## Overview

<table>
<thead>
<tr>
<th>Category</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>LibrarySpecificDemos</strong></td>
<td>tcl3dCg</td>
</tr>
<tr>
<td></td>
<td>tcl3dFTGL</td>
</tr>
<tr>
<td></td>
<td>tcl3dGauges</td>
</tr>
<tr>
<td></td>
<td>tcl3dOde</td>
</tr>
<tr>
<td></td>
<td>tcl3dOgl</td>
</tr>
<tr>
<td></td>
<td>tcl3dOglExt</td>
</tr>
<tr>
<td></td>
<td>tcl3dSDL</td>
</tr>
<tr>
<td></td>
<td>tcl3dTogl</td>
</tr>
<tr>
<td><strong>Tcl3DSpecificDemos</strong></td>
<td>rVis</td>
</tr>
<tr>
<td><strong>TutorialsAndBooks</strong></td>
<td>CodeSampler</td>
</tr>
<tr>
<td></td>
<td>GameProgrammer</td>
</tr>
<tr>
<td></td>
<td>NeHe</td>
</tr>
<tr>
<td></td>
<td>Nopper</td>
</tr>
<tr>
<td></td>
<td>RedBook</td>
</tr>
<tr>
<td><strong>OpenSceneGraph</strong></td>
<td>CubosLocos</td>
</tr>
<tr>
<td></td>
<td>FopingTutorials</td>
</tr>
<tr>
<td></td>
<td>NPS-Tutorials</td>
</tr>
<tr>
<td></td>
<td>OsgHelp</td>
</tr>
<tr>
<td></td>
<td>QuickStartGuide</td>
</tr>
<tr>
<td>Category</td>
<td>LibrarySpecificDemos</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>Root</td>
<td>Contents</td>
</tr>
<tr>
<td>Types</td>
<td>tcl3dCg tcl3dFTGL tcl3dGauges tcl3dOde tcl3dOgl tcl3dOglExt tcl3dSDL tcl3dTogl</td>
</tr>
</tbody>
</table>
This section contains Cg demo applications from several resources, that have been ported to Tcl3D. The examples cover vertex and fragment shader programming in Cg. Original sources from different sites. See the documentation for details.

Available demos

<table>
<thead>
<tr>
<th>QJuliaGPU</th>
<th>cgFireInTheSky</th>
<th>cgParticles</th>
<th>cgTeapot</th>
</tr>
</thead>
<tbody>
<tr>
<td>cgVertexExample</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This program ray traces the quaternion Julia set in a fragment shader using the sphere tracing method. The program draws a fullscreen quad where each fragment of the quad specifies a different ray. These rays are passed to the fragment shader which iteratively takes conservative steps along a ray as determined by a distance estimator for the set. The rays will either stop when close to an isosurface of the distance function (considered a hit), or leave the bounding sphere of the Julia set. If the ray is a hit, shading is performed by approximating the gradient of the distance function and using this as a surface normal.

A more complete description of the sphere tracing method can be found in John Hart's paper, "Ray Tracing Deterministic 3-D Fractals" (http://graphics.cs.uiuc.edu/~jch/papers/rtqjs.pdf).

Controls:

- left mouse button: rotate view
- middle mouse button: zoom in/out
- right mouse button: translate view
- m: toggle morph animation
- s: toggle shadows on/off
- r: reload shaders from disk
- i/I: increment/decrement 1st imaginary component of Julia set constant
- j/J: increment/decrement 2nd imaginary component of Julia set constant
### Constant

- **k/K:**  increment/decrement 3rd imaginary component of Julia set
- **l/L:**  increment/decrement real component of Julia set constant
- **-/+:**  change number of iterations used to test convergence of a point
- **b/n:**  change precision of rendering

By default the program will shift through a random constants for the Julia set within the cube $[-1,1]^4$. Increasing the number of iterations or the precision will increase the amount of detail seen in the rendering. The former more accurately determines whether a point is included in the set, whereas the latter intersects an isosurface of the distance function closer to the actual set. Both of these parameters run into precision or computation limits when set too high.

---

Original C++ and Cg code by Keenan Crane (kcrane@uiuc.edu)
See [http://www.cs.caltech.edu/~keenan/project_qjulia.html](http://www.cs.caltech.edu/~keenan/project_qjulia.html) for the original files.

Modified for Tcl3D by Paul Obermeier 2009/08/29
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: | cgFireInTheSky
---|---
Type: | tcl3dCg
Category: | LibrarySpecificDemos
Root: | Contents

Original files from: http://www.shadertech.com/shaders/FireInTheSky-src.zip

Original files are Copyright (c) 2002 Jason Jerald
This software is provided 'as-is', without any express or implied warranty.
In no event will the authors be held liable for any damages arising from the
use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented;
you must not claim that you wrote the original software.
If you use this software in a product, an acknowledgment in
the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such,
and must not be misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.
**Demo:** cgParticles

**Type:** tcl3dCg

**Category:** LibrarySpecificDemos

**Root:** Contents

---

cgParticles.tcl

Particle Effects using CG and OpenGL

Original files from: http://www.shadertech.com/shaders/ParticleSystem-src.zip

Original files are Copyright (c) 20002 Arkadiusz Waliszewski

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Modified for Tcl3D by Paul Obermeier 2005/11/07
See www.tcl3d.org for the Tcl3D extension.
Original files from: http://developer.nvidia.com/Cg
This is the example called interfaces_ogl as included in the Cg Toolkit.
Modified for Tcl3D by Paul Obermeier 2005/11/07
See www.tcl3d.org for the Tcl3D extension.
**Demo:** cgVertexExample

<table>
<thead>
<tr>
<th>Type:</th>
<th>tcl3dCg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>LibrarySpecificDemos</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

---

**Tcl3D demo: Simple Cg vertex shader**

```
cgVertexExample.tcl

Original files from: http://developer.nvidia.com/Cg
This is the example called runtime_ogl as included in the Cg Toolkit.

Modified for Tcl3D by Paul Obermeier 2005/11/07
See www.tcl3d.org for the Tcl3D extension.
```
This section contains FTGL demo applications written in Tcl3D. The examples cover the demo applications distributed with FTGL.
This demo demonstrates the different rendering styles available with FTGL. Press <n> to change the font rendering style. Press <enter> to enable edit mode.

Please contact me if you have any suggestions, feature requests, or problems.

Henry Maddocks
henryj@paradise.net.nz
http://homepages.paradise.net.nz/henryj/

Modified for Tcl3D by Paul Obermeier 2006/01/18
See www.tcl3d.org for the Tcl3D extension.
A test program showing the 5 different font rendering types.

ftglTest.tcl

C++ source changed by mrn@paus.ch/ max rheiner
original source: henryj@paradise.net.nz

Modified for Tcl3D by Paul Obermeier 2006/01/18
See www.tcl3d.org for the Tcl3D extension.
This section contains demo applications written with Tcl3D extensions packages. The examples cover the tcl3dGauges package, which was supplied by Victor G. Bonilla.

<table>
<thead>
<tr>
<th>Available demos</th>
</tr>
</thead>
<tbody>
<tr>
<td>gaugedemo</td>
</tr>
<tr>
<td>gaugetest</td>
</tr>
<tr>
<td>Demo:</td>
</tr>
<tr>
<td>---------------</td>
</tr>
<tr>
<td>Type:</td>
</tr>
<tr>
<td>Category:</td>
</tr>
<tr>
<td>Root:</td>
</tr>
</tbody>
</table>

Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dGauges
Filename: gaugedemo.tcl
Author: Paul Obermeier
Description: Demo program showing the use of the Tcl3D extension package gauge.
Demo: gaugetest

Type: tcl3dGauges

Category: LibrarySpecificDemos

Root: Contents

Tcl3D demo: Gaugetest

Test program for the Tcl3D extension package gauge.
The program allows to show the 4 gauges at different sizes.

Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dGauges

Filename: gaugetest.tcl

Author: Paul Obermeier
This section contains ODE demo applications written in Tcl3D. The examples cover some demo applications distributed with PyOde.

<table>
<thead>
<tr>
<th>Available demos</th>
</tr>
</thead>
<tbody>
<tr>
<td>odeGravity</td>
</tr>
<tr>
<td>odeJoints</td>
</tr>
</tbody>
</table>
## Demo: odeGravity

<table>
<thead>
<tr>
<th>Type:</th>
<th>tcl3dOde</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>LibrarySpecificDemos</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

### Description:
Tcl3D Ode example: Bodies influenced by gravity.
Based on PyODE Tutorial 1 By Matthias Baas.

### Copyright:
2006-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.
Tcl3D demos at a glance

Version 0.5.0, December 2010

Copyright © 2005-2010 by Paul Obermeier. All rights reserved.

---

**Demo:** odeJoints  
**Type:** tcl3dOde  
**Category:** LibrarySpecificDemos  
**Root:** Contents

---

Copyright: 2006-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dOde  
Filename: odeJoints.tcl  
Author: Paul Obermeier  
Description: Tcl3D Ode example: Connected bodies with joints  
Based on PyODE Tutorial 2 By Matthias Baas.
Tcl3D: Doing 3D with Tcl

**Type:** tcl3dOgl

**Category:** LibrarySpecificDemos

**Root:** Contents

This section contains [OpenGL](https://www.opengl.org) demo applications from several resources, that have been ported to Tcl3D. The examples cover basic OpenGL programming. Original sources from different sites. See the documentation for details.

### Available demos

| GearTrain | ModelViewMatrix | Sierpinski | animlogo |
| atlantis | drawReadPixels | gluCylinder | glutShapes |
| imgproc | molecules | multiview | platonic |
| spheres | tcl3dChaos | texanim | texgen |
| trislam |
# Gear Train Simulation - Q Solutions

**GearTrain.tcl**

**GearTrain Simulator** * Version: 1.00

Copyright (C) 1999 Shobhan Kumar Dutta All Rights Reserved.

<skdutta@del3.vsnl.net.in>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL SHOBHAN KUMAR DUTTA BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT
Tcl conversion Copyright Philip Quaife August 2005.

This file is placed in the public domain

Slightly modified for Tcl3D presentation by Paul Obermeier 2006/08/02
See www.tcl3d.org for the Tcl3D extension.
## Demo: ModelViewMatrix

**Type:** tcl3dOgl  
**Category:** LibrarySpecificDemos  
**Root:** Contents

### Tutorial OpenGL Transformation

Original C++ code by Song Ho Ahn (song.ahn@gmail.com)  
See [www.songho.ca/opengl/gl_transform.html](http://www.songho.ca/opengl/gl_transform.html) for the original files

Modified for Tcl3D by Paul Obermeier 2009/09/13  
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
**Demo:** Sierpinski

**Type:** tcl3dOgl

**Category:** LibrarySpecificDemos

**Root:** Contents

---

**Tcl3D demo: Sierpinski Tetrahedron**

Recursive depth 6 | Build | Optimize | View distance | Animat

4096 triangles: 45 msec to build

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4 0, Tcl 8.4.13)

---

**Copyright:** 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

**Module:** Tcl3D -> tcl3dOgl

**Filename:** Sierpinski.tcl

**Author:** Paul Obermeier

**Description:** Tcl3D demo displaying a 3D Sierpinski Tetrahedron.

Derived from a demo by Gerard Sookahet (tetra-3dc.tcl), which used the 3dcanvas package.

The original version is at: http://wiki.tcl.tk/11832.

Incorporates optimization functions by Philip Quaife. See the Tcl'ers Wiki http://wiki.tcl.tk/14820 for a description of his optimizations.
animlogo.tcl

The animated OpenGL logo

This file is part of the openGL-logo demo.
(c) Henk Kok (kok@wins.uva.nl)

Copying, redistributing, etc is permitted as long as this copyright notice and the Dutch variable names :) stay in tact.

