normalmente em diversas plataformas de aprendizado, independente da tecnologia ou da implementação empregada nessas plataformas. Além disso, *Reusable Learning* [31] acrescenta que a interoperabilidade de um recurso educacional digital representa o grau ao qual ele é capaz de executar adequadamente em múltiplos sistemas e poder ser usado de forma bem-sucedida no contexto de sua audiência potencial, ou seja, no contexto do público para o qual o recurso foi projetado.

A obtenção da interoperabilidade de recursos instrucionais digitais também pode ser entendida como um meio de perpetuar a existência destes, implicando em minimizar ou, se possível, eliminar esforços de adaptação desses recursos a mudanças tecnológicas na implementação das plataformas ou sistemas nos quais eles executam. Dessa forma, a propriedade da durabilidade é alcançada [35], a qual adicionalmente promove melhoramento da qualidade pedagógica e técnica do recurso educacional digital, ampliando suas possibilidades de reutilização bem-sucedida e contribuindo para a não obsolescência do recurso ao longo do tempo.

3.4 Modelos de representação de metadados de objetos educacionais

Para que os objetos educacionais possam ser compartilhados e reutilizados, se faz necessária a criação de mecanismos de classificação, armazenamento, pesquisa e recuperação eficientes dos mesmos. Para tal, fazemos uso de metadados que descrevam características relevantes aos objetos de forma que eles possam ser catalogados em repositórios de objetos educacionais reusáveis e posteriormente recuperados e gerenciados por sistemas de gerenciamento de aprendizagem (*Learning Management Systems – LMSs*) [35]. Metadados representam os dados sobre os dados [25], e são importantes para a identificação, organização e recuperação da informação digital [33]. Dessa maneira, através deles são atribuídos aos objetos educacionais benefícios como acessibilidade, interoperabilidade e durabilidade. Alguns dos padrões de metadados mais conhecidos são DC (*Dublin Core*) [12], LOM (*Learning Object Metadata*) [10] do IEEE LTSC e ARIADNE (*Alliance of Remote Instructional Authoring and Distribution Networks for Europe*) [6]. Tais padrões são descritos

brevemente a seguir.

3.4.1 Dublin Core

O Dublin Core, proposto pela DCMI (Dublin Core Metadata Initiative) é um dos padrões mais antigos, nascido a partir do consenso de um grupo interdisciplinar de profissionais composto por bibliotecários, lingüistas, museólogos, entre outros. O DC é um formato proposto para descrever qualquer tipo de recurso, incluindo coleções de documentos e de mídias não eletrônicas, sendo dessa forma caracterizado por ser mais estruturado e flexível, fornecendo um conjunto de 15 atributos do padrão ISO/IEC 11.179 para descrição de seus elementos de dados [25].

As principais vantagens da utilização do padrão Dublin Core podem ser enumeradas abaixo, segundo [25]:

- Simplicidade: conjunto de elementos fácil de ser entendido, podendo ser usado tanto por leigos quanto por especialistas em descrição de recursos.
- Interoperabilidade semântica: os elementos podem ser utilizados para descrever recursos de diversas áreas de conhecimento. Tal facilidade permite que sejam realizadas pesquisas sem se importar com especificidade dos diversos campos de conhecimento.
- Consenso internacional: o padrão DC vem sendo utilizado em projetos em cerca de 20 países.

Segundo Moura [25], o padrão DC é bastante utilizado para a descrição de uma variedade de recursos existentes na Internet, com o propósito de ser um meio de comunicação e de busca de informações. Tem sido adotado por importantes instituições e agências governamentais [11].

3.4.2 IEEE *Learning Object Metadata* (IEEE-LOM)

Outro padrão conhecido é IEEE-LOM (*Learning Object Metadata*), proposto pelo IEEE LTSC [18]. Essa especificação descreve metadados de objetos educacionais, definindo atributos que os descrevem de forma total e mais adequada [25]. Segundo Moura [25], “um dos objetivos da norma é o de facilitar a busca, avaliação, aquisição e uso de objetos

educacionais por parte de aprendizes, instrutores ou processos automatizados de software. Outro objetivo é a facilidade de compartilhamento e intercâmbio de OEs (incluindo a possibilidade de um OE ser utilizado por LCMSs - sistemas de gerência de conteúdo educacional - de fabricantes diferentes), possibilitando o desenvolvimento de catálogos, ao mesmo tempo em que se leva em consideração a diversidade de contextos culturais e de língua onde os OE e seus metadados possam ser empregados”.

A figura 2 (página 25) mostra a árvore de atributos completa de descrição de elementos de dados que compõem o padrão de metadados IEEE-LOM.

3.4.3 ARIADNE

O padrão ARIADNE [6] (*Alliance of Remote Instructional Authoring and Distribution Networks for Europe*) surgiu na Europa a fim de suprir as necessidades específicas educacionais e requerimentos de treinamento profissional da comunidade européia [25]. Este padrão tem como característica principal o suporte às diversidades lingüística e cultural e ao compartilhamento e reuso de recursos de conhecimento [25]. Em termos do conjunto de especificações, o padrão ARIADNE define sete categorias em que se agrupam os elementos de dados:

- **General**: grupo de informações gerais que descrevem o objeto educacional;
- **Semantics**: classificação semântica do objeto educacional;
- **Pedagogical**: descrição pedagógica e educacional do objeto;
- **Technical**: grupo que descreve requerimentos e características técnicas do objeto;
- **Conditions**: grupo de elementos que descreve condições para uso do objeto educacional;
- **Meta-metadata**: elementos que descrevem a instância do metadado;
- **Annotation**: grupo de elementos opcionais que descrevem pessoas ou organizações que fazem anotações sobre o objeto educacional;

A Figura 3 (página 26) ilustra o grafo de referência entre as especificações do IEEE, IMS, AICC, ADL SCORM (sendo este visto em 4.2, página 28), ARIADNE e Dublin Core, entendidas como sendo os principais padrões usados [20].

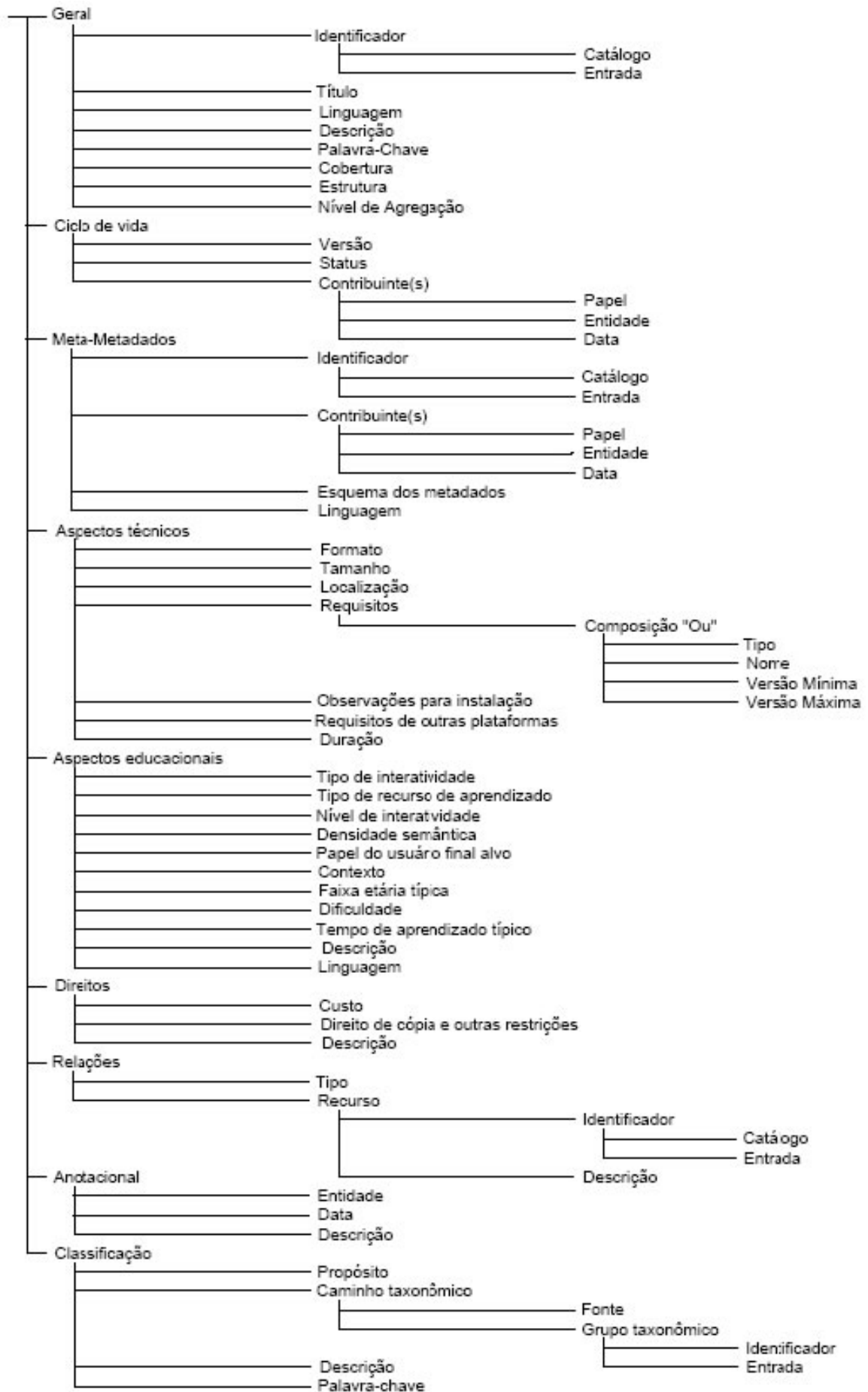


Figura 2: Árvore completa de atributos do padrão IEEE-LOM v1.0 [10]

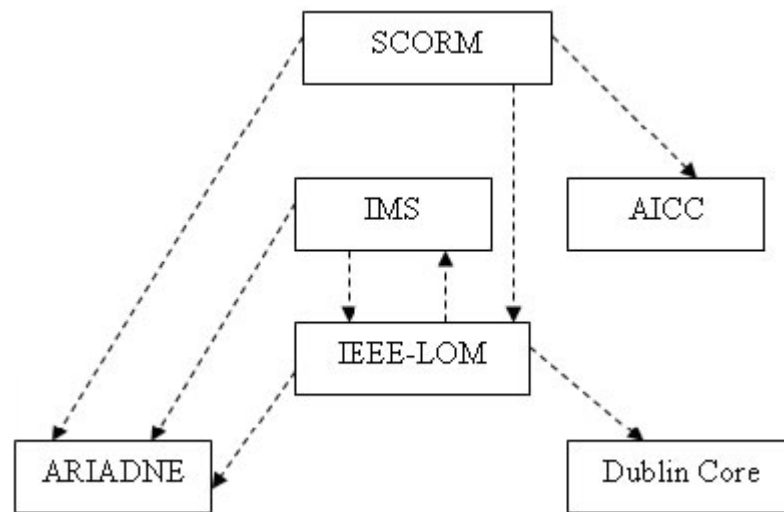


Figura 3: Grafo de referência entre as especificações do IEEE-LOM, IMS, AICC, SCORM, ARIADNE e Dublin Core, extraído de [20]

4 PADRONIZAÇÃO DE OBJETOS EDUCACIONAIS E O MODELO DE REFERÊNCIA SCORM

Nas seções seguintes, pretende-se conduzir uma discussão sobre os trabalhos desenvolvidos na área da padronização de material instrucional, focando inicialmente aspectos do *E-Learning*, passando logo após à abordagem com relação aos objetos educacionais através do modelo de referência SCORM.

4.1 Desenvolvimento de padrões para *E-Learning*

Ao longo dos últimos anos, a popularização dos serviços *Web* e a crescente utilização de ambientes de ensino a distância estimularam a organização dos recursos instrucionais através de ferramentas de classificação e indexação a fim de torná-los mais amplamente disponíveis para uso. Com isso, diversas tecnologias, como XML, XSD e RDF, passaram a ser utilizadas com o objetivo de estruturar tais recursos e prover mecanismos de busca, implementando o conceito de metadados, abordado no capítulo anterior. Contudo, a difusão de inúmeras soluções particulares baseadas em sistemas proprietários motivou esforços na direção da padronização do uso de tecnologias, abrangendo os processos de modelagem, desenvolvimento e distribuição de material instrucional via *Web* [16].

