**OpenGL Broadcast App Maint**

**Daily/Weekly Status Report**

**Project Manager:**Joshua Kolden, Crack Creative

**For week ending 01/16/2011**

Milestones

**Planned Dates**

**Actual Dates**

**Comments**

OpenGL Broadcast App Maint

3-4 weeks

In time

In progress

Accomplished this week

**(Task)** 1. (01/14/2011) Migrated Server code to CMake

**(Task)** 2. (01/15/2011) Tested/debugged server code using CMake on OSX. Linker error seen on XP. Not tested on linux.

**For week ending 01/23/2011**

**(Task)** 01/19/2011 – Completed compilation on WinXP (not tested yet)

**(Task)** 01/20/2011

• Started implementing two buffer system on server

**(Task)** 01/22/2011

• Completed implementation and testing of 2 buffer system on OSX

• Started fixed image resolution and fixed pixel aspect

Notes:

Created a GitHub repository for the project.

The latest server code has been checked into a private GitHub respository (OpenGLBAppMaint). I would have gladly created an account for Crack Creative but an email is required. If you have a GitHub account please send me the username so I can grant you access to the repository. Easier to use this than send a zip file each time.

**(contd…)**

**(Task)** 01/23/2011

• Continued with image padding

• Completed documentation of server functions.

Notes

Image padding seems a little involved. My understanding of this problem is that the spinning cube images must not change when window is resized. The JPEG library gives the ability to set pixel aspect values before compression may be more changes are needed to jpeg compression algorithm. I experimented with reading pixels from the entire window when window is reshaped. Task next week will be to compare buffers, when original window is read into a buffer and when the window is resized, then decide on how and where to pad. Basically, padding will be used to make sure that when pixels are converted to jpeg images, the row stride of an original buffer is the same size as the row stride of a buffer from a reshaped window.

**For week ending 01/30/2011**

**(Task)** 01/29/2011 –

• Decided to use ImageMagick library for image resizing. Installed ImageMagick library on OSX. [Remember to use ./configure –disable\_openmp]

• Completed coding of same pixel aspect ratio when openGL window is resized.

• Tested on OSX.

(NOTES)

Download ImageMagick source from imagemagick.org

On command line type –

./configure –disable\_openmp

make and sudo make install

**(Task)** 01/30/2011

• Started implementing single image resolution. Work in progress.

(Notes)

Added to GitHub repository – OpenGL-Broadcast-App-Maintenance

**For week ending 02/06/2011**

**(Task)** 02/05/2011 –

• Implemented fixed pixel aspect ratio.

• Working on getting fixed image resolution.

(NOTES)

For getting fixed pixel aspect ratio –

• Say the new window width and window ht are new\_win\_width and new\_win\_height.

• Let WINDOW\_WIDTH=256 and WINDOW\_HEIGHT=256 be the standard 256x256 window size.

• For fixed pixel aspect ratio –

total\_rows=WINDOW\_WIDTH; total\_cols=WINDOW\_HEIGHT;

if(new\_win\_width > new\_win\_height)

total\_rows += new\_win\_width/new\_win\_height;

else if(new\_win\_height > new\_win\_width)

total\_cols += new\_win\_height/new\_win\_width;

image.resize(“total\_cols X total\_rows);

/\* This gives fixed pixel aspect ratio \*/

On the device side, pad the image view with black color and scale the incoming image proportionally to fit the 256x256 device window.

**(Task)** 02/06/2011

• Completed fixed pixel aspect ratio and fixed resolution of images. Checked into GitHub.

(Notes)

Added to GitHub repository – OpenGL-Broadcast-App-Maintenance

**For week ending 02/13/2011**

**Summary -**

**1. Completed Linux Ubuntu 10.04 compilation of 2 buffer and fixed pixel aspect ratio/fixed image resolution**

**2. Completed changes to client for iPad compatibility.**

**3. Completed automatic client connection to UDT server when bonjour connection is detected.**

**(Task)** 02/08/2011 –

• Started linux compilation of 2 buffer and fixed pixel aspect ratio/fixed image resolution

• Started making changes to client for iPad compatibility.

**(Task)** 02/09/2011

• Continued linux compile

**(Task)** 02/11/2011

• Completed linux compile

• Continuing work on client iPad Compatibility.

**(Task)** 02/12/2011

• Completed changes to client for iPad compatibility. Tested on OSX.

• Started automatic client connection to server when bonjour is detected. Tested on OSX.

**(Task)** 02/13/2011

• Completed automatic client connection to server when bonjour is detected. Tested on OSX.

• Started windows xp compile.

(Notes)

Added to GitHub repository – OpenGL-Broadcast-App-Maintenance

Issues / Risks Identified

**(Description)**

**(Description)**