Original sources available at:
http://www.opengl.org/resources/code/samples/glut_examples/demos/demos.html

Modified for Tcl3D by Paul Obermeier 2006/08/02
See www.tcl3d.org for the Tcl3D extension.
atlantis.tcl

Copyright (c) Mark J. Kilgard, 1994. */
(c) Copyright 1993, 1994, Silicon Graphics, Inc. ALL RIGHTS RESERVED
Permission to use, copy, modify, and distribute this software for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice appear in supporting documentation, and that the name of Silicon Graphics, Inc. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

THE MATERIAL EMBODIED ON THIS SOFTWARE IS PROVIDED TO YOU "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SILICON GRAPHICS, INC. BE LIABLE TO YOU OR ANYONE ELSE FOR ANY DIRECT, SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER, INCLUDING WITHOUT LIMITATION, LOSS OF PROFIT, LOSS OF USE, SAVINGS OR REVENUE, OR THE CLAIMS OF
**Tcl3D demo: Speed test of `glDrawPixels` and `glReadPixels`**

<table>
<thead>
<tr>
<th>Format and type</th>
<th>Image size</th>
<th>Num calls</th>
<th>Test Run</th>
</tr>
</thead>
<tbody>
<tr>
<td>GL_RGBA</td>
<td>256x256</td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td>GL_UNSIGNED_BYTE</td>
<td>256x256</td>
<td>1000</td>
<td></td>
</tr>
</tbody>
</table>

- **Draw/Read of 1000 images:** 0.4355 1.9635 secs
- **Draw/Read of 1 image:** 0.4355 1.9636 ms/1000 (65536 pixels)
- **Draw/Read of 1 pixel:** 0.0066 0.0300 ms/1000
- **Saving image to GL_RGBA-256x256.png**

Test Run:

```tcl
Tcl3D demo testing the speed of the `glDrawPixels` and `glReadPixels` functions. The program generates a color gradient image of a specified size. If the image size is greater than 256x256, the color gradient is tiled. This image is then drawn into the framebuffer with `glDrawPixels` and read back with `glReadPixels` several times. The time needed for drawing and reading back is reported into a text widget and stdout (for batch processing). The format and type of the image data can be specified for testing the differences in speed. Currently the following formats and types are implemented:
- Formats: GL_RGB, GL_BGR, GL_RGBA, GL_BGRA.
- Types: GL_UNSIGNED_BYTE

Author: Paul Obermeier
Date: 2009-07-16
```
### gluCylinder

<table>
<thead>
<tr>
<th>Demo</th>
<th>gluCylinder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>tcl3dOgl</td>
</tr>
<tr>
<td>Category</td>
<td>LibrarySpecificDemos</td>
</tr>
<tr>
<td>Root</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Description:**

Tcl3D demo showing the use of gluQuadric routines to draw a cylinder.

---

**Copyright:**

2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

**Module:**

Tcl3D -> tcl3dOgl

**Filename:**

gluCylinder.tcl

**Author:**

Paul Obermeier

**Description:**

Tcl3D demo showing the use of gluQuadric routines to draw a cylinder.
Demo: glutShapes
Type: tcl3dOgl
Category: LibrarySpecificDemos
Root: Contents

<table>
<thead>
<tr>
<th>Tc13D demo: OpenGL GLUT shapes</th>
</tr>
</thead>
<tbody>
<tr>
<td>GearTrain.tcl</td>
</tr>
<tr>
<td>Sierpinski.tcl</td>
</tr>
<tr>
<td>animation.tcl</td>
</tr>
<tr>
<td>atlas.tcl</td>
</tr>
<tr>
<td>gluCylinder.tcl</td>
</tr>
<tr>
<td>glutShapes.tcl</td>
</tr>
<tr>
<td>multiview.tcl</td>
</tr>
<tr>
<td>spheres.tcl</td>
</tr>
</tbody>
</table>

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tc1 8.4.13)

Key-Escape Exit
Key-R Reset rotation
Key-Up/Down Decrease/Increase x rotation speed
Key-Left/Right Decrease/Increase y rotation speed

Copyright: 2006-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tc13D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tc13D -> tcl3dOgl
Filename: glutShapes.tcl
Author: Paul Obermeier
Date: 2006-12-01

Description: Tc13D demo showing all supported GLUT shapes.
Examples of various image processing operations coded as OpenGL accumulation buffer operations. This allows extremely fast image processing on machines with hardware accumulation buffers (RealityEngine, InfiniteReality, VGX).

This demo is part of the advanced glut demos. See http://www.opengl.org/resources/code/samples/glut_examples/advanced/advanced.html

Modified for Tcl3D by Paul Obermeier 2007/07/28
See www.tcl3d.org for the Tcl3D extension.
Demo: molecules
Type: tcl3dOgl
Category: LibrarySpecificDemos
Root: Contents

Tcl3D demo: Molecule viewer (4HHB.pdb)

Copyright: 2009-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dOgl
Filename: molecules.tcl
Author: Paul Obermeier
Description: Tcl3D demo displaying molecules as colored spheres.

The molecule description is read from a Protein Data Base file. See http://www.pdb.org for more information about PDB files. This site is also a resource for downloading PDB files.

Currently supported keywords are ATOM, HETATM and CONECT. Feel free to extend and optimize the PDB parser.

Atom color coding and atom radius are taken from the OpenSource molecule viewer QuteMol: http://qutemol.sourceforge.net/
**Demo:** multiview

**Type:** tcl3dOgl

**Category:** LibrarySpecificDemos

**Root:** Contents

---

**Tcl3D demo: Multiple viewports**

---

**Copyright:** 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

**Module:** Tcl3D -> tcl3dOgl

**Filename:** multiview.tcl

**Author:** Paul Obermeier

**Description:** Tcl3D demo showing the famous teapot in 4 different viewports on a single togl widget.
platonic.c - An OpenGL demonstration that draws the six platonic solids:
  The tetrahedron, the cube, the dodecahedron, the octahedron,
  the icosahedron and the teapotahedron. :-)  
The ray-traced image by Arvo and Kirk on the front cover of 
"An Introduction to Ray Tracing" (A. S. Glassner (ed.), 
Academic Press) inspired me to write this demo.
A menu with a number of options is tied to the left mouse 
button.

Author: Gustav Taxen, nv91-gta@nada.kth.se

Notes:
The code is not very pretty, nor is it optimized wrt OpenGL.
Should add shadows as well, but I'll save that for the next 
version...

Copyright (C) 1998 Gustav Taxen.
This is free software with ABSOLUTELY NO WARRANTY.

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, write to the Free Software Foundation, 675 Mass Ave, Cambridge, MA 02139, USA.

Original C code taken from:
http://www.student.nada.kth.se/~nv91-gta/OpenGL/projects/platonic/

Modified for Tcl3D by Paul Obermeier 2008/12/21
See www.tcl3d.org for the Tcl3D extension.

See http://design.osu.edu/carlson/history/lesson20.html about the history of the famous Utah teapot. This page also contains an image of the original ray-traced scene by Arvo and Kirk.
The image is also on the front page of Glassner's book "An Introduction to Ray Tracing".
For a mathematical description of the five platonic solids see http://en.wikipedia.org/wiki/Platonic_solid
Demo: spheres

Type: tcl3dOgl
Category: LibrarySpecificDemos
Root: Contents

Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dOgl
Filename: spheres.tcl
Author: Paul Obermeier
Description: Tcl3D demo displaying spheres in various modes.
**Tcl3D: Doing 3D with Tcl**

**Demo:** tcl3dChaos

**Type:** tcl3dOgl

**Category:** LibrarySpecificDemos

**Root:** Contents

---

**Copyright:** © 2007-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

**Module:** Tcl3D -> tcl3dOgl

**Filename:** tcl3dChaos.tcl

**Author:** Paul Obermeier

**Description:** Implementation of algorithm described on Wiki page "Simple Chaos Theory with Tcl" (http://wiki.tcl.tk/11887) using Tcl3D.

Interesting values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Iterations</th>
<th>Revert</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>300</td>
<td></td>
</tr>
<tr>
<td>6300</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.16)
Demo: texanim
Type: tcl3dOgl
Category: LibrarySpecificDemos
Root: Contents

Tcl3D demo: Texture animation with 3D textures.

Tcl3D demo showing the usage of a 3D texture for animation.
In the upper part of the window, a quad is drawn, which shows the actual texture animation.
In the lower half of the window, the 3D texture is visualized as a stack of quads. The sampling of the 3D texture is shown by a quad moving through the texture stack.
Either 4 predefined images can be used as textures or 4 choosable colors.

Author: Paul Obermeier
Date: 2009-01-16
### Demonstration of texture gen

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Original sources available at:
http://zavie.free.fr/opengl/#texturegen

Modified for Tcl3D by Paul Obermeier 2010/11/21

See www.tcl3d.org for the Tcl3D extension.
trislam.tcl

Purpose: Determine performance curves for various methods of pushing triangles and quads through the OpenGL pipeline

Copyright (c) 2004-2006, Geoff Broadwell; this script is released as open source and may be distributed and modified under the terms of either the Artistic License or the GNU General Public License, in the same manner as Perl itself. These licenses should have been distributed to you as part of your Perl distribution, and can be read using `perldoc perlartistic` and `perldoc perlgpl` respectively.

Rewritten in Python by Bob Free

Rewritten and extended for Tcl3D by Paul Obermeier, 2008
This section contains OpenGL demo applications from several resources, that have been ported to Tcl3D. The examples cover OpenGL extension programming. Original sources from different sites. See the documentation for details.

<table>
<thead>
<tr>
<th>Available demos</th>
</tr>
</thead>
<tbody>
<tr>
<td>OglBenchFBO</td>
</tr>
<tr>
<td>PhotoBooth</td>
</tr>
<tr>
<td>extensions</td>
</tr>
<tr>
<td>mandelbrot</td>
</tr>
</tbody>
</table>
**Demo:** OglBenchFBO  
**Type:** tcl3dOglExt  
**Category:** LibrarySpecificDemos  
**Root:** Contents

---

![3D Image](image-url)  
**Ogl_bench v1.0 - Copyright 2007 - Graphcomp**  
**Bob Free bfree@graphcomp.com**  
**http://graphcomp.com/opengl**

---

**Key- Escape Exit**  
**Key-F6 Start benchmark**  
**Key-Space Stop running benchmark**  
**Mouse-1 Rotate teapot**  
**Mouse-2 Rotate textured teapots**  

**Running on Linux with a GeForce FX Go5600/AGP8XSEZ (OpenGL 2.1.1 NVIDIA 180.14.09, Tcl 8.4.18)**
Demo: PhotoBooth

Type: tcl3dOglExt
Category: LibrarySpecificDemos
Root: Contents

Copyright (c) 2007, Libero Spagnolini
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the authors nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
Modifed for Tcl3D by Paul Obermeier 2007/04/14
See www.tcl3d.org for the Tcl3D extension.

The demo has been modified to allow up to 2 parameters to be changed interactively via a slider.
The parameter range of the two sliders can be provided as comment lines at the top of the shader source files.
Further enhancements include:
Loading of image files of any size via the "Load image" button. All image files with an extension of .jpg or .tga in the directory of the script are automatically recognized and inserted into the "Images" labelframe.
Add your own shader without modifying the Tcl script by adding a new file with extension .frag in the directory of the script.

A description of the effect shaders and the original sources are available at http://democracy.org/libero/photobooth/
extensions.tcl

Program to demonstrate the use of extensions.
Extensions used:
GL_ARB_multitexture
GL_EXT_point_parameters
GL_ARB_texture_compression
GL_EXT_texture_edge_clamp

Original C++ code by Dave Astle 2/1/2002
Original files from:
http://www.gamedev.net/reference/programming/features/oglext/demo.zip

Modified for Tcl3D by Paul Obermeier 2005/09/05
See www.tcl3d.org for the Tcl3D extension.
Mandelbrot shader using GPGPU techniques

Author: Gabriel Zachmann, June 2007

The code is derived from ../fbo_demo/saxpy.cpp

The original code can be found at:
http://zach.in.tu-clausthal.de/teaching/cg2_08/downloads/simple_glsl_demos.tar.gz

Modified and extended for Tcl3D by Paul Obermeier 2009/01/04
See www.tcl3d.org for the Tcl3D extension.
This section contains SDL demo applications written in Tcl3D. The examples cover joystick and CD programming with the help of the SDL library.

### Available demos

| cdplayer | joysticktest |
Demo: cdplayer
Type: tcl3dSDL
Category: LibrarySpecificDemos
Root: Contents

Copyright: 2006-2010 Paul Obermeier (obermeier@tcl3d.org)
See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMEOf ALL WARRANTIES.