Diversas organizações, grupos e comitês desenvolveram trabalhos de padronização para *E-Learning*. Dentre esses, destacam-se *Aviation Industry CBT Committee (AICC)* [4], *IEEE's Learning Technology Standardization Committee (LTSC)* [18], *IMS Global Learning Consortium* [19] e *US Department of Defense's Advanced Distributed Learning (ADL) initiative* [3]. O objetivo principal de tais organizações é estabelecer referências necessárias para interoperabilidade e conformidade entre diversos fornecedores [16], ou seja, garantir que produtos de *E-Learning* em conformidade com os padrões sejam compatíveis entre si e executem em quaisquer plataformas independentemente de fabricantes e características de implementação particulares das mesmas. Além disso, o emprego de padrões abertos é desejável, visto que desvincula dos avanços tecnológicos constantes a implementação e o funcionamento do material instrucional digital, livrando-o da possibilidade de tornar-se obsoleto e mantendo-o funcional ao longo do tempo.

Como foco de desenvolvimento dos padrões discutidos, duas áreas importantes relacionadas aos materiais instrucionais são visadas [16]:

- especificações de modelos para padronizar o formato dos conteúdos, sintaxe e semântica dos metadados;
- especificações referentes à arquitetura dos sistemas, componentes e tecnologias para prover suporte ao *E-Learning*;

4.2 O modelo de referência SCORM

O padrão SCORM (*Sharable Content Object Reference Model* – Modelo de Referência para Objetos de Conteúdo Compartilháveis) consiste em um modelo que descreve um conjunto de especificações técnicas e de referência para a apresentação de conteúdo de ensino e aprendizagem via *Web* (*E-learning*) [20]. Surgiu como uma iniciativa conjunta entre o departamento de defesa dos EUA e a união da indústria de tecnologia em 1997 com o objetivo de fomentar a adoção de um padrão único para os sistemas de educação a distância. Criou-se dessa união o consórcio ADL (*Advanced Distributed Learning Co-Labs*) [3], responsável pela apresentação do SCORM em 1999 como um padrão ideal, pois reuniu todos os outros padrões disponíveis no mercado, já citados na seção 4.1. A figura 4 (página 29) ilustra a organização do SCORM 2004 3rd *Edition* como um conjunto de especificações dessas organizações, que podem estar contidas ou referenciadas no modelo. A figura 3 (página 26) pode ser usada complementarmente para ilustrar, na forma de um grafo, as relações de inclusão ou referência entre os modelos, figurando o SCORM no topo do diagrama.

O padrão SCORM define um modelo de como se fazer e como se executar cursos baseados na *Web*. Suas normas são uma coleção de especificações, criando um abrangente e apropriado grupo de habilidades do ensino via *Web* [9] que confere certas propriedades, ou atributos, essenciais ao material educacional digital para ensino a distância [14][3], as quais são enumeradas abaixo.

- **Reusabilidade:** capacidade dos componentes instrucionais de serem usados em múltiplas aplicações, cursos e contextos;
- **Durabilidade:** capacidade dos componentes instrucionais de resistir às mudanças tecnológicas através do tempo - ou seja, manterem-se funcionando adequadamente à medida que novas tecnologias substituam as atuais - sem

necessidade de reprojeto, reconfiguração ou recodificação;

- **Acessibilidade:** capacidade dos componentes instrucionais de serem localizados e acessados a partir de múltiplos locais e serem entregues em outros locais;
- **Interoperabilidade:** capacidade dos componentes instrucionais de serem desenvolvidos em um sistema e serem usados em outro sistema;

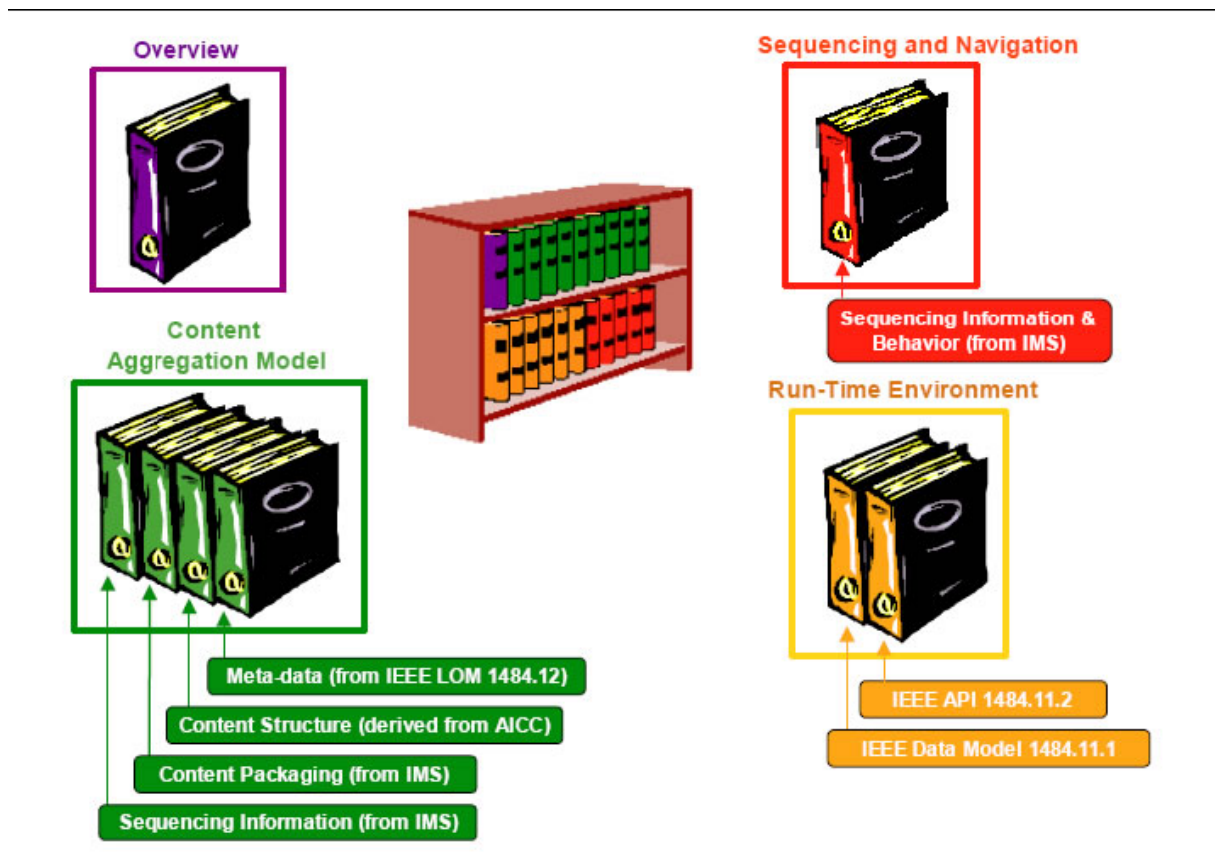


Figura 4: SCORM como conjunto de especificações. No caso desta figura, o SCORM 2004 3rd Edition é representado

Além dos benefícios promovidos pelo uso do SCORM com relação à produção de material educacional de forma modular, reusável, interoperável e durável, a adoção do padrão também promove uma mudança de paradigma no tratamento do conteúdo instrucional junto aos ambientes de educação a distância, ou Sistemas de Gerência de Aprendizado (LMS). Modelos de ambientes de EAD não conformantes com o padrão, e conseqüentemente não aderentes aos moldes orientados a objetos educacionais, seguem uma estrutura não flexível de manipulação de seu conteúdo instrucional, impondo ao mesmo uma organização monolítica [8]. Dessa forma, os cursos nos LMSs tradicionais são relegados a serem manipulados e distribuídos obrigatoriamente nesse próprio nível de granularidade (veja seção 3.2). Por outro

lado, um LMS conformante com o modelo de referência SCORM, possuindo portanto a capacidade de lidar com o conteúdo instrucional por meio de objetos educacionais, permite que os cursos agora sejam manipulados de forma diferente e flexível. A figura 5 (página 30) mostra o novo tipo de interação entre o sistema de gerência de aprendizado (LMS) e o conteúdo instrucional, onde os cursos agora podem ser montados com base nos objetos educacionais que os compõem, promovendo flexibilidade e personalização de acordo com as necessidades de aprendizado específicas e particulares de cada usuário do ambiente [16].

Percebemos portanto que o emprego do padrão SCORM torna concretas as propriedades essenciais do conceito de objetos educacionais, viabilizando a aplicação dessa inovadora e vantajosa abordagem aos processos de desenvolvimento e distribuição de conteúdo instrucional para os ambientes de educação a distância.

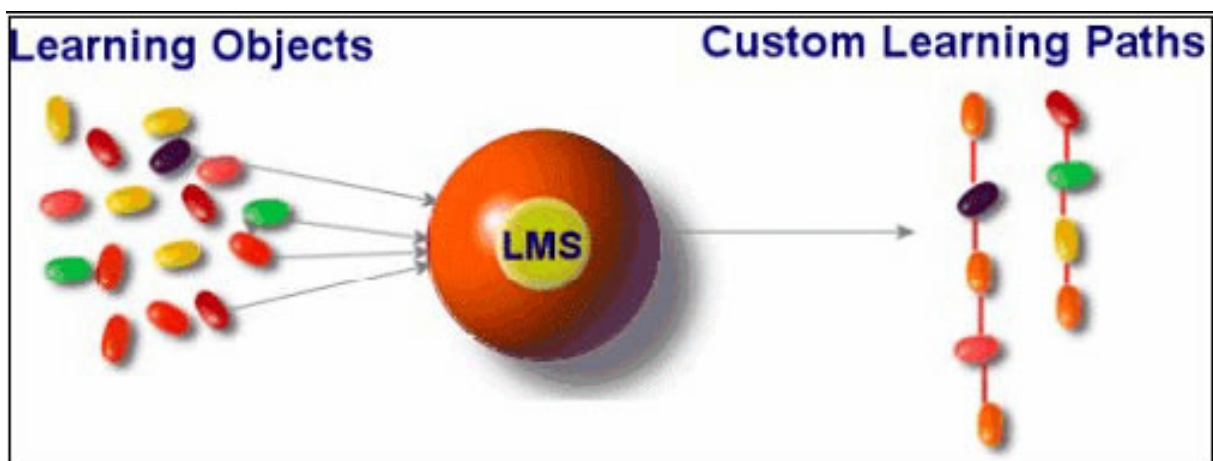


Figura 5: Interação entre LMS e objetos educacionais viabilizada pela adoção do modelo SCORM

De acordo com o mostrado na figura 4 (página 29), o conjunto de especificações SCORM 2004 Terceira Edição é dividido em quatro componentes, enumerados a seguir.

- Visão Geral (*Overview*);
- Modelo de Agregação de Conteúdo (*Content Model Aggregation*);
- Ambiente de Tempo de Execução (*Run-time Environment*);
- Seqüenciamento e Navegação (*Sequencing and Navegation*)

4.2.1 Modelo de Agregação de Conteúdo (*Content Aggregation Model*)

Formado pelas especificações de descrição de metadados definidas em IMS [19] e IEEE-LOM [10], o modelo de agregação de conteúdo determina a forma como os conteúdos de ensino devem ser criados e agrupados, sendo subdividido em três componentes:

- Modelo de Conteúdo (*Content Model*)
- Empacotamento de Conteúdo (*Content Packaging*)
- Dicionário de Metadados (*Metadata*)

♦ Modelo de Conteúdo (*Content Model*)

O modelo de conteúdo define a forma de representação do material instrucional, ou seja, dispõe um conjunto de elementos necessários para compor a estrutura do conteúdo instrucional, também considerada como granularidade (veja seção 3.2). Esses elementos são descritos a seguir.

