QIOPackager – Desenvolvimento de uma ferramenta automática conversora de objetos educacionais em pacotes de conteúdo SCORM® 2004 3rd edition

Ricardo Donato Iop

Universidade Federal de Santa Maria
rdiop@inf.ufsm.br

Roseclea Duarte Medina

Universidade Federal de Santa Maria
rose@inf.ufsm.br

Resumo: 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 gestão de aprendizagem (LMSs) adequados ao modelo de Referência SCORM 2004 3rd Edition.

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

IOPackager – Development of an automatic tool for conversion of learning objects into SCORM® 2004 3rd edition content packages

Abstract: 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;

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 (Moura, 2006). 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 (Fallon, 2002). 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 aprendizado *online* (Andrade, 2006), englobando funcionalidades dos sistemas tradicionais de *E-Learning*, porém ampliando suas capacidades no sentido da automatização completa dos eventos de treinamento via *Web*. Assim, com a expansão desordenada de diversas dessas plataformas desenvolvidas com finalidades e ferramentas próprias (Fallon, 2002), 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 (Lucena, 2006), 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 (Moura, 2006) (Kratz, 2006). 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 gerencie.

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 (BECTA, 2006), 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.

2 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.

2.1 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*) (Kratz, 2006). 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*) (ADL, 2007), responsável pela apresentação do SCORM em 1999 como um padrão ideal, pois reuniu todos os outros padrões disponíveis no mercado. 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* (Caetano, 2006) que confere certas propriedades, ou atributos, essenciais ao material educacional digital para ensino a distância (Fabre et al., 2006) (ADL, 2007), 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 reprojeção, 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;

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 Navigation*)

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

Formado pelas especificações de descrição de metadados definidas em IMS (IMS GLC, 2006) e IEEE-LOM (IEEE DSLOM, 2006), 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*)

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

Baseado no padrão AICC (AICC, 2007), 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 (Kratz, 2006). 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 (ADL SCORM, 2007).

- **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 (ADL SCORM, 2007).

- **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 (ADL SCORM, 2007).

2.1.3 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 (Dutra, 2006).

2.2 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*. 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 (Kratz, 2006). 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 (Kratz, 2006). Adicionalmente, metadados podem ser associados a qualquer elemento dentro do pacote, podendo descrever recursos individuais (SCOs ou *assets*) ou conjuntos deles (agregações de conteúdo).

2.2.1 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 organização, ou diagrama de estrutura de conteúdo (BECTA, 2006), 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>. 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).

3 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.

3.1 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. O editor gráfico de pacotes de conteúdo e metadados RELOAD (RELOAD, 2007) 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 (ADL SCORM, 2007) e em sistemas de gerência de aprendizado (LMSs).

3.2 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 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.

3.3 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..

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.

3.4 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* por meio da implementação de dois métodos da API definida para o ambiente de tempo de execução do SCORM, 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, 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 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*. 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 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.

3.5 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 (ADL Conformance Test Suite, 2007), 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 (ADL Conformance Test Suite, 2007). 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)

3.6 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.

4 Resultados e Validação da ferramenta iopackager

De acordo com o descrito anteriormente, 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 (ADL Conformance Test Suite, 2007). 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.

4.1 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.

4.2 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 (ADL Conformance Test Suite, 2007) 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 (Murphy, 2007).

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>* conforme com o padrão SCORM.

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 (Figura 1).

Figura 1: Página inicial do objeto educacional exemplo lançada pela simulação do ambiente de tempo de execução do utilitário validador

A figura 2 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.

Figura 2:

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

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 3 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.

Figura

3: Rótulo de conformidade emitido pelo validador da ADL para o SCO **sco.html**

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 a seguir (Figura 4).

Figura 4: Interface do validador mostrando a avaliação bem-sucedida da conformidade do manifesto e dos metadados do objeto educacional exemplo

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 5 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.

Figura 5: 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

5 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.

6 Referências Bibliográficas

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.

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.

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

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

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.

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.

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.

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.

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.

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.

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

DUTRA, R. L. S., TAROUÇO, 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.

FABRE, M. J. M., TAROUÇO, 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.

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

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.

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

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

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.

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

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

MCGREAL, R.. Learning Objects: A Practical Definition. Disponível em:
<http://itdl.org/Journal/Sep_04/article02.htm>. Acesso em: 20 de dezembro de 2006.

MOURA, S. L.. Uma arquitetura para Integração de Repositórios de Objetos de 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.

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.

PHP: Hypertext Preprocessor. Disponível em: <www.php.net>. Acesso em: 12 de janeiro de 2007.

RELOAD Project - Reusable eLearning Object Authoring & Delivery. Disponível em:
<<http://www.reload.ac.uk/>>. Acesso em: 18 de janeiro de 2007.

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.

Reusable Learning. Disponível em: <<http://www.reusablelearning.org/>>. Acesso em: 20 de dezembro de 2006.

SCHAEFER, M. T.. Demystifying metadata: initiatives for web document description.

Information Retrieval & Library Automation. v. 33. n. 11. p. 52. 1998

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.

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.

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.