Module: Tcl3D -> tcl3dSDL
Filename: cdplayer.tcl
Author: Paul Obermeier
Description: Tcl script implementing a simple CD player to test the CD related functions (SDL_CD*) of the Tcl3D SDL wrapping.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>joysticktest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>tcl3dSDL</td>
</tr>
<tr>
<td>Category:</td>
<td>LibrarySpecificDemos</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Joystick 1 (TWIN SHOCK JOYPAD)**

Axes 0 and 1

Axes 2 and 3

Hat 0

---

Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dSDL
Filename: joysticktest.tcl
Author: Paul Obermeier

Description: Tcl script to test the joystick related functions of the Tcl3D SDL wrapping.
<table>
<thead>
<tr>
<th>Type:</th>
<th>tcl3dTogl</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>LibrarySpecificDemos</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

The following demos from the Togl distribution have been ported to Tcl3D. Original sources available at: [http://sourceforge.net/projects/togl/](http://sourceforge.net/projects/togl/)

### Available demos

- [tcl3dDouble](#)
- [tcl3dFont](#)
- [tcl3dGears](#)
- [tcl3dTexture](#)
- [tcl3dToglFonts](#)
A Tcl3D widget demo with two windows, one single buffered and the other double buffered.

This is a version of the original Togl double demo written entirely in Tcl with the help of the Tcl3D package.

Copyright (C) 1996 Brian Paul and Ben Bederson (Original C/Tcl version)
Copyright (C) 2005 Paul Obermeier (Tcl3D version)
See the LICENSE file for copyright details.

Original sources available at: http://sourceforge.net/projects/togl/
Tcl3D: Doing 3D with Tcl

Demo: tcl3dFont
Type: tcl3dTogl
Category: LibrarySpecificDemos
Root: Contents

Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D -> tcl3dTogl
Filename: tcl3dFont.tcl
Author: Paul Obermeier

Description: Tcl script to select a font. The font is displayed in a Tk widget as well as in an OpenGL window. The font name in XLFD notation is shown in a text widget for copy/paste. This demo shows the usage of the "loadbitmapfont" command built into the Togl widget.
Note: The Tk font might look nicer, because font antialiasing is enabled. On Windows this can be toggled in the display property window (Appearance->Effects).

Tcl3D demos at a glance
Version 0.5.0, December 2010
Page 53 of 266
Copyright © 2005-2010 by Paul Obermeier. All rights reserved.
**Demo:** tcl3dGears  
**Type:** tcl3dTogl  
**Category:** LibrarySpecificDemos  
**Root:** Contents

---

**tcl3dGears.tcl**

Test Togl using GL Gears Demo

This is a version of the original Togl gears demo written entirely in Tcl with the help of the Tcl3D package.

Copyright (C) 1997 Philip Quaife (Original C/Tcl version)  
Copyright (C) 2005 Paul Obermeier (Tcl3D version)  
See the LICENSE file for copyright details.

Original sources available at: http://sourceforge.net/projects/togl/
Tcl3D: Doing 3D with Tcl

Tcl3dTexture
Type: tcl3dTogl
Category: LibrarySpecificDemos
Root: Contents

tcl3dTexture.tcl
Togl texture map demo

This is a version of the original Togl texture demo written entirely in Tcl with the help of the Tcl3D package.

Copyright (C) 1996 Brian Paul and Ben Bederson (Original C/Tcl version)
Copyright (C) 2005 Paul Obermeier (Tcl3D version)
See the LICENSE file for copyright details.

Original sources available at: http://sourceforge.net/projects/togl/

Copyright © 2005-2010 by Paul Obermeier. All rights reserved.
**Tcl3D demo: Togl bitmap font specification examples**

```tcl
loadbitmapfont
loadbitmapfont -family courier
loadbitmapfont -family times
loadbitmapfont -family fixed -size 12 -weight medium -slant regular
loadbitmapfont -family fixed -size 12 -weight bold -slant italic
loadbitmapfont -slant xyz
loadbitmapfont -weight xyz
loadbitmapfont -size 20
loadbitmapfont -size 20 -weight bold
loadbitmapfont -size 20 -slant italic
loadbitmapfont --*courier-bold-*--10--*---*---*---*---
loadbitmapfont -family 8x13
loadbitmapfont 8x13
loadbitmapfont -family a-b
loadbitmapfont a-b
loadbitmapfont -family
loadbitmapfont -family-weight-slant (Could not allocate font "-weight")
loadbitmapfont -unknownoption (Could not allocate font "-unknownoption")

Key: Escape Exit
```

Running on Windows NT with a GeForce RX 8500/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.16)

**Copyright:** 2006-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

**Module:** Tcl3D -> tcl3dTogl
**Filename:** tcl3dToglFonts.tcl
**Author:** Paul Obermeier
**Description:** Program demonstrating and testing the different possibilities of specifying a bitmap font for the Togl widget.
<table>
<thead>
<tr>
<th>Category:</th>
<th>Tcl3DSpecificDemos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
<tr>
<td>Types:</td>
<td>rtVis</td>
</tr>
<tr>
<td>Type:</td>
<td>rtVis</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>Category:</td>
<td>Tcl3DSpecificDemos</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Available demos

![rtVis](image-url)
Tcl3D: Doing 3D with Tcl

Demo: rtVis
Type: rtVis
Category: Tcl3DSpecificDemos
Root: Contents

Copyright:    2008-2010 Paul Obermeier (obermeier@tcl3d.org)
See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module:       Tcl3DSpecificDemos
Filename:     rtVis.tcl
Author:       Paul Obermeier
Description:  Ray Tracing visualization program.
The comments of the rtvis* procedures explain how to use the ray-tracing visualization commands.
bytearray.tcl

Tcl3D demo showing the use of the tcl3dByteArray2Vector function, introduced in Version 0.3.
The program texture maps an image generated with Tcl onto a quad.

Author: Paul Obermeier
Date: 2006-02-01
This program creates a checkerboard image in two ways. The first texture is created with an algorithm, as used in some of the RedBook examples (ex. checker.tcl). This algorithm has been converted 1:1 from C to Tcl. Very slow. The second image is created using the Img extension, which is essentially faster.

Author: Paul Obermeier
Date: 2006-09-22
Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D
Filename: imgViewer.tcl
Author: Paul Obermeier
Description: Tcl program to display images and stretch them in realtime with the use of OpenGL textures. The images can be read from files in all formats supported by the Img extension. The stretched image may also be written out to an image file.
Demo: modelViewer

Type:  
Category: Tcl3DSpecificDemos  
Root: Contents

![Tcl3D Model Viewer (a1.obj)](image_url)

Size (x,y,z): (5.42, 5.88, 2.22)

Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D
Filename: modelViewer.tcl
Author: Paul Obermeier
Description: Tcl program to display 3D model files in all formats supported by the Tcl3D extension.
Tcl3D demo: OpenGL execution modes

Tcl3D demo showing 3 possible modes of OpenGL execution:

**Normal mode:** Use the OpenGL functions as wrapped by SWIG. This is the fastest mode. If using an OpenGL function not available in the used driver implementation, this mode will dump core.

**Safe mode:** In this mode every OpenGL function is checked for availability in the driver before execution. If it’s not available, a message is printed out.

**Debug mode:** This mode checks the availability of an OpenGL function like the safe mode, and additionally prints out each OpenGL function before execution.

The program allows to insert an unavailable command in the display callback to see the impact on execution. Currently this command is set to "glFinishTextureSUNX", which is an old, not widely used extension and therefore should not be available in most driver implementations currently in the wild.

Author: Paul Obermeier
Module:       Tcl3DSpecificDemos
Filename:     rtVis.tcl
Author:       Paul Obermeier
Description: Ray Tracing visualization program.
              The comments of the rtvis* procedures explain how to use the ray-tracing visualization commands.
Copyright: 2005-2010 Paul Obermeier (obermeier@tcl3d.org)

See the file "Tcl3D_License.txt" for information on usage and redistribution of this file, and for a DISCLAIMER OF ALL WARRANTIES.

Module: Tcl3D
Filename: tcl3dInfo.tcl
Author: Paul Obermeier

Description: Tcl script to display OpenGL related information. When called without arguments, a window is opened with buttons to display OpenGL information for the following categories:
- General information (-info)
- Available OpenGL commands in Tcl (-cmd)
- Available OpenGL enumerations in Tcl (-enum)

The information texts can also be printed to stdout without opening a GUI, if calling this Tcl script with any of the above listed command line options.
To display all four categories, the option "-all" can be used.

Note: To retrieve all necessary information, an OpenGL context has to be established. So the batch mode needs a DISPLAY, too.
**Demo:** togInCanvas

**Type:**

**Category:** Tcl3DSpecificDemos

**Root:** Contents

---

**Tcl3D demo: Togl window in canvas**

This is the canvas background

Running on Windows NT with a GeForce FX Go6600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)

```
toglInCanvas.tcl
```

Tcl3D demo using a Togl window and some button widgets inserted into a canvas.

**Author:** Paul Obermeier

**Date:** 2006-12-08
Tcl3D demo: Manipulating image vectors (Test 5)

Key-1: Copy: Dest[bw] = Src[bw]
Key-2: Copy: Dest[r,g,b] = Src[r,g,b]
Key-3: Manip: Dest[bw] = -1 * Src(bw) + 255
Key-4: Manip: Dest[r,g,b] = -1 * Src(r,g,b) + 255
Key-5: Swap: Dest[r,g,b] = Src[r,g,b]
Key-Esc: Exit

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)

vectormanip.tcl

Tcl3D demo showing the use of the Vector manipulation functions, introduced in Version 0.3.2.
The program texture maps an image generated with Tcl (the source) onto the left quad. The source texture is manipulated with the vector functions according to the chosen method and mapped onto the right quad. See functions execMethod? below.

Author: Paul Obermeier
Date: 2006-08-15
Several demo applications from Kevin Harris' page have been ported to Tcl3D. The examples cover Cg, GLSL and OpenGL extension programming.

Original sources available at: [http://www.codesampler.com/oglsrc.htm](http://www.codesampler.com/oglsrc.htm)
Tcl3D: Doing 3D with Tcl

ogl_vertex_displacement  oglu_projtexture

Tcl3D demos at a glance  Version 0.5.0, December 2010  Page 73 of 266
Copyright © 2005-2010 by Paul Obermeier. All rights reserved.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>ogl_alpha_blending_framebuffer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>CodeSampler</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

---

Name: ogl_alpha_blending_framebuffer.cpp  
Author: Kevin Harris (kevin@codesampler.com)  
Last Modified: 03/25/05  
Description: This sample demonstrates how to perform alpha-blending in the frame-buffer. The sample renders a textured cube which is alpha-blended into the frame-buffer in such a way as to create a translucent effect.

Control Keys: b - Toggle blending

---

Original C++ code by Kevin Harris (kevin@codesampler.com)  
See www.codesampler.com for the original files  
OpenGL samples page 4: Alpha Blending in the Frame buffer  
http://www.codesampler.com/oglsrc/oglsrc_4.htm#ogl_alpha_blending_framebuffer  
Modified for Tcl3D by Paul Obermeier 2008/05/01  
See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>ogl_alpha_blending_texture</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>CodeSampler</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

---

**Tcl3D demo: CodeSampler’s Texture Alpha Blending**

**Key-Shortcuts:**
- **b** - Toggle blending
- **s** - Toggle usage of cull-mode sorting trick
- **Up Arrow** - Move the test cube closer
- **Down Arrow** - Move the test cube away

---

**Name: ogl_alpha_blending Texture.cpp**

**Author:** Kevin Harris (kevin@codesampler.com)

**Last Modified:** 03/25/05

**Description:** This sample demonstrates how to perform alpha blending using the alpha channel of a standard .tga texture. For proper alpha blending, the sample uses a cull-mode sorting trick to ensure the sides of the textured cube get rendered in back-to-front order.

**Control Keys:**
- **b** - Toggle blending
- **s** - Toggle usage of cull-mode sorting trick
- **Up Arrow** - Move the test cube closer
- **Down Arrow** - Move the test cube away

---

Original C++ code by Kevin Harris (kevin@codesampler.com)

See www.codesampler.com for the original files

OpenGL samples page 3: Alpha Texture Blending

[http://www.codesampler.com/oglsrc/oglsrc_3.htm#ogl_alpha_blending_texture](http://www.codesampler.com/oglsrc/oglsrc_3.htm#ogl_alpha_blending_texture)

Modified for Tcl3D by Paul Obermeier 2008/05/01

See www.tcl3d.org for the Tcl3D extension.

