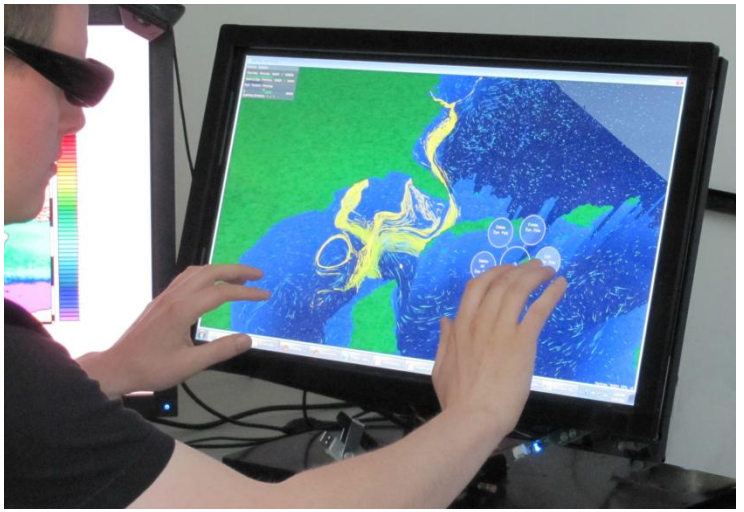
GLUT
Tips and Tricks

Thomas Butkiewicz, Ph.D

GLUT Code Organization

Keep you main() simple!



```
int main(int argc, char* argv[])
{
    glutInit(&argc, argv);
    if (STEREO_MODE)
        glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH | GLUT_STENCIL | GLUT_STEREO);
    else
        glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH | GLUT_STENCIL);

    glutInitWindowSize(1906, 1002);
    glutInitWindowPosition(0, 0);
    glutCreateWindow("SMT Flow - Feb 7th 2011");
    init();

    glutIdleFunc(idle);
    glutReshapeFunc(reshape);
    glutDisplayFunc(display);
    glutKeyboardFunc(keyboard);
    glutMouseFunc(mouseSelection);
    glutMotionFunc(motionFunc);

    glutMainLoop();

    return 0;
}
```

```
void init(void)
{
    leftMouseDown = false;
    rightMouseDown = false;

    srand(time(NULL));
    settings = new Settings();
    dataset = new NCDFdata(settings);

    glClearColor(MAIN_BACKGROUND_COLOR,1.0);
    glEnable(GL_DEPTH_TEST);
    glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    glEnable(GL_NORMALIZE);
    //glEnable(GL_LIGHTING);
    glEnable(GL_BLEND);
//  glShadeModel(GL_SMOOTH);
```

GLUT Code Organization

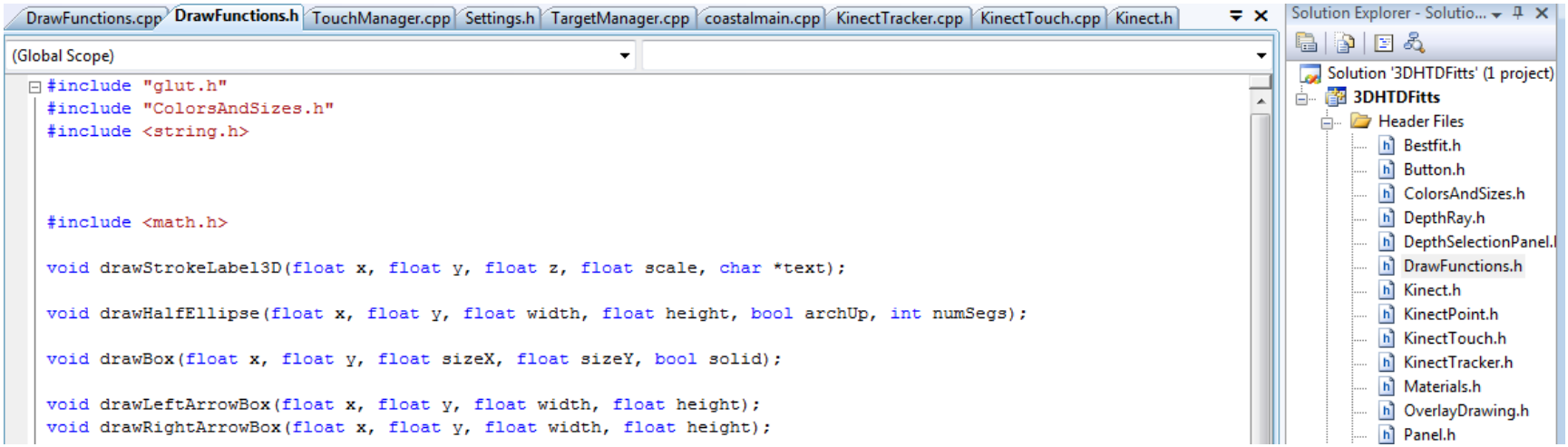
Keep display() understandable

```
void display()
{
    //update animations
    calcAnimations();

    //set up camera, etc
    glMatrixMode(GL_PROJECTION);
    glPushMatrix();
    glLoadIdentity();
    glOrtho(0.0, glutGet(GLUT_WINDOW_WIDTH), 0.0, glutGet(GLUT_WINDOW_HEIGHT), -1.0, 1.0);
    glMatrixMode(GL_MODELVIEW);
    glPushMatrix();
    glLoadIdentity();

    //draw scene
    drawScene();
    //draw overlay
    drawOverlay();

    //swap buffers
    glutSwapBuffers();

}//end display()
```

