

# Programa de modelagem 3D em VRML para web

## VRML 3d modeling software for Web

**Eng. Fernando Batista Bruno**

Mestrando do Programa de Pós-Graduação em Design (PGDESIGN), UFRGS – Brasil.  
fernando.bruno@ufrgs.br

**Dr. José Luis Farinatti Aymone**

PGDESIGN, Universidade Federal do Rio Grande do Sul (UFRGS)  
aymone@ufrgs.br

**Dr. Fábio Gonçalves Teixeira**

PGDESIGN, Universidade Federal do Rio Grande do Sul (UFRGS)  
fabio.teixeira@ufrgs.br

**Dra. Tânia Luisa Koltermann da Silva**

PGDESIGN, Universidade Federal do Rio Grande do Sul (UFRGS)  
tania.koltermann@ufrgs.br

**Abstract.** This work describes a software which is a VRML (Virtual Reality Modeling Language) modeler based on Web and a learning object for this language. The modeler, developed using PHP, HTML and JavaScript, runs directly on a website and it is able to show the model and its VRML code during the creation process, and to record it on the user machine. The software developed is able to model primitive forms, as box, cylinder and sphere, and faceset surfaces, helping users to model 3D objects and to understand VRML syntax. The model material is chosen according to color and transparency.

**Keywords.** Web; VRML; 3D Modeling; Virtual Reality.

## Introdução

A realidade virtual é uma tecnologia que pode ser aplicada em várias áreas de conhecimento, como engenharia, arquitetura, design e educação (Teixeira et al., 2007). O atual estágio de desenvolvimento da tecnologia de informação, aliado ao crescimento da utilização de conexões de banda larga à Internet, permite a construção de aplicações totalmente baseadas na Web.

O VRML – Virtual Reality Modeling Language – é uma linguagem utilizada para descrever ambientes tridimensionais virtuais através de formas primitivas (cilindros, esferas, paralelepípedos e cones), superfícies facetadas e reversas (Ames et al., 1996). Os arquivos VRML são constituídos de texto simples, escritos com codificação de caracteres UTF-8, que recebem a extensão WRL.

A versatilidade do VRML pode ser verificada através de sua estrutura, baseada em nós que podem ser aninhados e escritos com uma sintaxe simples – fatores que permitem sua implementação em diversas linguagens de programação.

Apesar de ser um formato projetado para rodar em navegadores de Internet, através de plugins (Nadeau, 1998) como o Cortona3D Viewer (Cortona3D, 2009), Octaga Player (Octaga, 2009) ou FreeWRL (Communications Research Centre Canada, 2009), não existem modeladores VRML baseados na Web. A maioria dos arquivos de realidade virtual encontrados na rede foi modelada em algum programa baseado em desktop, como o Autodesk 3ds Max (Autodesk Inc., 2009), Rhino (Robert McNeel & Associates, 2009), ou Blender (Blender Foundation, 2009).

O presente trabalho tem por objetivo apresentar a implementação de um modelador VRML baseado na Web, utilizando na sua parte lógica a linguagem PHP (2009). O PHP é uma linguagem modular muito utilizada na criação e exibição de conteúdo dinâmico na Internet. O PHP funciona de maneira interpretada (os programas não são compilados), rodando em conjunto com um servidor Web (Dall'Oglio, 2007).

Na parte de apresentação, é utilizada a linguagem HTML (World Wide Web Consortium, 1999), que é o formato padrão das páginas na Internet, assim como o JavaScript (Mozilla Foundation, 2009) – linguagem padrão para os client side scripts.

O modelador proposto pode ser utilizado como um objeto de aprendizagem da linguagem VRML para iniciantes.

## Modelador 3D

O desenvolvimento do modelador parte do trabalho iniciado por Pache (2002), que elaborou uma série de classes em PHP para a geração de formas primitivas em VRML. Tais classes foram disponibilizadas sob uma licença que restringe seu uso para fins educacionais e não-comerciais.

O software disponibilizado por Pache (2002) não possui uma interface para modelagem, e o resultado é enviado diretamente para um arquivo no servidor, não permitindo a sua visualização imediata. Assim, o foco deste trabalho está na implementação de novas classes que permitam a construção de formas mais complexas através de superfícies facetadas, assim como na elaboração de uma interface com área de visualização e painel de comandos, apresentada na Figura 1, onde seja possível selecionar parâmetros (forma, dimensões, cor e transparência) do modelo.