---
<table>
<thead>
<tr>
<th>Demo:</th>
<th>ogl_axis_aligned_billboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>CodeSampler</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

---

**Name:** ogl_axis_aligned_billboard.cpp  
**Author:** Kevin Harris (kevin@codesampler.com)  
**Last Modified:** 02/01/05  
**Description:** An example of axis aligned billboarding.

**Control Keys:**  
- **F1** - Toggle billboarding  
- **Up** - View moves forward  
- **Down** - View moves backward  
- **Left** - View strafes left  
- **Right** - View strafes right  
- **Left Mouse** - Perform looking  
- **Mouse** - Look about the scene

---

Original C++ code by Kevin Harris (kevin@codesampler.com)  
See www.codesampler.com for the original files  
OpenGL samples page 8: Axis-Aligned Billboards

Modified for Tcl3D by Paul Obermeier 2007/03/10  
See www.tcl3d.org for the Tcl3D extension.
Name: ogl_benchmark_sphere.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 04/21/05
Description: Renders a textured sphere using either Immediate Mode calls, Immediate Mode calls cached in a Display List, or as a collection of geometric data stored in an interleaved fashion within a Vertex Array.

Control Keys: Left Mouse Button - Spin the view.
F1 - Decrease sphere precision.
F2 - Increase sphere precision.
F3 - Use Immediate mode
F4 - Use a Display List
F5 - Use a Vertex Array
F6 - Perform Benchmarking
F7 - Toggle wire-frame mode.

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 9: Benchmarking Test App
Modified for Tcl3D by Paul Obermeier 2005/11/07
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: ogl_cg_multitexture
Type: CodeSampler
Category: TutorialsAndBooks
Root: Contents

---

**Name: ogl_cg_multitexture.cpp**
**Author: Kevin Harris**
**Last Modified: 04/26/05**
**Description:** This sample demonstrates how to blend two textures together with Cg using either OpenGL's native multi-texture support (using semantics) or by using Cg's special texture functions: cgGLSetTextureParameter, cgGLEnableTextureParameter, and cgGLDisableTextureParameter.

---

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 10: Multi-Texturing with Cg

Modified for Tcl3D by Paul Obermeier 2007/05/22
See www.tcl3d.org for the Tcl3D extension.

The original demo has been extended with a little GUI to allow switching between the two call semantics at runtime. To visualize, that a different shader program is active, the OpenGL semantics shader adds only half of the checker image color.
Demo: ogl_color_tracking

Type: CodeSampler
Category: TutorialsAndBooks
Root: Contents

Name: ogl_color_tracking.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 04/28/05
Description: This sample demonstrates color-tracking and two-sided lighting in OpenGL.

Color tracking allows us to substitute the color of our vertices for one or more of the material colors used by OpenGL's lighting equation. This feature is typically not used much anymore as since modelers today use textures to color their geometry - not vertex colors. Of course, this technique is alive and kicking in a billion lines of legacy code so it's good to understand this technique just in case you run across it.

Two-sided lighting basically means that we want OpenGL to light both sides of our geometry instead of just the front faces. Again, this feature is typically not used much anymore since it's very inefficient to light both sides of every triangle but there are some cases where this is helpful to know.
Control Keys: c - Toggle between a material color or color tracking the vertices
l - Toggle two-sided lighting

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 5: Color Tracking and Two-Sided lighting
http://www.codesampler.com/oglsrc/oglsrc_5.htm#ogl_color_tracking

Modified for Tcl3D by Paul Obermeier 2008/05/01
See www.tcl3d.org for the Tcl3D extension.
Name: ogl_fps_controls.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/01/05
Description: This sample demonstrates how to collect user input and build a custom view matrix for First Person Shooter style controls.

Control Keys:
- Up: View moves forward
- Down: View moves backward
- Left: View strafes left
- Right: View strafes right
- Left Mouse: Perform looking
- Mouse: Look about the scene
- Home: View moves up
- End: View moves down

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 5: First Person Shooter Controls
Modified for Tcl3D by Paul Obermeier 2005/11/05
See www.tcl3d.org for the Tcl3D extension.
This sample demonstrates how to create dynamic textures through off-screen rendering. The off-screen rendering step is accomplished using a frame-buffer and render-buffer object, which is created using OpenGL's EXT_framebuffer_object extension.

As a demonstration, a spinning textured cube is rendered to a frame-buffer object, which is in turn, used to create a dynamic texture. The dynamic texture is then used to texture a second spinning cube, which will be rendered to the application's window.

Control Keys: Left Mouse Button - Spin the large, black cube.
Right Mouse Button - Spin the textured cube being rendered into the p-buffer.

Note: The EXT_framebuffer_object extension is an excellent replacement for...
the WGL_ARB_pbuffer and WGL_ARB_render_texture combo which is normally used to create dynamic textures. An example of this older technique can be found here:

http://www.codesampler.com/oglsrc/oglsrc_7.htm#ogl_offscreen_rendering

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 14: Off-screen Rendering Using Frame-Buffer Objects

Modified for Tcl3D by Paul Obermeier 2007/02/25
See www.tcl3d.org for the Tcl3D extension.
Name: ogl_glslang_simple_vs2ps.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 04/21/05
Description: This sample demonstrates how to write vertex and fragment shaders using OpenGL's new high-level shading language GLslang.

Control Keys: F1 - Toggle usage of vertex and fragment shaders.

Note: The fragment shader has been changed slightly from what the fixed-function pipeline does by default so you can see a noticeable change when toggling the shaders on and off. Instead of modulating the vertex color with the texture's texel, the fragment shader adds the two together, which causes the fragment shader to produce a brighter, washed-out image. This modification can be switched back in the fragment shader file.

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 10: Simple Vertex & Fragment Shader (GLslang)
Modified for Tcl3D by Paul Obermeier 2005/11/05
See www.tcl3d.org for the Tcl3D extension.
Name: ogl_lighting.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/01/05
Description: This sample demonstrates the three basic types of lights that are available in OpenGL: directional, spot, and point.

Control Keys: l - Changes the light's type
w - Toggles wire frame mode

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 5: Lighting
http://www.codesampler.com/oglsrc/oglsrc_5.htm#ogl_lighting

Modified for Tcl3D by Paul Obermeier 2008/05/01
See www.tcl3d.org for the Tcl3D extension.
**Demo:** ogl_material

**Type:** CodeSampler

**Category:** TutorialsAndBooks

**Root:** Contents

---

**Name:** ogl_material.cpp

**Author:** Kevin Harris (kevin@codesampler.com)

**Last Modified:** 04/28/05

**Description:** This sample demonstrates how to use materials with lighting to produce different surface effects.

**Control Keys:** Left Mouse Button - Spin the view

---

**Original C++ code by Kevin Harris (kevin@codesampler.com)**

See www.codesampler.com for the original files

OpenGL samples page 5: Materials
http://www.codesampler.com ogl_material

**Modified for Tcl3D by Paul Obermeier 2008/04/28**

See www.tcl3d.org for the Tcl3D extension.
Name: ogl_multitexture_blending.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/08/05
Description: This sample demonstrates how to use the OpenGL extensions GL_ARB_multitexture and GL_ARB_texture_env_combine in conjunction with specially encoded vertex colors to blend three textures together.

This technique is very popular in terrain rendering engines which use it to blend dramatically different textures such as rock and grass together with out creating a noticeable edge. For example, with three textures consisting of stone, grass, and sand you can render a mountain that blends in patches of grass and sand at its base.

Of course, while this technique remains popular as a fall-back for older hardware, shaders make this task a lot easier and are quickly becoming the preferred method for terrain texture blending.

The technique basically consists of the following steps:
Step 1: Take the desired contribution of the three textures and encode them into the vertex's color such that the RGB portion of the color controls the interpolation between texture stages 0 and 1, and the color's ALPHA controls the interpolation between texture stages 1 and 2.

Step 2: Use GL_ARB_multitexture to apply three textures simultaneously to our geometry.

Step 3: Set the first texture on texture stage 0.

Step 4: During texture stage 1, use GL_INTERPOLATE_ARB to linearly interpolate between the output of stage 0 and the texture of stage 1 with GL_SRC_COLOR (i.e. the RGB part of the color).

Step 4: During texture stage 2, use GL_INTERPOLATE_ARB to linearly interpolate between the output of stage 1 and the texture of stage 2 with GL_SRC_ALPHA (i.e. the ALPHA part of the color).

Control Keys: F1 - Increase contribution of texture 0
              F2 - Decrease contribution of texture 0
              F3 - Increase contribution of texture 2
              F4 - Decrease contribution of texture 2
              F5 - Toggle wire-frame mode.
              Up  - View moves forward
              Down - View moves backward

Note: I tried to create an intuitive way to set the contribution of each texture at run-time using the function keys, but this system is still a little confusing since I only allow the contribution of texture 0 and texture 2 to be adjusted. This is due to the fact that the equation for encoding the blending info into the vertex color simply infers the contribution value of texture 1 based on the values for textures 0 and 2. Therefore, the contribution value of texture 1 must be indirectly set by adjusting the contributions of textures 0 and 2.

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 4: Multi-Texture Blending

Modified for Tcl3D by Paul Obermeier 2007/03/10
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: ogl_near_far_clip

Type: CodeSampler
Category: TutorialsAndBooks
Root: Contents

Name: ogl_near_far_clip.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/01/05
Description: This sample demonstrates how adjustments to OpenGL's near and far clip planes effect the view.

Control Keys:
- Up: View moves forward
- Down: View moves backward
- Left: View strafes left
- Right: View strafes right
- Left Mouse: Perform looking
- Mouse: Look about the scene
- F1: Increase near clip value
- F2: Decrease near clip value
- F3: Increase far clip value
- F4: Decrease far clip value

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 2: Near/Far Clipping Plane
**Demo:**
ogl_occlusion_query

**Type:** CodeSampler

**Category:** TutorialsAndBooks

**Root:** Contents

---

**Tcl3D demo: CodeSampler's Occlusion Query using the ARB extension**

- **Key-Escape Exit**
- **Running on Linux with a GeForce FX 5600/AGP/SE2 (OpenGL 2.1, NVIDIA 100.14.09, Tcl 0.4.10)**

---

**Name:** ogl_occlusion_query_arb.cpp

**Author:** Kevin Harris (kevin@codesampler.com)

**Last Modified:** 02/01/05

**Description:** This sample demonstrates how to use OpenGL's new extension, ARB_occlusion_query and NV_occlusion_query.

**Control Keys:** Left Mouse Button - Spin the view

---

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 7: Occlusion Query

Modified for Tcl3D by Paul Obermeier 2007/03/10
See www.tcl3d.org for the Tcl3D extension.

This sample integrates ARB_occlusion_query and NV_occlusion_query code into one file.
If called with no command line arguments, it uses the ARB_occlusion_query extension.
Use "nv" as parameter to use the NV_occlusion_query extension.
## Demo: ogl_planar_shadow

**Type:** CodeSampler  
**Category:** TutorialsAndBooks  
**Root:** Contents

---

**Name:** ogl_planar_shadow.cpp  
**Author:** Kevin Harris (kevin@codesampler.com)  
**Last Modified:** 02/01/05  
**Description:** This sample demonstrates how to create planar shadows under OpenGL.

Planar shadows are created by building a special projection matrix which flattens an object's geometry into a plane when rendered.

If the plane, which the geometry is flattened into, matches up with another planar surface like a floor or a wall, the flattened geometry can be made to resemble a shadow on that surface.

**Control Keys:**  
- **Up** - Light moves up  
- **Down** - Light moves down  
- **Left** - Light moves left  
- **Right** - Light moves right  
- Left Mouse Button - Spin the view  
- Right Mouse Button - Spin the teapot

---

Running on Linux with a GeForce FX Go5600/AGP/SSE2 (OpenGl 2.1.1, NVIDIA 100.14.09, Tcl 0.4.10)
Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 7: Planar Shadows
http://www.codesampler.com/oglsrc/oglsrc_7.htm#ogl_planar_shadow

Modified for Tcl3D by Paul Obermeier 2008/05/02
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

**Demo:** ogl_point_rotated_billboard

**Type:** CodeSampler

**Category:** TutorialsAndBooks

**Root:** Contents

---

**Key-Escape**  Exit  
**Key-P1**  Toggle billboarding  
**Key-Up|Down**  View moves forward|backward  
**Key-Left|Right**  View strafes to the left|right  
**Key-Home|End**  View elevates up|down  
**Billboarding is on**  

---

**Control Keys:**  
F1 - Toggle billboarding  
Up - View moves forward  
Down - View moves backward  
Left - View strafes left  
Right - View strafes right  
Left Mouse - Perform looking  
Mouse - Look about the scene  

---

**Name:** ogl_point_rotated_billboard.cpp  
**Author:** Kevin Harris (kevin@codesampler.com)  
**Last Modified:** 02/01/05  
**Description:** An example of point rotated billboarding.

