

# DWeb3D: Um toolkit para facilitar a criação e manipulação de cenas 3D em X3D

Felipe Quintella  
Tecgraf / DI / PUC-Rio  
R. M. S. Vicente, 225  
Rio de Janeiro, RJ, Brasil  
felipe@crossway.com.br

Luciano Soares  
Tecgraf / DI / PUC-Rio  
R. M. S. Vicente, 225  
Rio de Janeiro, RJ, Brasil  
lpsoares@tecgraf.puc-rio.br

Alberto Raposo  
Tecgraf / DI / PUC-Rio  
R. M. S. Vicente, 225  
Rio de Janeiro, RJ, Brasil  
abraoso@tecgraf.puc-rio.br

## Resumo

*O padrão X3D é uma solução aberta, suas qualidades e problemas são discutidos e correlacionados com soluções existentes. Neste processo detecta-se algumas necessidades das aplicações atuais e a complexidade do X3D ao lidar com essas questões. Como tentativa de demonstrar que algumas das complexidades do X3D podem ser amenizadas foi desenvolvido o DWeb3D. O DWeb3D é um toolkit que facilita o desenvolvimento de aplicações X3D dinâmicas, demonstrando que é possível agilizar o processo de desenvolvimento, dando acesso mais amplo aos desenvolvedores. O toolkit oferece ferramentas para lidar com a publicação, o sincronismo, a interatividade e o controle de múltiplos usuários, além de possibilitar a persistência do grafo.*

## Abstract

*The X3D standard is open, its qualities and problems are discussed and correlated with other solutions. In this process it was detected some necessities in current applications and the complexity of X3D to deal with these issues. As an attempt to demonstrate that the complexity of X3D in some aspects may be reduced, the DWeb3D toolkit was built. DWeb3d is a toolkit to help the development of dynamic X3D applications, showing that it is possible to simplify the development process, increasing the access to developers. The toolkit provides tools to deal with publishing, synchronism, interactivity, multiple users management and disk persistency.*

## 1 Introdução

Uma das primeiras soluções para ambientes 3D na web foi a VRML[2]. A VRML teve boa recepção por parte da comunidade web, porém sua adoção foi limitada. No início dos anos 2000, para substituir o VRML e tentar contornar suas limitações, surgiu o X3D [3, 4], aperfeiçoando

o VRML. Um dos objetivos desta pesquisa é analisar as possíveis causas deste fato e propor métodos que podem mudar este cenário. Paralelamente a essas tecnologias, existem tentativas de prover conteúdo 3D usando plugins específicos, como o Flash [1] e o Shockwave 3D [9]. Segundo Toni Parisi [6], desenvolvedores comerciais estão expressando crescente interesse em explorar o 3D em tempo real em aplicações web.

A abordagem proposta passará pelo projeto e desenvolvimento de um toolkit de desenvolvimento de aplicações X3D que englobe as seguintes características:

- Colaboração entre usuários;
- Interação com a GUI Web;
- Integração com outras aplicações;
- Persistência de dados.

## 2 DWeb3D

Nesta pesquisa foi desenvolvido um conjunto de ferramentas que reduza a curva de aprendizado e o tempo necessário para o desenvolvimento de uma aplicação X3D, ao mesmo tempo em que promova a reutilização de código. Assim o DWeb3D é um toolkit para auxiliar o desenvolvimento de aplicações colaborativas utilizando o X3D. Entende-se como um toolkit um conjunto de bibliotecas de desenvolvimento que lidem com diversos dos desafios encontrados no desenvolvimento de uma aplicação 3D web [5].

O desenvolvimento foi baseado na linguagem C# dentro da plataforma .NET. Para os casos de estudo foi desenvolvido um plugin de renderização para a Unity3D<sup>1</sup>. A Unity3D é um motor gráfico que se propõe a ser modularizável e extensível. Ela suporta uma grande gama de formatos para importação, possui um IDE (Integrated Development Environment) bem completo. A escolha foi feita porque a Unity3D possui uma linguagem de scripts, possui

<sup>1</sup><http://unity3d.com/>

um plugin para utilização na web e possibilitaria uma interface mais rica do que utilizar um browser X3D existente, uma vez que com estes a interface teria que ser feita em ECMAScript e seria mais complexa.