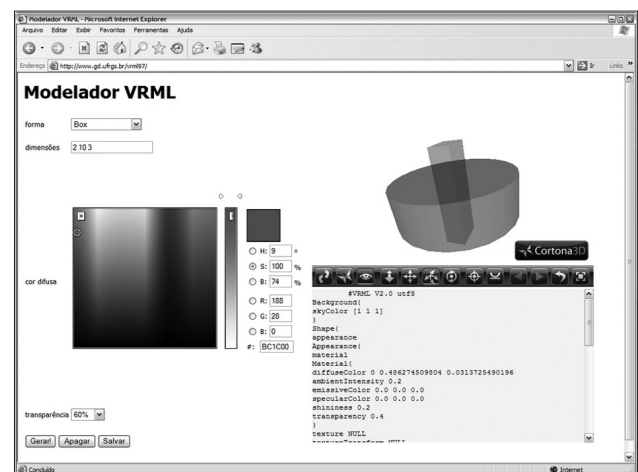


Figura 1: interface de modelagem

# SIGraDi 2009 sp

A seguir, são apresentadas as funcionalidades implementadas no Modelador 3D: Seletor de formas (Box, Sphere, Cylinder, IndexedFaceSet, IndexedLineSet), Seletor de cores, Transparência, Geração de modelos e Gravação.

## Seletor de formas

O seletor de formas oferece, além de formas primitivas (Box, Sphere e Cylinder), as novas implementações de superfícies facetadas (IndexedFaceSet) e linhas (IndexedLineSet).

A geometria dos modelos é definida através do seletor de formas, e suas dimensões são atribuídas através de um campo homônimo, que deve ser preenchido conforme as particularidades de cada uma das formas oferecidas.

### Forma: Box

A modelagem de paralelepípedos exige três parâmetros: largura, altura e profundidade – x, y e z, separados por um espaço. Para um cubo de lado 2.5:

Dimensões: 2.5 2.5 2.5

### Forma: Sphere

Para a construção de esferas, o único parâmetro necessário é o seu raio. Para uma esfera de raio 3:

Dimensões: 3

### Forma: Cylinder

Para desenhar um cilindro, os parâmetros são raio da base e altura, separados por um espaço. Para um cilindro de raio 0,5 e altura 6:

Dimensões: 0.5 6

### Forma: IndexedFaceSet

A modelagem de superfícies facetadas é feita através de duas etapas distintas.

A primeira é a definição dos vértices do modelo através de suas coordenadas (x y z) separadas por um espaço, sendo cada ponto limitado por uma vírgula (x0 y0 z0, x1 y1 z1, x2 y2 z2, ... , xn yn zn).

A segunda é a definição de conexões entre os vértices criados na primeira etapa, que originarão as faces, separados por uma vírgula. Cada ponto é identificado por um índice, iniciando em 0 – o primeiro ponto é 0, o segundo é 1, o terceiro é 2, prosseguindo desta maneira. Para indicar o fechamento de uma face, utiliza-se o índice -1.

Para separar as duas etapas, é utilizado um ponto-e-vírgula (pontos;conexões).

Para modelar um triângulo retângulo com catetos igual a 2:

Dimensões: 0 0 0,2 0 0,2 0 0;0,1,2,0,-1

### Forma: IndexedLineSet

O desenho de linhas segue o mesmo princípio do desenho de superfícies facetadas: duas etapas para definição dos vértices e das conexões, utilizando-se a mesma sintaxe. Para um quadrado de lado 1:

Dimensões: 0 0 0,1 0 0,1 1 0 0;0,1,2,3,0,-1

## Seletor de cores

Seletores de cores são um tema recorrente e bastante explorado por vários desenvolvedores independentes. Na Internet, podem ser encontrados vários tipos de seletores, licenciados de inúmeras maneiras e para diferentes usos. O seletor utilizado neste trabalho foi desenvolvido em JavaScript por Dyer (2007), e sua licença permite o uso irrestrito para qualquer fim (Figura 2). A meta do desenvolvimento deste seletor é ser o mais próximo possível do seletor de cores do Adobe Photoshop.

