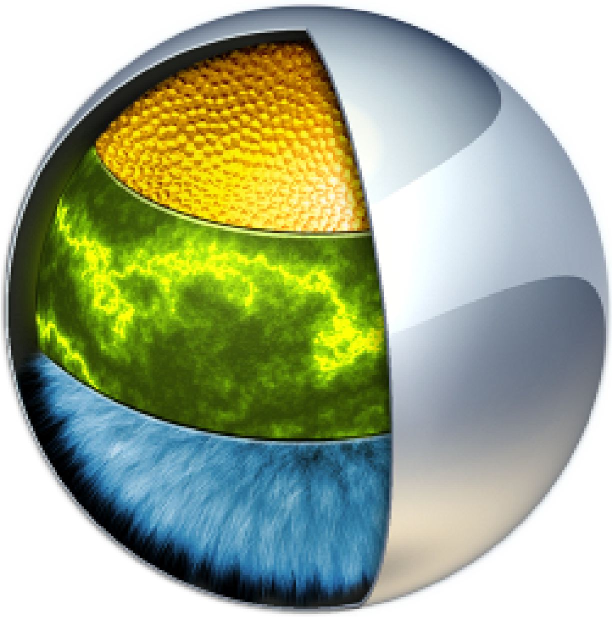

Cg Toolkit Release Notes

Release 3.1.0010

NVIDIA Corporation

March 20, 2012

1.1 NAME

cg_3_1_0010 - Cg Toolkit 3.1 February 2012

1.2 SUMMARY

The Cg Toolkit allows developers to write and run Cg programs using a wide variety of hardware and OS platforms and graphics APIs. Originally released in December 2002, the Toolkit now supports over 30 different DirectX and OpenGL profile targets. It provides a compiler for the Cg language, runtime libraries to use with the OpenGL and DirectX graphics APIs, support for CgFX effect files, example applications, and extensive documentation.

1.3 CONTENTS

1.3.1 Cg Runtime libraries

The Cg core runtime library for managing parameters and loading programs.

The CgGL runtime library for OpenGL based applications.

The CgD3D9 runtime library for DirectX 9 based applications.

The CgD3D10 runtime library for DirectX 10 based applications.

The CgD3D11 runtime library for DirectX 11 based applications.

1.3.2 Supported Profiles

OpenGL

gp5tcp NV_tessellation_program5 control program.

gp5tep NV_tessellation_program5 evaluation program.

gp5gp NV_geomemtry_program5.

gp5vp NV_vertex_program5.

gp5fp NV_fragment_program5.

gp4gp NV_geomemtry_program4.

gp4vp NV_vertex_program4.

gp4fp NV_fragment_program4.

glslg OpenGL Shading Language (GLSL) for OpenGL 2.0 geometry shader.

glslv OpenGL Shading Language (GLSL) for OpenGL 2.0 vertex shader.

glslf OpenGL Shading Language (GLSL) for OpenGL 2.0 fragment shader.

arbvp1 ARB_vertex_program 1.0.

arbfp1 ARB_fragment_program 1.0.

vp40 ARB_vertex_program + NV_vertex_program2 option.

fp40 ARB_fragment_program + NV_fragment_program2 option.

vp30 NV_vertex_program 2.0.

fp30 NV_fragment_program 1.0.

vp20 NV_vertex_program 1.0.

fp20 NV_register_combiners and NV_texture_shader.

DirectX 11.0

ds_5_0 HLSL11 Domain Shader.

hs_5_0 HLSL11 Hull Shader.

gs_5_0 HLSL11 Geometry Shader.

vs_5_0 HLSL11 Vertex Shader.

ps_5_0 HLSL11 Fragment Shader.

DirectX 10.0

gs_4_0 HLSL10 Geometry Shader.

vs_4_0 HLSL10 Vertex Shader.

ps_4_0 HLSL10 Fragment Shader.

DirectX 9.0c

hlslv HLSL9 Vertex Shader.

hlslf HLSL9 Fragment Shader.

vs_3_0 Vertex Shader 3.0.

ps_3_0 Pixel Shader 3.0.

DirectX 9

vs_2_x Extended Vertex Shader 2.0.

ps_2_x Extended Pixel Shader 2.0.

vs_2_0 Vertex Shader 2.0.

ps_2_0 Pixel Shader 2.0.

vs_1_1 Vertex Shader 1.1.

ps_1_3 Pixel Shader 1.3.

ps_1_2 Pixel Shader 1.2.

ps_1_1 Pixel Shader 1.1.

1.4 IMPROVEMENTS AND BUG FIXES

1.4.1 Improvements

Added Cg language support for uniform buffers.

Added OpenGL Unified Buffer Object (UBO) support for buffers.

