

Objetos de Aprendizado Reutilizáveis (RLOs): Conceitos, Padronização, Uso e Armazenamento

Luiz Antônio M. Pereira
lpereira@inf.puc-rio.br

Fábio A. Machado Porto
porto@inf.puc-rio.br

Rubens Nascimento Melo
rubens@inf.puc-rio.br

PUC-RioInf.MCC10/03 Março, 2003

Abstract

In this work we list and discuss the concepts and characteristics of Reusable Learning Objects (or simply Learning Objects – RLO or LO) that are being considered by the literature as an efficient solution to cost-effective instructional content development for e-learning. In order to establish a common structure to describe these learning objects, the IEEE Learning Technology Standards Committee (IEEE-LTSC) is developing the Learning Object Metadata standard (LOM) that is being referenced by other important standards such as IMS and SCORM. In this work we propose a LO data model and discuss aspects related to the storage, search and retrieval of LOs from databases. In order to put into perspective the use of LOs by applications, we also propose a four-level database architecture as a reference model. Finally, we discuss the automated and semi-automated construction of learning contents.

Keywords: Learning Objects, LO, RLO, IEEE-LOM, IMS, SCORM, LO management.

Resumo

No presente trabalho relacionamos e discutimos os conceitos e características dos Objetos de Aprendizado Reutilizáveis (*Reusable Learning Objects* ou, simplesmente, *Learning Objects* - RLOs ou LOs), que vêm sendo apontados pela literatura como uma solução eficiente para a diminuição do custo de desenvolvimento de cursos ministrados a distância. De forma a estabelecer uma estrutura comum de descrição desses objetos de aprendizado, o Comitê de Padrões de Tecnologia de Aprendizado (LTSC) do IEEE vem desenvolvendo o padrão de metadados de objetos de aprendizado (LOM), que é referenciado por outros importantes padrões como o IMS e o SCORM. Nesse trabalho propomos um modelo de dados de LOs e discutimos aspectos relacionados ao armazenamento, pesquisa e recuperação de LOs armazenados em bancos de dados. Com vistas a permitir o uso de LOs por aplicações, também propomos uma arquitetura de banco de dados em quatro níveis como um modelo de referência. Finalmente, discutimos a construção automática e semi-automática de conteúdos de aprendizado.

Palavras-chave: Objetos de aprendizado, *learning objects*, RLO, IEEE-LOM, IMS, SCORM, gerenciamento de LOs

1 - Introdução e Motivação

Muitos cursos oferecidos na forma tradicional, i.e., presenciais e liderados por instrutores, têm sido cogitados para serem convertidos para a forma eletrônica, com a finalidade de serem ministrados a distância [1], ou seja, com os alunos separados fisicamente entre si, separados dos professores e fora dos limites geográficos do estabelecimento de educação [2].

Embora as definições e características precisas de Ensino a Distância (EAD) não sejam um consenso entre pesquisadores [3], interessam-nos alguns aspectos comuns a todos os diversos entendimentos a respeito do assunto; EAD pode ser caracterizado modernamente, não só pela separação física e temporal entre professores e alunos, como também pela necessária interação entre eles mediada por alguma forma de tecnologia que permita a *colaboração* (entre alunos e entre professores e alunos). A infra-estrutura de comunicação, que é vista pelos pesquisadores como uma das mais efetivas, consiste das já bastante difundidas Internet e intranets.

A opção por se ministrar ou cursar um programa de ensino a distância considera inúmeros aspectos dos quais destacamos (1) o aspecto econômico, referente tanto à parte dos alunos quanto dos instrutores e instituições de ensino e que pode estar associado à abrangência geográfica e mercadológica, custos de transporte e estadia, etc., (2) o aspecto da padronização da forma e do conteúdo dos cursos e o não menos importante (3) aspecto social, expresso na preocupação dos governos em levarem o ensino aos mais recônditos lugares de um país.

Os dois primeiros aspectos assumem uma posição de destaque nesse trabalho, na medida em que estamos exatamente em busca de respostas para a seguinte pergunta: como poderemos desenvolver conteúdo instrucional rapidamente, com qualidade e a baixo custo? Uma possível resposta pode ser sintetizada em três expressões que são inter-relacionadas: divisão em módulos, padronização e reúso.

No presente trabalho trataremos de relacionar os conceitos e características dos Objetos de Aprendizado Reutilizáveis (*Reusable Learning Objects* ou, simplesmente, *Learning Objects* - RLOs ou LOs), que vêm sendo apontados pela literatura como uma solução eficiente para os problemas de padronização e baixo custo de desenvolvimento de conteúdo instrucional. Trataremos, também, das questões referentes à montagem de cursos (automatizada e semi-automatizada) a partir desses elementos abordando, inclusive, as necessidades de armazenamento, pesquisa e recuperação dos LOs.

Para tal, no capítulo 2 apresentaremos as definições mais comumente dadas para LOs, a história da evolução do conceito, suas características e a proposta da Cisco, entendida como uma das principais metodologias de desenvolvimento de LOs. Apresentaremos, também, a proposta de extensão dessa metodologia feita pelo Centro para Tecnologia Instrucional da Universidade Nacional de Singapura, como o objetivo de atender não somente as áreas de treinamento em empresas como, também, ensino de nível superior.

Diante do entendimento unânime de que LOs devem ser modulares e descritos de forma padronizada para que possam ser interoperáveis e mais facilmente

pesquisáveis e reutilizados, o IEEE desenvolveu o padrão LOM (metadados de LOs), que serve como infra-estrutura comum de outros importantes padrões como o IMS e o SCORM. Por essa razão pretendemos adotar o IEEE-LOM como o padrão a ser seguido nesse e em nossos estudos futuros. No capítulo 3 apresentaremos o IEEE-LOM, o IMS e o SCORM (o primeiro mais detalhadamente), o que nos permitirá conhecer melhor os padrões, entender a importância do IEEE-LOM e verificar que (e como) os três padrões se inter-relacionam fortemente.

No capítulo 4 estudaremos as possíveis relações que um LO pode manter com outro(s), iniciando pelas relações definidas no LOM e identificando outras que podem ser criadas e/ou modificadas ao longo da execução de um conteúdo instrucional.

No capítulo 5 apresentaremos os conceitos, terminologia e o modelo de dados de LOs que decidimos adotar em nossos estudos. Os agregados de LOs nos diversos níveis (igualmente entendidos como LOs) farão parte de bancos de dados de LOs. As características gerais de uma arquitetura em camadas para depósitos de LOs (na realidade um modelo de referência, baseado no modelo ANSI/SPARC) serão também apresentadas no capítulo 5. Esses modelos servirão de base para nossas pesquisas e implementações futuras.

No capítulo 6 mostraremos algumas características de nossa proposta para um ambiente de montagem (criação) e execução de conteúdo instrucional. Mais detalhes estão descritos em outros resultados de nossas pesquisas.

Finalmente, no capítulo 7, apresentaremos nossas conclusões e uma relação de trabalhos que compõem nossa (enorme) “*To Do List*”.

2 - Objetos de Aprendizado (LOs): Conceitos, Benefícios e Características

2.1 – Definições

Existem muitas definições para objetos de aprendizado reutilizáveis (*Learning Objects* ou *Reusable Learning Objects* – LO ou RLO), mas existe um consenso em torno do conceito básico de *porções reutilizáveis de conteúdo instrucional*. A indústria de multimídia lutou por uma definição de LOs por anos, antes que a necessidade de uma definição formal diminuísse diante do entendimento desse conceito pela indústria, entendimento esse que foi considerado mais importante que a definição em si. Três definições estão a seguir.

Um objeto de aprendizado ou objeto de aprendizado reutilizável é [4] uma coleção reutilizável de material usado para apresentar e dar apoio a um único objetivo de aprendizado ou [5] um *pequeno componente instrucional* que pode ser usado para suportar o aprendizado em *ambientes diferentes* ou ainda [6] qualquer entidade, digital ou não-digital (física), que pode ser usada para aprendizado, educação ou treinamento. Um pequeno componente instrucional é um módulo ou lição que se propõe a ensinar um

conceito específico, fato, procedimento, processo ou princípio. Ambientes diferentes implicam que um objeto de aprendizado pode ser usado em sistemas de gerência de aprendizado (*learning management systems* - LMS) ou sistemas de gerência de conteúdo de aprendizado (*learning content management systems* - LCMS) diferentes. Por exemplo, um módulo ou um LO desenvolvido para ensinar as vogais do alemão pode se tornar acessível através do IVLE (*Integrated Virtual Learning Environment*) [7] como também através de outros LCMS tais como o BlackBoard [8] e o WebCT [9].

Os elementos componentes dos LOs são tipicamente identificados (ou descritos) através de *tags* (etiquetas) XML definidas por um conjunto de padrões internacionais e diretrizes de especificação. LOs são igualmente catalogados através de *tags*, conforme os mesmos padrões e diretrizes, e armazenados em repositórios. As *tags* de identificação dos elementos e LOs auxiliam os mecanismos de busca na localização do objeto quando o usuário solicita uma busca no repositório.

Um instrutor pode montar um novo conteúdo através da definição de uma composição apropriada de LOs pré existentes e pode publicá-lo, colocando-o em um LCMS tal como o IVLE. A idéia aqui é minimizar a duplicação do esforço de produção desses módulos de aprendizado que é, em geral, bastante custosa, na medida em que os módulos podem conter apresentações multimídia, simulações e animações. LOs provêm, com isso, uma grande flexibilidade na organização de material de ensino.

2.2 - Benefícios do Uso de LOs

Relacionamos, nessa seção, um conjunto de benefícios proporcionados pelo uso de LOs em cursos a distância, tanto do ponto de vista dos autores dos cursos quanto do ponto de vista dos alunos.

Benefícios para os autores [10]:

- *Templates* podem ser desenvolvidos para garantir a consistência entre o projeto e o desenvolvimento de LOs dentro de uma organização;
- Usando-se *templates*, os autores podem escrever conteúdos efetivos e eficientes para treinamento baseado no trabalho ou na tarefa;
- Através de pesquisa nas *tags* XML dos metadados que descrevem os LOs um autor pode mais facilmente localizar e reusar um objeto;
- Autores podem combinar objetos antigos e novos para satisfazer às necessidades de suas audiências;
- O uso de objetos suporta tanto o reúso quanto à reorientação de propósito de qualquer recurso, de um pequeno elemento de mídia à toda a estrutura de um curso;
- A formatação e o estilo de apresentação dos objetos são definidos pelo sistema e preferências do aluno, ou seja, os autores não precisam despende tempo com questões de interface com o usuário;
- O mesmo banco de dados de objetos pode ser usado para aprendizado convencional, i.e., liderados por instrutores, para aprendizado através de CBT (com isolamento do aluno), por ferramentas de suporte à performance, por salas de aula virtuais, PDAs (*Personal Digital Assistants* - os *Palm Tops*), ou uma combinação dessas possibilidades.

Benefícios para os alunos:

- No caso de operações sem conexão, apenas os objetos e as seqüências de execução que fazem parte dos módulos sendo cursados no momento precisam ser carregados no nó de execução do aluno;
- É possível dispor-se de um ambiente onde os LOs possam ser organizados (quais e em qual seqüência) de forma automatizada, por demanda, através de definição, por parte do interessado, de parâmetros como custo, duração, grau de profundidade, pré e pós-requisitos etc.

2.3 – Breve História dos LOs

A história dos LOs não tem uma origem clara. Consta [4] que o termo LO começou a ser usado no início da década de 90. É difícil determinar, exatamente, quando e quem cunhou ou o termo “*learning object*”, mas o crédito por isso tem sido dado a Wayne Hodgins, desde então na AutoDesk.

Em 1992, Wayne estava assistindo a um de seus filhos a brincar com blocos de montagem Lego enquanto matutava sobre problemas ligados a estratégias de aprendizado. Wayne compreendeu, naquele momento, que a indústria precisava de blocos de construção para aprendizados que fossem *plug-and-play* e interoperáveis. Ele denominou esses blocos de construção de “*learning objects*”.

Entre 1992 e 1995 muitos grupos isolados começaram a trabalhar com o conceito inicial de LOs. O Grupo de Metadados de Objetos de Aprendizado (*Learning Object Metadata Group*) do Instituto Nacional de Ciência e Tecnologia (*National Institute of Science and Technology*) e a CEdMA (*Computer Education Management Association*) investigaram questões relativas a LOs, incluindo modularidade, orientação para o armazenamento em banco de dados¹ e uso de elementos de marcação (*tagging*), que hoje denominamos de metadados. Esses dois grupos, e alguns outros como eles, levantavam questões para considerações futuras.

De 1994 a 1996 muitos outros grupos se envolveram na questão. O IEEE, o IMS Global Learning Consortium Inc. e a ARIADNE (*Alliance of Remote Instructional Authoring and Distribution Networks for Europe*) desenvolveram ou iniciaram trabalhos na área de LOs. Nessa mesma época, a Oracle percebeu que LOs eram críticos para sua futura estratégia em aprendizado e, entre 1994 e 1995, iniciou pesquisas que resultaram na *Oracle Learning Application* (OLA), que foi uma tentativa inicial no sentido de se dispor de um ambiente de *autoria* (desenvolvimento) de cursos usando LOs.

Embora a OLA nunca tenha frutificado na Oracle, Tom Kelly e Chuck Barritt mudaram-se para a Cisco Systems e continuaram seus trabalhos em LOs, que culminaram com a publicação, em 1998, do *white paper* da Cisco sobre Objetos de Aprendizado Reutilizáveis. Esse trabalho, em conjunto com o trabalhos dos organismos de padronização e especificação da indústria, colaboraram bastante para trazer RLOs a um lugar considerado de destaque por toda a indústria a partir do ano 2000.

¹ Tradução dada para *database centricity*

Apesar da indústria estar consciente dos benefícios de se usar objetos de informação e de aprendizado padronizados, reutilizáveis e armazenáveis em bancos de dados, poucas empresas dispõem hoje em dia de processos e ferramentas de projeto, desenvolvimento e implantação de LOs [10].

Como continuidade dos esforços da Cisco na pesquisa em LOs, a mesma publicou em novembro de 2001 a versão 4.0 de sua estratégia de uso de RLOs, que apresentaremos mais adiante.

2.4 – Conteúdo, Construção e Uso dos LOs

Embora haja consenso na indústria de que é necessário criar-se conteúdos instrucionais modulares e padronizados, várias soluções distintas vêm sendo dadas para o problema da definição de conteúdos de LOs.

Duas das características mais discutidas de um LO são o seu tamanho e forma. Segundo a Cisco, o tamanho e forma de um LO devem ser definidos por cada organização. Essas decisões devem ser baseadas nas necessidades, ferramentas, processos e objetivos de negócio da organização. A Cisco, por exemplo, adotou uma hierarquia de dois níveis para LOs e RLOs (que serão vistos mais adiante) a serem classificados segundo os cinco tipos de informação propostos por M. David Merrill e Ruth C. Clark: conceito, fato, procedimento, processo e princípio. O tamanho dos objetos depende, então, do nível de hierarquia: do menor elemento de mídia pesquisável até coleções desses elementos que formam RLOs que, por sua vez, podem ser combinados para formarem uma lição (ou um RLO). Embora, segundo a Cisco, a questão do tamanho dos LOs deva ser deixada a critério da organização, um lembrete importante que podemos tomar emprestado da comunidade de Engenharia de *Software* é que objetos grandes são mais difíceis de serem reutilizados.

