

## O Desenvolvimento de Jogos Baseado em OA para E-Learning

Diego Passos Costa<sup>1</sup>, Patrícia Dourado<sup>1</sup>, Alex Pereira<sup>1</sup>, André Santanchè<sup>1</sup>

<sup>1</sup> NUPPEAD – Núcleo de Pesquisa e Projetos em Educação a Distância  
NUPERC – Núcleo de Pesquisa Interdepartamental em Redes de Computadores  
Universidade Salvador – UNIFACS  
Av Cardeal da Silva, 747 – Salvador – BA – Brasil

{diego.costa, patricia.almeida}@si.unifacs.br, alex0002@ig.com.br,  
santanche@unifacs.br

**Abstract.** *The current learning object (LO) standards result from a unification of several complementary standards, often developed in autonomous way. In spite of efforts to develop a unified standard, there are many limitations arising out of the organic way in which these systems were established and connected. In this paper we present two e-learning games we implemented by using learning object standards, which supplied subsidies for reflection about better practices in game development as LOs, and the limitations of current LO standards in this context.*

**Resumo.** *Os atuais padrões de objetos de aprendizagem (OA) resultam de uma unificação de diversos padrões complementares, desenvolvidos muitas vezes de forma autônoma. A despeito dos esforços para desenvolver um padrão unificado, existem diversas limitações decorrentes da forma orgânica como tais sistemas se estabeleceram e coligaram. Neste artigo apresentaremos dois jogos educativos que implementamos sobre os padrões de objetos de aprendizagem, que forneceram subsídios para uma reflexão sobre boas práticas para o desenvolvimento de jogos na forma de OAs e dos limites dos atuais padrões de OA neste contexto.*

### 1. Introdução

Os atuais padrões abertos de objetos de aprendizagem (OA) correspondem a um conjunto de iniciativas que se formaram em torno de uma noção inicial, definida pelo IEEE *Learning Technology Standards Committee* (LTSC). Tal noção se configurou no processo de elaboração do LOM – *Learning Object Metadata* [IEEE 2002] – um padrão de metadados para descrever OAs. A partir dos metadados, novas demandas surgiram e, conseqüentemente, padrões relacionados. A necessidade de empacotamento e distribuição de objetos complexos resultou no IMS CP (*IMS Content Package*) [Smythe e Jackl 2004] e a de seqüenciamento do conteúdo a ser consumido no IMS SS (*IMS Simple Sequencing*) [Norton e Panar 2003]. Em paralelo o *Aviation Industry CBT Committee* (AICC) definiu um padrão para a comunicação de módulos educacionais com ambientes de aprendizagem baseados na Web [McDonald et al. 2004] que, apesar de não estar necessariamente alinhado com os demais padrões, parecia uma abordagem interessante para os OAs.

O *Advanced Distributed Learning* (ADL) propôs em seguida uma perspectiva mais integrada sobre os padrões de OAs, através do SCORM – *Sharable Content Object Reference Model* [ADL 2006]. Ao invés de ser mais um padrão, o SCORM unificou outros esforços, estendendo quando necessário as especificações, de modo que elas pudessem se integrar.

Se por um lado os padrões relacionados a OAs são resultado de um interessante trabalho distribuído e colaborativo, por outro, observamos alguns efeitos colaterais nesta abordagem. Primeiro, que a unificação feita pelo SCORM parte de uma “colagem” de iniciativas algumas vezes independentes. Segundo, que a comunidade de usuários a quem tais padrões são endereçados ainda tem dificuldades em perceber o quadro geral do que exatamente se considera um OA, quais são suas características, possibilidades e limitações.

Tais observações foram especialmente assinaladas na pesquisa relatada neste artigo, que parte de dois projetos de jogos educativos implementados mediante o uso dos padrões de OAs, desenvolvidos em paralelo por nosso grupo de pesquisa. Tais jogos educativos exploraram a integração das diversas tecnologias unificadas pelo SCORM – inclusive na sua integração com um Ambiente Virtual de Aprendizagem (AVA), o Moodle – e servem como base para uma análise dos benefícios e limitações da proposta. O artigo relata nossas experiências com o desenvolvimento do jogo Banca do Quincas, ganhador do concurso da Rede Internacional Virtual de Educação (RIVED 2007) e do Desafio on-line, que foi desenvolvido para alunos do EAD da UNIFACS (Universidade Salvador) e foi implantado no AVA Moodle.