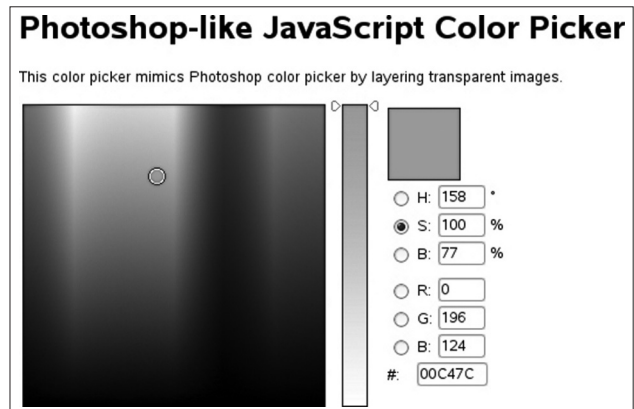


Figura 2: seletor de cores desenvolvido em JavaScript

O seletor oferece três tipos distintos de saída:

- HSB: Hue (°), Saturation (%), Brightness (%);
- RGB: Red, Green, Blue (0 – 255);
- Hexadecimal: #RRGGBB;

Para o modelador, é utilizada a saída RGB, que é o padrão do formato VRML. Porém, antes de ser utilizado, os componentes de cor devem ser transformados, já que no VRML a quantidade de cada componente varia entre 0 e 1 e não entre 0 e 255, como o seletor oferece.

O único parâmetro que pode ser alterado através deste seletor é a cor difusa do material (diffuse color). Outros parâmetros de cor, como cor emissiva, recebem um valor padrão.

## Transparência

A transparência do material pode ser atribuída através do campo de mesmo nome, que oferece uma caixa de seleção com opções que vão de 0% a 100%, com incrementos de 10%.

Através deste parâmetro, pode-se perceber que o mapeamento do material em formas primitivas, que o VRML considera como sólidos, ocorre somente nas faces cujos vetores normais apontem para a parte externa do modelo – as faces internas não são mapeadas.

Em superfícies facetadas, para que o material seja mapeado em ambas faces, é necessário informar que o objeto construído não é um sólido, através do parâmetro "SOLID FALSE", dentro do nó da geometria, o que é configurado automaticamente neste modelador.

## Geração de modelos

O botão Gerar envia os dados digitados no formulário para o núcleo do programa, onde são tratados e servem de parâmetro para a criação de objetos baseados nas classes existentes. Estes objetos são responsáveis pelo processamento das informações, transformando os dados enviados em nós VRML.

Quando o servidor termina de montar o ambiente VRML, um script PHP o envia para a máquina cliente, e o código-fonte (os nós, escritos em texto simples) fica armazenado em um cookie (arquivo temporário) na máquina cliente. Este procedimento faz com que o modelador sempre adicione novos modelos ao ambiente existente, ao invés de substituí-los, já que, toda vez que o gerador é acionado, os dados contidos neste cookie são enviados junto aos dados do formulário.

Durante este processo, o usuário tem acesso ao código-fonte do ambiente – que é o conteúdo do arquivo VRML (Figura 3).

```
#VRML V2.0 utf8
Background{
  skyColor [1 1 1]
}
Shape{
  appearance
  Appearance{
    material
    Material{
      diffuseColor 0.788235294118 0.196078431373 0
      ambientIntensity 0.2
      emissiveColor 0.0 0.0 0.0
      specularColor 0.0 0.0 0.0
      shininess 0.2
      transparency 0.2
    }
    texture NULL
    textureTransform NULL
  }
  geometry
  IndexedFaceSet{
    coord
```

Figura 3: código-fonte do arquivo VRML gerado

## Gravação

O botão Salvar permite que o usuário grave o arquivo gerado em sua máquina. O procedimento é bastante simples, pois o código-fonte do ambiente já está disponível em um cookie, bastando ao programa redirecionar a saída para um arquivo com o nome e o caminho que o usuário escolher.

## Ferramentas utilizadas

Durante o desenvolvimento e depuração do código do modelador, vários programas e ferramentas foram utilizados. Estas ferramentas podem ser classificadas em quatro grupos: Sistemas Operacionais, Web Browsers, Plugins e Ambientes de Programação, conforme é mostrado na Tabela 1.