Como já mencionamos, a Cisco desenvolveu e publicou sua própria estratégia de objetos de aprendizado reutilizáveis. Julgamos relevante estudarmos esse documento [10], extraindo do mesmo os conceitos mais importantes, que serão apresentados na seção a seguir. Logo após, na seção 2.4.2, apresentaremos a proposta do Centro para Tecnologia Instrucional da Universidade Nacional de Singapura, entendido por eles próprios como uma extensão do modelo da Cisco, concebida para atender, também, ao ensino de nível superior.

2.4.1 - A Proposta da Cisco Systems

A Cisco inicia seu documento reconhecendo a necessidade de abandonar-se a prática da criação e *entrega*² de cursos a distância compostos por blocos de treinamento grandes e inflexíveis, adotando-se objetos armazenáveis em banco de dados que podem ser reutilizados, pesquisados e modificados. A esse esforço de pesquisa a Cisco deu o nome de *Reusable Learning Object (RLO) Strategy*, que se encaixa em sua Arquitetura de Solução para *E-Learning*. Essa, por sua vez permite que a Cisco desenvolva, gere e entregue todo o tipo de informação e treinamento via Internet. Pelo fato dessa

² Usamos o termo entrega como tradução direta de *delivery*, que significa colocar um conteúdo à disposição dos alunos.

arquitetura ser independente de ferramenta e de plataforma, ela suporta uma variedade de aplicações de autoria tradicional e eletrônica, de entrega e de gerência de aprendizado.

A arquitetura de solução para *e-learning* da Cisco contém as seguintes camadas: acesso, plataforma de aplicação e infra-estrutura de rede. A figura 2.1 ilustra.

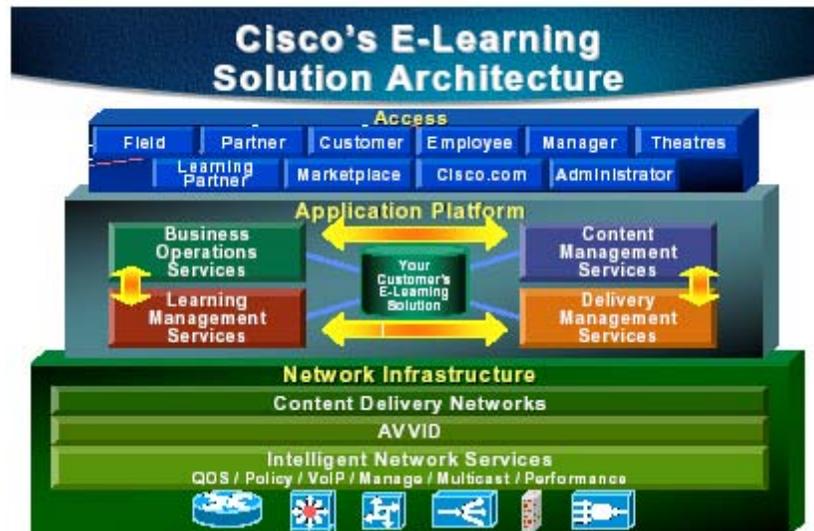


Figura 2.1 - Arquitetura de solução para *e-Learning* da Cisco [11]

Como podemos verificar na figura, a camada de acesso atende a múltiplas audiências. Cada uma dessas audiências pode ter seus próprios esquemas ou projetos de portais de acesso adaptados às suas próprias necessidades. Todos os usuários acessam a solução de *e-learning* através da camada de acesso.

A camada de infra-estrutura de rede serve para a distribuição dos serviços providos pela camada de aplicação. Essa camada deve garantir disponibilidade, escalabilidade e performance. Deve, também, permitir o tráfego de dados, voz e vídeo, assim como gerenciar a segurança, QoS e a distribuição de conteúdo.

A criação, armazenamento, gerência e entrega de um objeto residem na camada de plataforma de aplicação (intermediária às duas anteriormente apresentadas). Essa camada é dividida em quatro partes, para separar os processo de concepção, desenvolvimento, entrega e gerência em áreas de responsabilidade distintas. Essas partes são: serviços de operações de negócio, gerência de conteúdo, gerência de entrega e gerência de aprendizado. Nos deteremos mais na parte de gerência de conteúdo, que trata da maneira segundo a qual a informação é segmentada, agrupada e armazenada no sistema de gerência de conteúdo. Detalhes das demais partes podem ser encontrados em [11].

Os Serviços de gerência de conteúdo permitem que os provedores de conteúdo montem, registrem, gerenciem e publiquem seus conteúdos. A operação de registro de conteúdo (obrigatoriamente executada antes da publicação) consiste, além do envio do mesmo ao gerenciador, sua categorização e rotulação (descrição) através do uso de metadados. É importante que as ferramentas de construção de conteúdo, como Word,

PowerPoint, DreamWeaver, etc., se integrem perfeitamente a esses serviços. O gerenciador de armazenamento usado deve ser capaz de tratar tanto dados sem estrutura (imagens, executáveis, arquivos binários quaisquer) quanto dados estruturados. É importante que se disponha, também, de mecanismos de busca dos conteúdos armazenados (*mineradores* de objetos), para que se possa facilmente localizá-los e (re)utilizá-los na montagem de lições e cursos. Percebe-se, com isso, a relevância de se dispor de um esquema padronizado de classificação e descrição do conteúdo e, em função disso, abordaremos o assunto "metadados" em detalhes, mais adiante. O registro armazena a localização, os metadados descritivos e estruturais associados a um objeto de aprendizado específico e o objeto propriamente dito.

O ILSG (*Internet Learning Solutions Group* da Cisco) combina *objetos de informação reutilizáveis* (RIOs) para formar estruturas maiores, denominadas pela Cisco de RLOs³. Em outras palavras, um RLO encapsula um módulo que provê uma visão geral do RLO, um módulo que o sumariza, um módulo que permite a avaliação do aproveitamento por parte do aluno e de 5 a 9 (7 ± 2) RIOs. A figura 2.2 ilustra.

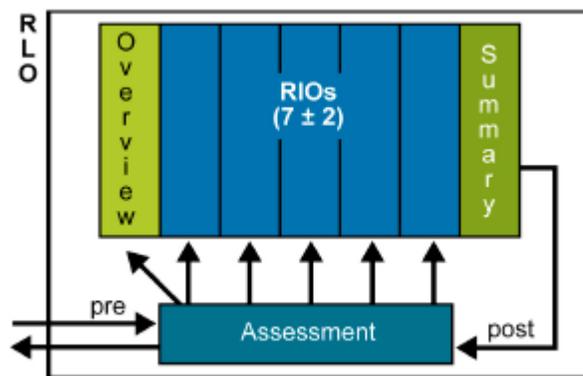


Figura 2.2 - Estrutura interna do RLO [10]

Um RLO cumpre um único objetivo dentro de uma seqüência definida de RLOs que formam, segundo a Cisco, uma *lição*. Cada RIO, por sua vez, é desenvolvido de forma a cumprir uma seqüência que atende ao objetivo do RLO. Um RIO compõe-se de itens de conteúdo, itens de prática e itens de avaliação, conforme ilustrado pela figura 2.3.

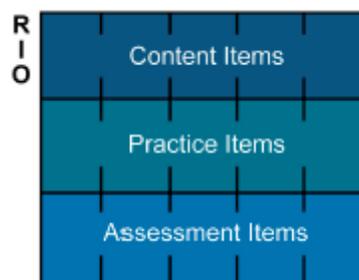


Figura 2.3 - Estrutura de um RIO [10]

³ RLOs são nomes genéricos dados a objetos de aprendizado reutilizáveis e não estão ligados, como vemos em todo esse trabalho, exclusivamente à arquitetura da Cisco Systems.

Para facilitar a padronização do conteúdo, o ILSG optou por adotar para os RLOs a classificação proposta por M. David Merrill e Ruth C. Clark: um conceito, um fato, um procedimento⁴, um processo⁵ ou um princípio (mais detalhes em [10]).

Um aluno pode decidir cursar o RLO por inteiro (lição). Como em qualquer lição tradicional, o RLO dá ao aluno o contexto de aprendizado necessário, o conhecimento e habilidades que eles precisam para conseguirem um dado objetivo, e também um mecanismo de avaliação.

Um administrador ou gerente de currículo pode combinar RLOs para formar estruturas maiores, tais como módulos, unidades e cursos. A estrutura RLO-RIO define uma hierarquia em dois níveis. A ILSG escolheu o rótulo de lição para RLOs e o de seção para cada RIO dentro de uma lição. RLOs se agrupam em estruturas maiores, dependendo do autor, dos requisitos de negócio e do empacotamento da oferta no sistema de gerência de aprendizado. Uma hierarquia que *pode* ser usada, do maior nível de agregação para o menor, é currículo, unidade, módulo, lição (RLO) e seção (RIO). A figura 2.4 ilustra.

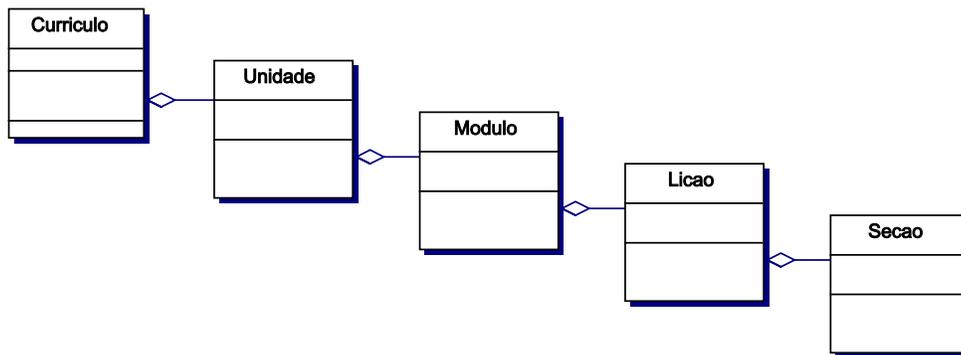


Figura 2.4 – Sugestão da Cisco Systems para níveis de agregação em conteúdos de aprendizado.

2.4.2 - A Proposta do CIT-NUS

O Centro para Tecnologia Instrucional da Universidade Nacional de Singapura decidiu dar uma definição ampla do que se constitui um LO. Essa definição modifica e estende a definição do escopo de LOs dada pela Cisco Systems, que foi criado especificamente para treinamento em engenharia e em outros campos científicos, de forma a atender ao propósito de aprimorar o ensino de disciplinas em uma universidade.

Segundo [5], um LO deve⁶ se constituir dos elementos apresentados resumidamente nos itens a seguir.

- **Visão Geral:** deve incluir as seguintes seções: uma introdução, importância da lição, objetivos, pré-requisitos, aplicação prática e linhas gerais do conteúdo.

⁴ Quando se deseja ensinar um procedimento executado em um trabalho.

⁵ Quando se deseja ensinar como um sistema funciona.

⁶ Uma disciplina na definição do conteúdo dos LOs deve existir para que se garanta a qualidade dos seus repositórios.

- **Apresentação do Conteúdo:** o conteúdo também pode ser classificado em categorias amplas como conceito, fato, procedimento, processo e princípio. Pontos de vista discordantes, escolas de pensamento, críticas e opiniões, que são comuns na natureza humana, podem ser classificados como conceitos. O conteúdo pode ser apresentado em uma variedade de modos de ensino. Neles se incluem lições lideradas por instrutores, tais como *web-casts* (difusões de conteúdos no contexto da *web*), aprendizado baseado em problemas, estudos de caso e *sites* de cursos interativos. Uma variedade de mídias pode ser usada, por exemplo, textos, vídeo, áudio, gráficos, animações 2D e 3D e simulações.
- **Prática:** a prática pode conter projetos práticos, trabalho de campo, projetos em equipe, estudos de casos, trabalhos em laboratório, atividades em pequenos grupos, questões de múltipla escolha, questões de “preencha os espaços em branco”, associações, ensaios e outros tipos de questionários. A prática pode ser apresentada em uma variedade de formas tais como jogos, quebra-cabeças, gráficos, vídeos, animações, simulações ou simplesmente textos.
- **Avaliação:** nesse item são recomendados pré e pós questionários. Componentes de avaliação são similares aos componentes de prática mencionados acima.
- **Sumário:** essa seção contém a revisão da lição, provê informação sobre a próxima⁷ lição e recursos adicionais que podem ser úteis para o entendimento melhor do tópico.

Independentemente da indiscutível diversidade de conceituação e de definição da estrutura e conteúdo dos LOs, é importante que a maneira de descrevermos as unidades de conteúdo seja padronizada, o que facilitará a busca e a interoperabilidade de conteúdos. Com essa motivação, o IEEE desenvolveu esforços no sentido de padronizar a estrutura dos descritores de LOs. Essa padronização é expressa em termos de um conjunto de metadados, que apresentamos a seguir.

3 – Metadados de LOs

Metadados de LOs são informações sobre os dados que compõem LOs, sejam eles físicos ou digitais. Como o número de objetos cresce exponencialmente e nossas necessidades por aprendizado se expandem nas mesmas proporções, a falta de informações sobre os conteúdos dos LOs já desenvolvidos impõe limitações críticas e fundamentais na nossa capacidade de pesquisar, gerenciar e usar LOs [6].

Com o intuito de disciplinar a descrição do conteúdo dos LOs, permitindo a uniformização e ampliando a qualidade da base de LOs e também a reutilização ampla dos mesmos (independentemente até do LCMS usado), várias organizações procuram criar padrões para metadados de LOs. São vários os existentes hoje em dia, dentre eles destacam-se o LOM (*Learning Object Metadata*) do IEEE *Learning Technology Standards Committee* (LTSC), o do IMS Global Learning Consortium e o SCORM

⁷ Se os objetos de aprendizado são reutilizáveis, parte dessa informação deve ser preenchida após a colocação dos LOs na seqüência desejada.

(*Sharable Content Object Reference Model*), que apresentaremos adiante. Na realidade, como veremos também adiante, o padrão do IMS se baseia no LOM, descrevendo-o em XML, e o SCORM adota a descrição em XML do IMS (chamada de *Metadata XML Binding*) para compor um de seus “livros”. A figura 3.3 mostra o grafo de referências entre padrões.

3.1 - LOM do IEEE

3.1.1 - Introdução

O IEEE é uma organização credenciada para desenvolvimento de normas. O LTSC foi instituído pelo IEEE, pela *Computer Society* e pelo *Standards Activity Board* com o objetivo de desenvolver normas, orientações e práticas recomendadas na área de aprendizado suportado por computador. Ao contrário de organismos não credenciados, as especificações produzidas pelo IEEE tornam-se normas. O credenciamento de organização para a produção de normas não implica em qualidade ou utilidade do seu produto. Significa que as normas foram produzidas seguindo um processo rigoroso de formulação.