Assim, organizamos as seções aqui apresentadas da seguinte maneira: na Seção 2 descrevemos os trabalhos relacionados; na Seção 3 detalhamos os jogos desenvolvidos conforme os padrões de OA; na Seção 4 apresentamos uma reflexão sobre boas práticas para o desenvolvimento de jogos em OAs, bem como uma análise das vantagens e limitações da atual integração dos padrões de OAs, especialmente no contexto de desenvolvimento de jogos; na Seção 5 expomos as considerações finais.

## **2. Trabalhos Relacionados**

### **2.1. Learning Object Metadata**

Com o intuito de descrever e classificar os OAs, de maneira que se possa projetar ferramentas aptas a encontrá-los e utilizá-los, o LTSC desenvolveu o padrão LOM [IEEE 2002]. O padrão IEEE LOM deriva de trabalhos anteriores desenvolvidos na fundação européia ARIADNE (<http://www.ariadne-eu.org>), e pelo IMS Global Learning Consortium (<http://www.imsglobal.org>) [Nilsson et al. 2006].

No padrão LOM, os OAs são descritos por meio de um conjunto de propriedades, às quais são dados valores que caracterizam o objeto. O padrão define um esquema que relaciona hierarquicamente as propriedades aceitas, o domínio dos valores que elas podem receber, bem como sua cardinalidade. Este esquema, denominado *BaseScheme* (esquema básico), reúne as propriedades consideradas fundamentais para a descrição dos objetos. Ele pode ser estendido para outros esquemas derivados, de acordo com necessidades específicas.

## 2.2. IMS CP

Objetos de Aprendizagem são usualmente armazenados dentro de estruturas de pacote. Um pacote é uma estrutura responsável pela agregação de múltiplos artefatos digitais em um arquivo único, preservando sua organização. Um dos padrões existentes para representação, empacotamento e distribuição dos OAs é o IMS CP (*IMS Content Package*), cuja estrutura será detalhada a seguir.

A estrutura de um pacote do padrão IMS CP é definida conforme a Figura 1.

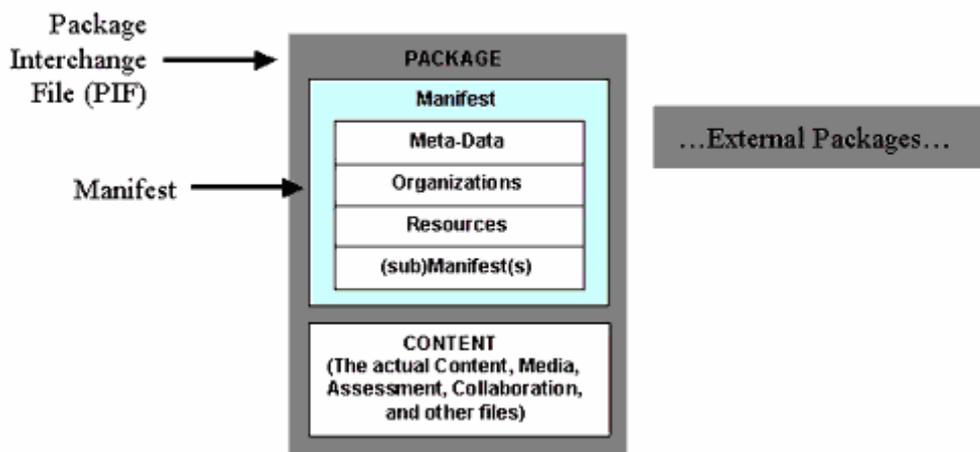


Figura 1. Estrutura de um pacote IMS CP [Smythe e Jackl 2004].

Na figura, a estrutura externa, indicada como Package, consiste em um arquivo compactado em formato ZIP. Dentro do ZIP, está um arquivo especial denominado manifesto (*manifest*) com quatro subdivisões:

- **Meta-Data** – metadados educacionais conforme o padrão LOM que descrevem o objeto como um todo.
- **Organização** – define uma estrutura organizacional hierárquica que funciona como um índice de tópicos e subtópicos associados ao conteúdo educacional.
- **Recursos** – contém referências para os artefatos que estão armazenados no ZIP e mapeiam as dependências entre eles.
- **Submanifestos** – esta seção é opcional, e contém manifestos subordinados, quando há pacotes dentro de pacotes.