---

Original C++ code by Kevin Harris (kevin@codesampler.com)  
See www.codesampler.com for the original files  
OpenGL samples page 8: Point-Rotated Billboards

Modified for Tcl3D by Paul Obermeier 2007/03/10  
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: ogl_point_sprites
Type: CodeSampler
Category: TutorialsAndBooks
Root: Contents

Name: ogl_point_sprites.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/01/05
Description: This sample demonstrates how to create point sprites using OpenGL's new GL_ARB_point_sprite extension, which can be used to create point-rotated billboards on the GPU.

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 6: Point Sprites

Modified for Tcl3D by Paul Obermeier 2005/11/08
See www.tcl3d.org for the Tcl3D extension.
Demo: ogl_polygon_offset

Type: CodeSampler
Category: TutorialsAndBooks
Root: Contents

---

Name: ogl_polygon_offset.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/01/05
Description: This sample demonstrates how to eliminate z-fighting when rendering polygons directly on top of other polygons.

Control Keys: Left Mouse Button - Spin the view
F1 - Increase Offset Factor
F2 - Decrease Offset Factor
F3 - Increase Offset Unit
F4 - Decrease Offset Unit

---

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 5: Polygon Offset

Modified for Tcl3D by Paul Obermeier 2007/03/05
See www.tcl3d.org for the Tcl3D extension.

See http://www.opengl.org/sdk/docs/man/xhtml/glPolygonOffset.xml for the glPolygonOffset command.
### Demo: ogl_skinning

**Type:** CodeSampler  
**Category:** TutorialsAndBooks  
**Root:** Contents

---

#### Name: ogl_cg_skinning.cpp ogl_glslang_skinning.cpp  
**Author:** Kevin Harris (kevin@codesampler.com)  
**Last Modified:** 04/28/05  
**Description:** This sample demonstrates how to skin a mesh on the hardware using a Cg or GLSL shader. To keep things simple, the skeletal system used in this sample is very simple and only consists of two bones or bone matrices.

Special thanks go out to Cyril Zeller, and Matthias Wloka of nVIDIA for their help in straightening out a few oddities that my sample was suffering from. In short, Cg works fine and I'm occasionally a big dummy! ;)

**Control Keys:**  
- Left Mouse Button - Spin the matrix for bone 0.  
- Right Mouse Button - Spin the matrix for bone 1.  
- Key-s - Start|Stop animation.  
- Key-Up|Down - Increase|Decrease distance.  
- Key-F1 - Toggle test geometry.  
- Key-F2 - Toggle wire-frame mode.  

---

Running on Linux with a GeForce FX Go5800/AGP/SE2 (OpenGL 2.1.1 NVIDIA 100.14.09, Tcl 0.4.10)

---

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 11: Matrix Palette Skinning on the Hardware

Modified for Tcl3D by Paul Obermeier 2005/11/05
See www.tcl3d.org for the Tcl3D extension.

This sample integrates Cg and GLSL code into one file.
If called with no command line arguments, it uses the Cg shader.
Use "glsl" as parameter to use the GLSL shader.
Demo: ogl_texture_addressing

Type: CodeSampler
Category: TutorialsAndBooks
Root: Contents

Name: ogl_texture_addressing.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 02/01/05
Description: This sample demonstrates the two methods of texture addressing that are available under OpenGL:

GL_REPEAT
GL_CLAMP
GL_MIRRORED_REPEAT_ARB ( GL_ARB_texture_mirrored_repeat )
GL_CLAMP_TO_BORDER_ARB ( GL_ARB_texture_border_clamp )
GL_CLAMP_TO_EDGE ( GL_SGIS_texture_edge_clamp )

Control Keys: F1 - Changes addressing method for the S coordinates
F2 - Changes addressing method for the T coordinates

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 3: Texture Addressing

Modified for Tcl3D by Paul Obermeier 2007/03/06
See www.tcl3d.org for the Tcl3D extension.
Name: ogl_cg_vertex_displacement.cpp
ogl_glslang_vertex_displacement.cpp
Author: Kevin Harris (kevin@codesampler.com)
Last Modified: 04/21/05
Description: This sample demonstrates how to perform mesh deformation or vertex displacement with OpenGL using a Cg or GLSL shader.

Control Keys: F1 - Increase flag motion
F2 - Decrease flag motion
F3 - Toggle wire-frame mode

Original C++ code by Kevin Harris (kevin@codesampler.com)
See www.codesampler.com for the original files
OpenGL samples page 11: Vertex Displacement or Mesh Deformation Shader

Modified for Tcl3D by Paul Obermeier 2005/11/05
See www.tcl3d.org for the Tcl3D extension.

This sample integrates the Cg and GLSL code into one file.
If called with no command line arguments, it uses the GLSL shader.
Use "cg" as command line parameter to use the Cg shader.
This program demonstrates how one would go about doing a projected texture. The sample here shows how a projected texture technique can be used to produce a light map.

The point is that even though you have very few vertices available for the fixed function pipeline lighting solution, you can achieve nice per pixel lighting even though the surface has only a handful of vertices.

This sample draws a cube, only allowing the inside being visible via culling front facing polys, and then projects the light map texture on the second texture stage all through the fixed function pipeline. The left mouse button will move the cube around and the right mouse button will move the projected light map around.
Several demo applications from Vahid Kazemi's page have been ported to Tcl3D. Original sources available at: [http://www.GameProgrammer.org](http://www.GameProgrammer.org)

### Available demos

<table>
<thead>
<tr>
<th>GL_Blanding</th>
<th>GL_Envmap</th>
<th>GL_Font</th>
<th>GL_Lighting</th>
</tr>
</thead>
<tbody>
<tr>
<td>GL_Motionblur</td>
<td>GL_Primitives</td>
<td>GL_Shadow</td>
<td>GL_Texturing</td>
</tr>
<tr>
<td>GL_Vewing</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Tcl3D demos at a glance**

Version 0.5.0, December 2010

Copyright © 2005-2010 by Paul Obermeier. All rights reserved.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>GL_Blending</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>GameProgrammer</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**GL_Blending.tcl**

Tutorial from www.GameProgrammer.org

Blending demo

Original code Copyright 2005 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/12

See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>GL_Envmap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>GameProgrammer</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**GL_Texturing.tcl**

Tutorial from www.GameProgrammer.org
Using Textures

Original code Copyright 2004 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/12
See www.tcl3d.org for the Tcl3D extension.
GL_Font.tcl

Tutorial from www.GameProgrammer.org
Bitmap fonts

Original code Copyright 2005 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/15
See www.tcl3d.org for the Tcl3D extension.
**Demo:** GL_Lighting

**Type:** GameProgrammer

**Category:** TutorialsAndBooks

**Root:** Contents

---

**GL_Lighting.tcl**

Tutorial from www.GameProgrammer.org

Turn the lights on!

Original code Copyright 2004 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/11

See www.tcl3d.org for the Tcl3D extension.

---

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)

Key-Escape  Exit  
Key-Up|Down Increase|Decrease rotation speed
Mouse-1|2  Start|Stop animation
### TCL3D Demo: GL_Motionblur

**Type:** GameProgrammer  
**Category:** TutorialsAndBooks  
**Root:** Contents

#### GL_Motionblur.tcl

Tutorial from www.GameProgrammer.org  
Using Textures

Original code Copyright 2006 by Vahid Kazemi

Modified for TCL3D by Paul Obermeier 2006/09/14

See www.tcl3d.org for the TCL3D extension.
GL_Primitives.tcl

Tutorial from www.GameProgrammer.org
OpenGL Primitives.

Original code Copyright 2004 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/11
See www.tcl3d.org for the Tcl3D extension.
### GL_Shadow

**Demo:** GL_Shadow

**Type:** GameProgrammer

**Category:** TutorialsAndBooks

**Root:** Contents

#### Key-Train Information
- Key-Escape: Exit
- Key-Up/Down: Increase/Decrease rotation speed
- Mouse-1/2: Start/Stop animation

Running on Windows NT with a GeForce FX 5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)

---

**GL_Shadow.tcl**

Tutorial from www.GameProgrammer.org

Stencil shadows.

Original code Copyright 2005 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/10

See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>GL_Texturing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>GameProgrammer</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**GL_Texturing.tcl**

Tutorial from [www.GameProgrammer.org](http://www.GameProgrammer.org)

Using Textures

Original code Copyright 2004 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/12

See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
GL_Viewing.tcl

Tutorial from www.GameProgrammer.org
Viewing and Transformations.

Original code Copyright 2004 by Vahid Kazemi

Modified for Tcl3D by Paul Obermeier 2006/09/11
See www.tcl3d.org for the Tcl3D extension.
Some of the NeHe OpenGL tutorials have been ported to run with Tcl3D. Currently 34 out of 48 lessons are available.
Original sources available at: http://nehe.gamedev.net/
Lesson01.tcl

NeHe's OpenGL Framework

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
NeHe's First Polygon Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson03.tcl

NeHe's Color Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>Lesson04</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>NeHe</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Tcl3D demo: NeHe's Rotation Tutorial (Lesson 4)**

- **Key-Escape Exit**
- **Key-F1**  Toggle window mode
- **Key-F12**  Create PDF file
- **Mouse-1|2**  Start|Stop animation

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)

Lesson04.tcl

NeHe's Rotation Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson05.tcl

NeHe's Solid Object Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson06.tcl

NeHe's Texture Mapping Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson07.tcl

NeHe's Textures, Lighting & Keyboard Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson08.tcl

Tom Stanis & NeHe's Blending Tutorial

This Code Was Created By Tom Stanis / Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
NeHe's Animated Blended Textures Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson10.tcl

Lionel Brits & NeHe's 3D World Tutorial

This Code Was Created By Lionel Brits & Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
### Lesson11

**Demo:** Lesson11

**Type:** NeHe

**Category:** TutorialsAndBooks

**Root:** Contents

---

**Lesson11.tcl**

bosco & NeHe's Waving Texture Tutorial

This Code Was Created By bosco / Jeff Molofee 2000  
A HUGE Thanks To Fredric Echols For Cleaning Up  
And Optimizing This Code, Making It More Flexible!  
If You've Found This Code Useful, Please Let Me Know.  
Visit My Site At nehe.gamedev.net  

Modified for Tcl3D by Paul Obermeier 2006/01/25  
See www.tcl3d.org for the Tcl3D extension.
Lesson12.tcl

NeHe's Display List Tutorial

This Code Was Created By Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: Lesson13
Type: NeHe
Category: TutorialsAndBooks
Root: Contents

Tcl3D demo: NeHe's Bitmap Font Tutorial (Lesson 13)

Active OpenGL Text With NeHe -

Key-Esc= Exit
Key-F1  Toggle window mode
Mouse-1/2 Start/Stop animation

Running on Windows NT with a GeForce FX Go6600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)

Lesson13.tcl

NeHe's Bitmap Font Tutorial

This Code Was Created By Jeff Molofee 2000
Modified by Shawn T. to handle (%3.2f, num) parameters.
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing The Base Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
This Code Was Created By Jeff Molofee 2000
Modified by Shawn T. to handle (%3.2f, num) parameters.
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing The Base Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/26
See www.tcl3d.org for the Tcl3D extension.
Lesson16.tcl

Chris Aliotta & NeHe's Fog Tutorial

This Code Was Created By Christopher Aliotta & Jeff Molofee 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson18.tcl

NeHe & TipTup's Quadratics Tutorial

This Code Was Created By Jeff Molofee and GB Schmick 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing This Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit Our Sites At www.tiptup.com and nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/01/25
See www.tcl3d.org for the Tcl3D extension.
Lesson19.tcl

NeHe's Particle Tutorial

This Code Was Created By Jeff Molofee 2000
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/03/14
See www.tcl3d.org for the Tcl3D extension.
Lesson20.tcl

NeHe's Masking Tutorial

This Code Was Created By Jeff Molofee 2000
And Modified By Giuseppe D'Agata (waveform@tiscalinet.it)
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/03/14
See www.tcl3d.org for the Tcl3D extension.
Lesson21.tcl