Os cerca de 20 grupos de estudos constituídos no LTSC classificam-se nas categorias de atividades gerais, atividades relacionadas ao aprendiz, atividades relacionadas ao conteúdo, dados e metadados, sistemas de gerência e aplicações.

O LOM tem suas origens nos projetos ARIADNE⁸ e IMS⁹ (desse último tratamos também nesse trabalho) e também se baseia em estudos de metadados desenvolvidos pelo grupo do Dublin Core¹⁰ [6].

Atualmente o IEEE-LOM encontra-se em desenvolvimento (é uma proposta de norma), sendo que o documento consultado (P1484) nesse trabalho foi o de versão 12.1/D6.4. A versão do esquema base de metadados que essa proposta descreve/especifica é a LOM V1.0.

3.1.2 - Objetivos

Um dos objetivos da norma é o de facilitar a busca, avaliação, aquisição e uso de objetos de aprendizado por parte de aprendizes, instrutores ou processos automatizados de *software*. Outro objetivo é o de facilitar o compartilhamento e intercâmbio de LOs (incluindo a possibilidade de um LO ser utilizado por LCMSs de fabricantes diferentes), possibilitando o desenvolvimento de catálogos e inventários ao mesmo tempo em que se leva em consideração a diversidade de contextos culturais e de língua onde os LOs e seus metadados possam ser empregados.

Através da especificação de um esquema conceitual comum, a norma garante que *bindings* (representações desses metadados através de linguagens conhecidas, como XML, por exemplo) entre metadados de LOs terão melhor grau de interoperabilidade semântica.

⁸ <http://www.ariadne-eu.org>

⁹ <http://www.imsproject.org>

¹⁰ <http://dublincore.org>

A intenção desse padrão é especificar-se um esquema base que pode ser usado para facilitar, por exemplo, a colocação automática e adaptável de LOs em seqüência por parte de agentes de *software*.

3.1.3 - Estrutura Básica dos Metadados

A futura norma especifica um esquema conceitual de dados que define a estrutura da instância de metadados de um objeto de aprendizado. Uma instância de metadados de um LO descreve as características relevantes do LO ao qual se aplica e é composta de elementos de dados. Esses elementos de dados compõem uma *hierarquia* (nós intermediários ou agregações¹¹ e folhas), que no primeiro nível enquadram-se nas seguintes categorias:

1. **de características gerais**, que agrupa as informações gerais que descrevem o LO como um todo;
2. **de ciclo de vida**, que agrupa as características relacionadas à história e estado corrente desse LO e aquelas que afetaram esse LO durante sua evolução;
3. **de meta-metadados**, agrupa as informações sobre a instância de metadados propriamente dita (ao invés do conteúdo sendo descrito);
4. **técnicas**, agrupa as características e requisitos técnicos do LO;
5. **educacionais**, agrupa as características educacionais e pedagógicas do LO;
6. **de direitos**, agrupa os direitos de propriedade intelectual e condições de uso do LO;
7. **de relações**, agrupa as características que dizem respeito às possíveis relações entre LO e outros LOs;
8. **de anotação**, agrupando os elementos que contêm comentários sobre o uso educacional do LO e sobre por quem e quando os comentários foram feitos;
9. **de classificação**, que agrupa os elementos que descrevem o LO com relação a um sistema de classificação específico.

A figura 3.1 mostra a árvore completa do LOM.

O LOM V1.0 é apresentado em uma tabela onde as linhas contêm os elementos de dados descritos/especificados em seus diversos níveis hierárquicos e onde as colunas são, para cada elemento de dado, como abaixo:

1. o número (1, 1.1, 1.1.2 ...);
2. o nome;
3. a explanação (descrição em prosa do elemento);
4. o número de valores permitidos;
5. a ordem dos valores permitidos (não especificada, ordenado e não ordenado);
6. o espaço de valores (domínio e/ou norma aplicável);
7. tipo de dado;
8. exemplo(s).

¹¹ No LOM V1.0, nós intermediários não possuem campos de valores ou tipos de dados

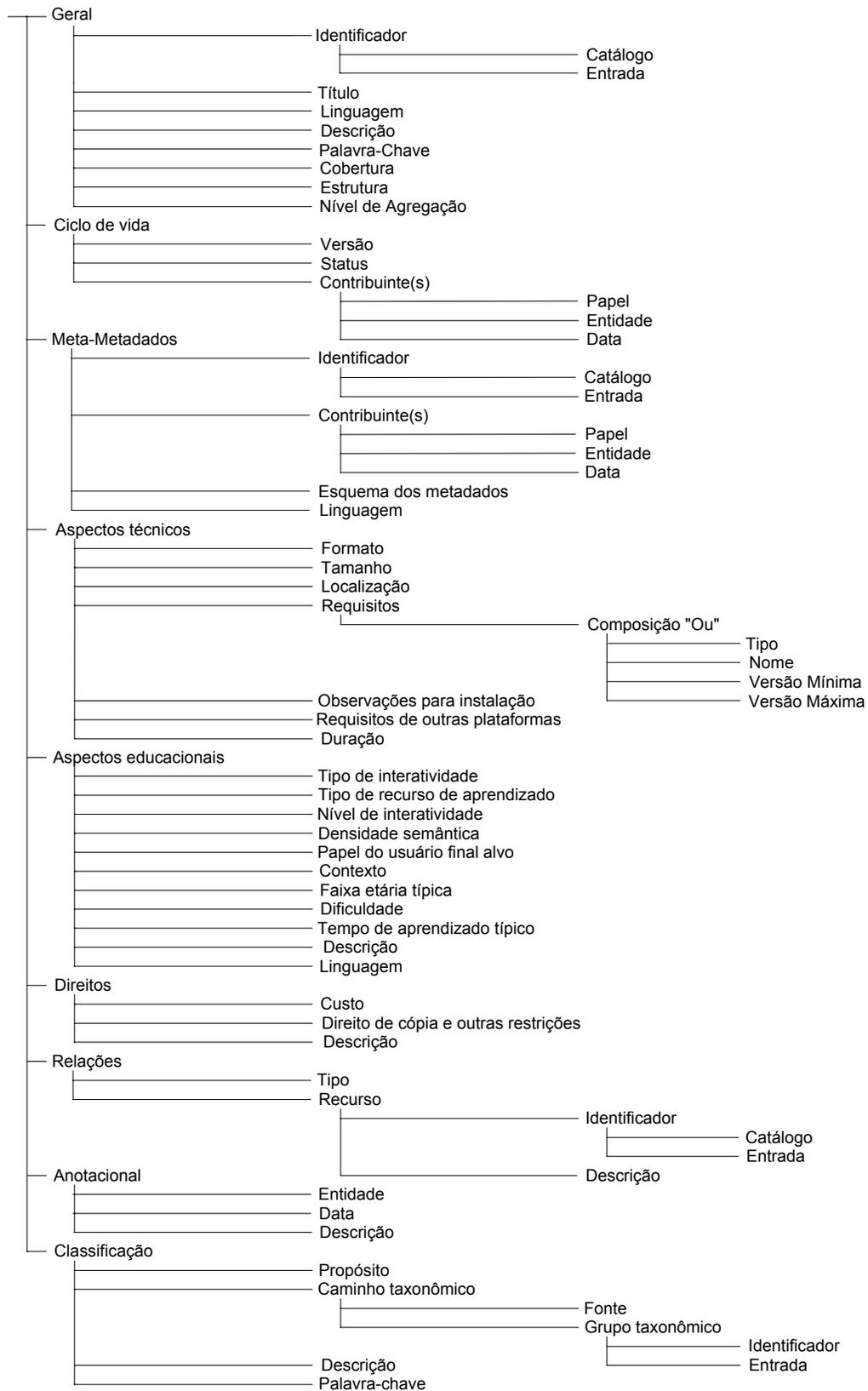


Figura 3.1 – Árvore completa do LOM

Apresentamos na tabela 3.1 a seguir a árvore dos elementos de dados do LOM V1.0, onde comentamos e exemplificamos os nós da figura 3.1. O campo “mult” diz

respeito à multiplicidade de ocorrência do respectivo atributo: “0..1” significa 0 ou 1, “*” significa 0 ou muitos.

No.	Nível 1	Nível 2	Nível 3	Nível 4	Mult	Definições/Comentários/Exemplos
1	Geral				0..1	
1.1		Identificador			*	
1.1.1			Catálogo		0..1	Nome ou designador do esquema de identificação ou catalogação. Ex. "ISBN", "ARIADNE"
1.1.2			Entrada		0..1	O valor do identificador segundo o esquema acima.
1.2		Título			0..1	Nome dado ao LO. Ex. "A vida e os trabalhos de Leonardo da Vinci"
1.3		Linguagem			*	Código da linguagem segundo a ISO 639:1988 e sub-código do país, conforme a ISO 3166-1:1977. Ex.: "pr-BR".
1.4		Descrição			*	Descrição textual do conteúdo do LO. Ex.: ("pt", "Nesse vídeo clip, a vida e obra de Leonardo da Vinci são rapidamente apresentadas. O foco é em sua produção artística, mais notadamente a Mona Lisa")
1.5		Palavra-Chave			*	Palavra-chave que descreve o tópico do LO. Ex.: ("pt", "Mona Lisa")
1.6		Cobertura			*	O tempo, cultura, geografia ou região aos quais o LO se aplica. Ex.: ("pt", "Século 16 França")
1.7		Estrutura			0..1	Estrutura organizacional do LO. Pode ser: <ul style="list-style-type: none"> • "atômica", para um LO indivisível; • "coleção", para um conjunto de objetos sem nenhum relacionamento especificado entre eles; • "em rede", um conjunto de objetos com relacionamentos não especificados; • "hierárquica", um conjunto de objetos que possuem relacionamentos que podem ser representados como uma árvore e • "linear", um conjunto de objetos completamente ordenados como, por exemplo, conectados através de relacionamentos "anterior" e "posterior".
1.8		Nível de Agregação			0..1	A granularidade funcional do LO tomando valores de 1 a 4 e correspondendo, por exemplo, aos valores "raw" (bruto, menor nível de agregação), "lição", "curso" e "conjunto de cursos" ou "programa", que daria direito a um certificado. Objetos do nível 4 podem conter objetos do nível 3 ou podem conter outros objetos do nível 4 recursivamente.
2	Ciclo de vida				0..1	Essa categoria descreve a história e o estado corrente do LO e das entidades que afetaram o LO durante sua evolução
2.1		Versão			0..1	Ex.: ("pt", "1.2.alfa"),
2.2		Status			0..1	"draft", "final", "revised", "unavailable".
2.3		Contribuinte(s)			*	As entidades (pessoas, organizações) que contribuíram para os estados do LO ao longo de seu ciclo de vida.
2.3.1			Papel		0..1	Tipo de contribuição. Ex.: autor, publicador, iniciador, terminador, validador, editor, etc.
2.3.2			Entidade		*	Identificador de uma informação sobre uma entidade (vCard), conforme definição pelas RFCs 2425 e 2426.
2.3.3			Data		0..1	Data da contribuição
3	Meta-Metadados				0..1	Essa categoria descreve o registro de metadados propriamente dito.
3.1		Identificador			*	
3.1.1			Catálogo		0..1	Nome ou designador do esquema de identificação ou catalogação, retirado do repertório da ISSO/IEC 10646-1:2000. Ex.: "Ariadne", "URI"
3.1.2			Entrada		0..1	Identificador retirado do catálogo selecionado. Ex.: "KUL532", "http://www.ieee.org/descriptions/1234"
3.2		Contribuinte(s)			*	As entidades (pessoas, organizações) que contribuíram para os estados da instância dos metadados ao longo de seu ciclo de vida.
3.2.1			Papel		0..1	
3.2.2			Entidade		*	

3.2.3			Data		0..1	
3.3		Esquema dos metadados			*	Ex.: "LOMv1.0"
3.4		Linguagem			0..1	
4	Aspectos técnicos				0..1	Descreve as características e requisitos técnicos do LO.
4.1		Formato			*	Tipos dos dados de todos os componentes desse LO (tipos MIME baseados no IANA, vide RFC2048:1996, ou "non-digital"). Ex.: "video/mpeg", "application/x-toolbook", "text/html", "non-digital".
4.2		Tamanho			0..1	... do LO em bytes.
4.3		Localização			*	String que pode ser usada para acessar o LO, como uma URL ou método que mapeia em uma URL.
4.4		Requisitos			*	Requisitos técnicos necessários para o uso desse LO. Se existem repetições desse item significa que há múltiplos requisitos e que todos eles são necessários (AND).
4.4.1			Composição "Ou"		*	Grupamento de múltiplos requisitos. O requisito composto é satisfeito quando um dos requisitos componentes é satisfeito.
4.4.1.1				Tipo	0..1	Tipo da tecnologia necessária de <i>hardware</i> , <i>software</i> , rede, etc. ¹²
4.4.1.2				Nome	0..1	Nome do requisito. Ex.: "MS-DOS", "Windows", "Netscape"
4.4.1.3				Versão Mínima	0..1	Valores de acordo com a ISO/IEC 10646-1:2000
4.4.1.4				Versão Máxima	0..1	Idem
4.5		Observações para instalação			0..1	("en", "Unzip the zip file and launch index.html in your web browser.")
4.6		Requisitos de outras plataformas			0..1	Informação sobre outro <i>hardware/software</i> necessário que não podem ser descritos no item 4.4. Ex.: ("en", "sound card"), ("en", "runtime X")
4.7		Duração			0..1	
5	Aspectos educacionais				*	Descreve as características-chave educacionais ou pedagógicas do LO, informações que são úteis para professores, gerentes, autores e aprendizes.
5.1		Tipo de interatividade			0..1	Modo de aprendizado predominante nesse LO. Pode ser "active", "expositive", ou "mixed".
5.2		Tipo de recurso de aprendizado			*	Tipo específico do LO: pode ser "exercise", "simulation", "questionnaire", "diagram", "figure", "graph", "index", "slide", dentre outros.
5.3		Nível de interatividade			0..1	Pode variar de "very low" a "very high"
5.4		Densidade semântica			0..1	O grau de concisão de um LO. A densidade semântica de um LO pode ser estimada em termos de seu tamanho, abrangência ou duração, para LOs com tempos bem definidos de exibição. A densidade semântica de um LO é independente de sua dificuldade. Varia de "very low" a "very high".
5.5		Papel do usuário final alvo			*	Principal usuário (usuário alvo) para o qual o LO foi desenvolvido, primeiro os mais dominantes, podendo ser "reacher", "author", "learner" e "manager".
5.6		Contexto			*	O principal ambiente dentro do qual o LO foi projetado para ser usado. Pode ser "school", "higher education", "training" e "other".
5.7		Faixa etária típica			*	Ex.: "0-9", "17-23", "18-", ("en", "suitable for children over 7").
5.8		Dificuldade			0..1	De "very easy" a "very difficult".
5.9		Tempo de aprendizado típico			0..1	
5.10		Descrição			*	Comentário sobre como esse LO deverá ser usado.
5.11		Linguagem			*	A linguagem humana dominada pelo usuário alvo desse LO.
6	Direitos				0..1	Descreve os direitos de propriedade intelectual e condições de uso do LO.