Além do manifesto, ficam dentro do ZIP todos os demais artefatos que compõem o objeto complexo.

Para exemplificarmos vamos considerar o objeto de aprendizagem Banca do Quincas, cujo propósito será detalhado mais adiante. Esse jogo é composto por diferentes tipos de arquivo como: páginas HTML, imagens, arquivo para definição de folhas de estilo. Para implementá-lo, utilizamos a ferramenta *Reload Editor* (<http://www.reload.ac.uk/>), cuja finalidade é organizar, anotar e realizar o empacotamento de objetos de aprendizagem, conforme os padrões IMS CP e SCORM.

Seguindo o padrão IMS CP, a Figura 2 (esquerda) ilustra todos os artefatos relacionados ao jogo Banca do Quincas em um único arquivo ZIP, que também empacota o respectivo arquivo manifesto (*imsmanifest*), responsável por descrever os artefatos presentes. A Figura 2 (direita) ilustra o mesmo objeto, sendo editado pelo Reload Editor. Identificamos no seu painel à esquerda do Reload os artefatos que estão

contidos no arquivo ZIP, e à direita, a estrutura interna do arquivo de manifesto, onde podemos observar uma estrutura hierárquica que organiza internamente o OA e o registro dos principais artefatos e as suas dependências. O controle de dependências serve para quando se reusar um determinado artefato, que o mesmo seja tratado de maneira íntegra sem perder suas características.

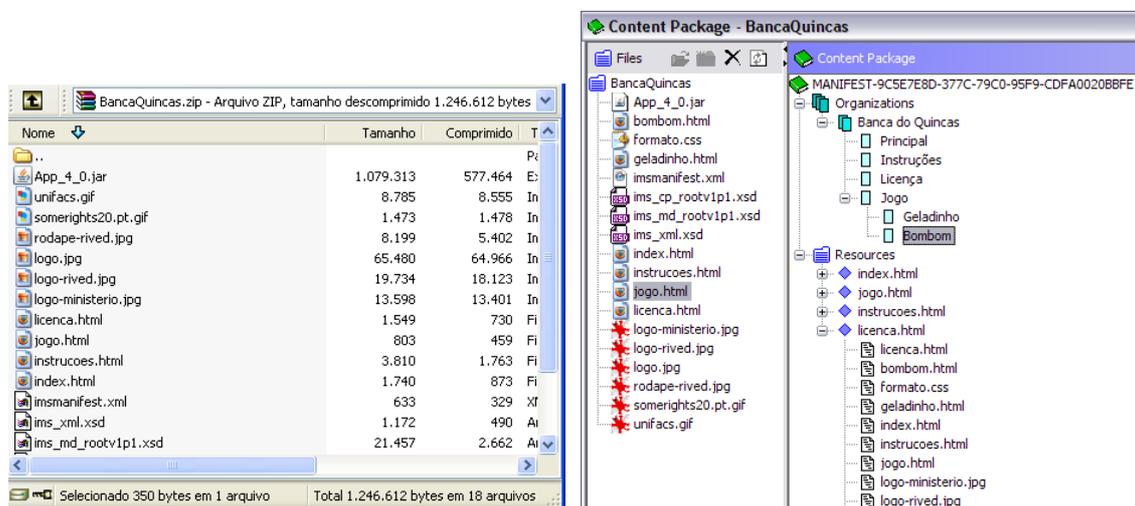


Figura 2. Objeto de Aprendizagem na perspectiva de um software para ZIP e do Reload Editor.

### 2.3. AICC

O *Aviation Industry CBT Committee* – AICC (<http://aicc.org>) definiu um padrão para a comunicação de módulos educacionais com ambientes de aprendizagem baseados na Web [McDonald et al. 2004]. Este padrão define API baseada em ECMAScript (uma generalização do JavaScript). Através desta API um OA pode solicitar serviços e informações do ambiente de aprendizagem e vice-versa.