Sistema Operacional	MS Windows XP	OpenSUSE 11
Web Browser	MS Internet Explorer 6, 7 e 8 (beta); Mozilla Firefox 3; Google Chrome	Mozilla Firefox 3; Opera 9.62
Plugin	Cortona 3D Viewer	FreeWRL
Ambiente de Programação	Notepad ++; VRML Pad 2	Quanta Plus

## Considerações finais

O modelador ainda está no estágio inicial de desenvolvimento, necessitando de uma interface mais completa e intuitiva para tornar mais simples a sua utilização.

Apesar desta limitação, o modelador apresenta uma grande variedade de possibilidades, mostrando o que é possível ser realizado quando o VRML é aliado a uma linguagem de programação robusta, com suporte a objetos e classes.

Para iniciantes, o modelador proposto serve também como objeto de aprendizagem, auxiliando na compreensão da sintaxe da linguagem e dos dados de entrada para a modelagem das diversas formas (Box, Sphere, Cylinder...) em VRML.

Com a continuidade deste projeto, outras implementações podem ser incorporadas, entre elas:

- Transformações de translação e rotação;
- Câmeras;
- Editor de materiais com texturas;
- Splines e NURBS (Non Uniform Rational Basis Splines).

## Referências

- Ames, A. L., Nadeau, D. R., Moreland, J. L.: 1997, The VRML 2.0 Sourcebook, John Wiley & Sons, Inc..
- Autodesk Inc.: 2009, Autodesk 3ds Max 2010, disponível em <<http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13571450>>, acesso em 11 set. 2009.
- Blender Foundation: 2009, Blender 2.49b, disponível em <<http://www.blender.org/download/get-blender/>>, acesso em 11 set. 2009.
- Communications Research Centre Canada: 2009, FreeWRL 1.22.4, disponível em <<http://freewrl.sourceforge.net/download.html>>, acesso em 10 set. 2009.
- Cortona3D: 2009, Cortona3D Viewer 6.0, disponível em <<http://www.cortona3d.com/cortona>>, acesso em 10 set. 2009.
- Dall'Oglio, P.: 2007, PHP: Programando com Orientação a Objetos, Novatec, São Paulo.
- Dyer, J.: 2007, Photoshop-like JavaScript Color Picker, disponível em <<http://johndyer.name/file.axd?file=colorpicker.zip>>, acesso em 10 set. 2009.
- Mozilla Foundation: 2009, Core JavaScript 1.5 Reference, disponível em <[https://developer.mozilla.org/en/Core\\_JavaScript\\_1.5\\_Reference](https://developer.mozilla.org/en/Core_JavaScript_1.5_Reference)>, acesso em 11 set. 2009.
- Nadeau, D. R.: 1998, VRML 98: Introduction to VRML 97, San Diego Supercomputer Center, disponível em <<http://www.sdsc.edu/~nadeau/Courses/VRML98/vrml97.zip>>, acesso em 10 set. 2009.
- Octaga: 2009, Octaga Player, disponível em <[http://www.octaga.com/index.php?option=com\\_content&task=view&id=17&Itemid=229](http://www.octaga.com/index.php?option=com_content&task=view&id=17&Itemid=229)>, acesso em 11 set. 2009.
- Pache, C.: 2002, VRML, disponível em <<http://www.phpclasses.org/browse/package/759.html>>, acesso em 10 set. 2009.
- Robert McNeel & Associates: 2009, Rhino 4.0 SR6 Evaluation, disponível em <<http://download.rhino3d.com/rhino/4.0/eval/>>, acesso em 11 set. 2009.
- PHP: 2009, PHP: Hypertext Preprocessor, disponível em <<http://php.net/downloads.php>>, acesso em 11 set. 2009.
- Teixeira, F. G., Silva, R. P., Silva, T. L. K., Hoffmann, A. T., Aymone, J. L. F.: 2007, HyperCAL3D - Modelador de Sólidos para Geometria Descritiva, Graphica 2007, disponível em <[http://www.degraf.ufpr.br/artigos\\_graphica/HyperCAL.pdf](http://www.degraf.ufpr.br/artigos_graphica/HyperCAL.pdf)>, acesso em 10 set. 2009.
- World Wide Web Consortium: 1999, HTML 4.01 Specification, disponível em <<http://www.w3.org/TR/html4/>>, acesso em 11 set. 2009.