NeHe's Line Tutorial

This Code Was Created By Jeff Molofee 2000
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/03/14
See www.tcl3d.org for the Tcl3D extension.
Lesson22.tcl

NeHe's GL_ARB_multitexture & Bump Mapping Tutorial

This Code Was Created by Jens Schneider (WizardSoft) 2000
Lesson22 to the series of OpenGL tutorials by NeHe-Production

This Code is loosely based upon Lesson06 by Jeff Molofee.
contact me at: schneide@pool.informatik.rwth-aachen.de

Basecode Was Created By Jeff Molofee 2000
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/16
See www.tcl3d.org for the Tcl3D extension.
Lesson23.tcl

NeHe & TipTup's Environment Mapping Tutorial

This Code Was Created By Jeff Molofee and GB Schmick 2000
A HUGE Thanks To Fredric Echols For Cleaning Up
And Optimizing The Base Code, Making It More Flexible!
If You've Found This Code Useful, Please Let Me Know.
Visit Our Sites At www.tiptup.com and nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/27
See www.tcl3d.org for the Tcl3D extension.
**Demo:** Lesson24

<table>
<thead>
<tr>
<th>Type</th>
<th>NeHe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root</td>
<td>Contents</td>
</tr>
</tbody>
</table>

---

**Lesson24.tcl**

NeHe's Token, Extensions, Scissoring & TGA Loading Tutorial

This Code Was Created By Jeff Molofee 2000
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/25
See www.tcl3d.org for the Tcl3D extension.
Lesson25.tcl

Piotr Cieslak & NeHe's Morphing Points Tutorial

This Code Was Created By Pet & Commented/Cleaned Up By Jeff Molofee
If You've Found This Code Useful, Please Let Me Know.
Visit NeHe Productions At http://nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2007/03/03
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Lesson26.tcl

Banu Octavian & NeHe's Stencil & Reflection Tutorial

This code has been created by Banu Octavian aka Choko - 20 may 2000 and uses NeHe tutorials as a starting point (window initialization, texture loading, GL initialization and code for keypresses) - very good tutorials, Jeff. If anyone is interested about the presented algorithm please e-mail me at boct@romwest.ro

Code Commenting And Clean Up By Jeff Molofee (NeHe)
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/16
See www.tcl3d.org for the Tcl3D extension.
Lesson27.tcl

"Banu Octavian & NeHe's Shadow Casting Tutorial"

This code has been created by Banu Octavian aka Choko - 20 may 2000
and uses NeHe tutorials as a starting point (window initialization,
texture loading, GL initialization and code for keypresses) - very good
tutorials, Jeff. If anyone is interested about the presented algorithm
please e-mail me at boct@romwest.ro
Attention!!! This code is not for beginners.

Modified for Tcl3D by Paul Obermeier 2007/02/27
See www.tcl3d.org for the Tcl3D extension.
Lesson28.tcl

David Nikdel & NeHe's Bezier Tutorial

This Code Was Published By Jeff Molofee 2000
Code Was Created By David Nikdel For NeHe Productions
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/29
See www.tcl3d.org for the Tcl3D extension.
NeHe & Evan 'terminate' Pipho's TGA Loading Tutorial

Loading Uncompressed and Compressed .TGA Files with the Img extension.

This Code Was Created By Evan Pipho
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/16
See www.tcl3d.org for the Tcl3D extension.
Lesson36.tcl

Dario Corno's Radial Blur & Rendering To A Texture Tutorial

If You've Found This Code Useful, Please Let Me Know. Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/23
See www.tcl3d.org for the Tcl3D extension.
### Sami Hamlaoui's Cel-Shading Code

Note: The original article for this code can be found at:
http://www.gamedev.net/reference/programming/features/celshading

If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/22
See www.tcl3d.org for the Tcl3D extension.
NeHe's Volumetric Fog Tutorial

This Code Was Created By Jeff Molofee 2003
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/27
See www.tcl3d.org for the Tcl3D extension.
Lesson45.tcl

Paul Frazee's Vertex Buffer Object Tutorial

Code Commenting And Clean Up By Jeff Molofee (NeHe)
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/17
See www.tcl3d.org for the Tcl3D extension.
Lesson46.tcl

NeHe & MainRoach's FSAA Tutorial

This Code Was Created By Jeff Molofee 2001
and Colt McAnlis ( MainRoach ).
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/13
See www.tcl3d.org for the Tcl3D extension.

This demo uses the multisampling options built into tcl3dTogl starting from version 0.3.2.
Another way to set the number of samples is via the driver specific GUI under Windows, or by setting the environment variable __GL_FSAA_MODE under Linux.
**Tcl3D** demo: NeHe & Owen Bourne's Cg Vertex Shader Tutorial (Lesson 47)

Lesson47.tcl

NeHe & Owen Bourne's Cg Vertex Shader Tutorial

If You've Found This Code Useful, Please Let Me Know. Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/09/05

See www.tcl3d.org for the Tcl3D extension.
Lesson48.tcl

NeHe & Terence J. Grant's ArcBall Rotation Tutorial

Authors Name: Terence J. Grant

NeHe Productions 1997-2004
If You've Found This Code Useful, Please Let Me Know.
Visit My Site At nehe.gamedev.net

Modified for Tcl3D by Paul Obermeier 2006/08/31
See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Type:</th>
<th>Nopper</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Available demos

- Example02
- Example03
- Example04
- Example05
OpenGL 3.3 with GLEW - Example 02

@author Norbert Nopper norbert@nopper.tv
@version 1.0

Homepage: http://nopper.tv

Copyright Norbert Nopper

Modified for Tcl3D by Paul Obermeier 2010/09/01
See www.tcl3d.org for the Tcl3D extension.
OpenGL 3.3 with GLEW - Example 03

@author Norbert Nopper norbert@nopper.tv
@version 1.0

Homepage: http://nopper.tv

Copyright Norbert Nopper

Modified for Tcl3D by Paul Obermeier 2010/09/01
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: Example04
Type: Nopper
Category: TutorialsAndBooks
Root: Contents

OpenGL 3.3 with GLEW - Example 04
@author Norbert Nopper norbert@nopper.tv
@version 1.0
Homepage: http://nopper.tv
Copyright Norbert Nopper
Modified for Tcl3D by Paul Obermeier 2010/09/01
See www.tcl3d.org for the Tcl3D extension.
OpenGL 3.3 with GLEW - Example 05

@author Norbert Nopper norbert@nopper.tv
@version 1.0

Homepage: http://nopper.tv

Copyright Norbert Nopper

Modified for Tcl3D by Paul Obermeier 2010/09/01
See www.tcl3d.org for the Tcl3D extension.
The Redbook describing OpenGL Version 1.4 contains 72 examples written in C. 67 of them have been successfully converted into equivalent Tcl3D scripts and the results compared on several operating systems and computers against the C version. Three of the missing five examples (surfpoints, tess, tesswin) deal with tesselation, which is currently not supported. The other two test programs (aaindex, fogindex) not yet ported deal with color index mode, which is not yet implemented in the tcl3dTogl widget.

<table>
<thead>
<tr>
<th>combiner</th>
<th>convolution</th>
<th>cube</th>
<th>cubemap</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
</tr>
<tr>
<td>dof</td>
<td>double</td>
<td>drawf</td>
<td>feedback</td>
</tr>
<tr>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
<td><img src="image8.png" alt="Image" /></td>
</tr>
<tr>
<td>fog</td>
<td>fogcoord</td>
<td>font</td>
<td>hello</td>
</tr>
<tr>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
</tr>
<tr>
<td>histogram</td>
<td>image</td>
<td>light</td>
<td>lines</td>
</tr>
<tr>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
<td><img src="image15.png" alt="Image" /></td>
<td><img src="image16.png" alt="Image" /></td>
</tr>
<tr>
<td>list</td>
<td>material</td>
<td>minmax</td>
<td>mipmap</td>
</tr>
<tr>
<td><img src="image17.png" alt="Image" /></td>
<td><img src="image18.png" alt="Image" /></td>
<td><img src="image19.png" alt="Image" /></td>
<td><img src="image20.png" alt="Image" /></td>
</tr>
<tr>
<td>model</td>
<td>movelight</td>
<td>multisamp</td>
<td>multtex</td>
</tr>
<tr>
<td><img src="image21.png" alt="Image" /></td>
<td><img src="image22.png" alt="Image" /></td>
<td><img src="image23.png" alt="Image" /></td>
<td><img src="image24.png" alt="Image" /></td>
</tr>
<tr>
<td>mvarray</td>
<td>pickdepth</td>
<td>picksquare</td>
<td>planet</td>
</tr>
<tr>
<td>pointp</td>
<td>polyoff</td>
<td>polys</td>
<td>quadric</td>
</tr>
<tr>
<td>-------</td>
<td>---------</td>
<td>-------</td>
<td>---------</td>
</tr>
<tr>
<td>robot</td>
<td>scene</td>
<td>select</td>
<td>shadowmap</td>
</tr>
<tr>
<td>smooth</td>
<td>stencil</td>
<td>stroke</td>
<td>surface</td>
</tr>
<tr>
<td>teapots</td>
<td>texbind</td>
<td>texgen</td>
<td>texprox</td>
</tr>
<tr>
<td>texsub</td>
<td>texture3d</td>
<td>texturesurf</td>
<td>torus</td>
</tr>
<tr>
<td>trim</td>
<td>unproject</td>
<td>varray</td>
<td>wrap</td>
</tr>
</tbody>
</table>
### Tcl3D demo: OpenGL Red Book example aapoly

**aapoly.tcl**

An example of the OpenGL red book modified to work with Tcl3D.

The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program draws filled polygons with antialiased edges. The special GL_SRC_ALPHA_SATURATE blending function is used.
Pressing the 't' key turns the antialiasing on and off.

---

<table>
<thead>
<tr>
<th>Demo:</th>
<th>aapoly</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>RedBook</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)
## Tcl3D: Doing 3D with Tcl

**Demo:** aapolyStride

**Type:** RedBook

**Category:** TutorialsAndBooks

**Root:** Contents

### aapoly.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program draws filled polygons with antialiased edges. The special GL_SRC_ALPHA_SATURATE blending function is used.
Pressing the 't' key turns the anti-aliasing on and off.
aargb.tcl

An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program draws shows how to draw anti-aliased lines. It draws two diagonal lines to form an X; when 'r' is typed in the window, the lines are rotated in opposite directions.
accanti.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Use the accumulation buffer to do full-scene antialiasing
on a scene with orthographic parallel projection.
accpersp.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Use the accumulation buffer to do full-scene antialiasing on a scene with perspective projection, using the special routines accFrustum() and accPerspective().
alpha.tcl

An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program draws several overlapping filled polygons to demonstrate the effect order has on alpha blending results. Use the 't' key to toggle the order of drawing polygons.
alpha3D.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates how to intermix opaque and alpha blended polygons in the same scene, by using glDepthMask. Press the 'a' key to animate moving the transparent object through the opaque object. Press the 'r' key to reset the scene.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program uses evaluators to draw a Bezier curve.
bezmesh.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program renders a lighted, filled Bezier surface,
using two-dimensional evaluators.
bezsurf.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program renders a wireframe Bezier surface, using two-dimensional evaluators.
Tcl3D: Doing 3D with Tcl

An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

Demonstrate the different blending functions available with the OpenGL imaging subset. This program demonstrates use of the glBlendEquation call.

The following keys change the selected blend equation function:

'a' -> GL_FUNC_ADD
's' -> GL_FUNC_SUBTRACT
'r' -> GL_FUNC_REVERSE_SUBTRACT
'm' -> GL_MIN
'x' -> GL_MAX

blendeqn.tcl

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program texture maps a checkerboard image onto
two rectangles.

If running this program on OpenGL 1.0, texture objects are
not used.
clip.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates arbitrary clipping planes.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>colormat</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>RedBook</td>
</tr>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

After initialization, the program will be in
ColorMaterial mode. Interaction: pressing the
mouse buttons will change the diffuse reflection values.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program uses the color matrix to exchange the color channels of an image.

Red  -> Green
Green -> Blue
Blue  -> Red
colortable.tcl

An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

Invert a passed block of pixels. This program illustrates the use of the glColorTable() function.
combiner.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program renders a variety of quads showing different effects of texture combiner functions.