### 2.4. SCORM

O *Shareable Content Object Reference Model* (SCORM) [ADL 2006] é uma especificação – criada pela ADL (*Advanced Distributed Learning* – <http://www.adlnet.gov>) – com o objetivo de sistematizar o modo como o conteúdo para aprendizagem on-line é criado ou distribuído. A especificação SCORM alcançou maturidade no domínio do *e-learning* e recebe suporte de ambientes virtuais de aprendizagem bastante difundidos, tais como o Moodle ([moodle.org](http://moodle.org)) e o Sakai ([sakaiproject.org](http://sakaiproject.org)). O Padrão faz uso de XML e integra: metadados, com uma extensão e adaptação do IEEE LOM; empacotamento, com uma extensão e adaptação do IMS CP; e comunicação, com uma extensão e adaptação do AICC.

## 3. Estudo de Caso – Jogos em Objetos de Aprendizagem

### 3.1. Banca do Quincas e o Concurso RIVED

O RIVED - Rede Interativa Virtual de Educação (<http://www.rived.mec.gov.br>) é um programa da Secretaria de Educação a Distância – SEED, cujo objetivo é a produção de conteúdos pedagógicos digitais na forma de OAs ([http://www.rived.mec.gov.br/site\\_objeto\\_lis.php](http://www.rived.mec.gov.br/site_objeto_lis.php)). O conceito de objetos de

aprendizagem apresentado pelo RIVED é: “recurso que possa ser reutilizado para dar suporte ao aprendizado”, este, contudo, deve ser digital.

Para intensificar e incentivar o desenvolvimento dos objetos de aprendizagem foi criado o Concurso RIVED de Produção de Objetos de Aprendizagem, ao qual o público alvo envia suas propostas de objetos de aprendizagem para seleção.

Já mencionamos que, como parte das pesquisas realizadas por nosso grupo desenvolvemos um jogo de simulação denominado Banca do Quincas, que foi enviado para o concurso RIVED 2007. O concurso RIVED não exige que sejam adotados os padrões abertos de OA, entretanto, como este projeto tem um escopo mais amplo, além do RIVED, decidimos pautá-lo nestes padrões.

A Banca do Quincas é um jogo de simulação no qual se administra uma banca que vende produtos. O objetivo do jogo é tentar obter o maior lucro possível com as vendas. Podemos obter uma visão inicial do jogo pela Figura 3, em que identificamos as opções de produtos a serem vendidas e as possibilidades de navegação no jogo.



**Figura 3.** Tela inicial do jogo Banca do Quincas.

A lógica e estrutura deste jogo foram baseadas em um jogo clássico do passado, denominado *Lemonade*, que foi um grande sucesso, tendo gerado muitas publicações na área educacional a respeito do assunto [Santanchè 2002]. Este jogo também é uma espécie de simplificação da simulação empresarial, que é uma estratégia educativa muito motivadora e de grande sucesso no âmbito do ensino superior.

Diversamente do *Lemonade*, o produto a ser vendido na Banca do Quincas pode ser configurado para atender a diferentes contextos. O jogo vem com duas pré-configurações para a venda de geladinhos e bombons.

O jogo Banca do Quincas desenvolve-se de forma linear e cíclica ao longo de diversas telas. Quando jogado em equipe, pode ser definido um número limitado de rodadas para o jogo. A Figura 4 ilustra quatro principais etapas do jogo. Ao iniciar uma rodada é apresenta-se ao aluno a situação climática (tempo) corrente (etapa A), pois o resultado das vendas é sensível à variação do tempo, uma vez que afeta a decisão de compra dos consumidores. Três tipos de tempo são possíveis: ensolarado, chuvoso, quente e seco.

Na etapa B o usuário deverá especificar quanto deseja comprar de cada um dos ingredientes que irão compor o produto final, mantendo-se atento ao seu saldo financeiro e ao custo unitário dos ingredientes. Na etapa C, ele terá que definir quantas unidades do produto final deseja produzir, levando-se em consideração o seu estoque de

ingredientes. Após definir o preço unitário, inicia-se o processo de vendas (não ilustrado na figura). Ao final da rodada, o usuário confere os resultados das vendas (etapa D).



Figura 4. Telas ilustrando as principais etapas do jogo Banca do Quincas.