¹² Embora o padrão considere, para esse campo (4.4.1.1), as tecnologias de *hardware*, de rede, etc. os valores relacionados no domínio são, exclusivamente, "sistema operacional" e "*browser*".

6.1		Custo			0..1	Especifica de é necessário um pagamento pelo uso do LO. Pode ser "Yes" ou "No".
6.2		Direito de cópia e outras restrições			0..1	Se existem restrições de copyright, dentre outras. Pode ser "Yes" ou "No".
6.3		Descrição			0..1	Comentários sobre as condições de uso do LO.
7	Relações				*	Essa categoria define as possíveis relações entre esse LO e outros. Múltiplas relações são definidas por múltiplas ocorrências dessa categoria. Se há relações com mais de um LO então deve der definida uma instância para cada objeto.
7.1		Tipo			0..1	Baseadas nas categorias do Dublin Core: "ispartof", "haspart", "isversionof", "hasversion", "isformatof", "hasformat", "references" (referencia), "isreferencedby", "isbasedon", "isbasisfor", "requires" e "isrequiredby"
7.2		Recurso			0..1	O LO alvo que a relação anterior referencia.
7.2.1			Identificador		*	Rótulo globalmente único que especifica o LO alvo.
7.2.1.1				Catálogo	0..1	Nome ou designador do esquema de identificação ou catalogação, retirado do repertório da ISSO/IEC 10646-1:2000. Ex.: "Ariadne", "ISBN"
7.2.1.2				Entrada	0..1	O valor segundo o esquema adotado. Ex.: "2-7342-0318", "LEAO875", "http://www.ieee.org/"
7.2.2			Descrição		*	Descrição do LO alvo. Ex.: ("en","The QuickTime movie of the Mona Lisa on the web site of the Louvre museum.")
8	Anotacional				*	Essa categoria provê comentários sobre o uso educacional do LO e sobre quem e quando essas informações foram criadas.
8.1		Entidade			0..1	Análogo a 2.3.2
8.2		Data			0..1	Análogo a 2.3.3
8.3		Descrição			0..1	Ex.: ("en", "I have used this video clip with my students. They really enjoy being able to zoom in on specific features of the painting. Make sure they have a broadband connection or the experience becomes too cumbersome to be educationally interesting.")
9	Classificação				*	Essa categoria descreve onde o LO se enquadra segundo um sistema de classificação em particular
9.1		Propósito			0..1	O propósito da classificação do LO, podendo ser "discipline", "idea", "prerequisite", "educational objective", "accessibility", "restrictions", "educational level", "skill level", "security level", "competency".
9.2		Caminho taxonômico			*	O caminho taxonômico em uma dato sistema de classificação. Cada nível subseqüente é um refinamento da definição do nível anterior.
9.2.1			Fonte		0..1	O nome do sistema de classificação. Ex.: ("en","ACM"), ("en","MESH"), ("en","ARIADNE")
9.2.2			Grupo taxonômico ¹³		*	Um termo particular dentro de uma taxonomia. Ex.: [{"12",("en","Physics")}, {"23",("en","Acoustics")}, {"34",("en","Instruments")}, {"45",("en","Stethoscope")}] Um segundo grupo taxonômico para o mesmo LO poderia ser [{"56",("en","Medicine")}, {"67",("en","Diagnostics")}, {"34",("en","Instruments")}, {"45",("en","Stethoscope")}]
9.2.2.1				Identificador	0..1	O identificador do grupo taxonômico, segundo o repertório da ISO/IEC 10646-1:2000. Ex.: "320", "4.3.2", "BF180"
9.2.2.2				Entrada	0..1	O rótulo textual do grupo taxonômico.: Ex.: ("pt", "Ciências Médicas")
9.3		Descrição			0..1	Ex.: ("en","A medical instrument for listening called a stethoscope.")
9.4		Palavra-chave			*	Ex.: ("en", "diagnostic instrument")

Tabela 3.1 - Resumo da descrição/definição do IEEE LOM V1.0

¹³ Tradução para "taxon" retirada do Dicionário Inglês-Português, Antônio Houaiss, 1a. edição, 1a. tiragem, 1982.

3.1.4 - Conformidade e Mecanismos de Extensão do LOM V1.0

Todos os elementos de dados são opcionais. Isso significa que uma instância LOM que esteja no padrão pode incluir, por exemplo, valor apenas para um elemento de dados da cláusula 4. Como, para essa cláusula, o esquema base LOM V1.0 impõe um relacionamento de agregação, caso se decida definir o elemento 4.4.1.4, é preciso que se tenha também definidos os componentes 4.4.1, 4.4 e 4 (a árvore não pode ter folhas soltas).

Uma instância de metadados em conformidade *estricta* com o LOM só pode conter elementos de dados LOM. Uma instância de metadados em conformidade com o LOM pode conter elementos de dados estendidos. Uma instância LOM que não contém valor para algum elemento de dados LOM é, ainda sim, uma instância em conformidade.

Com a finalidade de maximizar a interoperabilidade semântica, elementos de dados estendidos não devem substituir elementos de dados na estrutura padrão do LOM. Por exemplo, uma organização não deve introduzir um novo elemento "nome" que substitua o elemento 1.2: Geral.Título.

Alguns tipos de extensão ao esquema base LOM V1.0 podem ser feitos utilizando-se o item de número 9 - Classificação, na medida em que dentro desse item qualquer outro sistema de classificação pode ser usado.

Extensões devem reter os domínios e tipos de dados do esquema base. Extensões não poderão definir tipos de dados ou domínios para os elementos de dados agregados (os que não são folhas da hierarquia).

3.2 - O IMS da IMS Global Learning Consortium, Inc.

3.2.1 - Introdução

A IMS foi formada em 1997 como um projeto dentro da *US National Learning Infrastructure Initiative* (NLII). Com o passar dos anos, entretanto, ela cresceu, tornando-se um consórcio internacional de mais de 250 organizações. O foco inicial em educação superior foi ampliado para incluir escolas e treinamento comercial. Esses membros contribuintes, além da rede de participantes, representam quase todos os setores na comunidade global de EAD.

As primeiras especificações de metadados foram publicadas em 1999. Desde então, uma lista de especificações foram rascunhadas e especificações mais novas são publicadas com bastante frequência.

No momento, a especificação de metadados mais atual é a versão 1.2.1 (*final*) de maio de 2001 [12]. Esse documento descreve os nomes, definições, organização e restrições dos elementos de metadados do padrão IMS e é composto de duas partes: (1) uma versão do IEEE-LOM, que não a atual, padrão no qual se baseia, e (2) um número de modificações e extensões aprovadas pelo setor técnico da IMS. As modificações relacionadas no documento já se encontram implementadas na versão do IEEE-LOM que apresentamos anteriormente, na medida em que o corpo técnico da IMS também

participa intensamente das discussões do padrão do IEEE. As extensões ao LOM são tratadas a seguir.

A IMS tem, também publicado, a codificação do seu padrão (usa o IEEE-LOM) em XML [13].

3.2.2 – DTDs e Schemas

Os arquivos XML contendo as instâncias dos metadados dos LOs são validados contra o padrão IMS de duas formas: os nomes dos elementos, modelo de conteúdo e os atributos dos elementos podem ser definidos em uma DTD (que pode ser colocada externamente ou internamente ao arquivo de metadados) ou através de um XML Schema. Esse último é o preferido dos autores pelo fato das linguagens de esquema serem extensíveis, permitindo que quem desenvolve adicione às linguagens informações adicionais como tipos de dados, herança e regras de apresentação. Isso faz com que linguagens de esquema sejam mais poderosas do que DTDs. Os programas de registro de LOs (são editores XML, em última instância) devem fazer uso desses esquemas para guiar o processo de criação dos metadados, colocando os elementos corretos nos lugares corretos. A especificação corrente em XML do IMS tratada em [13] define a DTD e o *W3C XML Schema* através dos arquivos *imsmd_rootv1p2.dtd* e *imsmd_rootv1p2.xsd*, respectivamente.

Após as considerações iniciais sobre XML, DTDs e XML Schema encontramos, sob o título “Descrição Narrativa da Ligação XML”, a descrição detalhada em XML, com exemplos, dos elementos do IEEE-LOM.

Um exemplo de codificação dos metadados de um LO encontra-se no apêndice A.

3.2.3 - Extensões

O *draft* do padrão IEEE-LOM pode não capturar adequada e completamente todos os metadados necessários para descrever um LO e seus usos [12]. O LOM permite extensões para elementos e estruturas de metadados proprietários que devem ser feitas de forma a nunca sobrescrever (substituir) qualquer elemento ou cláusula do padrão (vimos isso no item 3.1.4). A decisão de implementar extensões ao LOM não deve ser dada sem razões fortes porque extensões podem comprometer a interoperabilidade, ressaltando-se que é altamente desejável que haja consenso no uso específico das mesmas e que qualquer proposta de uso ou uso efetivo de extensões devem ser levados ao conhecimento do grupo de trabalho de metadados da IMS.

Existem duas formas gerais de extensões ao padrão IMS [12]:

1. Através da adição de um ou mais elementos já definidos no IEEE-LOM à estrutura, em locais outros onde não os definidos no padrão, na medida em que o LOM contém alguns elementos com definições que não são específicas a um contexto particular. O reuso desses elementos em outros contextos pode ser considerado, contanto que suas definições não sejam alteradas (vide exemplo do elemento `<role>` a seguir);
2. Através da adição de novos elementos pertencentes a outros *namespaces*.

A especificação XML/IMS [13] dos metadados define um modo de tratarmos uniformemente todas as extensões proprietárias definidas pelos usuários. Esse tratamento pode ser feito de duas formas diferentes, dependendo do tipo de arquivo de controle (DTD, XSD ou XDR) escolhido para verificar a instância de metadados. A primeira maneira é através do uso de DTDs e a segunda é através do uso de definições em *XML Schema*.

Mais detalhes sobre capacidade e formas de extensão do modelo podem ser encontrados em [13].

Extensões Usando DTDs:

A IMS estabelece o uso do elemento `<extension>` para definições de extensões usando uma DTD. Esse elemento é opcional *para cada ramo/folha da árvore* da estrutura de metadados e pode ser usado mais de uma vez em um mesmo elemento de dados.

Exemplo:

```
<coverage>
  <langstring>1880-1900</langstring>
  <extension>
    <role>Date Range</role>
  </extension>
</coverage>
```

Especificar extensões através de mais de um arquivo DTD é problemático [13]. A IMS recomenda usar linguagem de definição em *XML Schema* e sua capacidade de definir-se *namespaces* ao se especificar extensões.

Extensões Usando Definições em XML Schema:

O IMS suporta metadados extensíveis usando a linguagem de definição XML Schema através da especificação de outros elementos LOM, como uma extensão, ou como novos elementos (definidos em *namespaces* específicos). Elementos do *namespace* usado tipicamente contêm um prefixo significativo, tal como "adl" (para representar a iniciativa ADL), para identificar unicamente suas extensões. A IMS recomenda o uso das Recomendações W3C para *namespaces*¹⁴.

Um exemplo do uso do elemento LOM `<role>` como extensão a outro elemento LOM é dado a seguir.

```
<coverage>
  <langstring>1880-1900</langstring>
  <role>Date Range</role>
</coverage>
```

O uso do elemento de extensão de *namespace* é ilustrado a seguir:

```
<context>
  <langstring xml:lang="en">Military Training</langstring>
```

¹⁴ <http://www.w3.org/TR/1999/REC-xml-names-19990114>

```
<adl:type>
  <title>Roman military tactics</title>
  <langstring xml:lan="en">This example discusses how
  the Romans defined many aspects of modern
  warfare.</langstring>
</adl:type>
</context>
```

3.3 - Sharable Content Object Reference Model (SCORM)

3.3.1 - Introdução

A *Advanced Distributed Learning* (ADL) foi iniciada em 1997 para acelerar o desenvolvimento em grande escala de *software* de aprendizado dinâmico, a custo aceitável, para atender às necessidades de educação e treinamento dos militares e da força de trabalho americanos do futuro [5] e [14]. A ADL tem o suporte do Departamento de Defesa (DoD) e da *White House Office of Science and Technology Policy*. O SCORM, o documento desenvolvido pela ADL, provê um conjunto de especificações técnicas inter-relacionadas, construídas com base nos trabalhos desenvolvidos pela AICC (*Aviation Industry CBT Committee*), IMS e IEEE, visando a criação de um modelo unificado de conteúdo. A iniciativa ADL também definiu requisitos de alto-nível para conteúdo de aprendizado, tais como capacidade de reúso, acessibilidade, durabilidade e interoperabilidade.

Cada nova versão do SCORM, além de introduzir novos conceitos, incorpora as mudanças efetuadas em seus documentos-base, como o IMS e o IEEE-LOM.

A versão 1.2 do SCORM inclui exemplos de códigos que implementam aspectos do SCORM. Esses exemplos básicos servem para facilitar e acelerar implementações mais sofisticadas.

3.3.2 - Organização do SCORM

O SCORM trata cada especificação referenciada individualmente como um "livro" separado (possivelmente composto por mais de um volume). Versões futuras do SCORM poderão adicionar novos "livros" à coleção SCORM. A presente versão consiste de três livros, conforme apresentado a seguir.

O livro 1 contém uma visão geral da iniciativa ADL, os conceitos fundamentais por trás do SCORM e um sumário das especificações técnicas e orientações contidas nas demais seções.

O livro 2 contém orientações para identificar-se e agregar-se recursos em conteúdos de aprendizado estruturados. Esse livro descreve uma nomenclatura para conteúdo de aprendizado, descreve o *SCORM Content Packaging* e referencia o IMS (*IMS Learning Resource Meta-Data Information Model*) e o LOM. Juntas, essas especificações formam o *SCORM Content Aggregation Model - CAM*.

O livro 3, o “Ambiente SCORM de Tempo de Execução”, contém instruções para iniciar-se, comunicar-se com e acompanhar-se conteúdos de aprendizado em ambientes baseados na *web*. Esse livro é resultante das funcionalidades de tempo de

execução definidas pela AICC em seu "Instruções para Inter-Operabilidade CM1001". A ADL colaborou com membros e participantes da AICC com os objetivos de desenvolver-se uma especificação comum de execução e de API e de adotar-se os elementos de dados baseados na *web* da AICC.

A figura abaixo ilustra o padrão SCORM como uma coleção de especificações.