- ***Asset***: elemento análogo ao definido na classificação da granularidade de OEs (seção 3.2), um *asset* corresponde a um recurso digital mínimo, como texto, imagem, som, uma página HTML ou qualquer outro formato *Web*. Ou seja, um *asset* equivale à representação SCORM da granularidade *content asset* presente no modelo de agregação abordado seção 3.2.
- ***Sharable Content Object (SCO)***: o objeto de conteúdo compartilhável representa um conjunto de um ou mais *assets*, e obrigatoriamente inclui um mecanismo de comunicação com o sistema de gerência de aprendizado (LMS), sendo esta última característica abordada na seção 4.2.2.
- ***Content Aggregation***: a agregação de conteúdo consiste na organização seqüencial do conteúdo instrucional, constituído por *assets* e SCOs, na ordem em que ele deve ser acessado ou exibido. A figura 7 (página 35) ilustra um exemplo de hierarquia de itens que formam a agregação de conteúdo.

♦ Empacotamento de Conteúdo (*Content Packaging*)

O empacotamento de conteúdo é um conjunto de regras e normas para reunir o

conteúdo educacional em blocos, ou pacotes, de forma bem definida e estruturada com o objetivo de facilitar sua distribuição e posterior uso junto aos ambientes de aprendizado (LMSs). Esse assunto, devido a sua relevância ao enfoque do presente trabalho, será visto com mais detalhes na seção 4.3.

♦ Dicionário de Metadados (*Metadata*)

Dentro do Modelo de Agregação de Conteúdo do padrão SCORM, os metadados são utilizados com a finalidade de descrever de forma coerente e precisa cada componente do modelo de conteúdo (*assets*, *SCOs* e *content aggregation*), de modo que eles possam ser arquivados e pesquisados de forma rápida e eficiente [20].

4.2.2 Ambiente de Tempo de Execução (*Run-time Environment*)

Baseado no padrão AICC [4], o ambiente de tempo de execução tem por objetivo permitir que os conteúdos de ensino possam ser visualizados em diferentes sistemas de gerência de aprendizado (LMSs) e administrar o comportamento dos conteúdos educacionais [20]. Dessa forma, o ambiente de execução define protocolos de comunicação e realiza a interação entre o navegador *Web* do usuário executando o conteúdo educacional e o LMS que o disponibiliza. Possui as funções de iniciar a execução do objeto, definir e recuperar informações sobre suas variáveis de estado, monitorar seu andamento e progresso corrente, tratar eventuais erros de execução, entre outras. As funcionalidades definidas para o ambiente de tempo de execução são reunidas em três elementos, descritos a seguir.

- **Mecanismo de lançamento (*Launch*)**: mecanismo presente em todos os LMSs que possui a responsabilidade de enviar os recursos de ensino ao navegador do aluno (navegador *Web*). Os componentes do Modelo de Conteúdo que podem ser enviados pelo LMS são os *assets* e os *SCOs* [2].
- **Adaptador de API (*Application Program Interface adapter*)**: mecanismo ou o meio pelo qual os *SCOs* e o LMS se comunicam. A finalidade da *API Adapter* é facilitar a interoperabilidade e a reutilização dos conteúdos de ensino, criando uma forma padrão de troca de informações, encapsulando os detalhes da comunicação entre o aluno e o ambiente de EAD [2].

- **Modelo de Dados (*Data Model*):** criado para permitir que diferentes LMSs possam processar e manipular um conjunto definido de informações provenientes de qualquer SCO. Desta maneira, é capaz de proporcionar aos SCOs uma forma padrão de acessar os dados do LMS. O modelo de dados é um protocolo para possibilitar a comunicação entre o SCO e o LMS através uma linguagem que ambos entendam [2].

A figura 6 (página 33) mostra os componentes envolvidos na comunicação entre a plataforma LMS e um SCO do ponto de vista da implementação cliente-servidor. O servidor LMS inicia o SCO, o qual executa no navegador do cliente (*browser*), via adaptador API (*API adapter*). Esta interface é parte da plataforma LMS (representada pela área hachurada) residente no lado do cliente e intermedia a comunicação com o LMS através de métodos definidos pela API e implementados em *JavaScript* no SCO. Portanto, um SCO é capaz de estabelecer a comunicação com o servidor LMS via conjunto de funções *Javascript* definidas na API e transferir informações através do Modelo de Dados (*Data Model*).

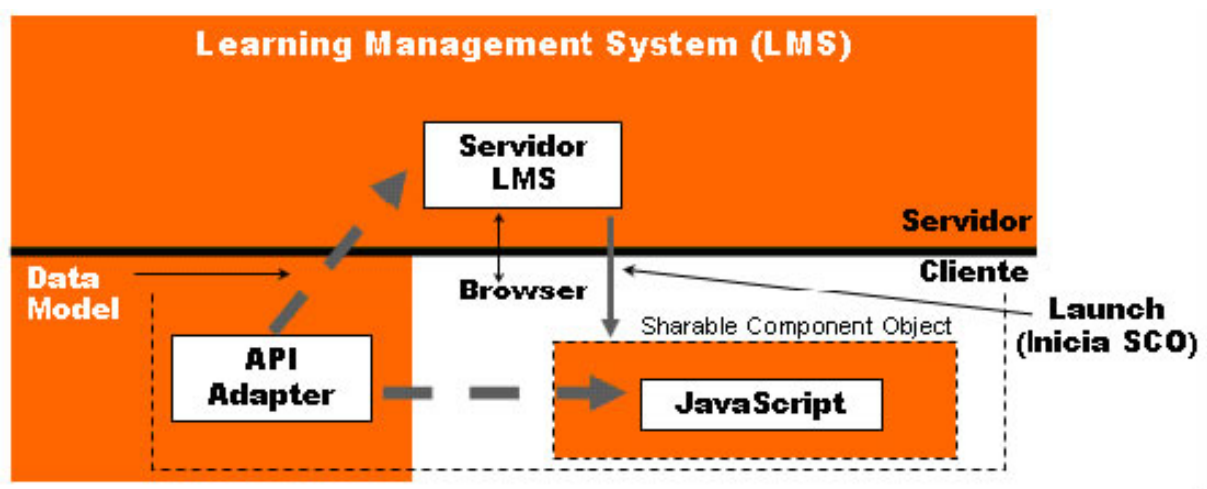


Figura 6: Ambiente de tempo de execução de uma plataforma de E-Learning, extraído de [20]

4.2.3 Seqüenciamento e Navegação (*Sequencing and Navigation*)

Descreve como os conteúdos SCORM podem ser seqüenciados e como um LMS compatível deve interpretar regras de seqüenciamento. Por meio dessas regras, juntamente com que foi definido na agregação de conteúdo, um ambiente de aprendizado (LMS) é capaz de determinar o que e quando deve ser entregue e rastrear o progresso do estudante durante o

curso [13].

4.3 Empacotamento de conteúdo (*content packaging*) no modelo SCORM

Um pacote SCORM é formado basicamente por um documento XML, chamado **manifesto**, que descreve os conteúdos e a organização do bloco, e pelos arquivos físicos que constituem os recursos educacionais, que podem ser SCOs ou *assets* (veja figura 7, página 35). Com o objetivo de facilitar a distribuição pela *Web*, todos esses componentes são armazenados em arquivos do tipo PIF (*Package Interchange File*), que podem ter diversos formatos, sendo os mais comuns zip, jar, rar, arj, tar e cab [20]. Existem duas formas ou perfis de organização do pacote. O *Resource Package Application Profile* (perfil de aplicação de pacote de recurso) define apenas recursos como integrantes do pacote, enquanto o *Content Aggregation Package Application Profile* (perfil de aplicação de pacote de agregação de conteúdo) determina que o pacote é constituído, além dos recursos, também pela estrutura hierárquica que descreve a agregação do conteúdo do pacote [20] (como visto anteriormente em 4.2.1). Adicionalmente, metadados podem ser associados a qualquer elemento dentro do pacote, podendo descrever recursos individuais (SCOs ou *assets*) ou conjuntos deles (agregações de conteúdo).

4.3.1 Estrutura do arquivo XML de manifesto do pacote SCORM

Todo pacote SCORM deve conter um arquivo manifesto no seu nível raiz com o nome obrigatório de **imsmanifest.xml**, sob pena de não ser possível desempacotá-lo e usá-lo em um LMS. Esse arquivo XML especial, herdado das especificações IMS e IEEE-LOM, descreve o pacote através de informações que incluem metadados, organizações e recursos. Resumidamente, o manifesto reúne tudo a respeito do conteúdo instrucional, e de como ele deve ser acessado e usado, devendo listar todos os arquivos que compõem o conteúdo do objeto.

Metadados são usados na descrição de recursos, *assets* e SCOs através da *tag* XML `<metadata>`, a qual pode aninhar outras diversas *tags* XML que descrevem os vários atributos referentes ao objeto educacional. A figura 2 (página 25) mostra a árvore completa cobrindo todos os atributos de metadados empregados no modelo de agregação de conteúdo do padrão

SCORM.

A organização, ou diagrama de estrutura de conteúdo [7], é um dos três elementos-chave contidos no manifesto de um pacote de conteúdo. Consiste de uma descrição de como o conteúdo deve ser apresentado ao aprendiz, definindo sua ordem de apresentação e controlando a hierarquia de conteúdos. A seção *organizations*, representada pela tag XML `<organizations>`, é composta de uma ou mais organizações, representadas cada qual pela tag XML `<organization>`. A figura 7 (página 35) mostra graficamente o conteúdo de uma organização através da árvore de itens. Cada item da hierarquia pode referenciar um recurso ao qual está relacionado, e que se encontra descrito na seção de recursos do manifesto (será vista a seguir).

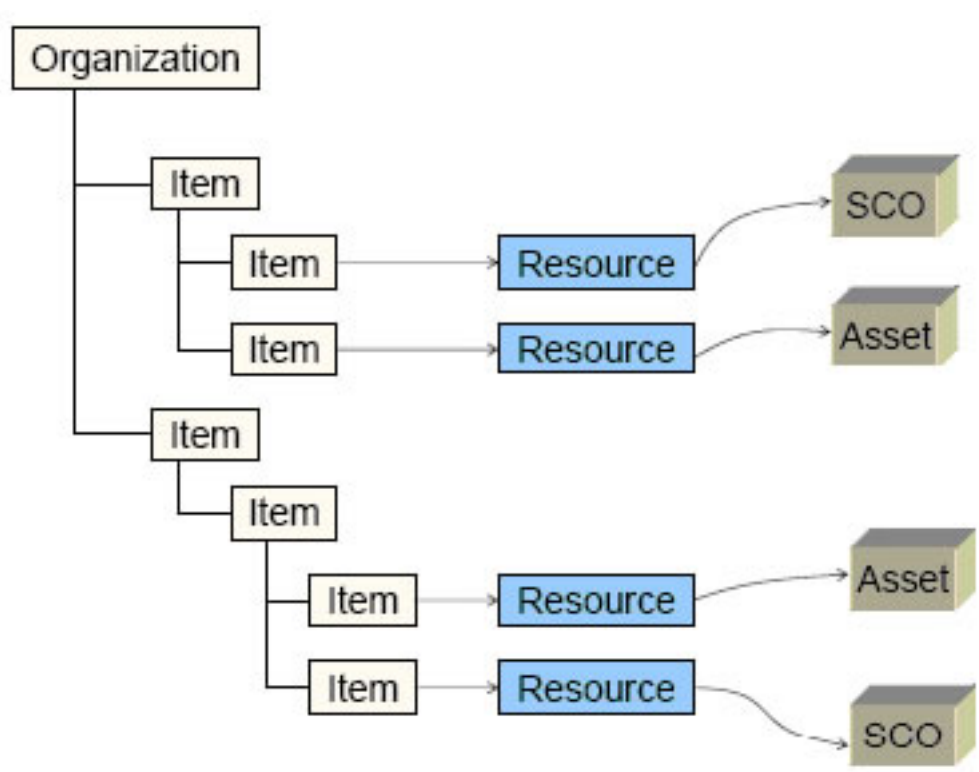


Figura 7: Seção Organization do arquivo XML de manifesto mostrando o mapeamento de recursos na agregação de conteúdo de um pacote SCORM

Um recurso, no contexto do empacotamento de conteúdo SCORM, consiste de uma referência a um arquivo ou a uma coleção de arquivos (elementos de mídia quaisquer) relacionados que compõem aquele recurso. Cada recurso é descrito individualmente pela *tag* `<resource>`, e pode ser acompanhado de metadados associados, através da *tag* `<metadata>`, e de outros dois elementos, cujas *tags* são `<file>` e `<dependency>`. Um elemento arquivo (*file*) consiste de uma referência a um arquivo físico não definido anteriormente como um recurso (*resource*). Uma dependência, por sua vez, corresponde a uma referência a um recurso já definido. O arquivo de manifesto contém uma seção *resources*, indicada pela *tag* XML `<resources>`, a qual deve conter o(s) recurso(s) relacionado(s) ao objeto educacional (um ou mais elementos `<resource>`). A figura 8 (página 36) mostra a estrutura interna de um pacote de conteúdo SCORM, ilustrando todos os seus elementos integrantes vistos nesta seção. A representação do arquivo XML do manifesto é encontrada na figura 9 (página 37).