Este objeto foi construído atendendo ao padrão SCORM para objetos de aprendizagem da seguinte maneira: (i) todos os recursos que envolvem o objeto – incluindo planilhas, imagens, páginas e *applet* – foram empacotados no formato IMS CP (adaptação SCORM); (ii) os metadados do objeto seguem o padrão LOM; (iii) para o desenvolvimento da interface entre o objeto e um ambiente de aprendizagem na Web, planeja-se utilizar o padrão AICC, que exigirá a integração entre Java e Javascript. Além disso, o jogo roda na forma de uma *applet* Java, a fim de atender um dos requisitos definidos pelo SCORM de que os objetos devem ser capazes de rodar na Web.

Seguindo a filosofia dos OAs, o jogo Banca do Quincas foi projetado para a realização de reuso sistemático. A base para viabilizar este tipo de reuso é a possibilidade de se customizar no objeto diversos aspectos do funcionamento do jogo, através de um componente Java que funciona como planilha eletrônica – o Jeks (<http://www.etecks.com/jeks/>). A planilha não tem somente a responsabilidade de armazenar os dados básicos de funcionamento, mas toda a lógica do jogo, isto é, valores das variáveis e os cálculos matemáticos e estatísticos foram implementados diretamente na planilha.

A escolha da planilha eletrônica como interface de customização se deu por ser este um aplicativo muito usado hoje em dia, o que a torna uma interface conhecida e de fácil entendimento, facilitando as customizações médias e avançadas. Todas as tecnologias para desenvolvimento de aplicações Web possuem limitações e as *applets* Java não constituem uma exceção. Entretanto, comparada a outras tecnologias largamente usadas na Web – como o Adobe Flash –, a *applet* Java mostrou-se mais adequada ao projeto, permitindo uma clara separação entre os diversos elementos que compõem o objeto de aprendizagem, a saber: animações, imagens, código do objeto, e principalmente, o elemento de customização.

Hoje, o jogo Banca do Quincas faz parte de um projeto maior chamado LOGames, uma iniciativa OpenSource de criar um repositório de desenvolvimento de jogos na forma de OA. Mais detalhes sobre o projeto estão no site: <http://purl.org/net/bancaquincas>.

### 3.2. Desafio On-line

O Desafio On-line é um jogo de computador projetado para funcionar como um OA dentro de um AVA. Tal jogo funciona como uma “cola” entre diversas atividades propostas no AVA. Por ter sido desenvolvido dentro do formato SCORM, o Desafio On-line originalmente foi projetado para rodar em qualquer AVA, entretanto, devido a limitações técnicas, descritas mais adiante, seu escopo está atualmente limitado ao Moodle.

O Desafio On-line tomou como base um jogo existente e de grande sucesso: Carmen Sandiego. Nele o aluno/jogador assume o papel de uma pessoa em busca de um emprego, e posteriormente, de uma ascensão dentro da empresa, até alcançar o cargo de diretor executivo. Como pode ser observado na Figura 5 (tela A), o aluno recebe instruções em formato multimídia (vídeo) que informam qual o desafio da fase corrente. O aluno pode escolher a opção de ver o desafio em formato de texto (tela B na figura). Após receber as instruções do desafio da respectiva fase, o aluno deve cumprir as tarefas que lhe foram designadas para avançar de estágio, e assim subir mais um cargo na organização. Antes de submeter qualquer tarefa, o aluno tem instruções de como fazê-las (tela C na figura). Em seguida, ele responde um questionário (tela D da figura) que o habilita a submeter a tarefa da fase (tela E da figura).

Este jogo foi projetado para interagir intensamente com o AVA, uma vez que algumas atividades – tal como a submissão de tarefas –, são feitas pelo ambiente que, além disso, armazena o score do aluno no jogo. Um OA tem a condição de entregar ao AVA para armazenamento informações importantes sobre o usuário, como por exemplo, seu score e desempenho em determinada tarefa. Com isso, é possível acompanhar mais de perto o desenvolvimento do aluno e identificar possíveis dificuldades. Por outro lado, ainda existem restrições de interação objeto/ambiente nos atuais padrões. No Desafio On-line, por exemplo, o objeto não conseguia obter do AVA algumas informações de desempenho do aluno externas ao objeto. Para contornar este problema, foi necessária a criação de rotinas específicas no Moodle, fora do padrão AICC, comprometendo a reusabilidade do objeto em outros tipos de AVA.