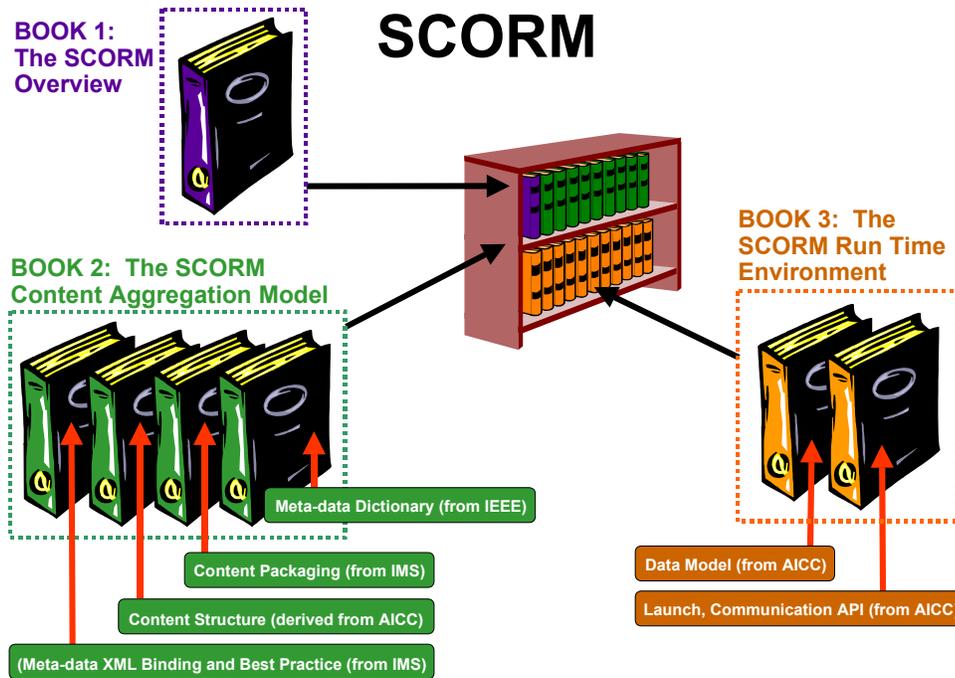


Figura 3.2 - O SCORM como uma coleção de especificações [14]

3.4 – Grafo de Referências das Especificações

A figura 3.3 abaixo ilustra o grafo de referências entre as especificações do IEEE, IMS, AICC, SCORM, ARIADNE e Dublin Core, entendidos como os principais padrões usados/estudados hoje em dia. Na figura, a seta tracejada deve ser interpretada como “referencia” ou “usa”.

Podemos perceber, pelo texto anterior e pela figura 3.3, que os padrões IEEE-LOM e IMS têm um papel importante no contexto, referenciando ou ao outro e referenciando os demais. Esses dois e o SCORM (nessa ordem) definem uma espécie de “espinha dorsal”, seguindo um esquema *construtivo*, no sentido de que o padrão representado mais acima na figura “embrulha” o mais abaixo.

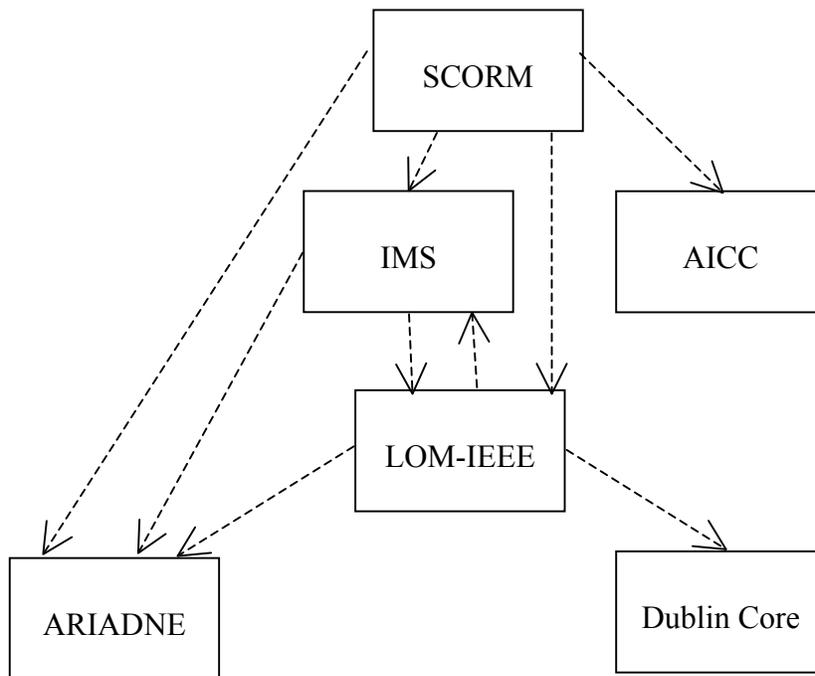


Figura 3.3 - Grafo de referências entre as especificações do IEEE, IMS, AICC, SCORM, ARIADNE e Dublin Core. A seta tracejada deve ser interpretada como “referencia” ou “usa”.

4 - Associações Entre LOs

4.1 – Tipos de Associações

Seguindo outros grupos de pesquisa e o que especifica o LOM, entendemos que LOs podem ser compostos por outros conteúdos instrucionais de diversos níveis de agregação (atômico ou outros LOs), formando um novo conteúdo em um dos níveis (atômico, lição, curso e programa; vide item 1.8 do LOM, tabela 3.1). As associações ou relacionamentos eventualmente existentes entre LOs (por exemplo, as especificadas nos itens 1.7 e 7 do LOM) em um determinado nível devem ser mantidas quando os LOs são agregados formando LOs de nível acima.

Classificamos as possíveis associações que possam existir entre LOs em três tipos: associações estruturais, associações semânticas e associações de tempo de execução. A figura 4.1 ilustra nossa idéia de que um determinado LO pode se relacionar com outro(s) LO(s) através de um conjunto de relações (classe Relacionamento) e que cada instância dessas relações pode ser de um dos três tipos e possui atributos (dentre eles o sub-tipo da associação, como veremos adiante) e, possivelmente, operações.

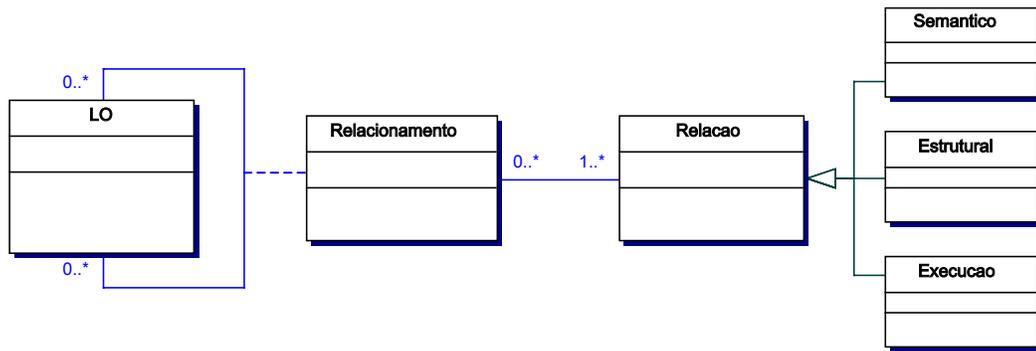


Figura 4.1 – Tipos de relacionamentos entre LOs.

Os itens 1.7 e 7 do LOM (estrutura e relações, respectivamente, vide tabela 3.1) especificam as possíveis associações que podem existir entre LOs. Apresentaremos, a seguir, cada um desses tipos com mais detalhes.

4.2 – Associações Estruturais

Desse grupo de associações fazem parte as que estabelecem (obviamente) a estrutura de LOs nos diversos níveis de agregação possíveis. As relações estruturais devem ser conhecidas necessariamente antes da execução de um conteúdo (e, tipicamente, já o são em tempo de construção do conteúdo), pois definem a seqüência em que os objetos são executados. Segundo o IEEE LOM V1.0 elas podem estabelecer uma estrutura de LOs de um dos seguintes tipos:

- **Atômica:** na realidade significa a ausência total de associação, que indica que o LO é indivisível;
- **Coleção:** para um conjunto de objetos sem nenhum relacionamento especificado entre eles. Essa estrutura sugere que os conteúdos de aprendizado podem ser executados segundo uma ordem qualquer;
- **Hierárquica:** um conjunto de objetos que possuem relacionamentos que podem ser representados como uma árvore. Essa estrutura sugere que os componentes devem ser executados segundo uma ordem preestabelecida como, por exemplo, *bottom-up*, *left-right*;
- **Linear:** um conjunto de objetos completamente ordenados como, por exemplo, conectados através de relacionamentos “anterior” e “posterior”;
- **Em rede:** um conjunto de objetos com relacionamentos, porém que não formam uma estrutura puramente hierárquica ou linear, formando, com isso uma estrutura híbrida.

As relações estruturais podem ser representadas graficamente (e genericamente) através de grafos, onde os nós representam as execuções dos conteúdos e onde as arestas definem as possíveis transições, indicando o fluxo de execução. Em outras palavras, as relações estruturais definem a estrutura de um programa de aprendizado.

4.3 - Associações Semânticas

Vimos que o IEEE-LOM enumera, no item 7.1, possíveis relações entre LOs, que são baseadas nas categorias definidas no padrão Dublin Core. As relações lá enumeradas são classificadas como relações “todo/parte”, “de versão”, “de transformação de formato”, “de referência”, “de criação” e “de dependência”, cujas definições encontram-se abaixo [15]:

- **Relações todo/parte:** são aquelas onde um LO é uma parte física ou lógica de outro. Os possíveis valores são "*ispartof*" e "*haspart*".
- **Relações de versão:** são aquelas onde um LO é um estado histórico ou edição de um outro LO elaborado pelo mesmo criador. Os possíveis valores são "*isversionof*" e "*hasversion*".
- **Relações de transformação de formato:** são aquelas onde um LO foi derivado de outro através de uma reprodução ou tecnologia de reforma que não é fundamentalmente uma interpretação mas sim que objetiva ser uma representação. Os possíveis valores são "*isformatof*" e "*hasformat*".
- **Relações de referência:** são aquelas onde o autor de um LO cita, reconhece, discute ou simplesmente refere-se a um outro LO. Os valores correspondentes são "*references*" e "*isreferencedby*".
- **Relações de criação:** são aquelas onde um LO é uma execução, produção, derivação, tradução, adaptação ou interpretação de outro LO. Os possíveis valores são "*isbasedon*" e "*isbasisfor*".
- **Relações de dependência:** são aquelas onde um LO requer um outro LO para seu funcionamento, entrega ou para compor seu conteúdo e não pode ser usado sem que o outro esteja presente. Os valores correspondentes são "*requires*" e "*isrequiredby*".

Segundo o padrão Dublin Core, comunidades especializadas poderão necessitar estender as relações de referência para expressarem tipos específicos de referências ou de imposições de autoria. Em trabalho futuro pretendemos estudar a suficiência das relações todo-parte e de dependência para definirmos pré e pós requisitos de LOs. Como já mencionamos, isso permitirá que processos automatizados sugiram seqüências possíveis de LOs na montagem de um conteúdo.

As relações acima podem ser previamente estabelecidas em tempo de construção do conteúdo sendo, nesse caso, informadas durante o registro do LO na base de LOs. Podemos, também, imaginar um ambiente onde essas relações possam ser “mineradas” por um mecanismo qualquer de busca com características de inteligência.

As relações semânticas entre objetos permitem, por exemplo, que conjuntos desses objetos sejam também relacionados como possíveis integrantes de um conteúdo quando um deles é escolhido através de um mecanismo de busca. Exemplificamos: suponhamos que nosso mecanismo de busca tenha relacionado um objeto (através de pesquisa por palavra-chave, por exemplo) que consiste de outros LOs que são componentes necessários do que foi selecionado. Obviamente, os componentes necessários serão também relacionados como resultado da pesquisa. Suponhamos, ainda, que no registro de qualquer um dos LOs relacionados conste que haja objetos aos quais estes façam referência. Nesse caso, os objetos referenciados *poderão* ser também relacionados, a título de fonte de consulta adicional.

A tabela 4.1 abaixo apresenta essas relações, com as possíveis alternativas que se configuram nas fases de modelagem e de execução de um conteúdo, frutos da seleção de um LO. Imaginamos uma situação em que o objeto de aprendizado identificado como LO1 seja selecionado pelo mecanismo de busca, e desse LO conste um elemento de relação com um dos valores "*ispartof*", "*haspart*", "*isversionof*", "*hasversion*", "*isformatof*", "*hasformat*", "*references*", "*isreferencedby*", "*isbasedon*", "*isbasisfor*", "*requires*" ou "*isrequiredby*", referenciando um outro objeto LO2. As perguntas que procuramos responder são: a escolha desse objeto abrirá caminhos de execução alternativos? Abrirá caminhos obrigatórios em seqüência ou em paralelo? Não terá nenhuma consequência além da necessidade de execução do LO? ...

Relação	Consequência
LO1 <i>ispartof</i> LO2	Nenhuma
LO1 <i>haspart</i> LO2	LO2 deverá ser também selecionado
LO1 <i>isversionof</i> LO2	LO2 poderá ser também selecionado como uma alternativa
LO1 <i>hasversion</i> LO2	LO2 poderá ser também selecionado como uma alternativa
LO1 <i>isformatof</i> LO2	LO2 poderá ser também selecionado como uma alternativa
LO1 <i>hasformat</i> LO2	LO2 poderá ser também selecionado como uma alternativa
LO1 <i>references</i> LO2	LO2 poderá ser também selecionado como uma seção que poderá ser consultada
LO1 <i>isreferencedby</i> LO2	LO2 poderá ser também selecionado como uma seção que poderá ser consultada
LO1 <i>isbasedon</i> LO2	LO2 poderá ser também selecionado como uma seção que poderá ser consultada
LO1 <i>isbasisfor</i> LO2	LO2 poderá ser também selecionado como uma seção que poderá ser consultada
LO1 <i>requires</i> LO2	LO2 deverá ser também selecionado
LO1 <i>isrequiredby</i> LO2	Nenhuma

Tabela 4.1 – Consequências da inclusão de um LO que possui relação com outro LO

Assim que LO1 e LO2 (este somente se for necessário ou se for uma alternativa) forem incorporados ao conteúdo, uma associação de obrigatoriedade ou de alternativa entre os dois deverá ser, também, instanciada.

4.4 – Associações de Tempo de Execução

As relações de tempo de execução podem ser definidas/estabelecidas e/ou ser alteradas em tempo de execução do conteúdo.

Examinamos, nos itens 4.2 e 4.3, alguns detalhes das relações entre LOs definidas no padrão IEEE-LOM, estabelecidas durante a criação/registo do LO ou, dinamicamente, durante a estruturação dos LOs para formarem um novo conteúdo. Do ponto de vista de quem executa o conteúdo, essas relações são consideradas estáticas, já que permanecem inalteradas durante a execução do conteúdo. Seria interessante que considerássemos um modelo mais flexível, onde parte das relações possa ser estabelecida em tempo de execução, como resultado de condições que se evidenciam em um determinado instante ou por escolha do aluno diante de opções que a ele são

apresentadas. Isso é o que examinaremos no próximo item, investigando duas relações, uma delas que depende do contexto de execução e outra que depende da vontade do executor (da escolha) no momento da execução: são as relações condicionais e as mutuamente excludentes.

4.4.1 – Relações Condicionais

Essas relações definem condições para que ações sejam executadas [16]. Uma relação condicional pode ser entendida, também, como uma condição associada a um conjunto de componentes, e ações que serão aplicadas a um (outro) conjunto de componentes quando a condição é satisfeita [17] ou, ainda, é uma relação lógica entre proposições p e q na forma “se p então q ” onde, se p é verdadeiro então q não poderá ser falso. Por exemplo,

Após o término da execução da seção (LO) “A”, se o executor teve aproveitamento igual ou superior a 80%, então iniciar a execução da seção “B”.

ou

Após o término da execução da seção “A”, se o executor teve aproveitamento igual ou superior a 80%, então ele está apto a iniciar a execução da seção “B”.

são relações condicionais. Tipicamente essas relações são identificadas pelas palavras “a menos que”, “se”, “enquanto” e “conquanto”.