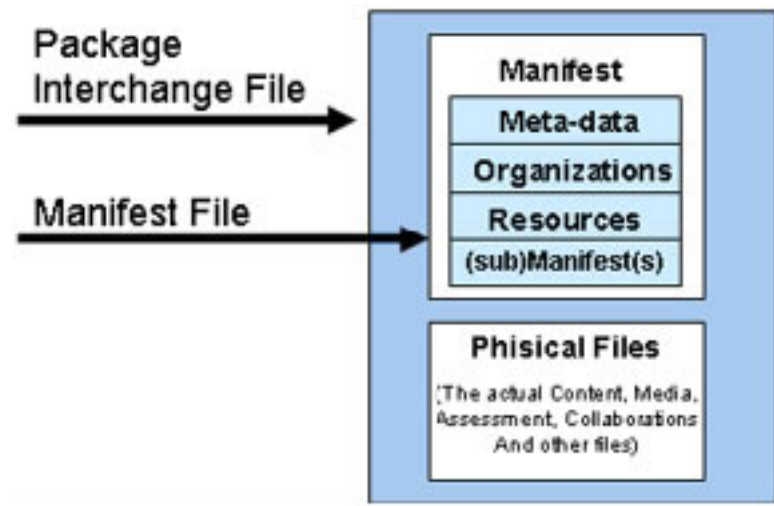


Figura 8: Componentes de um pacote de conteúdo SCORM

```
<manifest>
  <metadata/>
  <organizations/>
  <resources>
    <resource identifier="RESOURCE1" adlcp:scormType="sco" type="webcontent"
      href="lesson1.htm">
      <file href="lesson1.htm"/>
    </resource>
    <resource identifier="RESOURCE2" adlcp:scormType="sco" type="webcontent"
      href="intro1.htm">
      <file href="intro1.htm"/>
    </resource>
    <resource identifier="RESOURCE3" adlcp:scormType="asset"
      type="webcontent" href="content1.htm">
      <file href="content1.htm"/>
    </resource>
    <resource identifier="RESOURCE4" adlcp:scormType="sco" type="webcontent"
      href="summary1.htm">
      <file href="summary1.htm"/>
    </resource>
  </resources>
</manifest>
```

Figura 9: Exemplo do arquivo XML de manifesto mostrando os elementos *metadata*, *organizations* (ambos neste caso vazios) e *resources*

5 PROPOSTA DA FERRAMENTA IOPACKAGER

Dada a relevância do processo de padronização no emprego de tecnologias de comunicação e informação, especialmente com relação às tendências recentemente discutidas e implementadas no âmbito educacional, torna-se claro que novos rumos da educação a distância começam a ser explorados e gradualmente aceitos. Dessa forma, uma das questões principais levantadas pelo autor deste trabalho é como tornar acessível o empacotamento de conteúdo educacional para o maior público desenvolvedor possível, de forma que os benefícios do uso de objetos educacionais sejam amplamente difundidos e que realmente agreguem valor na promoção de uma educação de qualidade. Sendo parte do contexto educacional, o desenvolvimento adequado de objetos educacionais para *E-Learning* não deve se restringir a profissionais da área técnica, mas sim abranger também, e principalmente, profissionais da educação, como professores e educadores, garantindo que eles tenham autonomia e total suporte tecnológico para realizar seu trabalho.

5.1 Características gerais e vantagens do empacotador

Com base no exposto, a proposta deste trabalho é desenvolver uma ferramenta que consiste de um programa PHP capaz de converter automaticamente um objeto educacional fornecido pelo usuário em um pacote de conteúdo conformante com a norma SCORM 2004 Terceira Edição. Como parte da ferramenta, assim como componente integrante do pacote de conteúdo SCORM, a criação da seção de metadados, utilizada para descrever o objeto educacional, será provida ao usuário através de uma interface gráfica. O objeto educacional de entrada da ferramenta deverá estar organizado em uma estrutura de diretórios qualquer contendo todos os arquivos do material instrucional. A saída correspondente será um pacote de agregação de conteúdo, cujos detalhes estão em 5.3. O editor gráfico de pacotes de conteúdo e metadados RELOAD [29] foi de grande utilidade na elaboração das especificações deste empacotador e na geração automática de manifestos XML exemplos para finalidade de testes, principalmente no desenvolvimento do modelo de comunicação que os pacotes produzidos irão implementar. Outras características da ferramenta seguem abaixo:

- implementa modelo que dispensa interface gráfica na definição de recursos de

pacote de conteúdo;

- isenta o usuário da necessidade de conhecer detalhes técnicos do padrão SCORM para adequar seu material instrucional à especificação;
- permite que materiais educacionais tradicionais, ou seja, não padronizados, sejam armazenados, pesquisados e executados (visualizados) em ambientes conformantes com o padrão SCORM 2004 Terceira Edição [2] e em sistemas de gerência de aprendizado (LMSs).

5.2 Desenvolvimento da interface de criação de metadados

De modo que o objeto educacional a ser empacotado seja apropriadamente classificado para busca e armazenamento em LMSs SCORM, a geração de metadados associados é necessária. Para isso, é disponibilizada uma interface na qual campos correspondentes a cada um dos atributos definidos em IEEE-LOM (figura 2, página 25) e empregados pelo SCORM estão presentes para serem preenchidos. A implementação é simples, consistindo apenas de um formulário contendo todos os referidos campos de atributos. Após o usuário completar a inserção dessas informações, elas serão convertidas na estrutura XML equivalente conformante com o SCORM 2004 Terceira Edição para posteriormente serem agregadas ao manifesto do pacote de agregação como metadados descritivos do objeto educacional.

5.3 Desenvolvimento do gerador de pacotes de agregação

O objetivo principal do processo de criação de pacotes é gerar o arquivo XML de manifesto **imsmanifest.xml** que descreve a estrutura do objeto educacional. Portanto, a funcionalidade do gerador se concentra em montar o arquivo XML à medida que identifica os componentes deste durante a análise dos arquivos de entrada. Como anteriormente visto, o pacote é composto por organizações, recursos e metadados. As questões de implementação que envolvem o componente organizações será visto na seção 5.4.

O processo de classificação dos arquivos componentes do objeto educacional será com base na regra a seguir. Arquivos baseados em texto, cujas extensões podem ser, por exemplo, html, css, php, jsp entre outros, serão definidos individualmente como recursos, que poderão

ser formados por elementos *<file>*, ou seja, referências a arquivos que não são recursos, ou elementos *<dependency>*, que são referências a outros arquivos já definidos como recursos. As demais mídias *Web*, como arquivos de imagens, animações, entre outros, serão considerados apenas arquivos de dependência comuns referenciados por recursos, ou seja, elementos *<file>*.

O procedimento de classificação e descrição dos recursos consiste em varrer o diretório de entrada que compõe o objeto educacional, analisar cada arquivo buscando localizar suas dependências (ou seja, que outros arquivos ele referencia dentro da estrutura de diretórios) se ele for baseado em texto, e a partir disso acrescentá-los à seção *<resources>* do arquivo de manifesto. A localização de dependências será através de busca por expressões regulares baseada em um conjunto pré-estabelecido de padrões de reconhecimento, os quais podem ser associados separadamente a cada extensão de arquivo baseado em texto que a ferramenta é capaz de distinguir. Essa característica será configurável, ou seja, será possível definir internamente à ferramenta extensões de arquivo com seus respectivos grupos individuais de padrões de reconhecimento de arquivos referenciados.

5.4 Desenvolvimento do modelo de agregação e do mecanismo de lançamento do objeto educacional

Ainda como parte integrante do processo de criação do arquivo manifesto, e tendo sido devidamente identificados e descritos os recursos na seção *resources*, o passo seguinte do empacotador consiste em prover o meio de comunicação do objeto educacional sendo empacotado com o ambiente de aprendizado (LMS), de modo que ele possa ser lançado por este e visualizado. Neste sentido, a proposta deste trabalho restringe-se apenas a incluir o mecanismo de *Launch* (seção 4.2.2) por meio da implementação de dois métodos da API definida para o ambiente de tempo de execução do SCORM (seção 4.2.2), os quais são *Initialize()*, cuja finalidade é estabelecer a comunicação entre LMS e objeto educacional, lançando este logo após, e *Terminate()*, método cuja chamada indica que o objeto não está mais sob o monitoramento do LMS, sendo encarregado de finalizar a comunicação iniciada por *Initialize()*. É importante mencionar que quaisquer outros métodos do modelo de dados (também em 4.2.2), por serem dependentes da natureza do objeto educacional e necessitarem de intervenção humana para serem programados, não serão oferecidos como funcionalidade

do IOPackager, visto que a premissa básica deste é implementar uma solução genérica de empacotamento.

Um detalhe importante do Modelo de Agregação de Conteúdo (seção 4.2.1) envolvido na geração do manifesto e ainda não abordado será melhor discutido aqui. Para os propósitos do presente projeto e visando simplificar a representação do objeto educacional, todos os recursos identificados pela ferramenta e incluídos na seção *resources* do manifesto serão por padrão do tipo *asset* (veja seção 4.2.1). O objetivo dessa decisão é tornar possível a criação de um mecanismo genérico de comunicação para quaisquer objetos que viessem a ser empacotados. A explicação é que obrigatoriamente, para que a conformidade com o SCORM seja obtida, todos os SCOs definidos na seção *resources* devem ser providos de mecanismos de comunicação individuais, ou seja, eles devem possuir a habilidade de serem lançados e monitorados pelo LMS. Não haveria motivos para definir arbitrariamente diversos recursos como SCOs, pois, dentro das limitações da solução proposta por este trabalho, arruinaria a adaptação genérica de um objeto qualquer ao padrão SCORM de forma que ele tivesse a habilidade de ser lançado e visualizado. Portanto, apenas um SCO será definido e incluído à seção *resources*, o que será visto a seguir.

A implementação da comunicação dar-se-á da seguinte forma. Do ponto de vista do manifesto, uma única organização será criada (veja a seção 4.3.1) e incluída à seção *organizations*, referenciando o único SCO definido em *resources*, que será o arquivo **sco.html**. A função deste arquivo é abrigar a implementação dos métodos *JavaScript* da API de comunicação do objeto educacional e redirecionar a execução para a página principal do objeto educacional, definida pelo usuário do empacotador. Isso é necessário porque não haverá nenhuma outra referência aos recursos do OE a não ser através do arquivo *sco.html*. Ou seja, após estabelecida a comunicação por meio das chamadas de API, o LMS encontrará apenas um SCO pertencente ao OE (o arquivo *sco.html*) e irá lançá-lo. Este SCO, por sua vez, deve invocar os arquivos do OE propriamente dito, o que fará através de uma página principal do objeto educacional, definida pelo usuário do empacotador. Após a visualização do objeto ser completada, o SCO será notificado por meio de métodos *JavaScript* implementados no próprio arquivo *sco.html*, invocará *Terminate()* e encerrará a comunicação com o LMS. É importante salientar que, embora o SCO esteja apto para trocar informações com o LMS, não consta da proposta deste trabalho realizar isso. A finalidade da comunicação descrita aqui, como já foi afirmado, é permitir que o objeto convertido ao SCORM seja apenas visualizado

e prover que o pacote gerado seja conformante com a especificação. Nenhuma informação será trocada, de fato, com o sistema de gerência de aprendizado.