O objeto Desafio On-line foi produto de uma composição de vários subprodutos, alguns deles também OAs. Neste sentido, a interface Web foi utilizada como plataforma comum onde tudo se integra. Na tela A é possível ver um dos vídeos em Flash inseridos no jogo. Na tela C é usado um módulo em Javascript para fazer uma animação no terminal do computador. A tela D mostra um sub-OA, que foi produzido por uma ferramenta chamada eLearning XHTML editor – eXe (<http://exelearning.org>). O eXe é uma ferramenta de autoria *open source*, que é capaz de exportar seu produto para o formato SCORM. Dentre outras coisas, o eXe pode ser usado para a construção de questionários, que foram integrados ao objeto Desafio On-line pela capacidade do padrão SCORM de empacotar OAs (sub-OAs) dentro de um OA.

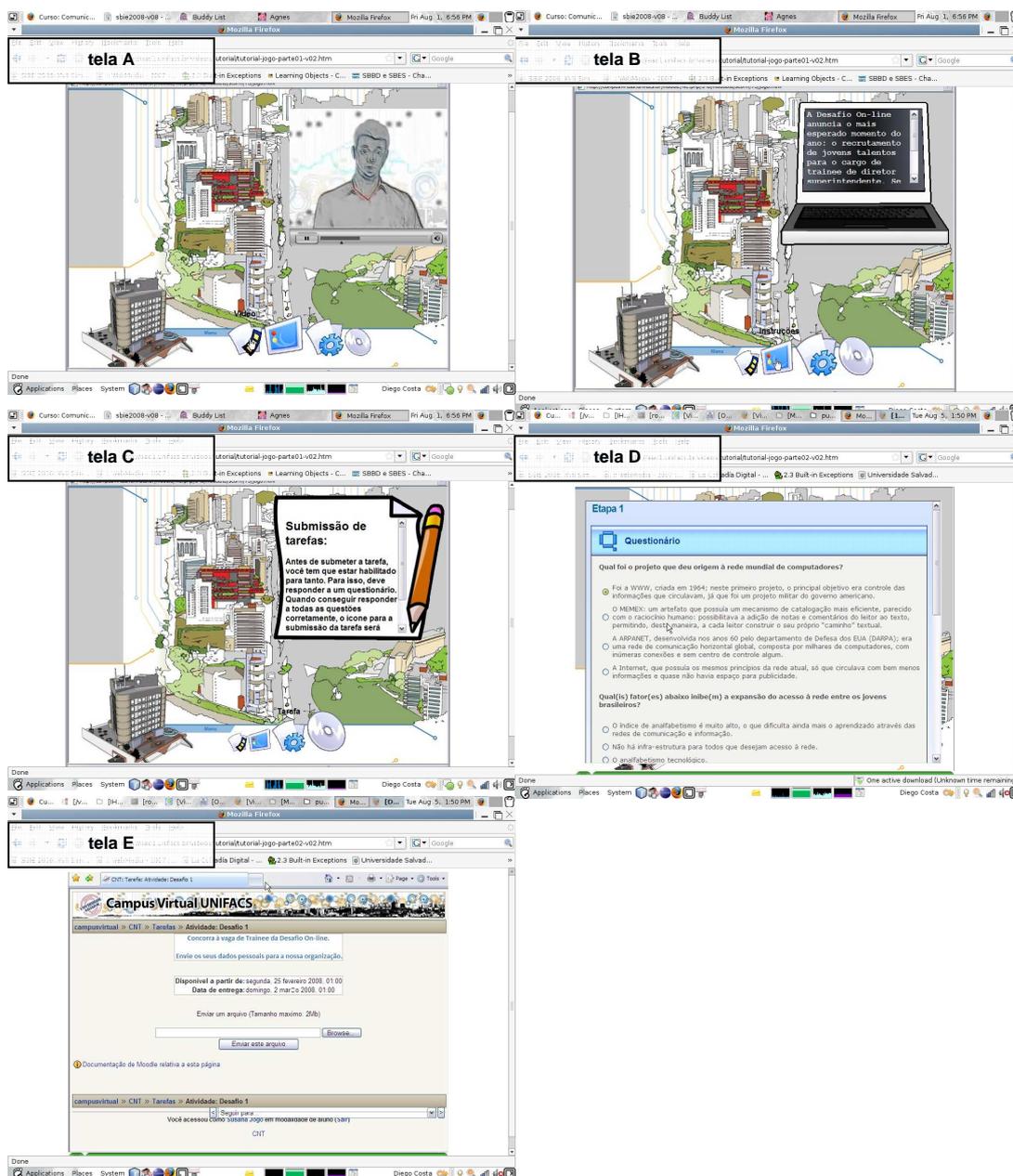


Figura 5. Telas ilustrando as principais etapas do jogo Desafio On-line.