Essas relações podem ser representadas graficamente através de sentenças lógicas que *guardam* as flechas (arestas do grafo orientado que define a lição) que estabelecem a relação de ordem de execução entre dois LOs, definindo, com isso, condições para que LOs sejam executados. A figura 4.2 ilustra.

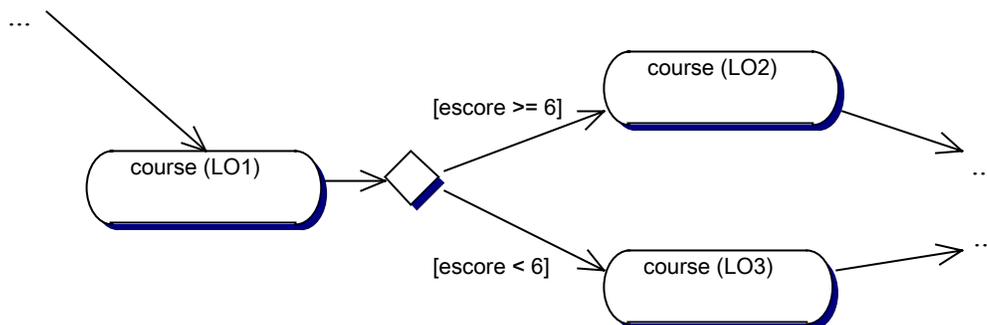


Figura 4.2 – Possível representação gráfica de relações condicionais.

As sentenças lógicas são avaliadas *em tempo de execução* do conteúdo, com base nos fatos/estados nelas expressos, permitindo que o *coordenador de execução*¹⁵

¹⁵ A definição de uma arquitetura adequada à execução a distância de conteúdos instrucionais será tratada em trabalho futuro.

escalone adequadamente os LOs. Na figura 4.2, os rótulos das atividades significam as operações de execução dos conteúdos especificados entre parênteses. Cada LO deve colocar a respectiva operação de execução à disposição do executor.

4.4.2 – Relações de Exclusão Mútua

São relações que representam alternativas excludentes de percurso através do grafo de execução. São definidas em tempo de modelagem e não estão associadas a fatos ou estados de execução, ou seja, a escolha do caminho depende, unicamente, da vontade do executor.

Podemos representar a exclusão mútua de alternativas através de uma linha tracejada que corta as rotas alternativas, rotuladas pela expressão de restrição “{XOR}”, no estilo da UML [18]. A figura 4.3 ilustra.

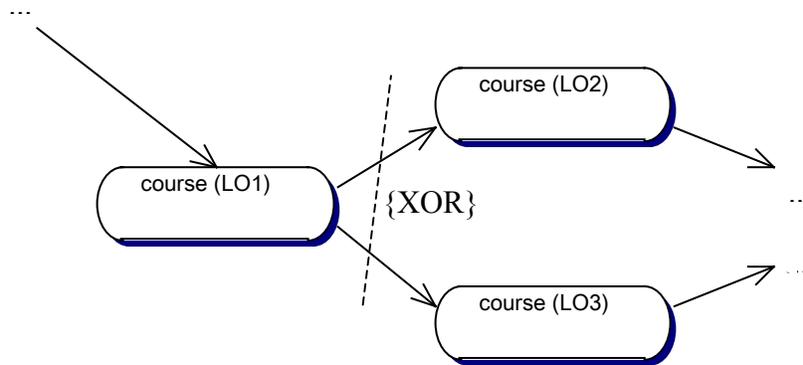


Figura 4.3 – Possível representação gráfica de relações mutuamente excludentes.

Nesse caso o aluno poderá optar por executar o conteúdo do LO2 *ou* do LO3, mas nunca dos dois. Uma outra forma de representarmos o trecho do diagrama de atividades da figura 4.3, mais próxima do modelo de execução, está apresentado na figura 4.4.

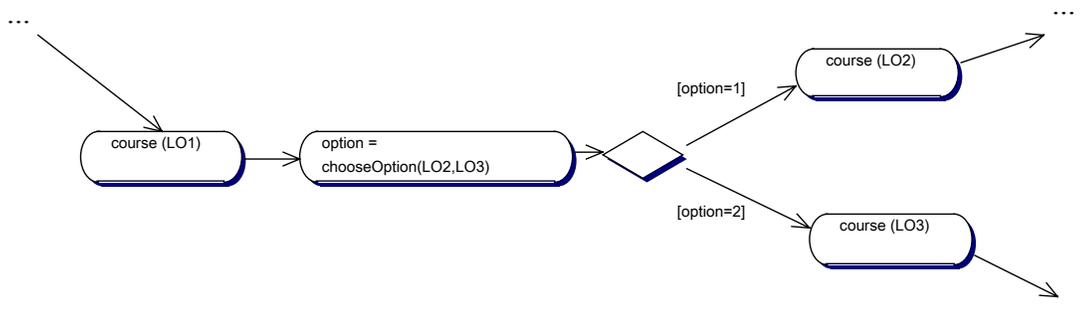


Figura 4.4 – Forma alternativa de apresentação gráfica de relações mutuamente excludentes.

As relações desses dois grupos transformam-se, portanto, em relações estruturais quando o conteúdo é executado.

5 – Conceitos, Modelo de Dados e Arquitetura de BDs de LOs Adotados Pelo TecBD

5.1 – Introdução

Ao longo do ano de 2002 procedemos a uma série de reuniões para apresentação e discussão dos conceitos e padrões mais amplamente aceitos pelos grupos da academia e indústria ligados à pesquisa e produção de conteúdo instrucional para EAD. Esses aspectos encontram-se descritos nos capítulos anteriores do presente trabalho. Nessas reuniões, o principal resultado esperado foi estabelecermos conceitos, terminologia e modelos de uso comum nas pesquisas futuras do TecBD. O que foi aceito pelo grupo é apresentado nas seções a seguir.

5.2 – Conceitos e Modelo de Classes de LOs

A figura 2.4 ilustra a sugestão dada pela Cisco Systems para níveis de agregação em conteúdos de cursos. Baseados na terminologia já usada pelo grupo, permitimo-nos estabelecer relações entre esta e a sugestão dada pela Cisco:

Terminologia Sugerida Pela Cisco	Terminologia já em Uso Pelo TecBD
Currículo	Programa
Unidade	Curso
Módulo	Módulo
Lição	Aula
Seção	Tópico

Tabela 5.1 – Correspondência entre a terminologia sugerida pela Cisco e a usada pelo grupo do TecBD.

A questão se o grupo adotará ou não os quatro níveis de agregação (granularidade funcional de LOs) definidos pelo LOM (item 1.8), ou seja, de 1 a 4, podendo corresponder a conteúdo bruto, lição, curso e programa, respectivamente, ficou para ser discutida em reuniões e trabalhos futuros visando a definição de metodologia(s) de desenvolvimento de conteúdo modular.

Com relação aos componentes básicos dos LOs, em seus níveis elementares, entendemos que:

- Um *item de conteúdo* (CI) é um objeto composto por um conjunto de dados armazenados em uma unidade física (e.g. um arquivo .PPT), podendo ter ou não um objeto instrucional bem definido. Seus atributos e operações típicos são:
 - Atributos:
 - sizeMb : float;
 - type : string {"PPT", "DOC", ...};
 - content : blob;
 - Operações:
 - course() : void;

- `getHeaderInfo()`¹⁶ : string;
- Um *item de exercício* ou *de prática* (EI) é um objeto que apresenta os quesitos de exercício para um LO elementar. É composto de uma unidade física (e.g. um arquivo .DOC). Seus atributos e operações típicos são:
 - Atributos:
 - `sizeMb`:float;
 - `type`:string {"PPT", "DOC", ...};
 - `content`:blob;
 - Operações:
 - `course()`:void, que deve permitir a visualização, escolha e aplicação do(s) quesito(s) de exercício na tecnologia escolhida;
 - `getHeaderInfo()`:string;
- Um *item de avaliação* (AI) é um objeto que produz o resultado da aplicação do(s) quesito(s) de avaliação segundo uma métrica definida. Seus atributos e operações típicos são:
 - Atributos:
 - `sizeMb`:float;
 - `type`:string {"EXE", "CLASS", ...};
 - `content`:blob;
 - Operações:
 - `course()` : void, que deve permitir a visualização, escolha e aplicação do(s) quesito(s) de avaliação na tecnologia escolhida;
 - `getHeaderInfo()`:string;
- Um *Objeto Atômico de Aprendizado* (ALO) é composto de *um* ou mais CIs, zero ou mais EIs e zero ou mais AIs. Corresponde a uma seção ou tópico.

Como dissemos, os componentes que apresentamos acima são os componentes básicos e correspondem ao nível atômico, ou menor nível de agregação, dos conteúdos instrucionais. Esses elementos podem se agregar recursivamente formando conteúdos em níveis mais altos. No item 1.7 do LOM (vide tabela 3.1 e item 4.2 – Associações Estruturais desse trabalho) estão relacionadas as possibilidades segundo as quais objetos de aprendizado podem se relacionar para formarem a estrutura de um conteúdo. A estrutura deve sugerir se há ou não uma ordem de execução dos componentes. Analisando as formas de estruturação de conteúdo apresentadas no item 4.2 verificamos que qualquer ordem de execução lá sugerida se enquadra em duas únicas categorias; quando não há ordem de execução específica e quando há uma ordem de execução. Dessa forma, definimos, ainda:

- Um Objeto de Aprendizado Seqüencial (SLO) é uma seqüência ordenada de um número arbitrário de ALOs, SLOs e NSLOs.
- Um Objeto de Aprendizado Não-Seqüencial (NSLO) é um conjunto (não ordenado) de um número arbitrário de ALOs, SLOs e NSLOs.
- ALOs, SLOs e NSLOs são descritos por descritores LOM.
- SLOs e NSLOs podem ser entendidos/vistos como entidades com significância instrucional (que possuem um objetivo instrucional bem definido).

¹⁶ A operação `getHeaderInfo()` executará as operações de exibição dos metadados eventualmente existentes nos conteúdos.

Chamamos a atenção para o detalhe de que um determinado conteúdo poderá conter diversas porções, cada uma com uma ou outra característica. Poderemos, ainda, adicionar uma restrição para o caso de um conteúdo onde não haja uma ordem definida de execução de seus componentes, mas que estes não possam ser executados em paralelo.

Essas possibilidades estão ilustradas na figura 5.1 abaixo

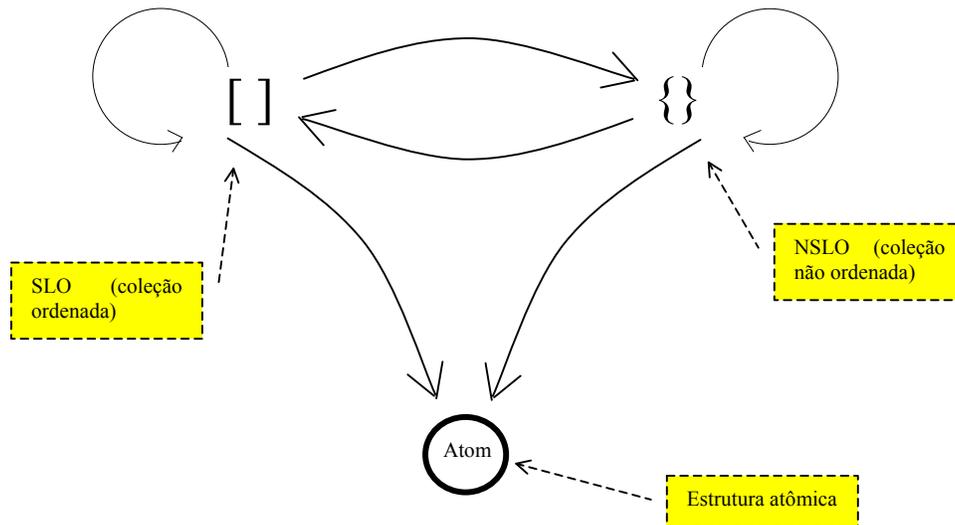


Figura 5.1 - “Aparelho Reprodutor de LOs”. As setas contínuas indicam “contém zero ou mais”.

Acreditamos que essa seja uma forma conveniente de tratarmos uniformemente granularidades diferentes de objetos de aprendizado, flexibilizando, *au goût du client*, o conceito de LO.

O modelo de classes de LOs decorrente é apresentado na figura 5.2.

Interessa-nos, também, a definição de uma arquitetura para um ambiente de armazenamento e recuperação de LOs. Descreveremos adiante a nossa proposta de um modelo de referência de arquitetura baseado no modelo ANSI/SPARC para SGBDs, igualmente composto de três níveis de abstração. Antes de apresentarmos nossa proposta, faremos uma breve descrição do modelo de referência ANSI/SPARC.

5.3 - O Modelo de Referência ANSI/SPARC

Um modelo de referência é um modelo idealizado de arquitetura, com o objetivo de ser o ponto de partida para a definição de uma *arquitetura padronizada* de sistemas dentro de um determinado escopo. É desenvolvido como um precursor para qualquer atividade de padronização [19]. Pode ser entendido como uma estrutura conceitual cujo propósito é dividir o trabalho de padronização em fragmentos gerenciáveis, e para mostrar, em um nível geral, como esses fragmentos estão relacionados uns com os outros.