5.5 Validação do IOPackager

Para a validação da ferramenta empacotadora, será usado um utilitário desenvolvido e disponibilizado pela ADL, o SCORM® 2004 3rd *Edition Conformance Test Suite* [1], a fim de verificar a conformidade dos objetos educacionais gerados ao padrão SCORM 2004 3rd *Edition*. Outra motivação fundamental para usar o software validador se deve à sua funcionalidade de simular o comportamento de um LMS, sendo de grande valia no sentido de avaliar o funcionamento correto dos pacotes produzidos, bem como sua interação com os ambientes de aprendizado.

O SCORM® 2004 3rd *Edition Conformance Test Suite* define 3 modalidades de conformidade a serem avaliadas: *Learning Management System (LMS)*, *Sharable Content Object (SCO)* e *Content Package (CP)*. Os objetos educacionais produzidos pela ferramenta proposta neste trabalho serão submetidos para avaliação quanto às modalidades SCO e CP. Cada modalidade possui um rótulo de conformidade, aos quais são associadas categorias de conformidade [1]. Para ser conformante na respectiva categoria, o objeto educacional empacotado deve atender todos os requisitos especificados para a mesma, os quais podem ser encontrados para referência na íntegra em “*View the ADL SCORM 2004 3rd Edition Conformance Matrix*”, seção acessada pela tela principal do utilitário. Para a validação do IOPackager, serão visados 2 rótulos de conformidade, os quais, seguidos de suas respectivas categorias associadas, estão descritos abaixo.

Rótulo de conformidade 1:

SCO SCORM 2004 3rd Edition Conformant

(conformante com *Sharable Content Object* SCORM 2004 Terceira edição)

Categoria de conformidade associada:

- **SCO Run-Time Environment Version 1.0 (SCO RTE 1.0)**

(SCO Ambiente de Tempo de Execução versão 1.0)

Rótulo de conformidade 2:

CP SCORM 2004 3rd Edition Conformant

(conformante com *Content Package SCORM 2004* Terceira edição)

Categorias de conformidade associadas:

- **Content Package Content Aggregation Model Version 1.0 (CP CAM 1.0)**
(Modelo de Agregação de Conteúdo de Pacote de Conteúdo versão 1.0)
- **Content Package Run-Time Environment Version 1.0 (CP RTE 1.0)**
(Ambiente de Tempo de Execução de Pacote de Conteúdo versão 1.0)

5.6 Linguagem e ferramentas utilizadas no desenvolvimento do empacotador

A linguagem PHP 5.2 [28] será utilizada na implementação do projeto pelas seguintes características:

- orientada a objetos, permitindo melhor estruturação e organização modular do programa em classes;
- portátil, desenvolvida para ser executada em qualquer plataforma;
- flexível e dinâmica para desenvolvimento de aplicações *Web*;
- inclui amplo suporte a DOM (*Document Object Model*), manipulação de documentos XML e expressões regulares (PCRE – *Pearl Compatible Regular Expressions*) através de bibliotecas.

6 IMPLEMENTAÇÃO DO IOPACKAGER

O desenvolvimento da ferramenta IOPackager visa prover três capacidades fundamentais a um objeto educacional convencional adequado ao modelo SCORM através do processo de empacotamento: a capacidade de ser armazenado, pesquisado através de metadados descritivos associados ao objeto, e visualizado em ambientes de aprendizado e LMSs conformantes com o modelo SCORM. Para alcançar isso, o ponto-chave do projeto é resolver o problema focando primariamente o aspecto da conformidade com o padrão no desenvolvimento dos componentes que implementam as funcionalidades principais da ferramenta. Essas funcionalidades equivalem a prover uma interface de inserção de metadados do objeto a ser empacotado e posteriormente gerar sua representação adequada às normas conformantes, prover um mecanismo de lançamento do objeto a partir do ambiente SCORM que o gerencia, e descrever uma estrutura interna de recursos conformante para que os arquivos do objeto empacotado estejam acessíveis e corretamente referenciados pelo pacote.

6.1 Modelagem da solução

A ferramenta IOPackager consiste de um programa orientado a objetos escrito em linguagem PHP [28]. Ela pode ser entendida como a reunião de tarefas distintas desempenhadas por seus componentes, ou classes, cujos respectivos funcionamentos são brevemente descritos abaixo. A figura 10 (página 45) apresenta as classes do IOPackager e suas relações de dependência.

- A classe **Manifesto** coordena todo o processo de empacotamento, incluindo a utilização dos serviços das demais classes na geração das seções de metadados, organizações e recursos do arquivo *imsmanifest.xml*;
- A classe **AnaliseArquivos** faz o trabalho de varredura do diretório de empacotamento, inspeciona seus arquivos, aplica as regras de classificação e definição de recursos e produz todas as estruturas de dados necessárias para que a seção de recursos do manifesto seja gerada;
- A classe **DocumentoXML** utiliza a biblioteca de manipulação de documentos

DOM (*Document Object Model*) derivada do PHP, atuando como uma camada de abstração e oferecendo um meio conveniente de carregar, criar, manipular e salvar documentos XML através de objetos;

- A classe **InterfaceUsuario** é responsável pela geração das páginas de interface com o usuário do IOPackager, recolhe suas informações recebidas do navegador e as repassa à classe Manifesto para que sejam processadas no empacotamento;
- A classe **CreateZip**, empregada porém não desenvolvida pelo autor deste trabalho, abriga os métodos necessários à criação de arquivos de extensão zip, sendo este o procedimento final da geração do pacote SCORM;

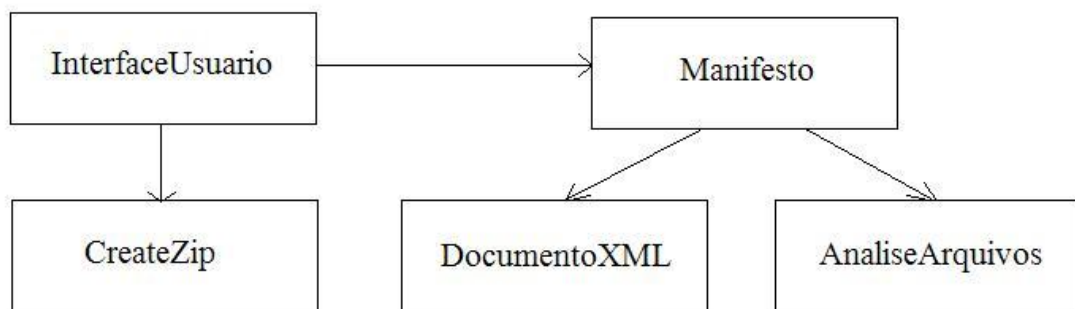


Figura 10: Diagrama ilustrando as relações de dependência entre as classes do IOPackager

6.2 Geração da seção de metadados do arquivo XML do manifesto

A implementação da interface de obtenção dos metadados do usuário e sua posterior representação em formato conformante através de XML foi possível com o emprego de um modelo (*template*) que descreve a seção `<metadata>` do arquivo de manifesto. Esse modelo corresponde à estrutura XML da seção de metadados (definida no Modelo de Agregação de Conteúdo [2]), na qual todos os atributos estão presentes e cujos valores estão rotulados com um identificador único. Após receber o formulário preenchido, cujos campos são todos opcionais, a classe Manifesto identifica cada um dos atributos de metadados informados e analisa o modelo, buscando substituir cada rótulo ali presente pelo respectivo valor correspondente a um campo informado no formulário. Posteriormente, os elementos não preenchidos são removidos, restando no modelo apenas o que foi fornecido pelo usuário.

Para que as informações de metadados sejam corretamente validadas, alguns cuidados devem ser tomados no momento do preenchimento com relação a restrições de tipo de informação associadas a alguns campos. Situados ao lado de cada um desses campos, quando se faz necessário, a interface fornece exemplos de dados corretos e instruções quanto ao formato esperado de preenchimento.

6.3 Implementação da classificação de arquivos em recursos de pacote

O objetivo do componente de análise e classificação dos arquivos de entrada do IOPackager é decidir quais arquivos são considerados recursos e quais não o são a fim de produzir a seção *<resources>* do arquivo de manifesto do pacote SCORM.

Do ponto de vista da conformidade com o padrão, o Modelo de Agregação de Conteúdo SCORM estabelece que a seção de recursos contenha, sob a forma de recursos ou referências de um recurso, todos os arquivos que foram incluídos no pacote. Ou seja, um arquivo pode ser definido como um recurso, caso em que pode possuir dependências de outros arquivos, ou ser apenas referenciado por um outro arquivo, caso em que ele é considerado parte de um recurso.

Tendo em vista o exposto, a proposta implementada para a ferramenta IOPackager com relação à geração da seção de recursos do arquivo de manifesto é a que segue:

- um arquivo qualquer deve ser descrito como recurso ou ser apenas referenciado por outro arquivo;
- um arquivo definido como recurso (descrito pela *tag <resource>*) pode listar outros arquivos dos quais depende (*<file>* e *<dependency>*), como mostra a figura 11 (página 47);
 - é possível somente a recursos listar arquivos referenciados ou dependentes;
 - todo arquivo definido como recurso pode possuir dependências (referenciar outros arquivos) e se o fizer, estes devem constar como elementos referenciados (figura 11);
 - se um arquivo A definido como recurso referenciar outro arquivo B que também é recurso, A contém B como sendo sua dependência por meio da *tag XML <dependency>*;
 - se um arquivo A definido como recurso referenciar outro arquivo B que não é

recurso, A contém B como sendo seu componente por meio da *tag* XML `<file>`;

```
<resource identifier="R_ID1" adlcp:scormType="sco" href="index_1.htm">
  <file href="index_1.htm"/>
  <dependency identifierref="DEP_R_ID1"/>
</resource>
```

Figura 11: Elemento *resource* na seção de recursos do manifesto XML

O IOPackager define internamente algumas informações para finalidade de consulta sobre extensões de arquivo conhecidas, que podem ser baseadas em texto ou não. O objetivo é predeterminar que os arquivos cujas extensões são conhecidas serão definidos como recursos. A separação é feita tendo em vista que somente os arquivos baseados em texto são possíveis de serem examinados quanto às suas dependências (arquivos referenciados), enquanto que os não baseados em texto mais comuns, como imagens e animações, por exemplo, geralmente não fazem referência a arquivos externos. Dessa forma, pode-se estipular, de forma simples diretamente no código-fonte da ferramenta mas não por meio de interface de usuário, quais extensões de arquivo são conhecidas do IOPackager, subdivididas entre a lista das extensões baseadas em texto, doravante designada Lista EBT, e a lista das extensões não baseadas em texto, doravante chamada de Lista ENBT. Cada uma das extensões da lista EBT conterá um grupo de padrões de busca por expressões regulares que será usado na identificação das dependências de cada arquivo de entrada cuja extensão seja coincidente à armazenada na lista.

O algoritmo que implementa as regras de empacotamento, realizado pela classe *AnaliseArquivos*, é descrito a seguir. Inicialmente, os arquivos integrantes do objeto educacional a ser empacotado são varridos e uma representação da árvore de diretórios é armazenada. Através do percurso dessa árvore, cada arquivo é classificado como sendo conhecido - cuja extensão está em alguma das listas EBT ou ENBT - ou não. Se o arquivo for conhecido e baseado em texto, a entrada na lista EBT correspondente à sua extensão será lida e as suas dependências serão identificadas com base no grupo de expressões regulares definido ali. Se a extensão do arquivo for conhecida e não baseada em texto, nenhuma dependência é localizada.

Terminada a localização das dependências de cada arquivo e a definição de recursos, o próximo passo é determinar, para cada arquivo que possui referências (que é um recurso), se estas são também recursos ou não. Se forem, a referência será um elemento `<dependency>`.

Caso contrário, será um elemento `<file>`.

6.4 Geração do arquivo `imsmanifest.xml` e a finalização do pacote SCORM

Após analisados os arquivos de entrada e definidos os recursos, a seção `<resources>` já está disponível para o manifesto. A seção `<organizations>`, por sua vez, consta de um modelo fixo (*template*), onde está referenciado o único SCO presente no pacote, o arquivo `sco.html`, cuja função é redirecionar a visualização do objeto educacional, após este ter sido lançado pelo LMS, para o arquivo principal indicado pelo usuário do IOPackager através da interface. Por meio da classe Manifesto, cada uma das 3 seções que compõem o manifesto, as quais são 3 documentos separados gerados pela classe DocumentoXML, são então adicionadas a um único documento XML chamado `imsmanifest.xml`, que deverá ser salvo na raiz do pacote SCORM, juntamente com todos os arquivos do objeto educacional. Por fim, todos os arquivos são reunidos e devidamente compactados com o auxílio da classe CreateZip em um pacote Zip.