#### 4. Desenvolvimento de Jogos em OA

A experiência no desenvolvimento e aplicação destes dois jogos permitiu uma reflexão e sistematização de boas práticas para o desenvolvimento de jogos na forma de OA. Tais práticas são apresentadas a seguir:

**Capacidade de Customização** – Quando um jogo se torna um OA, a customização passa a ser um aspecto essencial, dada a necessidade de os educadores adaptarem a dinâmica do jogo à sua realidade local. Por exemplo, no jogo Banca do Quincas é importante que o produto a ser vendido possa se adaptar ao contexto do jogador. A adaptação pode acontecer em diversos níveis, desde parâmetros simples, que são

informados ao jogo, até a possibilidade de se modificarem estruturas completas de funcionamento e apresentação, como acontece na Banca do Quincas.

**Ferramentas Especializadas × Customização** – É comum o uso de ferramentas especializadas para o desenvolvimento de OAs, tal como o Adobe Flash. Se por um lado tais ferramentas podem aumentar a produtividade em determinadas tarefas (e.g., criação de animações), por outro, elas usualmente produzem artefatos monolíticos, exigindo que o “reusuário” tenha acesso às ferramentas especializadas e saiba utilizá-las se quiser fazer adaptações no objeto. Duas estratégias interessantes para alcançar um meio termo são: tornar externos artefatos auxiliares e trabalhar com a composição de blocos menores. Estas estratégias são detalhadas nos tópicos a seguir.

**Artefatos Auxiliares Externos** – No objeto Banca do Quincas todos os artefatos de mídia (e.g., imagens e animações) foram colocados como artefatos externos. Tal procedimento permite que o “reusuário” faça adaptações apenas substituindo arquivos, sem precisar ter conhecimento da implementação do objeto, nem utilizar ferramentas especializadas.

**Composição de Blocos Menores** – Como foi relatado anteriormente, o objeto Desafio On-line utilizou a estratégia de compor artefatos e OAs menores dentro de um OA maior. Três vantagens desta estratégia são: a possibilidade de se combinarem múltiplas mídias e formas de interação, o uso de ferramentas especializadas para cada contexto e a facilidade de reuso (do todo ou das partes) e customização do objeto, que deixa de ser um bloco monolítico.

Também no desenvolvimento dos OAs, nos defrontamos com limitações relacionadas aos atuais padrões, que são desafios para pesquisas futuras:

**Customizações Permanentes** – O reuso de jogos requer a possibilidade não apenas de configuração dos mesmos, como também de armazenar suas configurações, de modo que possam ser compartilhados jogos customizados. Este foi o intuito de se utilizar planilhas para a customização do jogo Banca do Quincas. Entretanto, se por um lado é uma decisão de projeto criar um OA adaptável, por outro, os atuais padrões são mais voltados à publicação de artefatos acabados, e têm sérias limitações para tornarem tais adaptações permanentes, por dois motivos: linguagens para Web têm restrições de segurança para gravação em disco; o formato de pacote dos objetos não prevê modificação do conteúdo. Uma possível saída – que tentamos adotar no nosso projeto –, foi utilizar a interface de comunicação do AICC entre o OA e o ambiente em que ele está inserido, transferindo a responsabilidade pela persistência para o ambiente. No entanto, tal interface é bastante limitada a um conjunto de variáveis pré-estabelecidas, criando sérias restrições para dados a serem persistidos.