## 2.1 Estrutura do DWeb3D

O toolkit DWeb3D se divide em módulos organizados por função, cada módulo representa uma biblioteca dinâmica. Isto possibilita um uso mais racional, permitindo assim a aplicação usar somente as bibliotecas dinâmicas necessárias.

**Model:** Módulo base do funcionamento do toolkit, possuindo o grafo e todas as suas classes que representam a estrutura do X3D. Também nele estão as classes principais de leitura e escrita de arquivos X3D, permitindo a completa manipulação do grafo. Ele é o responsável pela leitura dos arquivos X3D e a escrita do grafo em formato XML. Já existem classes responsáveis pela utilização da estrutura num ambiente asp.net.

**ObjectSync:** Este é o pacote responsável pela sincronização de vários objetos do grafo através da rede. A interface define quais propriedades serão sincronizadas e permite disparar o sincronismo a partir de uma única chamada de função. O importante desta biblioteca é que ela lida com toda a camada de rede e torna bastante simples a tarefa de fazer a sincronização.

**DWebServer:** Biblioteca responsável pela criação de um servidor. Ela somente recebe as sincronizações e repassa para os outros clientes. A principal característica é que ela faz uso das que possuem capacidades de transmitir as sincronizações. Ela mantém uma cópia própria do grafo. Vale lembrar que a sincronização sempre é feita do cliente para o servidor.

**Unity Render:** É um plugin de renderização que permite utilizar o grafo X3D dentro do ambiente da Unity3D. O que ele faz é criar objetos Unity através das representações do X3D e mantê-los sincronizados com os objetos do grafo.

**Util:** Este pacote contém código de auxílio interno do toolkit, como conversões e outros códigos reutilizados dentro do toolkit.

**Unity WebPlayer:** Executável com os scripts desenvolvidos para o DWeb3D, de forma a servir de demonstração da interação do plugin Unity com o motor gráfico e de como colocar o plugin para funcionar. O WebPlayer poderá ser utilizado como um substituto de um browser.

## 3 Implementação das metas no DWeb3D

De forma a ilustrar a adequação do toolkit aos objetivos propostos, é apresentado como ele pode resolver os problemas existentes e como sua estrutura foi pensada para estes casos.

### 3.1 Colaboração

Para se alcançar a meta de possibilitar a colaboração alguns fatores são importantes: facilitar a criação de um servidor; possibilitar a sincronização das posições dos objetos entre as visualizações de diversos clientes; possibilitar a criação de novos objetos via interface do cliente.

Para criar uma aplicação básica, cria-se o objeto, diz-se qual a cena padrão dele, define-se que ele é o servidor padrão (podemos ter vários na mesma máquina) e registra-se o servidor no gerente que o vai organizar. As diversas funções de baixo nível como criar socket, tratar de threads, e criar loops de leitura necessários para um servidor estão englobadas no DWeb3D e, portanto, o desenvolvedor não precisa lidar com elas. Desta forma o tamanho e a complexidade do código produzido são em muito reduzidos.

Para a sincronização, foi criado o módulo ObjectSync, onde os objetos a serem sincronizados implementem uma interface específica. Esta interface permite definir quais campos queremos que sejam sincronizados no objeto e com isso podemos promover o sincronismo através de uma simples chamada de função.

### 3.2 Persistência e carga

A persistência é feita também através dos mecanismos que permitem a colaboração. O objeto em ambiente web será exposto, ele registra um cookie que identifica quem é o usuário. Ele também lê o cookie do usuário caso ele exista e junto à classe UserManager, detecta quem é o usuário no servidor e permite que este recupere as características de sua última visita.

### 3.3 Interação com outras aplicações

Para que a aplicação X3D possa interagir com outras aplicações dois caminhos existem: Através de webservices com ECMAScript; Visualizador X3D que permita a adição de scripts.

Para o DWeb3D foi decidido seguir o segundo caminho convertendo o grafo X3D num grafo da Unity 3D de forma que seja possível publicar na web um visualizador. Desta forma os scripts podem alterar o grafo, que por sua vez se sincronizam e alteram o grafo no servidor (Figura 1).

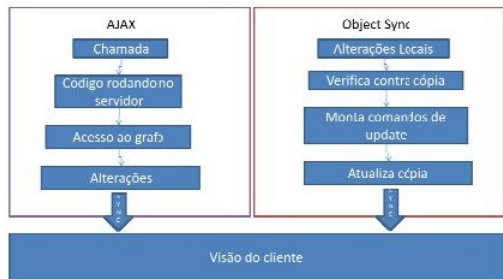


Figura 1. Esquema de sincronismo.