7 RESULTADOS E VALIDAÇÃO DA FERRAMENTA IOPACKAGER

De acordo com o descrito na seção 5.5 (página 42), a validação da ferramenta foi voltada à conformidade dos objetos empacotados com as normas da ADL, segundo os rótulos de conformidade definidos no utilitário validador [1]. A importância da validação segundo este utilitário é fundamental, pois dessa forma é assegurado que o pacote gerado executará em ambientes de aprendizado e LMSs conformantes da forma correta e esperada, garantindo inclusive que a própria ferramenta IOPackager pode ser empregada na geração de pacotes confiáveis quanto à conformidade e, conseqüentemente, quanto à funcionalidade.

7.1 Finalidade das categorias de conformidade

A meta buscada com a conformidade em relação às categorias definidas no utilitário validador da ADL é prover que a funcionalidade do pacote gerado seja garantida. Dessa forma, uma breve explanação é feita sobre os rótulos de conformidade e suas respectivas categorias visadas para a validação do IOPackager.

- **SCO SCORM 2004 3rd Edition Conformant - SCO Run-Time Environment Version 1.0 (SCO RTE 1.0)** : esta categoria diz respeito à validação da interação entre um LMS e um ou mais SCOs testados isoladamente, avaliando a capacidade destes de estabelecer comunicação e ser lançado pelo LMS.
- **CP SCORM 2004 3rd Edition Conformant - Content Package Content Aggregation Model Version 1.0 (CP CAM 1.0)** : esta categoria se refere à validação do arquivo de manifesto de um pacote quanto às suas características de ser bem-formado (*well-formed*) e válido em relação aos documentos de controle IMS (*IMS Controlling Documents*) e de funcionalidades específicas a elementos SCORM agregados à especificação IMS (*SCORM Application Profile*).
- **CP SCORM 2004 3rd Edition Conformant - Content Package Run-Time Environment Version 1.0 (CP RTE 1.0)**: esta categoria é idêntica à primeira descrita (SCO RTE 1.0), com a diferença de que é aplicada a SCOs pertencentes a um pacote, e não tomados isoladamente como no outro caso.

7.2 Exemplo de empacotamento de objeto educacional e validação do pacote correspondente

O objetivo desta seção é demonstrar a aplicação da ferramenta IOPackager no empacotamento de um objeto educacional tomado livremente como exemplo. Logo após, a validação do pacote correspondente será realizada por meio do utilitário validador de pacotes SCORM da ADL [1] e os respectivos resultados serão mostrados.

O objeto educacional MR. MURPHY'S GARDEN OF MATHEMATICAL DELIGHTS consiste de uma coleção de exercícios que envolvem conceitos matemáticos na resolução de problemas, podendo ser encontrado para download em [26].

As informações necessárias submetidas ao IOPackager para a geração deste pacote foram o caminho onde está localizado o diretório do objeto, seu arquivo principal a ser lançado, que no caso deste exemplo é **index.html**, o nome desejado para o pacote a ser gerado e alguns metadados associados para ilustrar a criação da seção `<metadata>` conformante com o padrão SCORM, que pode ser vista na figura 12 (página 51).

Após o pacote de conteúdo SCORM ter sido gerado, ele é submetido a validação com relação às 3 categorias visadas: **SCO RTE 1.0**, **CP CAM 1.0** e **CP RTE 1.0**.

Para a categoria SCO RTE 1.0, os resultados mostram que o objeto é capaz de estabelecer comunicação com um LMS e ser lançado por este, o que o permite ser visualizado normalmente. A figura 13 (página 52) demonstra o lançamento do objeto educacional pelo ambiente de tempo de execução do validador, consistindo simplesmente da sua página inicial sendo exibida pelo navegador após o SCO **sco.html** ter sido testado e invocado, ação esta feita a partir da parte esquerda da interface do validador visto na figura 14 (página 52). Ainda nesta imagem, à direita, aparecem resumidamente os resultados indicando a realização com êxito do teste do SCO **sco.html**, os quais revelam que as chamadas dos métodos de comunicação *Initialize()* e *Terminate()* da API (em cor verde) foram bem-sucedidas. A figura 15 (página 53) mostra os resultados detalhados e, destacado na parte inferior, o parecer final de conformidade atribuído ao SCO **sco.html**, indicando a conformidade com a categoria SCO RTE 1.0.

Para as categorias CP CAM 1.0 e CP RTE 1.0, as propriedades conformantes verificadas, além da validação do mecanismo de comunicação do SCO já descrita acima, são a validade do arquivo de manifesto em relação aos documentos de controle de conformidade e a validade dos metadados associados ao objeto, como mostra a figura 16 (página 54). Nessa

imagem, aparecem, à direita da interface, assinalados em cor verde (pois passaram no teste) os itens verificados referentes ao manifesto na parte central e referentes aos metadados na parte inferior. Lembrando que nestas categorias, a análise é realizada sobre todo o pacote, avaliado como o conjunto de todos os componentes. Por fim, a figura 17 (página 54) mostra o rótulo de conformidade, ou seja, conformante com as categorias CP CAM 1.0 e CP RTE 1.0, para o objeto educacional empacotado, indicando que o teste teve êxito e que o pacote de conteúdo SCORM 2004 3rd Edition é conformante.

```

<metadata>
  <schema>ADL SCORM</schema>
  <schemaversion>2004 3rd Edition</schemaversion>
  <imsmd:lom xmlns="http://ltsc.ieee.org/xsd/LOM">
    <imsmd:general>
      <imsmd:title>
        <imsmd:string>MR. MURPHY'S GARDEN OF MATHEMATICAL DELIGHTS</imsmd:string>
      </imsmd:title>
      <imsmd:language>en</imsmd:language>
      <imsmd:description>
        <imsmd:string>Solução de problemas matemáticos para ensino básico</imsmd:string>
      </imsmd:description>
      <imsmd:keyword>
        <imsmd:string>Matemática</imsmd:string>
      </imsmd:keyword>
    </imsmd:general>
    <imsmd:educational>
      <imsmd:interactivityType>
        <imsmd:source>LOMv1.0</imsmd:source>
        <imsmd:value>mixed</imsmd:value>
      </imsmd:interactivityType>
      <imsmd:learningResourceType>
        <imsmd:source>LOMv1.0</imsmd:source>
        <imsmd:value>exercise</imsmd:value>
      </imsmd:learningResourceType>
      <imsmd:interactivityLevel>
        <imsmd:source>LOMv1.0</imsmd:source>
        <imsmd:value>medium</imsmd:value>
      </imsmd:interactivityLevel>
      <imsmd:intendedEndUserRole>
        <imsmd:source>LOMv1.0</imsmd:source>
        <imsmd:value>learner</imsmd:value>
      </imsmd:intendedEndUserRole>
      <imsmd:context>
        <imsmd:source>LOMv1.0</imsmd:source>
        <imsmd:value>school</imsmd:value>
      </imsmd:context>
      <imsmd:typicalAgeRange>
        <imsmd:string>até 15 anos</imsmd:string>
      </imsmd:typicalAgeRange>
      <imsmd:difficulty>
        <imsmd:source>LOMv1.0</imsmd:source>
        <imsmd:value>easy</imsmd:value>
      </imsmd:difficulty>
      <imsmd:language>en</imsmd:language>
    </imsmd:educational>
  </imsmd:lom>
</metadata>

```

Figura 12: Metadados gerados pelo IOPackager para o objeto educacional empacotado do exemplo

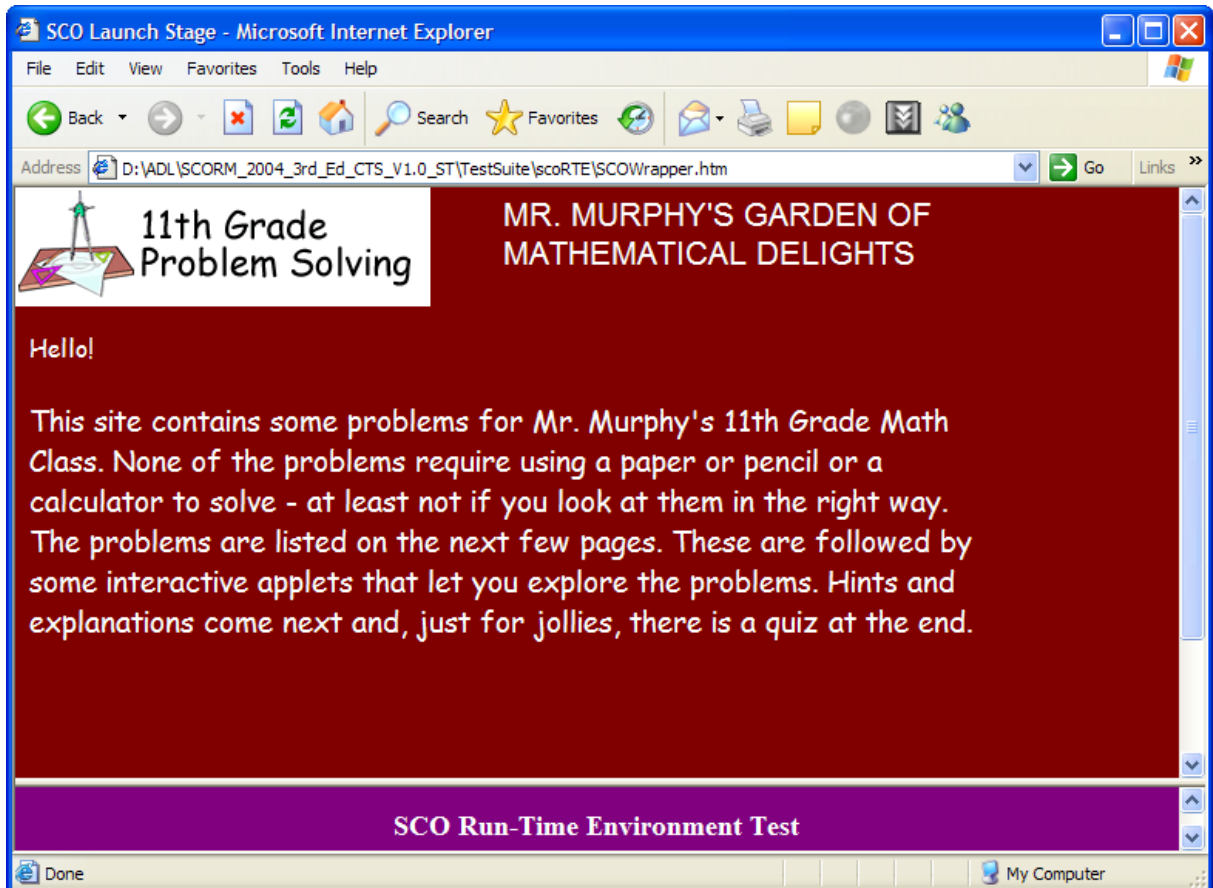


Figura 13: Página inicial do objeto educacional exemplo lançada pela simulação do ambiente de tempo de execução do utilitário validador