**Perspectiva Coletiva** – Dado que os objetos são construídos para a Web, naturalmente surge a necessidade de interação de diferentes objetos num mesmo ambiente. Esta necessidade surgiu no desenvolvimento do jogo Desafio On-line, no qual era necessária a criação de objetos com papéis distintos capazes de interagir. De um lado estavam os objetos dos alunos, registrando seu desempenho nas atividades, do outro, os objetos dos professores que deveriam ser capazes de ter acesso aos dados registrados pelos objetos dos alunos. Desse modo, os professores seriam capazes de acompanhar o desempenho dos alunos e dar-lhes um *feedback* através do próprio ambiente do jogo. Todavia, a interface de comunicação do jogo com o AVA, baseado no padrão AICC, parte de uma

perspectiva individual; assim, não é possível fazer dois objetos interagirem em um ambiente, pois cada objeto vê apenas a sua instância particular do ambiente. Na implementação do jogo Desafio On-line isto foi um empecilho para a interação de dois objetos. Esta limitação também afetou o desenvolvimento do jogo Banca do Quincas, dado que o propósito original seria a criação de dois objetos com perspectivas diferentes do jogo, um para customização, outro para o jogo em si. A impossibilidade de se criarem dois objetos interativos exigiu a criação de um único objeto, no qual o usuário deve escolher o caminho a ser tomado no início.

## **5. Considerações Finais**

Os objetos de aprendizagem são uma excelente opção para a construção e distribuição de jogos educativos. Eles possibilitam a criação de pequenos jogos auto-contidos – tal como a Banca do Quincas –, que podem ser combinados com outros recursos. Permitem também a criação de jogos que são a composição de outros objetos e são capazes de interagir com o AVA, tal como o Desafio On-line.

A experiência na criação desses jogos utilizando a tecnologia do OAs permitiu a definição de boas práticas para o seu desenvolvimento, enfatizando aspectos como reusabilidade, combinação de diversos recursos/ferramentas e composição. Por outro lado, apontamos desafios para a evolução dos atuais padrões de OA, para superar limitações decorrentes principalmente da forma orgânica de como estes padrões se estabeleceram.

## **6. Agradecimentos**

Este trabalho foi parcialmente financiado pelos projetos LabMultiflex (FAPESB-Infra 2006), MediaBank (HP Digital Publishing), WebMaps II (CNPq) e Bio-CORE (CNPq).

## **Referências**

- ADL (2006). Sharable Content Object Reference Model (SCORM) 2004 – 3rd Edition – Overview. [www.adlnet.gov/downloads/DownloadPage.aspx?ID=237](http://www.adlnet.gov/downloads/DownloadPage.aspx?ID=237), acessado em 10/2007.
- Boulic, R. and Renault, O. (1991) “3D Hierarchies for Animation”, In: *New Trends in Animation and Visualization*, Edited by Nadia Magnenat-Thalmann and Daniel Thalmann, John Wiley & Sons Ltd., England.
- IEEE L.T.S.C. (2002). Draft Standard for Learning Object Metadata – IEEE 1484.12.1-2002. [ltsc.ieee.org/wg12/files/LOM\\_1484\\_12\\_1\\_v1\\_Final\\_Draft.pdf](http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf), acessado em 09/2007.
- McDonald, W. A., Hyde, J., e Montgomery, A. (2004). CMI Guidelines for Interoperability AICC. [www.aicc.org/docs/tech/cmi001v4.pdf](http://www.aicc.org/docs/tech/cmi001v4.pdf), acessado em 10/2007.
- Norton, M. e Panar, A. (2003). IMS Simple Sequencing Information and Behavior Model – Version 1.0 Final Specification. IMS Global Learning Consortium, Inc. [www.imsglobal.org/simplesequencing/ssv1p0/imsss\\_infov1p0.html](http://www.imsglobal.org/simplesequencing/ssv1p0/imsss_infov1p0.html), acessado em 08/2008.
- Santanchè, Simone A. C. Modelo de Simulação Empresarial Baseada em Componentes de Software. Salvador, outubro de 2002. Dissertação de Mestrado.
- Smythe, C. e Jackl, A. (2004). IMS Content Packaging Information Model – Version 1.1.4 Final Specification. IMS Global Learning Consortium, Inc. [www.imsglobal.org/content/packaging/cpv1p1p4/imscp\\_infov1p1p4.html](http://www.imsglobal.org/content/packaging/cpv1p1p4/imscp_infov1p1p4.html), acessado em 08/2007.