The first row renders an untextured polygon (so you can compare the fragment colors) and then the 2 textures.
The second row shows several different combiner functions on a single texture: replace, modulate, add, add-signed, and subtract.
The third row shows the interpolate combiner function on a single texture with a constant color/alpha value, varying the amount of interpolation.
The fourth row uses multitexturing with two textures and different combiner functions.
The fifth row are some combiner experiments: using the scaling factor and reversing the order of subtraction.
for a combination function.
convolution.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Use various 2D convolutions filters to find edges in an image.
cube.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates a single modeling transformation,
glScalef() and a single viewing transformation, gluLookAt().
A wireframe cube is rendered.
An example of the OpenGL red book modified to work with Tc13D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tc13D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates cube map textures.
Six different colored checker board textures are created and applied to a lit sphere.

Pressing the 'f' and 'b' keys translate the object forward and backward.
dof.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates use of the accumulation buffer to
create an out-of-focus depth-of-field effect. The teapots
are drawn several times into the accumulation buffer. The
viewing volume is jittered, except at the focal point, where
the viewing volume is at the same position, each time. In
this case, the gold teapot remains in focus.
double.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This is a simple double buffered program.
Pressing the left mouse button rotates the rectangle.
Pressing the middle mouse button stops the rotation.
### drawf.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Draws the bitmapped letter F on the screen (several times).
This demonstrates use of the `glBitmap()` call.
feedback.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates use of OpenGL feedback. First,
a lighting environment is set up and a few lines are drawn.
Then feedback mode is entered, and the same lines are
drawn. The results in the feedback buffer are printed.
### Demo: fog

<table>
<thead>
<tr>
<th>Type</th>
<th>RedBook</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root</td>
<td>Contents</td>
</tr>
</tbody>
</table>

#### Demo Scripts
- colormain.tcl
- colorable.tcl
- combiner.tcl
- convolution.tcl
- cube.tcl
- cubemap.tcl
- dot.tcl
- double.tcl
- draw.tcl
- feedback.tcl
- fog.tcl
- fogcoord.tcl
- font.tcl
- helo.tcl
- histogram.tcl
- image.tcl
- light.tcl
- lines.tcl
- list.tcl
- material.tcl

---

#### fog.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program draws 5 red spheres, each at a different z distance from the eye, in different types of fog.
Pressing the f key chooses between 3 types of fog: exponential, exponential squared, and linear.
In this program, there is a fixed density value, as well as fixed start and end values for the linear fog.
### Demo: fogcoord

<table>
<thead>
<tr>
<th>Type:</th>
<th>RedBook</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

#### fogcoord.tcl

An example of the OpenGL red book modified to work with Tcl3D.

The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.

See file LICENSE for complete license information.

This program demonstrates the use of explicit fog coordinates. You can press the keyboard and change the fog coordinate value at any vertex. You can also switch between using explicit fog coordinates and the default fog generation mode.

Pressing the 'f' and 'b' keys move the viewer forward and backwards.

Pressing 'c' initiates the default fog generation.

Pressing capital 'C' restores explicit fog coordinates. Pressing '1', '2', '3', '8', '9', and '0' add or subtract from the fog coordinate values at one of the three vertices of the triangle.

---

Running on Windows NT with a GeForce FX Go5600/AGP/SSE2 (OpenGL 1.4.0, Tcl 8.4.13)
font.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Draws some text in a bitmapped font. Uses glBitmap() and other pixel routines. Also demonstrates use of display lists.
hello.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This is a simple, introductory OpenGL program.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Compute the histogram of the image. This program illustrates the use of the glHistogram() function.
image.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates drawing pixels and shows the effect of `glDrawPixels()`, `glCopyPixels()`, and `glPixelZoom()`.
Interaction: moving the mouse while pressing the mouse button will copy the image in the lower-left corner of the window to the mouse position, using the current pixel zoom factors.
There is no attempt to prevent you from drawing over the original image. If you press the 'r' key, the original image and zoom factors are reset. If you press the 'z' or 'Z' keys, you change the zoom factors.
light.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates the use of the OpenGL lighting model. A sphere is drawn using a grey material characteristic.
A single light source illuminates the object.
lines.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates geometric primitives and their attributes.
### Demo: list

<table>
<thead>
<tr>
<th>Type</th>
<th>RedBook</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**list.tcl**

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates how to make and execute a display list. Note that attributes, such as current color and matrix, are changed.
An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program demonstrates the use of the GL lighting model. Several objects are drawn using different material characteristics. A single light source illuminates the objects.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Determine the minimum and maximum values of a group of pixels.
This demonstrates use of the glMinmax() call.
mipmap.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates using mipmaps for texture maps.
To overtly show the effect of mipmaps, each mipmap reduction level has a solidly colored, contrasting texture image.
Thus, the quadrilateral which is drawn is drawn with several different colors.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program demonstrates modeling transformations
movelight.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates when to issue lighting and
transformation commands to render a model with a light
which is moved by a modeling transformation (rotate or
translate). The light position is reset after the modeling
transformation is called. The eye position does not change.

A sphere is drawn using a grey material characteristic.
A single light source illuminates the object.

Interaction: pressing the left mouse button alters
the modeling transformation (x rotation) by 30 degrees.
The scene is then redrawn with the light in a new position.
multisamp.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program draws shows how to use multisampling to
draw anti-aliased geometric primitives. The same
display list, a pinwheel of triangles and lines of
varying widths, is rendered twice. Multisampling is
enabled when the left side is drawn. Multisampling is
disabled when the right side is drawn.

Pressing the 'b' key toggles drawing of the checkerboard
background. Antialiasing is sometimes easier to see
when objects are rendered over a contrasting background.

This demo uses the multisampling options built into tcl3dTogl starting
from version 0.3.2.
Another way to set the number of samples is via the driver specific GUI under
Windows, or by setting the environment variable __GL_FSAA_MODE under Linux.
multitex.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.
mvarray.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates multiple vertex arrays,
specifically the OpenGL routine glMultiDrawElements().
An example of the OpenGL red book modified to work with Tcl3D.

The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

Picking is demonstrated in this program. In rendering mode, three overlapping rectangles are drawn. When the left mouse button is pressed, selection mode is entered with the picking matrix. Rectangles which are drawn under the cursor position are "picked." Pay special attention to the depth value range, which is returned.
**Demo:** picksquare

**Type:** RedBook

**Category:** TutorialsAndBooks

**Root:** Contents

---

**picksquare.tcl**

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2003, Paul Obermeier.
See file LICENSE for complete license information.

Use of multiple names and picking are demonstrated.
A 3x3 grid of squares is drawn. When the left mouse button is pressed, all squares under the cursor position have their color changed.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program shows how to composite modeling transformations
to draw translated and rotated models.
Interaction: pressing the d and y keys (day and year)
alters the rotation of the rotation of the planet around the sun.
pointp.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates point parameters and their effect
on point primitives.
250 points are randomly generated within a 10 by 10 by 40
region, centered at the origin. In some modes (including the
default), points that are closer to the viewer will appear larger.

Pressing the 'l', 'q', and 'c' keys switch the point
parameters attenuation mode to linear, quadratic, or constant,
respectively.
Pressing the 'f' and 'b' keys move the viewer forward
and backwards. In either linear or quadratic attenuation
mode, the distance from the viewer to the point will change
the size of the point primitive.
Pressing the '+' and '-' keys will change the current point
size. In this program, the point size is bounded, so it
will not get less than 2.0, nor greater than GL_POINT_SIZE_MAX.
polyoff.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates polygon offset to draw a shaded polygon and its wireframe counterpart without ugly visual artifacts ("stitching").
polys.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates polygon stippling.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates the use of some of the gluQuadric* routines. Quadric objects are created with some quadric properties and the callback routine to handle errors. Note that the cylinder has no top or bottom and the circle has a hole in it.
An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program shows how to composite modeling transformations to draw translated and rotated hierarchical models. Interaction: pressing the s and e keys (shoulder and elbow) alters the rotation of the robot arm.
scene.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates the use of the GL lighting model.
Objects are drawn using a grey material characteristic.
A single light source illuminates the objects.
select.tcl

An example of the OpenGL red book modified to work with Tcl3D.

The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This is an illustration of the selection mode and name stack, which detects whether objects which collide with a viewing volume. First, four triangles and a rectangular box representing a viewing volume are drawn (drawScene routine). The green triangle and yellow triangles appear to lie within the viewing volume, but the red triangle appears to lie outside it. Then the selection mode is entered (selectObjects routine). Drawing to the screen ceases. To see if any collisions occur, the four triangles are called. In this example, the green triangle causes one hit with the name 1, and the yellow triangles cause one hit with the name 3.
shadowmap.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.
smooth.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates smooth shading.
A smooth shaded polygon is drawn in a 2-D projection.
An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program demonstrates use of the stencil buffer for masking nonrectangular regions. Whenever the window is redrawn, a value of 1 is drawn into a diamond-shaped region in the stencil buffer. Elsewhere in the stencil buffer, the value is 0. Then a blue sphere is drawn where the stencil value is 1, and yellow torii are drawn where the stencil value is not 1.
Demo: stroke
Type: RedBook
Category: TutorialsAndBooks
Root: Contents

stroke.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates some characters of a stroke (vector) font. The characters are represented
by display lists, which are given numbers which correspond to the ASCII values of the characters.
Use of glCallLists() is demonstrated.
surface.tcl

An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program draws a NURBS surface in the shape of a symmetrical hill. The 'c' keyboard key allows you to toggle the visibility of the control points themselves. Note that some of the control points are hidden by the surface itself.
## Tcl3D Demo: OpenGL Red Book example teapots

### File: teapots.tcl

An example of the OpenGL red book modified to work with Tcl3D.

The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates lots of material properties.
A single light source illuminates the objects.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates using glBindTexture() by creating and managing two textures.
texgen.c

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program draws a texture mapped teapot with automatically generated texture coordinates. The texture is rendered as stripes on the teapot.
Initially, the object is drawn with texture coordinates based upon the object coordinates of the vertex and distance from the plane x = 0. Pressing the 'e' key changes the coordinate generation to eye coordinates of the vertex. Pressing the 'o' key switches it back to the object coordinates. Pressing the 's' key changes the plane to a slanted one (x + y + z = 0). Pressing the 'x' key switches it back to x = 0.
**Demo:** texprox

<table>
<thead>
<tr>
<th>Type</th>
<th>RedBook</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category</td>
<td>TutorialsAndBooks</td>
</tr>
<tr>
<td>Root</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**texprox.tcl**

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

The brief program illustrates use of texture proxies.
This program only prints out some messages about whether certain size textures are supported and then exits.
texsub.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program texture maps a checkerboard image onto
two rectangles. This program clamps the texture, if
the texture coordinates fall outside 0.0 and 1.0.
If the s key is pressed, a texture subimage is used to
alter the original texture. If the r key is pressed,
the original texture is restored.
This program demonstrates using a three-dimensional texture. It creates a 3D texture and then renders two rectangles with different texture coordinates to obtain different "slices" of the 3D texture.
texturesurf.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program uses evaluators to generate a curved surface and automatically generated texture coordinates.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates the creation of a display list.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program draws a NURBS surface in the shape of a symmetrical hill, using both a NURBS curve and pwl (piecewise linear) curve to trim part of the surface.
unproject.tcl

An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

When the left mouse button is pressed, this program reads the mouse position and determines two 3D points from which it was transformed. Very little is displayed.
An example of the OpenGL red book modified to work with Tcl3D.
The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc.
The Tcl3D sources are Copyright (c) 2005, Paul Obermeier.
See file LICENSE for complete license information.

This program demonstrates vertex arrays.
wrap.tcl

An example of the OpenGL red book modified to work with Tcl3D. The original C sources are Copyright (c) 1993-2003, Silicon Graphics, Inc. The Tcl3D sources are Copyright (c) 2005, Paul Obermeier. See file LICENSE for complete license information.

This program texture maps a checkerboard image onto two rectangles. This program demonstrates the wrapping modes, if the texture coordinates fall outside 0.0 and 1.0. Interaction: Pressing the 's' and 'S' keys switch the wrapping between clamping and repeating for the s parameter. The 't' and 'T' keys control the wrapping for the t parameter.