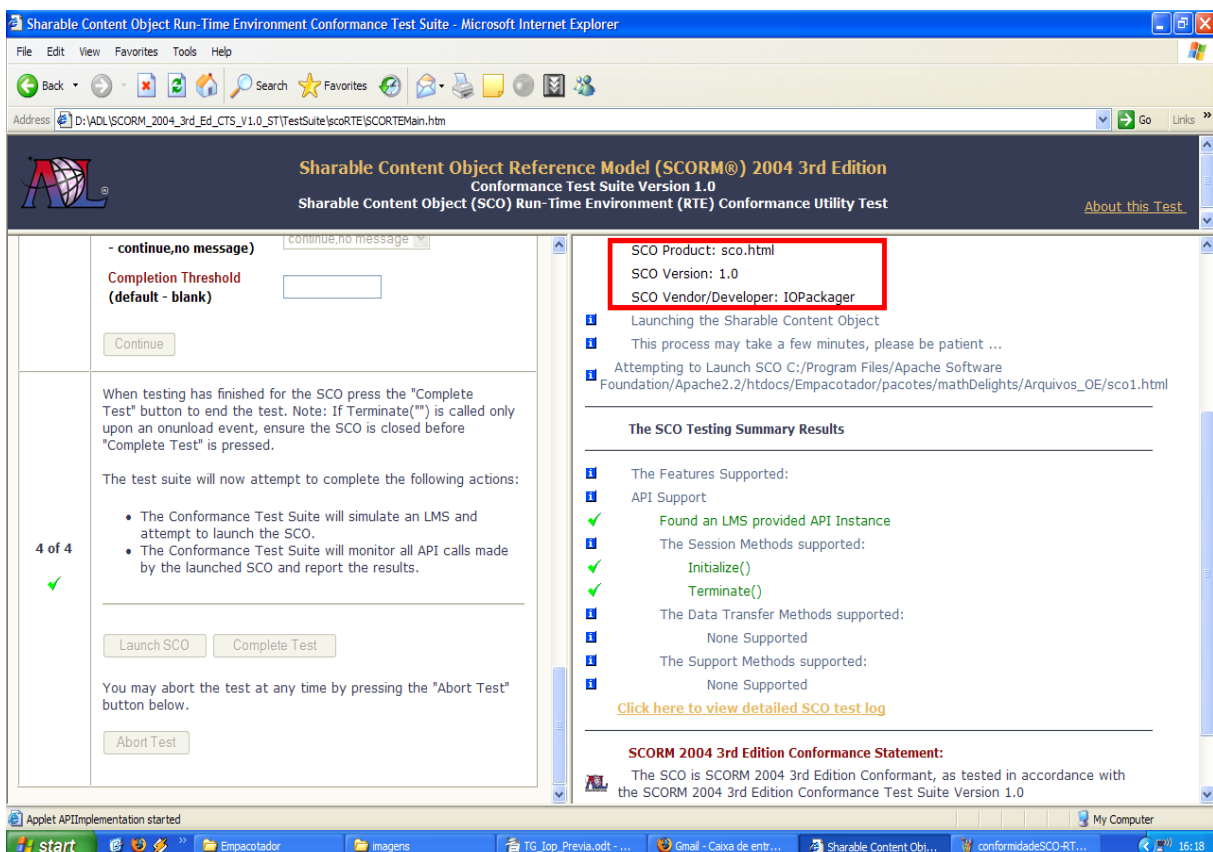


Figura 14: Interface do validador de conformidade da ADL após o teste do SCO do objeto educacional exemplo

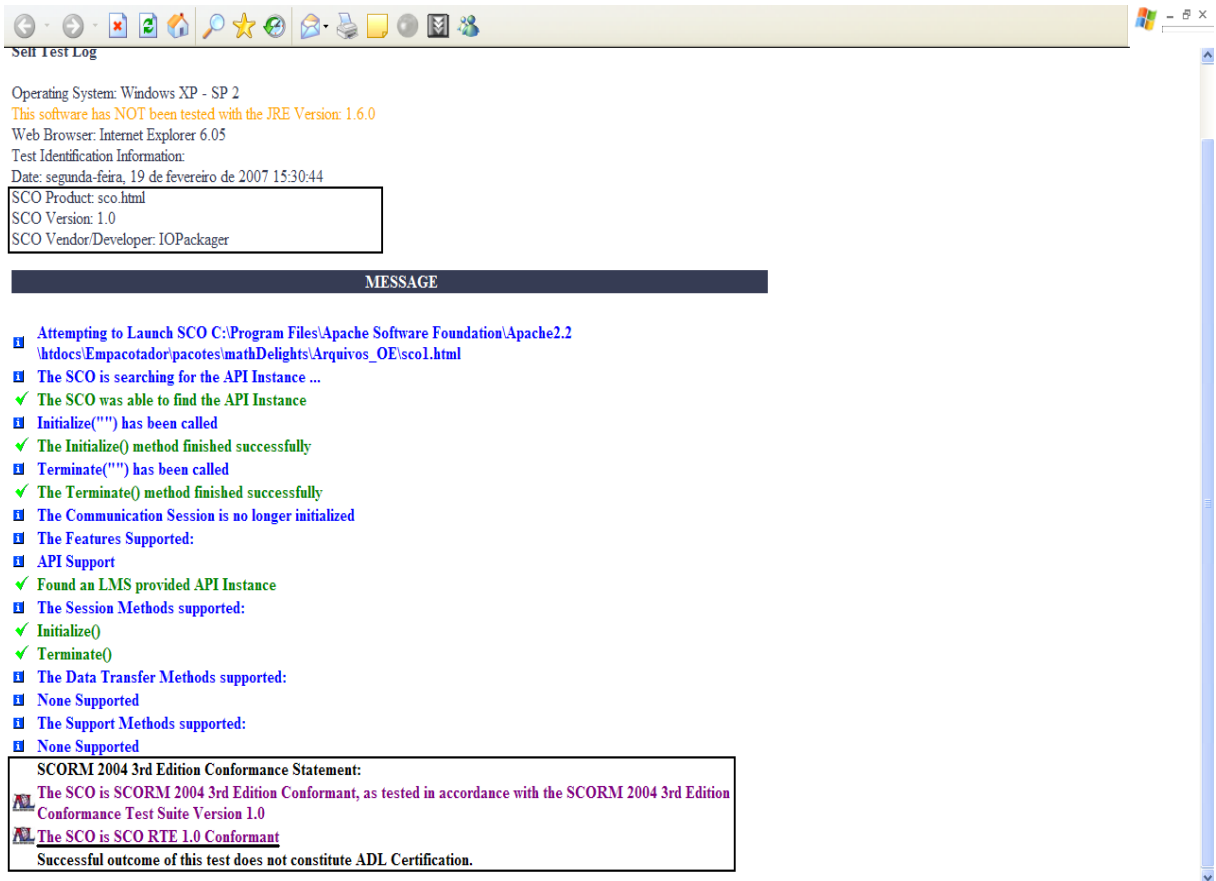


Figura 15: Rótulo de conformidade emitido pelo validador da ADL para o SCO sco.html

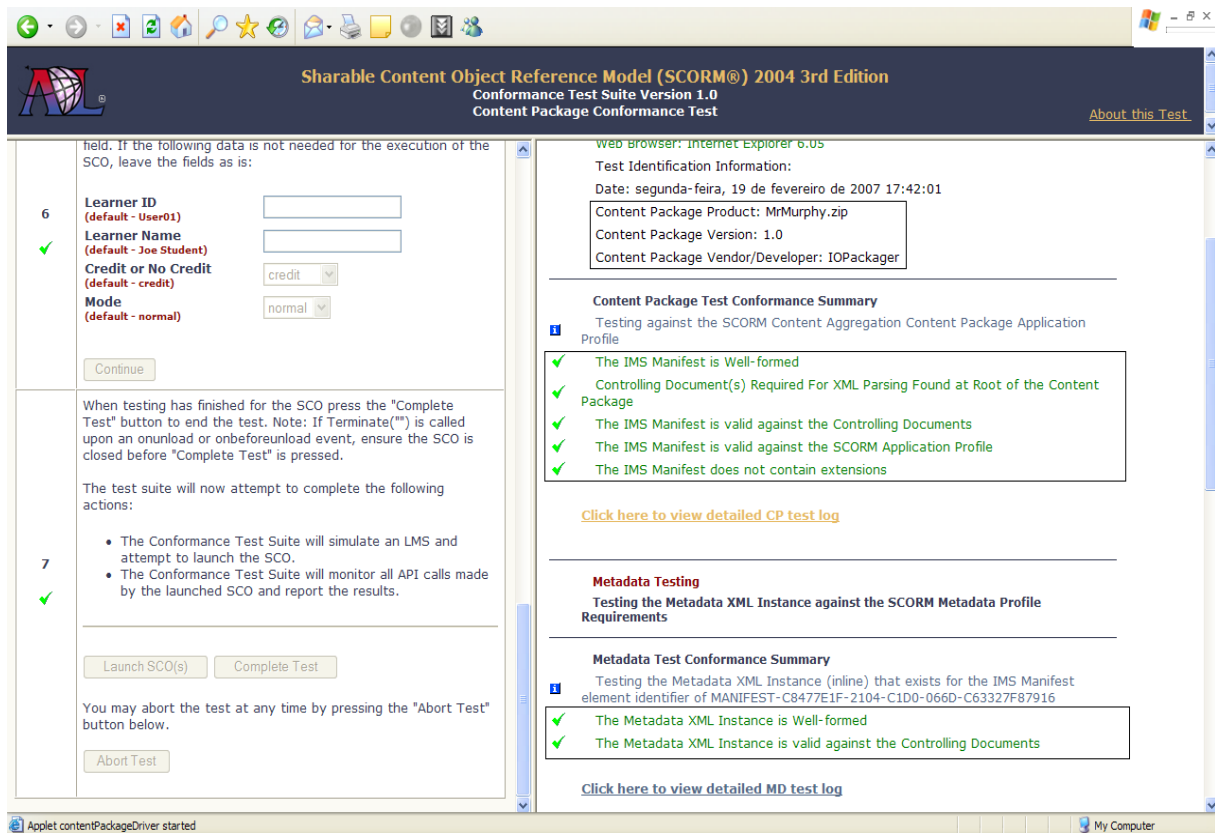


Figura 16: Interface do validador mostrando a avaliação bem-sucedida da conformidade do manifesto e dos metadados do objeto educacional exemplo

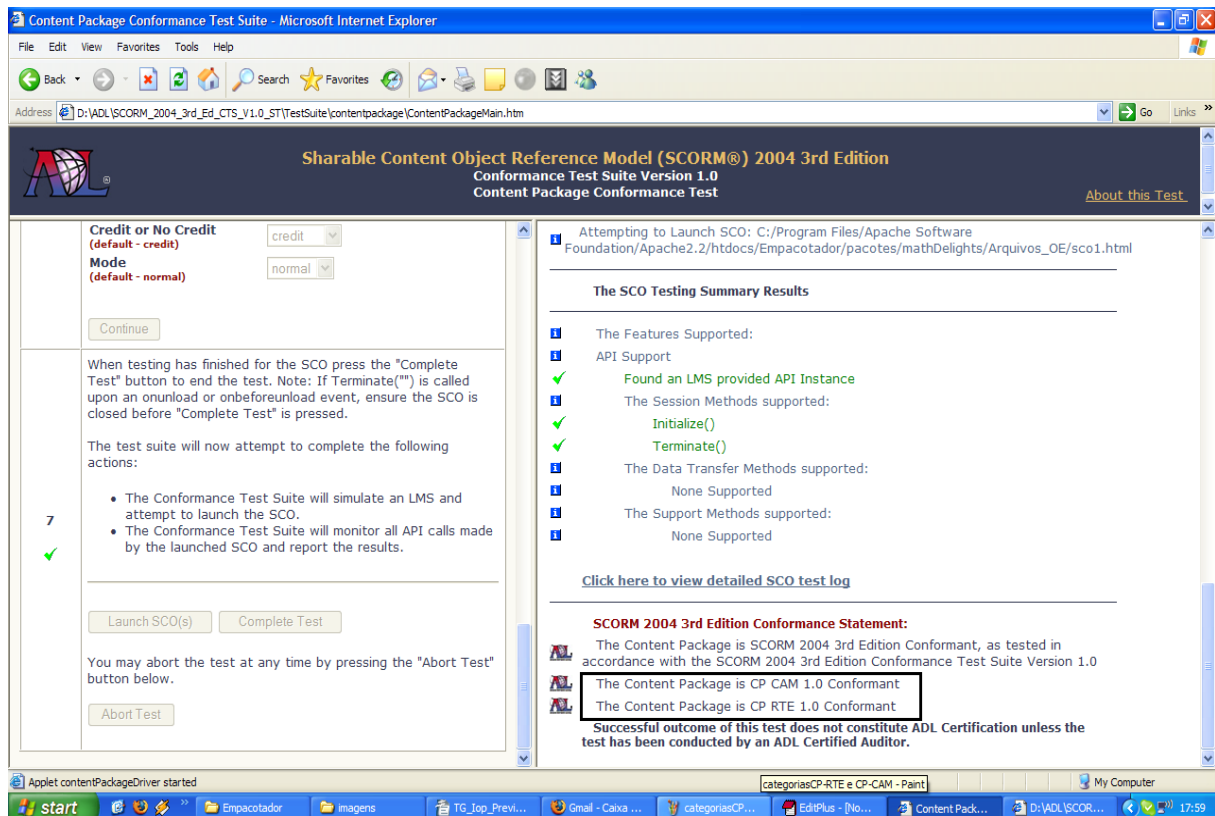


Figura 17: Interface do validador mostrando o parecer final emitido ao pacote do objeto educacional exemplo gerado pelo IOPackager com relação às categorias CP CAM 1.0 e CP RTE 1.0

8 CONCLUSÃO

Este trabalho objetivou a criação do IOPackager, que consiste de uma ferramenta genérica de empacotamento de conteúdo instrucional no formato SCORM 2004 Terceira Edição. Como parte integrante do projeto, foi apresentada a fundamentação teórica abrangendo áreas pertinentes ao problema, que inclui a educação a distância para a *Web* – também referida como *E-Learning* – com enfoque em sistemas de gerência de aprendizado (*Learning Management Systems*), objetos educacionais digitais e seu papel na autoria de material instrucional reusável, a relevância e os benefícios da padronização de conteúdo e de seu gerenciamento, e o padrão SCORM como modelo de referência estudado e empregado no processo de empacotamento proposto. Com relação aos resultados obtidos, e partindo do que foi exposto e demonstrado, os objetivos da ferramenta IOPackager foram plenamente alcançados, como mostram as considerações a seguir.