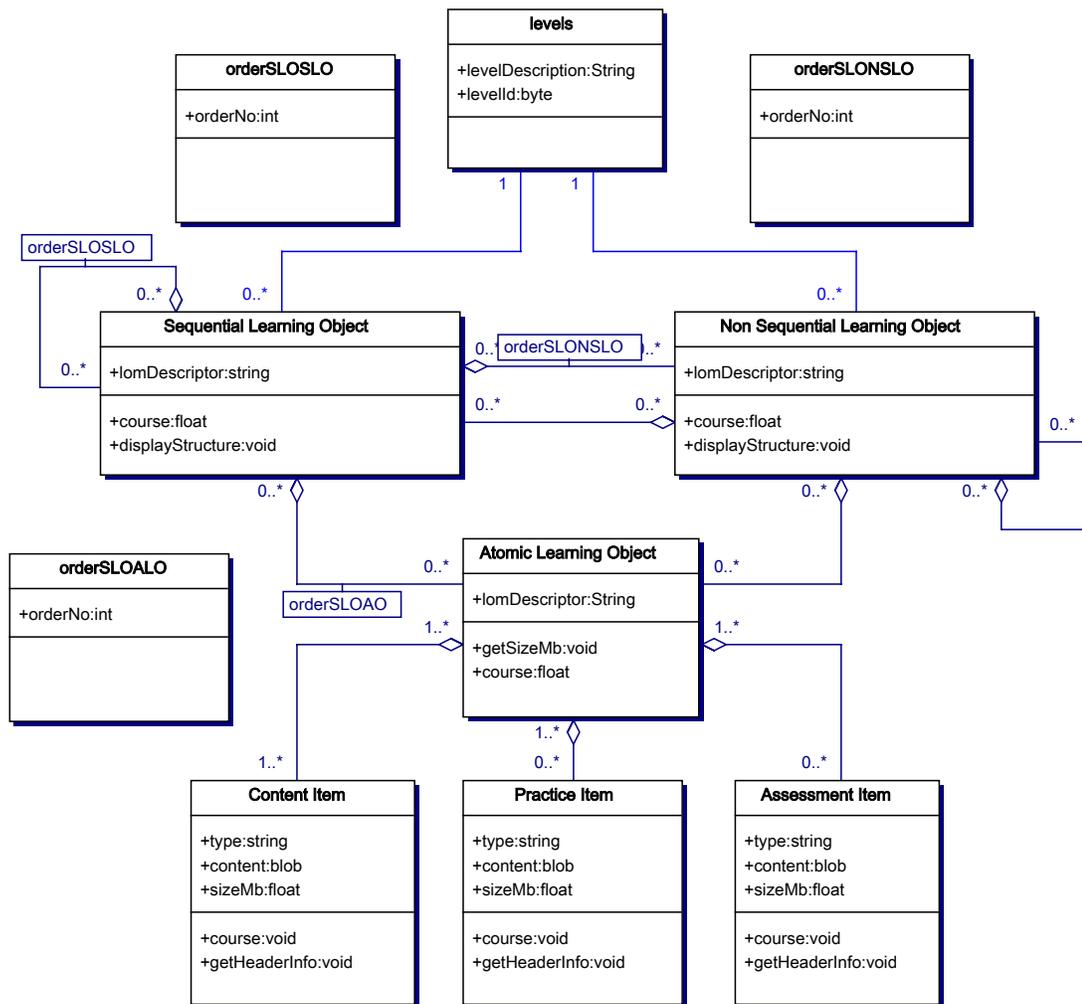


Figura 5.2 – Modelo de classes de LOs.

Um modelo de referência e, portanto, uma arquitetura no qual se referencia, podem ser baseados (1) em componentes, onde são definidos os componentes juntamente com seus inter-relacionamentos, (2) em funções, onde são identificados os atores e os casos de uso a cada um deles associados e (3) em dados, onde estes e suas estruturas são especificados, juntamente com a arquitetura das unidades funcionais que os utilizam. Na medida em que os dados são os elementos primordialmente enfocados quando se estuda/projeta sistemas de gerência de bancos de dados, a abordagem adotada na arquitetura ANSI/SPARC é a baseada em dados.

A arquitetura ANSI/SPARC reconhece três visões de dados: a *visão interna*, que é a visão do sistema ou da máquina, a *visão conceitual* (também chamada de lógica ou visão do empreendimento) e a *visão externa*, ou visão do usuário. Essas visões definem camadas distintas de abstração, onde a visão externa é a de mais alto nível, seguida da visão conceitual, e onde a visão interna, que enfoca mais os detalhes da tecnologia de armazenamento, ocupa a camada de mais baixo nível. A visão externa, por representar diferentes visões de usuários, é representada na figura 5.3 abaixo por múltiplas instâncias.

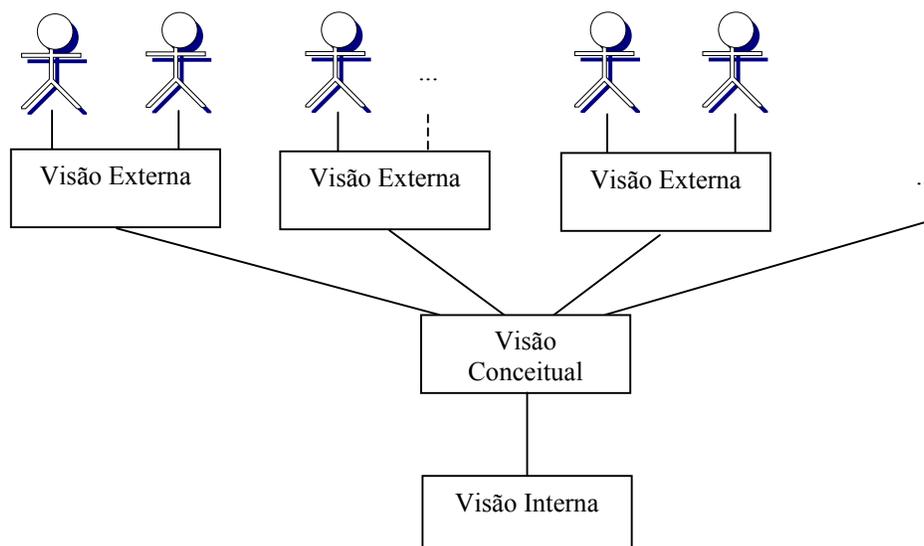


Figura 5.3 - A arquitetura ANSI/SPARC

À cada visão e instância de visão externa corresponde um esquema (interno, conceitual ou externo) que a define.

A visão interna diz respeito à definição física e à organização dos dados, onde são tratadas as questões de localização dos dados em diferentes dispositivos de armazenamento e de mecanismos de acesso aos mesmos. Nesse nível lidamos, portanto, com

- estruturas complexas de armazenamento;
- primitivas de acesso a disco;
- controle de memória principal;
- ênfase: otimização.

Na visão conceitual ou lógica os detalhes da visão interna não são considerados tendo-se, com isso, uma visão abstrata do banco de dados. Nesse nível lida-se com os dados e seus relacionamentos, abordando todos os aspectos que dizem respeito ao negócio (abordagem conceitual completa). Nesse nível lidamos, portanto, com

- descrição dos dados, suas relações e regras de integridade;
- conjunto simplificado de estruturas;
- conjunto de operadores alto nível;
- ênfase: descrição lógica dos dados.

A visão externa opera mapeamentos da visão conceitual para as possíveis diversas visões dos atores. Esses mapeamentos correspondem ao que os atores apreendem do BD, ou seja, não revelam os detalhes dos níveis abaixo. Nesse nível lidamos, portanto, com

- visões individualizadas sobre nível lógico;
- dependente do tipo de usuário;
- ênfase: segurança, simplificação.

Os esquemas das diversas visões são definidos através de linguagens próprias cujas sintaxes podem ser bem parecidas.

O componente que permite os mapeamentos entre as diversas visões é o dicionário/diretório de dados, que é um banco de metadados, armazenando as definições dos dados e os esquemas das diversas visões.

Essas três visões sugerem a necessidade de, pelo menos, três papéis de usuários de um SGBD (que podem corresponder a um ou mais indivíduos) e para os quais se deve dispor de conjuntos adequados/específicos de ferramentas:

1. O administrador do banco de dados, que é responsável pela manutenção do esquema interno;
2. O administrador do empreendimento, que mantém o esquema conceitual e que conhece os dados do negócio, suas restrições e as relações entre eles e,
3. O administrador de aplicativos, responsável pela manutenção dos esquemas externos.

5.4 – Um Modelo de Referência para SGBDs de LOs

O modelo de referência ANSI/SPARC sugere-nos uma forma interessante de organizarmos conceitualmente os depósitos de objetos de aprendizado e das relações que estes, possivelmente, mantêm entre si.

Os níveis ou visões de um gerenciador de LOs são denominados de “nível de componentes atômicos”, “nível de LOs”, “nível semântico” e “nível externo”.

O nível de componentes atômicos diz respeito à organização das estruturas de armazenamento e mecanismos de acesso aos dados das unidades elementares de conteúdo (*blobs* que compõem os ALOs), e de textos em XML que contém os metadados e as relações entre LOs no padrão IMS. Nesse nível estamos interessados em definir as formas ideais de armazenamento de *blobs* e textos XML, considerando as características do SGBD usado. Em nossa abordagem entendemos que os depósitos de LOs devem poder utilizar, como infra-estrutura, tecnologias de mercado capazes de armazenar, localizar e recuperar objetos complexos (ou que por sobre elas sejam desenvolvidas camadas que implementem essas funcionalidades).

O nível de LOs estabelece a visão de objetos de aprendizado, considerando as relações de agregação entre ALOs (conteúdos no nível atômico) e SLOs/NSLOs e entre esses dois últimos (vide figura 5.1).

O nível de estruturas de conteúdo considera as possíveis associações entre LOs (no nível abaixo), estabelecendo as visões de lição, módulo, unidade e currículo.

Um quarto nível, o nível externo, ainda é colocado sobre esses três níveis, provendo a visão de aplicação-fim, ou seja, a visão que os alunos terão do ambiente ao executar um conteúdo.

Nível	Visão	Usuários	Linguagens	Aplicações
Nível externo	Aplicação	Alunos	LOQL, WfQL	Execução de conteúdo (LMSs), ferramentas de busca de conteúdo, ferramentas de montagem de conteúdo sob demanda.
Semântico	Seção, lição, módulo, unidade e currículo	Desenvolvedores de aplicações gerais.	LOQL	Ferramentas (semi) automáticas de pesquisa e montagem de conteúdo, <i>middlewares</i> (integradores implementados através de <i>web services</i> , por exemplo).
LOs (conteúdo e estrutura)	ALOs, SLOs e NSLOs	Provedores de conteúdo (quem cria, edita, revê, etc., os arquivos de material).		Registradores de LOs, editores de conteúdo.
Componentes atômicos	Dados e seus relacionamentos	Administradores da base	DDL do SGBD usado	<i>wrappers</i>

Tabela 5.2 – O Modelo de Referência de LOs

Da análise do modelo de referência ANSI/SPARC e das funções necessárias relacionadas em [19], adaptando às características funcionais de nossa aplicação, chegamos à relação dos usuários, linguagens e ferramentas utilizadas em casa nível, que apresentamos na tabela 5.2.

5.5 – Bases de LOs Distribuídos

Motivados pelo projeto PGL (*Partnership in Global Learning*), estamos estudando a arquitetura de um ambiente distribuído onde, em linhas gerais, vários parceiros poderão contribuir com a criação de objetos de aprendizado para um *pool* de LOs. Estes LOs serão armazenados nos sítios de seus respectivos criadores que estarão interligados através de uma infra-estrutura de comunicação de dados. Haverá um ou mais catálogos dos LOs publicados (ou um mecanismo de pesquisa equivalente) que os autores de programas de aprendizado poderão consultar usando um mecanismo de consulta ainda a ser definido. Independentemente da localização física dos LOs, deverá ser possível que os autores de programas possam formar conteúdos, pela colocação dos LOs pesquisados no(s) catálogo(s) em seqüências apropriadas. Esse processo poderá ser feito manualmente ou (semi-)automaticamente.

Nesse item estaremos nos referindo ao ambiente que temos em mente para o PGL, procurando classificá-lo segundo o que foi definido em [19] quanto às três perspectivas de análise de modelos arquitetônicos para SGBDs distribuídos.

Pela breve descrição e relação de requisitos que temos do ambiente, podemos verificar que, quanto à autonomia, as bases de LOs comporão um sistema semi-autônomo, necessariamente cada base fazendo parte de uma federação que possuirá, para consulta, uma ou mais cópias (réplicas) dos metadados. Quanto à distribuição, trata-se de um sistema completamente distribuído, na medida em que não vemos a necessidade de qualquer distinção entre papéis de cada um dos nós de produção e

armazenamento de LOs da rede. Quanto à heterogeneidade, pelo fato de estarmos lidando, exclusivamente, com a camada externa do modelo da arquitetura de referência do SGBD, definindo um modelo de dados único e em posição de também definir o protocolo de comunicação e linguagem de consulta, entendemos que nosso ambiente é homogêneo.

6 – O Processo de Montagem do Conteúdo de Um Programa

Nesse capítulo apresentamos as linhas gerais do um ambiente de ensino a distância que estamos idealizando para o PGL. Maiores detalhes desse ambiente serão dados em trabalho futuro, que tratará da arquitetura e das funcionalidades do ambiente.

Concebemos um ambiente de ensino a distância onde existem objetos de aprendizado disponíveis em “cestas” de LOs (outra forma de tratamento que demos à base de LOs, com o intuito de ressaltar que estes não são armazenados segundo uma ordenação em particular) compartimentada por níveis de agregação dos conteúdos (seção, liça, etc.) e integradas de tal forma que são “vistas” como uma só pelos desenvolvedores de aplicações gerais. Esses objetos têm seus metadados visíveis por todos os nós da rede do PGL e são disponíveis para pesquisa, execução e/ou *download* através de um protocolo a ser definido (*web services*, por exemplo).

Vislumbramos duas aproximações para montagem de conteúdo:

1. Fornece-se ao “wizard” de montagem de conteúdo um conjunto de palavras-chave, pré e pós requisitos e outras restrições (i.e. custo, idioma, nível, etc.) para que ele busque os LOs na rede e os monte em seqüência ou
2. Uso de uma linguagem de consulta de objetos de aprendizado (“LOQL”), que permite que definamos um grafo de pesquisa e cujo resultado é um grafo onde os vértices são os LOs achados.

Para a segunda aproximação caberiam estudos a respeito de possíveis sintaxes, certamente baseadas em OQLs já existentes. Recomendamos um esforço nesse sentido em nossa conclusão. Seguem alguns detalhes segundo concebemos como uma primeira aproximação.

- Passo 1. Na primeira etapa é feita uma pesquisa (por um conjunto de valores/faixas de valores de metadados), cujo resultado é um conjunto de LOs que satisfazem ao(s) critério(s) de pesquisa. Nessa fase são também relacionados os LOs que foram apontados, através de relações expressas em seus metadados (relações *haspart/hasversion*, por exemplo), pelos LOs selecionados como também necessários ou equivalentes. São, então, eliminadas as referências em duplicidade.
- Passo 2. A segunda etapa consiste da definição da seqüência dos LOs selecionados através de pré e pós requisitos. Objetos cujos pós requisitos satisfazem aos pré requisitos de outro(s) são colocados, no grafo, de forma a anteceder-los em tempo de execução. Ligações (relações de precedência) entre LOs logicamente vizinhos são, então, estabelecidas.

Passo 3. Ao final desse processo existirão, possivelmente, lacunas entre LOs (onde não há, por exemplo, LOs selecionados que cumpram todos os pré-requisitos de um determinado LO resultando, com isso, em uma falha de seqüência). As falhas deverão ser preenchidas, através de nova busca, com outros LOs a serem retirados da cesta. Se forem encontrados LOs que permitem as ligações, preenchendo as lacunas, os LOs de ligação serão incorporados ao conteúdo nas posições convenientes. Do contrário, os caminhos com as lacunas (e os respectivos LOs) serão retirados do conteúdo. Se não houver caminho possível entre o início e o fim de um conteúdo, este não poderá ser oferecido.

Cabem algumas observações:

- As relações expressas nos metadados podem definir, também, percursos obrigatórios e/ou alternativos pelos LOs durante a execução do conteúdo;
- Um subconjunto dos possíveis percursos sobre esse grafo pode escolhido pelo desenvolvedor ou os vários caminhos possíveis podem ser deixados para escolha do aluno em tempo de execução;
- O material é descrito (registrado) como um LO e, como tal, será armazenado na base de LOs;
- O desenvolvedor, então, publica o material como um a lição, um módulo, uma unidade ou um currículo.