Added OpenGL GLSL version 110 and 120 translation support.

New tessellation examples added.

New uniform buffer examples added.

VC10 projects added for examples.

1.4.2 Documentation

Note: The Cg Users Manual has **not** been updated for this release.

Updated reference manual for new entry points.

Updated Cg standard library documentation.

Release history documentation added.

1.4.3 Bug Fixes

Fixed buffer emulation in DX9, *arbvp1* and *arbfp1* profiles.

Fixed f3dtex2D standard library issue.

Added samplerRBUF to standard library.

Fixed issues with implicit cast of int to uint.

Fixed cgc compiler issue for **$COL0**.

Fixed *cgGetParameterResourceIndex* to return the proper resource index.

Fixed var bindings info in *GLSL* and *fp20* profiles.

Improved the DX10 and DX11 examples.

Added support for non-uniform matrix argument in the *glslf* profile.

Increased *GLSL* output color array to 8.

Added **CENTROID**, **FLAT** and **NOPERSPECTIVE** semantic support to *gp4* and *gp5* profiles.

Fixed an issue for int and uint in *GLSL* profile versions before 130.

Fixed unsigned integer uniform usage in *GLSL* profiles.

Fixed SAMPLEPOS usage in the *gp5fp* profile.

Fixed the tex2dbias usage in the *arbfp1* profile.

1.4.4 New API

Below is the complete list of the new entry points in Cg 3.1. See the Cg Reference Manual for further details.

cgGetFirstUniformBufferParameter

cgGetNamedEffectUniformBuffer

cgGetNamedProgramUniformBuffer

cgGetNamedUniformBufferParameter

cgGetProfileSibling

cgGetProgramOutputVertices

cgGetUniformBufferBlockName

cgGetUniformBufferParameter

cgIsBuffer

cgSetProgramOutputVertices

cgSetUniformBufferParameter

cgGLCreateBufferFromObject

cgGLDetectGLSLVersion

cgGLGetContextGLSLVersion

cgGLGetContextOptimalOptions

cgGLGetGLSLVersion

cgGLGetGLSLVersionString

cgGLSetContextGLSLVersion

cgGLSetContextOptimalOptions

cgD3D10CreateBufferFromObject

cgD3D10CreateBuffer

cgD3D10GetBufferObject

cgD3D11CreateBufferFromObject

cgD3D11CreateBuffer

cgD3D11GetBufferObject

1.5 COMPATIBILITY NOTES

Although the 3.1 release of Cg is generally compatible with previous releases, several improvements and other changes may affect existing applications. This section details these potential compatibility issues.

Cg 3.1 supports defining constant blocks with the uniform keyword. Use of a BUFFER semantic on simple variables and structs has been deprecated. By default cgc will now issue a warning when the BUFFER semantic is used (and an error when BUFFER is used on a struct with a GLSL profile). The -no_uniform_blocks compiler flag can be used to disable these warnings and errors, but it also removes support for the new uniform keyword method of defining constant blocks.

There aren't any other known compatibility issues with programs written against Cg 3.0. For programs written against Cg 2.2 or earlier, refer to the Compatibility Notes section of the release notes for Cg 3.0.

1.5.1 Deprecated

Manual pages have been removed.

D3D8 support has been removed.

Mac OS X 10.4 support has been removed.

1.6 KNOWN ISSUES

1.6.1 Runtime

None.

1.6.2 Compiler

None.

1.7 DOWNLOAD

Windows x86/x86-64 installer for Windows XP, Vista and Win7.

Mac OS X ppc/i386/x86_64 dmg for Leopard, Snow Leopard and Lion.

Linux x86 tgz tarball, rpm for RedHat and deb for Debian and Ubuntu.

Linux x86-64 tgz tarball, rpm for RedHat and deb for Debian and Ubuntu.

1.8 LICENSE

The docs directory contains a file **license.pdf** providing a non-exclusive, world-wide, royalty free license for redistributing Cg with your applications. See this license for details.

1.8.1 Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS". NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

1.8.2 Trademarks

NVIDIA and the NVIDIA logo are trademarks or registered trademarks of NVIDIA Corporation. Microsoft, Windows, the Windows logo, and DirectX are registered trademarks of Microsoft Corporation. OpenGL is a trademark of SGI. Other company and product names may be trademarks of the respective companies with which they are associated.

NVIDIA Corporation
2701 San Tomas Expressway
Santa Clara, CA 95050
www.nvidia.com

1.9 SEE ALSO

2.1.0009, *2.1.0012*, *2.1.0016*, *2.1.0017*

2.2.0004, *2.2.0006*, *2.2.0010*, *2.2.0017*

3.0.0007, *3.0.0015*, *3.0.0016*

3.1.0010