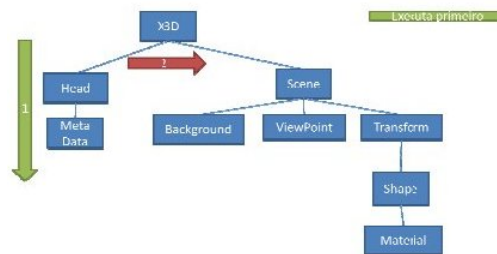


Figura 2. Ordem de renderização do grafo.

### 3.4 Interação com a GUI web

Um dos campos em maior expansão na web são as aplicações RIA<sup>2</sup> [7], que são aplicações que possibilitam uma interface melhor com o usuário. A evolução dos browsers e dos computadores pessoais tem permitido que muitas das aplicações antes feitas para rodar em desktop migrem para a web. As aplicações muito comumente alcançam esta meta através de aplicações de técnicas Ajax<sup>3</sup> [8].

## 4 Casos de estudo

Como forma de organizar os testes e as demonstrações foram criados pequenos projetos de testes, cada um focado em uma funcionalidade específica.

### 4.1 Grafo de Cena

As principais funções disponíveis no pacote Model do X3D, que encapsula o grafo e a estrutura do X3D são: Carga de um arquivo X3D; Exposição da estrutura da cena; Renderização da estrutura alterada novamente para um arquivo.

Pode-se notar que o modelo encapsula todo o processo de carga e renderização necessário para lidar com um arquivo X3D em código .NET. No final temos um grafo hierárquico com todos os nós da cena. A renderização por sua vez é feita através de uma função recursiva em profundidade como ilustrada na Figura 2.

A Figura 3 ilustra o resultado da renderização feita pelo modelo. Ela serve como forma de mostrar a correta carga e renderização.

### 4.2 Colaboração

O sistema de colaboração deve ser capaz de aceitar vários usuários e sincronizar as mensagens provenientes deles. Para isto foi definida uma cena simples que será carregada no servidor e nos dois clientes. Em seguida um dos

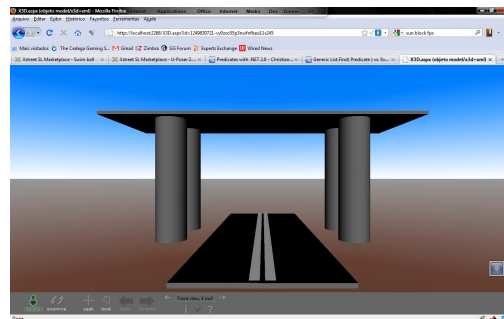


Figura 3. Renderização do modelo.

clientes irá modificar um objeto e sincronizar e o segundo cliente verá as modificações.

Uma vez conectados, cliente e servidor ficam transparentes para a aplicação. Desta forma, basta fazer as alterações necessárias e sincronizar. Para finalizar, foi criado um objeto fora do grafo para que se pudesse simular uma operação de conversa entre dois usuários Figura 4.



Figura 4. Usuários visualizando uma cena.

### 4.3 Persistência e carga

A aplicação de demonstração utiliza o segundo nível de persistência, que é a persistência em disco. Para isto, o servidor ganhou um método especial de salvamento. O código desenvolvido, define uma cena padrão e chama o método para salvá-la. Este método verifica a existência de uma base de dados padrão no disco, caso ela não exista será criada.

<sup>2</sup>Rich Internet Application

<sup>3</sup>Asynchronous Javascript And XML

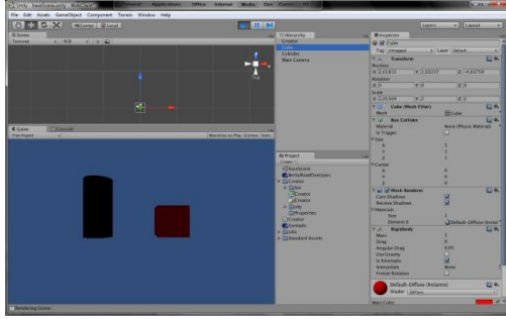


Figura 5. A Unity 3D depois de renderizado.