Ilustramos na figura 6.1 adiante um curso hipotético em Computação Gráfica que objetiva formar os alunos em técnicas de Síntese de Imagens.

A figura 6.1 representa, usando diagrama de atividades da UML, o grafo de execução sugerido pelo mecanismo de busca e planejamento, como resultado do seu processamento. O círculo cheio representa o ponto de entrada no curso, o símbolo composto pelos círculos concêntricos representam seu final. O curso consiste dos LOs (seções) de Geometria Computacional, que, por sua vez, usa (obrigatoriamente) os LOs de Geometria Analítica e de Complexidade de Algoritmos (esses dois últimos são definidos através, das relações *ispartof/haspart*, como partes do primeiro). Esse LO é pré-requisito para o LO “Algoritmos de Síntese de Imagens”, que se encontra disponível em duas versões, sendo que apenas uma delas será usada. Cada uma dessas versões possui sua seqüência própria de partes. As opções podem ser feitas pelos alunos, em tempo de execução, ou pelo professor, em tempo de modelagem. Independentemente da opção feita (que na figura é representada pela transição em negrito), o LO de laboratório em síntese de imagens deverá ser executado antes do curso terminar.

Os três passos descritos acima descrevem o processo de forma bastante superficial. Em uma primeira análise, antevemos a necessidade adicional de pesquisarmos mecanismos que permitam a eliminação de LOs em duplicata, fatoração de pré/pós requisitos comuns e o tratamento de referências circulares.

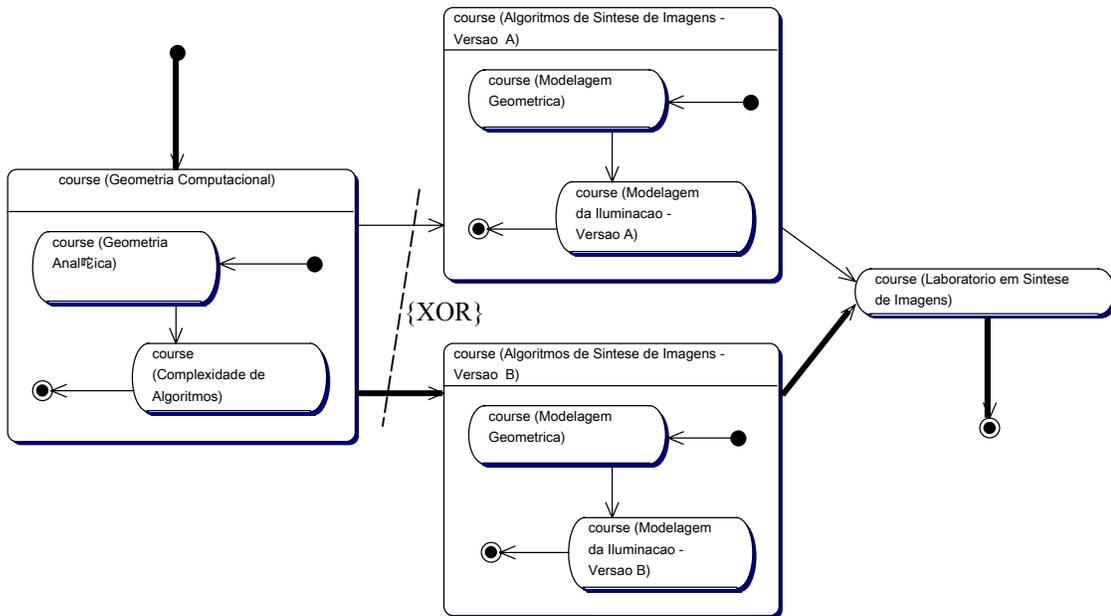


Figura 6.1 – Representação gráfica de um curso de formação em Técnicas de Síntese de Imagens por Computador

Percebemos, também, a necessidade de um mecanismo de especificação e codificação de pré e pós requisitos que, possivelmente, deverão estar organizados segundo taxonomias. O problema aumenta em magnitude quando a cesta de LOs engloba mais de uma área do conhecimento. Evidentemente a necessidade de especificação dos pré e pós requisitos requer que haja, no conjunto metadados, elementos de dados destinados aos seus armazenamentos. Cremos que seja necessário estendermos o padrão LOM com tal finalidade, na medida em que não vimos na especificação, explicitamente, elementos de dados para armazenar essa informação. Estudos nesse sentido merecem ser desenvolvidos.

7 – Conclusões e Trabalhos Futuros

O presente trabalho é um resultado parcial dos esforços do grupo de pesquisadores do TecBD da PUC-Rio na pesquisa de tecnologias de bancos de dados complexos, distribuídos e heterogêneos. A principal motivação do grupo é o projeto PGL (*Partnership in Global Learning*) que objetiva, em linhas bastante gerais, a definição de um ambiente colaborativo de criação e difusão e aplicação a distância de conteúdos de aprendizado.

Dentro desse contexto, interessa-nos pesquisar infra-estruturas de armazenamento de informações de negócio e de controle em processos colaborativos em ambientes distribuídos (*workflows* distribuídos). Interessa-nos, também, pesquisar tecnologias de gerência de objetos complexos associáveis e distribuídos.

A primeira parte dos nossos interesses foi satisfeita através da preparação de uma monografia sobre *workflows* distribuídos que já foi concluída. Nesse trabalho apresentamos aspectos básicos que julgamos relevantes para a definição de tecnologia

de banco de dados para a gerência de objetos de aprendizado reutilizáveis (RLOs). Com esses dois resultados pretendemos fundamentar, sob a ótica de banco de dados, os estudos da tecnologia do ambiente do PGL, que serão desenvolvidos em um próximo trabalho.

No presente trabalho apresentamos os principais conceitos e características de uma forma relativamente nova de preparar programas de aprendizado que são ministrados a distância. Essa forma envolve a criação e descrição de pequenos componentes de conteúdo instrucional para futuro uso em cursos que serão formados por uma seqüência conveniente desses componentes. Uma das grandes vantagens da “quebra” de um conteúdo em componentes é a possibilidade de reutilizarmos esses componentes em outros programas. Apresentamos os principais padrões utilizados para descrever esses componentes, objetivando a facilidade de busca e interoperabilidade. Esses componentes se dispõem, dentro de um determinado contexto, relativamente uns aos outros atendendo a um conjunto de associações que devem/podem ser definidas em tempo de modelagem e execução do curso. Estudamos, também essas relações. Por fim definimos um modelo de dados de LOs e propusemos um modelo de referência para arquiteturas de gerenciadores de objetos de aprendizado baseado no modelo de referência ANSI/SPARC para SGBDs.

Pelo fato de estarmos interessados, nesse trabalho, em uma abordagem mais “em largura” dos diversos assuntos tratados, não nos detivemos em muitos detalhes, respondemos a todos os questionamentos, ou sequer abordamos aspectos (alguns deles bastante importantes) tais como :

- Como implementar eficientemente relações de pré e pós condições entre LOs, possivelmente através de extensões do padrão IMS?
- Como armazenar (e possivelmente migrar para outros nós de execução) dos estados de execução dos LOs, levando-se em consideração, também, que dos estados também fazem parte as decisões tomadas pelos executores?
- Qual/quais seria(m) as linguagens de definição dos esquemas nos diversos níveis conceituais das bases de LOs?
- Seriam as relações *Requires* e *IsRequired* suficientes para implementar os conceitos de pré e pós condições?
- Como seriam as interfaces entre os gerenciadores locais de objetos de forma a facilitar a distribuição?
- ...

Essa (e certamente outras muitas) questões que vêm surgindo ao longo dos trabalhos de pesquisa da equipe do TecBD ficarão por conta de trabalhos futuros.

Bibliografia

- [1] Phillips, V., Selecting an Online Course Authoring System: Corporate Markets, <http://www.geteducated.com/articles/corpauth.htm>
- [2] The British Council, Educação a Distância , 2001, em http://www.britishcouncilpt.org/education/distance_p.htm, acesso em abril/2002.
- [3] Sizilio, G. R. M. A., Técnicas de Modelagem de Workflow Aplicadas à Autoria e Execução de Cursos de Ensino a Distância. Dissertação de Mestrado, Universidade Federal do Rio Grande do Sul, Instituto de Informática, novembro de 1999.
- [4] Jacobsen, P. E-learning Magazine, <http://www.elearningmag.com/elearning/article/articleDetail.jsp?id=5043> , acesso em maio/2002.
- [5] National University of Singapore, Centre for Instructional Technology, Courseware Development/EDtech, <http://courseware.nus.edu.sg/Standards/rlo.asp>, acesso em maio/2002.
- [6] IEEE P1484.12.1/D6.4 Draft Standard for Learning Object Metadata, IEEE, 4 de março de 2002.
- [7] National University of Singapore, Centre for Instructional Technology, <http://ivle.nus.edu.sg/ivle/background.htm>, acesso em maio/02.
- [8] Blackboard site, <http://company.blackboard.com/>, acesso em maio/02.
- [9] WebCT, Inc., <http://www.webct.com/>, acesso em maio/02.
- [10] Barrit, C., Reusable Learning Object Strategy, version 4.0, Cisco Systems, november 2001.
- [11] Cisco Systems, Model of An E-Learning Solution Architecture for the Enterprise, April 2001.
- [12] IMS Global Learning Consortium, Inc., especificação do padrão IMS baixada de <http://www.imsproject.org> em junho/02
- [13] IMS Global Learning Consortium, Inc., IMS Learning Resource Meta-Data XML Binding. Especificação do padrão IMS baixada de <http://www.imsproject.org> em junho/02
- [14] Advanced Distributed Learning Sharable Content Object Reference Model Version 1.2 - The SCORM Overview. Introdução ao SCORM, baixada de <http://www.adlnet.org> em junho/02.

- [15] Dublin Core Metadata Initiative, Relation Element Working Draft, 1997, em <http://dublincore.org/documents/relation-element/>, consultado em 14/9/2002.
- [16] Béjar, R. M., García, R. V., Diseño e implementación de un sistema semisupervisado de reconocimiento de language natural, Universidad de Murcia.
- [17] Pistori, J., Modelo Formal para Aplicações Multimídia, disponível em <http://www.inf.ufsc.br/~jefepist/mfam/>, acesso em setembro/2002
- [18] OMG Unified Modeling Language Specification, Version 1.4, setembro de 2001.
- [19] Özsü, T., Valduriez, P., Princípios de Sistemas de Bancos de Dados Distribuídos, tradução da 2a. edição americana, 2001, Editora Campus.

Apêndice A – Exemplo de Uso do Padrão IMS

Um exemplo de representação em XML dos metadados de um LO segundo o padrão IMS encontra-se abaixo, apresentado no arquivo *metadav1p2plex.zip*, baixado do *site* da IMS.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v3.5 (http://www.xmlspy.com) by Thor Anderson on July 9,
2001 -->
<!-- use the line below and comment out the XML Schema namespace declarations when
you want to check validity against the DTD -->
<!--<!DOCTYPE lom SYSTEM "imsmd_rootv1p2pl.dtd" -->
<lom xmlns="http://www.imsglobal.org/xsd/imsmd_rootv1p2pl"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmd_rootv1p2pl
http://www.imsglobal.org/xsd/imsmd_rootv1p2pl.xsd">
  <general>
    <title>
      <langstring xml:lang="en">Human Resource Management</langstring>
    </title>
    <catalogentry>
      <catalog>URI</catalog>
      <entry>
        <langstring xml:lang="en">
http://www.algonquincollege.com/distance/certfs.html</langstring>
        </entry>
      </catalogentry>
    </catalogentry>
    <catalog>POOL</catalog>
    <entry>
      <langstring xml:lang="en"> 000SamplePoolRefernece#</langstring>
    </entry>
    </catalogentry>
    <description>
      <langstring xml:lang="en">This course deals with personnel functions,
including concepts, principles and practices; techniques of personnel administration
(roles, staffing, human resources, employee development and incentives for effective
performance); and management's responsibility in personnel
administration.</langstring>
    </description>
  </general>
  <lifecycle>
    <version>
      <langstring>MGT2310</langstring>
    </version>
    <contribute>
      <role>
        <source>
          <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
        <value>
          <langstring xml:lang="x-none">Publisher</langstring>
        </value>
      </role>
      <centity>
        <vcard>
          BEGIN: vCard
          ORG: Algonquin College
          END: vCard
        </vcard>
      </centity>
      <date>
        <datetime>2001-02-01</datetime>
      </date>
    </contribute>
  </lifecycle>
  <metametadata>
    <contribute>
      <role>
        <source>
          <langstring xml:lang="x-none">LOMv1.0</langstring>
        </source>
      </role>
    </contribute>
  </metametadata>
</lom>
```

```

        <value>
          <langstring xml:lang="x-none">Creator</langstring>
        </value>
      </role>
    </centity>
    <vcard>
      BEGIN:vCard
      Fisher;Sue
      END:vCard
    </vcard>
  </centity>
</contribute>
<contribute>
  <role>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Validator</langstring>
    </value>
  </role>
  <centity>
    <vcard>
      BEGIN: vCard
      ORG: POOL Project
      END: vCard
    </vcard>
  </centity>
  <date>
    <datetime>2001-02-01</datetime>
  </date>
</contribute>
<metadatascheme>CanCore 1.0</metadatascheme>
<metadatascheme>IEEELOM:1.0</metadatascheme>
<language>en</language>
</metametadata>
<technical>
  <format>text/html</format>
  <location type="URI">
    http://www.algonquincollege.com/distance/certfs.html
  </location>
  <otherplatformrequirements>
    <langstring xml:lang="en">Group Work is required for computer
conferencing</langstring>
  </otherplatformrequirements>
</technical>
<educational>
  <intendedenduserrole>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Learner</langstring>
    </value>
  </intendedenduserrole>
  <context>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">University Undergraduate</langstring>
    </value>
  </context>
  <typicalagerange>
    <langstring xml:lang="en">18-99</langstring>
  </typicalagerange>
  <language>en</language>
</educational>
<rights>
  <cost>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">yes</langstring>
    </value>
  </cost>

```

```

<copyrightandotherrestrictions>
  <source>
    <langstring xml:lang="x-none">LOMv1.0</langstring>
  </source>
  <value>
    <langstring xml:lang="x-none">yes</langstring>
  </value>
</copyrightandotherrestrictions>
<description>
  <langstring xml:lang="en">Registration $332.85 Can. Contact Carole Smith,
smith@algonquinc.on.ca</langstring>
</description>
</rights>
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Discipline</langstring>
    </value>
  </purpose>
  <taxonpath>
    <source>
      <langstring xml:lang="en">Eric</langstring>
    </source>
    <taxon>
      <entry>
        <langstring xml:lang="en">Human Resources</langstring>
      </entry>
      <taxon>
        <entry>
          <langstring xml:lang="en">Management</langstring>
        </entry>
      </taxon>
    </taxon>
  </taxonpath>
</classification>
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">PedagogicType</langstring>
    </value>
  </purpose>
  <taxonpath>
    <source>
      <langstring xml:lang="x-none">CanCore</langstring>
    </source>
    <taxon>
      <entry>
        <langstring xml:lang="en">Course</langstring>
      </entry>
    </taxon>
  </taxonpath>
</classification>
</lom>

```