Os esforços principais concentraram-se em desvincular o usuário do IOPackager de quaisquer conhecimentos técnicos necessários ao procedimento de criação de pacotes de conteúdo SCORM funcionais e conformantes, o que foi obtido de forma simples e bem-sucedida. Outro aspecto de relevância indiscutível para o funcionamento correto do IOPackager foi o da conformidade com as normas estabelecidas pelo modelo de referência SCORM na estrutura do pacote e na geração do documento manifesto **imsmanifest.xml**. Através dos testes de validação dos pacotes produzidos pelo IOPackager, tornou-se claro que os objetivos quanto à conformidade foram satisfatoriamente alcançados de forma direta e objetiva. Dessa forma, com o auxílio do utilitário validador da ADL, foi possível obter resultados que certificaram a validade dos pacotes testados e asseguraram seu funcionamento em ambientes de aprendizado e sistemas de gerência conformantes com o padrão SCORM nos quais possam vir a ser armazenados e executados.

Portanto, o IOPackager pode ser considerado de valia para o processo de empacotamento de conteúdo visando conformidade com o modelo de referência SCORM 2004 Terceira Edição, tendo em vista a ênfase e o crescente interesse nessa área por parte de profissionais tanto da área tecnológica, no que diz respeito a padronização e compatibilidade de conteúdo, quanto da área pedagógica, com relação a aumento de qualidade, diminuição de custos e mais facilidade na autoria de materiais educacionais digitais.

9 REFERÊNCIAS BIBLIOGRÁFICAS

[1] ADL - Advanced Distributed Learning. SCORM® 2004 3rd Edition Conformance Test Suite. Disponível em: <<http://www.adlnet.gov/scorm/20043ED/cts.cfm>>. Acesso em: 12 de janeiro de 2007.

[2] ADL - Advanced Distributed Learning. Sharable Content Object Reference Model (SCORM) 2004 3rd Edition Documentation Suite. Disponível em: <<http://www.adlnet.gov/downloads/311.cfm>>. Acesso em: 11 de janeiro de 2007.

[3] ADL - Advanced Distributed Learning. Disponível em: <<http://www.adlnet.org>>. Acesso em: 26 de dezembro de 2006.

[4] AICC - Aviation Industry CBT Committees. Disponível em: <<http://www.aicc.org>>. Acesso em: 26 de dezembro de 2006.

[5] ANDRADE, M.V., BRASILEIRO, F.V. Sistemas de gerenciamento de aprendizagem: uma metodologia de avaliação. Disponível em: <<http://fgsnet.nova.edu/cread2/pdf/Andrade.pdf>>. Acesso em: 16 de dezembro de 2006.

[6] ARIADNE - Alliance of remote instructional authoring and distribution networks for Europe website. Disponível em: <<http://ariadne.unil.ch>>. Acesso em: 15 de outubro de 2006.

[7] BECTA - British Educational Communications and Technology Agency. Packaging and publishing learning objects: Best practice guidelines. Disponível em: <http://www.becta.org.uk/page_documents/industry/content_packaging.pdf>. Acesso em: 29 de dezembro de 2006.

[8] BOYLE, T.. Design principles for authoring dynamic, reusable learning objects. Disponível em: <<http://www.ascilite.org.au/conferences/auckland02/proceedings/papers/028.pdf>>. Acesso em: 18 de dezembro de 2006.

[9] CAETANO, L.. Uma pincelada sobre SCORM. Disponível em: <http://universus.datasus.gov.br/file.php?file=/1/docs_comuns/ApresentSCORM.pdf>. Acesso em: 28 de dezembro de 2006.

[10] Draft Standard for Learning Object Metadata – IEEE. Disponível em: <http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf>. Acesso em: 15 de outubro de 2006.

[11] Dublin Core 2001. International Conference on Dublin Core and Metadata Applications 2001 October 22-26, 2001 National Institute of Informatics Tokyo, Japan. Disponível em:

<<http://www.nii.ac.jp/dc2001>>. Acesso em: 15 de outubro de 2006.

[12] Dublin Core Metadata Initiative. Disponível em:
<<http://dublincore.org/documents/dces>>. Acesso em: 15 de outubro de 2006.

[13] DUTRA, R. L. S., TAROUCO, L. M. R.. Objetos de aprendizagem: uma comparação entre SCORM e IMS Learning Design. Disponível em:
<http://www.cinted.ufrgs.br/renote/jul2006/artigosrenote/a1_20138.pdf>. Acesso em: 28 de dezembro de 2006.

[14] FABRE, M. J. M., TAROUCO, L. M. R., TAMUSIUNAS, F. R.. SCORM (Sharable Content Object Reference Model). Disponível em:
<<http://www.cinted.ufrgs.br/files/tutoriais/scorm/scorm.htm>>. Acesso em: 26 de dezembro de 2006.

[15] FALLON, C.; BROWN, C.. E-learning Standards: A Guide to Purchasing, Developing, and Deploying Standards-Conformant e-Learning. St. Lucie Press. 2002.

[16] GIRARDI, R. A. A.. Framework para coordenação e mediação de Web Services modelados como Learning Objects para ambientes de aprendizado na Web. Disponível em:
<www2.dbd.puc-rio.br/pergamum/tesesabertas/0220942_04_cap_02.pdf>. Acesso em: 26 de dezembro de 2006.

[17] Granularity. Reusable Learning. Disponível em:
<<http://www.reusablelearning.org/index.asp?id=28>>. Acesso em: 21 de dezembro de 2006.

[18] IEEE Learning Technology Standards Committee – LTSC. Disponível em:
<<http://ltsc.ieee.org/wg12>>. Acesso em: 15 de outubro de 2006.

[19] IMS Global Learning Consortium, Inc. Disponível em: <<http://www.imsproject.org>>. Acesso em: 26 de dezembro de 2006.

[20] KRATZ, R.A. Fábrica de adequação de conteúdo de ensino para Objetos de Aprendizagem Reutilizáveis (RLOs) respeitando a norma SCORM. Disponível em:
<<http://www.inf.unisinos.br/~crespo/arquivos/dissertacoes/dissertacaoKratz.pdf>>. Acesso em: 01 de dezembro de 2006.

[21] Learnativity Foundation. Disponível em: <<http://www.learnativity.org/>>. Acesso em: 20 de dezembro de 2006.

[22] LONGMIRE, W.. A Primer On Learning Objects. Disponível em:
<<http://www.learningcircuits.org/mar2000/primer.html>>. Acesso em: 15 de outubro de 2006.

[23] LUCENA, B.. Novas Tecnologias no E-learning: Desafios e Oportunidades para o Design. Disponível em:
<http://www.abed.org.br/publicue/cgi/cgilua.exe/sys/start.htm?UserActiveTemplate=1por&in_foid=883&sid=135>. Acesso em: 16 de dezembro de 2006.

- [24] MCGREAL, R.. Learning Objects: A Practical Definition. Disponível em: <http://itdl.org/Journal/Sep_04/article02.htm>. Acesso em: 20 de dezembro de 2006.
- [25] MOURA, S. L.. Uma arquitetura para Integração de Repositórios de Objetos de Aprendizagem baseada em Mediadores e Serviços Web. Disponível em: <http://www.maxwell.lambda.ele.puc-rio.br/cgi-9bin/PRG_0599.EXE/7891_1.PDF?NrOcoSis=23240&CdLinPrg=pt>. Acesso em: 14 de outubro de 2006.
- [26] Mr. Murphy's Garden of Mathematical Delights. Disponível em: <http://www.reusablelearning.com/examples/mathdelights_sources/mathdelights_original.zip>. Acesso em: 19 de fevereiro de 2007.
- [27] PEREIRA, L. A. M., PORTO, F.A.M., MELO, R.N.. Objetos de Aprendizado Reutilizáveis (RLOs): Conceitos, Padronização, Uso e Armazenamento. Disponível em: <ftp://ftp.inf.puc-rio.br/pub/docs/techreports/03_10_pereira.pdf>. Acesso em: 18 de dezembro de 2006.
- [28] PHP: Hypertext Preprocessor. Disponível em: <www.php.net>. Acesso em: 12 de janeiro de 2007.
- [29] RELOAD Project - Reusable eLearning Object Authoring & Delivery. Disponível em: <<http://www.reload.ac.uk/>>. Acesso em: 18 de janeiro de 2007.
- [30] Reusable Information Object (RIO) Strategy of CISCO. Disponível em: <http://www.cisco.com/warp/public/779/ibs/solutions/learning/whitepapers/el_cisco_rio.pdf>. Acesso em: 15 de outubro de 2006.
- [31] Reusable Learning. Disponível em: <<http://www.reusablelearning.org/>>. Acesso em: 20 de dezembro de 2006.
- [32] SALVADOR, V.. Hiperídia interativa - a educação do futuro, no presente. Tecnologia Educacional. v. 22. n. 123/124. p. 22-23.
- [33] SCHAEFER, M. T.. Demystifying metadata: initiatives for web document description. Information Retrieval & Library Automation. v. 33. n. 11. p. 52.
- [34] TAROUCO, L.. Suporte de Redes e Computadores para Educação à Distância. Disponível em: <<http://www.pgie.ufrgs.br/webfolioead/biblioteca/artigo9/artigo9.html>>. Acesso em: 15 de dezembro de 2006.
- [35] TAROUCO, L. M. R., FABRE, M. J. M., TAMUSIUNAS, F. R.. Reusabilidade de objetos educacionais. Disponível em: <http://www.cinted.ufrgs.br/renote/fev2003/artigos/marie_reusabilidade.pdf>. Acesso em: 17 de janeiro de 2007.
- [36] TAROUCO, L. M. R.; FABRE, M. J. M.; DUTRA, R. L. S.. Interoperabilidade entre

objetos educacionais e sistemas de gerenciamento de aprendizado. Disponível em: <<http://www.cinted.ufrgs.br/ppt/interopObjEduc/sld001.htm>>. Acesso em: 14 de outubro de 2006.

[37] TAROUCO, L.M.R., MEHLECKE, Q.T.C.. Ambientes de Suporte para Educação a Distância: A mediação para aprendizagem cooperativa. Disponível em: <http://www.cinted.ufrgs.br/renote/fev2003/artigos/querte_ambientes.pdf>. Acesso em: 16 de dezembro de 2006.

[38] WILEY, D. A.. Learning object design and sequencing theory. Disponível em: <<http://davidwiley.com/papers/dissertation/dissertation.pdf>>. Acesso em: 14 de outubro de 2006.

[39] RIVED - Rede Internacional Virtual de Educação. O que é Objeto de Aprendizagem. Disponível em: <<http://rived.proinfo.mec.gov.br/curso/cofre/atividades.htm>>. Acesso em: 14 de janeiro de 2007.