#### 4.4 Interação com outras aplicações

Outro exemplo é o servidor interagir com outra aplicação, através da cena X3D com o código .NET num outro servidor. Para demonstrar o funcionamento do plugin, uma cena simples foi usada. Com a implementação feita podemos acompanhar o processo de renderização da Unity3D na Figura 5.

Na Figura 5 pode-se notar que os nós criados são selecionáveis e que eles apresentam as características dos nós da cena X3D. O processo que ocorreu foi uma conversão do modelo do grafo X3D com o grafo da Unity.

O DWeb3D abordou o problema da interação com outras aplicações focando na integração com o Unity3D. Para uma possível interação com outro motor gráfico, o DWeb3D implementa um esquema de plugins de renderização. Isto significa que, bastando reescrever o código de renderização, é possível aproveitar todos os outros mecanismos do toolkit como carga, manipulação do grafo, etc.

#### 4.5 Interação com o GUI Web

A interação da GUI web pode ser feita via chamadas Ajax comuns no .Net que executam código no servidor. O código estará no mesmo ambiente do servidor X3D, e através do gerente SceneServers pode ter acesso ao grafo do servidor e alterá-lo afetando assim a cena.

### 5 Conclusões

Foram abordados alguns pontos fracos detectados no X3D, especialmente na comparação direta com outras tecnologias 3D na web. Os aspectos abordados foram: colaboração, persistência, interação com outras aplicações e interação com o GUI web.

Pudemos verificar nos exemplos deste trabalho como a utilização do DWeb3D reduz a quantidade de código necessário para se obter aplicações 3D. Esta redução de código não só facilita o processo de desenvolvimento inicial, mas também traz uma série de vantagens para aqueles

que precisam desenvolver estes códigos, não precisarem lidar com as camadas básicas, que são trabalhosas e complexas, ele poderá se focar nas funcionalidades desejadas em seu software.

O DWeb3D pode se apresentar como uma ferramenta útil para facilitar o desenvolvimento de aplicações X3D complexas e se adéqua a princípios citados anteriormente, onde um toolkit como estes, ao simplificar desenvolvimento, promove a utilização da tecnologia.

### 6. Agradecimentos

O Tecgraf é um grupo prioritariamente financiado pela Petrobras. Alberto Raposo é financiado parcialmente pelo CNPq (processo 472967/2007-0) e pela FAPERJ (Bolsa Jovem Cientista do Nosso Estado).

### Referências

- [1] J. Allaire. Macromedia Flash MX next-generation rich client. Whitepaper, Adobe, 2002. [www.adobe.com/devnet/flash/whitepapers/richclient.pdf](http://www.adobe.com/devnet/flash/whitepapers/richclient.pdf).
- [2] W. Consortium. Virtual Reality Modeling Language. Disponível em: <http://www.web3d.org/x3d/specifications/vrml/ISO-IEC-14772-VRML97/> Acesso em: 10 julho 2009, 1997.
- [3] W. Consortium. Extensible 3D (X3D). Disponível em: <http://www.web3d.org/x3d/specifications/ISO-IEC-19775-1.2-X3D-AbstractSpecification/> Acesso em: 11 julho 2009, 2008.
- [4] Y. Doi and K. Kagawa. An X3D generator plug-in for Eclipse in a Web-based Educational System for Programming. In *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pages 2523–2528, Chesapeake, VA: AACE, 2006. EDMEDIA.
- [5] S. Greenberg. Toolkits and interface creativity. *Multimedia Tools Appl.*, 32(2):139–159, 2007.
- [6] T. Parisi. Ajax3D: The Open Platform for Rich 3D Web Applications. Whitepaper, Media Machines, Inc, 2006.
- [7] Y. S. Park, J. H. Lee, H. R. Choi, H. S. Kim, J. U. Jung, and J. Y. Park. Development of an RIA-based user interface for promotion of effectiveness in marine transportation. In *ACS'08: Proceedings of the 8th conference on Applied Computer Science*, pages 366–372, Stevens Point, Wisconsin, USA, 2008. World Scientific and Engineering Academy and Society (WSEAS).
- [8] R. Riordan. *Head First Ajax*. O'Reilly Media, 2008.
- [9] R. J. D. SANTOS, A. L. BATAIOLA, and R. P. DUBIELA. Aspectos Fundamentais da Criação de jogos em Shockwave 3D. *WJogos / SBGames 2004*, Simpósio Brasileiro de Jogos de Computador e Entretenimento Digital, 2004.