If running this program on OpenGL 1.0, texture objects are not used.
<table>
<thead>
<tr>
<th>Category:</th>
<th>OpenSceneGraph</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
<tr>
<td>Types:</td>
<td>CubosLocos FopingTutorials NPS-Tutorials OsgHelp QuickStartGuide</td>
</tr>
</tbody>
</table>
Some of the OpenSceneGraph tutorials from CubosLocos have been ported to run with Tcl3D. Original sources available at: [http://www.cuboslocos.com/](http://www.cuboslocos.com/)

### Available demos

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="earth.png" alt="Earth" /></td>
<td><img src="solar.png" alt="Sun" /></td>
</tr>
</tbody>
</table>

**Type:** | **CubosLocos**
---|---
Category: | OpenSceneGraph
Root: | Contents

Copyright © 2005-2010 by Paul Obermeier. All rights reserved.
Tcl3D: Doing 3D with Tcl

Demo: earth
Type: CubosLocos
Category: OpenSceneGraph
Root: Contents

Original C++ code by Katja Treiber and Matthias Schmidt.
See www.cuboslocos.com for the original files.

Modified for Tcl3D by Paul Obermeier 2009/08/30.
See www.tcl3d.org for the Tcl3D extension.
**Demo:** solar

<table>
<thead>
<tr>
<th>Type:</th>
<th>CubosLocos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**solar.tcl**

Original C++ code by Katja Treiber and Matthias Schmidt.
See www.cuboslocos.com for the original files.

Modified for Tcl3D by Paul Obermeier 2009/06/10.
See www.tcl3d.org for the Tcl3D extension.
Some of the OpenSceneGraph tutorials from Franclin Foping have been ported to run with Tcl3D.


### Available demos

<table>
<thead>
<tr>
<th>Tuto11</th>
<th>Tuto5</th>
<th>Tuto6</th>
<th>Tuto7</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Tuto11" /></td>
<td><img src="image2.png" alt="Tuto5" /></td>
<td><img src="image3.png" alt="Tuto6" /></td>
<td><img src="image4.png" alt="Tuto7" /></td>
</tr>
<tr>
<td>Tuto8</td>
<td>Tuto9</td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image5.png" alt="Tuto8" /></td>
<td><img src="image6.png" alt="Tuto9" /></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Tuto11.tcl: A billboarding effect

This tutorial will extend the previous one on lighting by adding two billboard quads. These are also textured and shaded.

Original C++ code by Franclin Foping.

Modified for Tcl3D by Paul Obermeier 2009/06/10.
See www.tcl3d.org for the Tcl3D extension.
Tcl3D demo: Foping's OSG tutorial #5 (Draw a shape)

Tuto5.tcl: Draw a shape

Original C++ code by Franclin Foping.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.

Original C++ code by Franclin Foping.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>Tuto7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>FopingTutorials</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Tuto7.tcl: Texturing and positioning


Modified for Tcl3D by Paul Obermeier 2009/03/20. See www.tcl3d.org for the Tcl3D extension.
**Demo:**

<table>
<thead>
<tr>
<th>Type:</th>
<th>FopingTutorials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Tuto8.tcl:** Draw a shape with fog.

Original C++ code by Franclin Foping.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
Tuto9.tcl: OSG Lighting example

This simple example will show how to easily shade your scene. We will be making use of 2 light sources, one is red and the other one is green. We will also render light markers to help you locate light source in the scene. This is helpful for debugging purposes.


Modified for Tcl3D by Paul Obermeier 2009/03/20. See www.tcl3d.org for the Tcl3D extension.
Some of the OpenSceneGraph tutorials from Joseph Sullivan have been ported to run with Tcl3D. Original sources available at: http://www.openscenegraph.org/projects/osg/wiki/Support/Tutorials/

Available demos

- NPS02_GeometryTest
- NPS03_TexturedGeometry
- NPS04_TestState
- NPS05_TestTutorial
- NPS06_SimpleHUD
- NPS07_TestSwitchDOF
- NPS08_TestUpdateCallback
- NPS09_TestKeyboard
- NPS10_TestManualCamera
Tcl3D demo: Sullivan's OSG tutorial #2 (GeometryTest)

Key-Escaps Exit
Key-f Save SceneGraph to file
House Trackball

NPS02_GeometryTest.tcl

Original C++ code by Joseph Sullivan.
for the original files.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>NPS03_TexturedGeometry</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>NPS-Tutorials</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Demo: Sullivan's OSG tutorial #3 (TexturedGeometry)**

- Key-Esc: Exit
- Key-f: Save SceneGraph to file
- Mouse: Trackball

Running on Windows NT 5.1 with a GeForce FX Go5600/AGP/SSE2 (OSG 2.8.2, Tcl/Tk 8.4.16)

NPS03_TexturedGeometry.tcl

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
**NPS04_TestState**

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
## Tcl3D demo: Sullivan's OSG tutorial #5 (TestTutorial)

<table>
<thead>
<tr>
<th>Demo:</th>
<th>NPS05_TestTutorial</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>NPS-Tutorials</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

### Description

Running on Windows NT 5.1 with a GeForce FX Go5600/AGP/SSE2 (OSG 2.8.2, Tcl/Tk 8.4.16)

**Key-Exit:** Exit

**Key-f:** Save SceneGraph to file

**Mouse:** Trackball

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
Original C++ code by Joseph Sullivan.
for the original files.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
Tcl3D demo: Sullivan's OSG tutorial #7 (TestSwitchDOF)

Original Tank

Damaged State Articulated

### NPS07_TestSwitchDOF.tcl

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>NPS08_TestUpdateCallback</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>NPS-Tutorials</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Tcl3D demo: Sullivan's OSG tutorial #8 (TestUpdateCallback)

Demo scripts
- NPS02_GeometryTest.tcl
- NPS03_TextureGeom.tcl
- NPS04_TestScale.tcl
- NPS05_TestTutorial.tcl
- NPS06_SimpleHUD.tcl
- NPS07TestClassOFI.tcl
- NPS08_TestUpdateCallback.tcl
- NPS09_TestKeyboard.tcl
- NPS10_TestManualCamera.tcl

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
### Demo: NPS09_TestKeyboard

<table>
<thead>
<tr>
<th>Type:</th>
<th>NPS-Tutorials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Tcl3D demo: Sullivan's OSG tutorial #9 (TestKeyboard)**

- **Key-Escape**: Exit
- **Key-Left|Right**: Rotate turret left|right
- **Key-f**: Save SceneGraph to file
- **Mouse**: Trackball

Running on Windows NT 5.1 with a GeForce FX Go5600/AGP/SSE2 (OSG 2.8.2, Tcl/Tk 8.4.16)

---

**NPS09_TestKeyboard.tcl**

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>NPS10_TestManualCamera</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>NPS-Tutorials</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**Tcl3D demo: Sullivan’s OSG tutorial #10 (TestManualCamera)**

![Image of a tank in a 3D environment with a menu showing the files NPS02_GeometryTest.tcl, NPS03_Tutorial.tcl, etc.]

- **Key-Escape Exit**
- **Key-f** Save SceneGraph to file
- **Key-v** Toggle view mode
- **Mouse** Trackball

Running on Windows NT 5.1 with a GeForce FX Go5600/AGP/SSE2 (OSG 2.8.2, Tcl/Tk 8.4.16)

NPS10_TestManualCamera.tcl

Original C++ code by Joseph Sullivan.

Modified for Tcl3D by Paul Obermeier 2009/05/01.
See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.
Some of the OpenSceneGraph tutorials from Peter Wraae Marino and Michael Bach Jensen have been ported to run with Tcl3D. Original sources available at their [OsgHelp](http://www.osghelp.org) website.

### Available demos

<table>
<thead>
<tr>
<th>AntialiasedLines</th>
<th>AutoTransform</th>
<th>Billboard</th>
<th>Geometry</th>
</tr>
</thead>
<tbody>
<tr>
<td>GeometryTextured</td>
<td>HelloWorld</td>
<td>Light</td>
<td>MultiTextures</td>
</tr>
<tr>
<td>Shapes</td>
<td>StatisticsDisplay</td>
<td>UpdateCallback</td>
<td>Wireframe</td>
</tr>
</tbody>
</table>
## AntialiasedLines

<table>
<thead>
<tr>
<th>Demo: AntialiasedLines</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: OsgHelp</td>
<td></td>
</tr>
<tr>
<td>Category: OpenSceneGraph</td>
<td></td>
</tr>
<tr>
<td>Root: Contents</td>
<td></td>
</tr>
</tbody>
</table>

### AntialiasedLines.tcl

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo</th>
<th>AutoTransform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>OsgHelp</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

AutoTransform.tcl

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
### Demo:

<table>
<thead>
<tr>
<th>Billboard</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type:</strong></td>
<td>OsgHelp</td>
</tr>
<tr>
<td><strong>Category:</strong></td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td><strong>Root:</strong></td>
<td>Contents</td>
</tr>
</tbody>
</table>

#### Billboard.tcl

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Tcl3D demos at a glance  Version 0.5.0, December 2010  Page 254 of 266
Copyright © 2005-2010 by Paul Obermeier. All rights reserved.

Demo: Geometry
Type: OsgHelp
Category: OpenSceneGraph
Root: Contents

Original C++ code by Peter Wraae Marino and Michael Bach Jensen.
See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20.
See www.tcl3d.org for the Tcl3D extension.
### Demo:

<table>
<thead>
<tr>
<th>Type:</th>
<th>GeometryTextured</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category:</td>
<td>OsgHelp</td>
</tr>
<tr>
<td>Root:</td>
<td>OpenSceneGraph</td>
</tr>
</tbody>
</table>

**GeometryTextured.tcl**

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>HelloWorld</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>OsgHelp</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

**HelloWorld.tcl**

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

Demo: Light
Type: OsgHelp
Category: OpenSceneGraph
Root: Contents

Original C++ code by Peter Wraae Marino and Michael Bach Jensen.
See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20.
See www.tcl3d.org for the Tcl3D extension.
Tcl3D: Doing 3D with Tcl

<table>
<thead>
<tr>
<th>Demo:</th>
<th>MultiTextures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>OsgHelp</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
### Demo: Shapes

**Type:** OsgHelp  
**Category:** OpenSceneGraph  
**Root:** Contents

---

#### Shapes.tcl

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See [www.osghelp.com](http://www.osghelp.com) for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See [www.tcl3d.org](http://www.tcl3d.org) for the Tcl3D extension.

---

**Running on Windows NT 5.1 with a GeForce FX 5600/AGP/SSE2 (OSG 2.8.2, Tcl/Tk 8.6b1.1)**

---

**Key-Escape Exit**  
**Key-f** Save SceneGraph to file  
**Mouse** Trackball
Tcl3D: Doing 3D with Tcl

## Demo: StatisticsDisplay

**Type:** OsgHelp  
**Category:** OpenSceneGraph  
**Root:** Contents

---

**StatisticsDisplay.tcl**

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
### UpdateCallback

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
Wireframe.tcl

Original C++ code by Peter Wraae Marino and Michael Bach Jensen. See www.osghelp.com for the original files.

Modified for Tcl3D by Paul Obermeier 2010/03/20. See www.tcl3d.org for the Tcl3D extension.
Some of the OpenSceneGraph examples from Paul Martz's Quick Start Guide have been ported to run with Tcl3D. Book and original sources available at: [http://www.skew-matrix.com/OSGQSG/](http://www.skew-matrix.com/OSGQSG/)

Available demos

- Callback
- Lighting
- Picking
Callback Example, Using an update callback to modify the scene graph

Modified for Tcl3D by Paul Obermeier 2009/10/20.
See www.tcl3d.org for the Tcl3D extension.
<table>
<thead>
<tr>
<th>Demo:</th>
<th>Lighting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type:</td>
<td>QuickStartGuide</td>
</tr>
<tr>
<td>Category:</td>
<td>OpenSceneGraph</td>
</tr>
<tr>
<td>Root:</td>
<td>Contents</td>
</tr>
</tbody>
</table>

Lighting Example, Basic light and material control

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.
Picking Example, Using the osgUtil Intersection classes and osgGA NodeKit

Code derived from an OSG example. Original comment block follows.

C++ source file - (C) 2003 Robert Osfield, released under the OSGGPL.

Simple example of use of osgViewer::GraphicsWindow + SimpleViewer that provides the user with control over view position with basic picking.

Modified for Tcl3D by Paul Obermeier 2009/03/20.
See www.tcl3d.org for the Tcl3D extension.