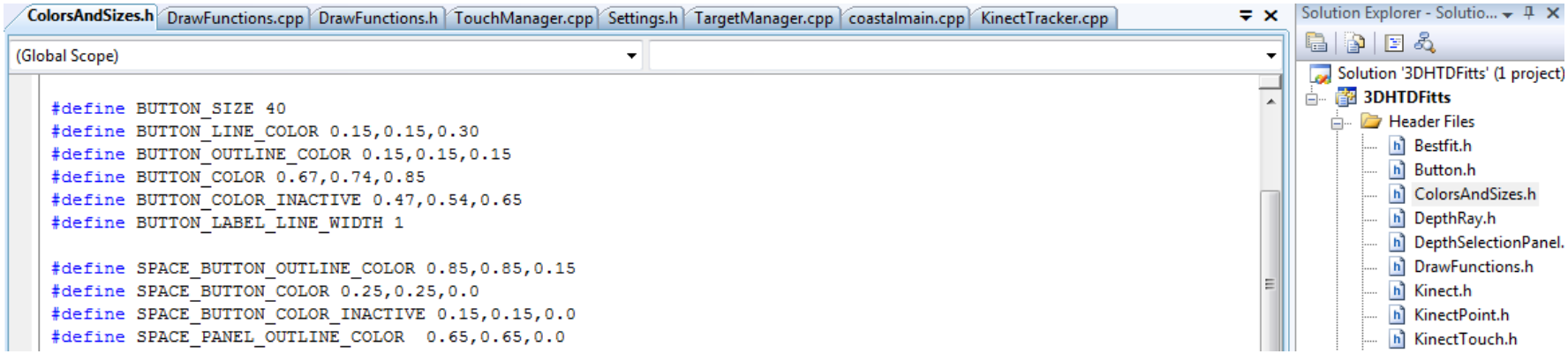
GLUT Code Organization

Organize your drawing functions

Save flexible functions for reuse later:



Use #define's to adjust look and feel across entire program(s):



Misc GLUT tips & tricks

To get current window size:

glutGet(GLUT_WINDOW_WIDTH);

glutGet(GLUT_WINDOW_HEIGHT);

Misc GLUT tips & tricks

Get the vertical transformation out of the way first.

```
void mouseSelection(int button, int state, int x, int y)
{
    y = glutGet(GLUT_WINDOW_HEIGHT) - y;

    if (state == GLUT_UP && button == GLUT_WHEEL_UP)
    {
        //zoom in
        settings->camera->MoveForward(settings->camera->moveSpeed*10);
        zoomIn = true;
    }
    else if (state == GLUT_UP && button == GLUT_WHEEL_DOWN)
    {
        //zoom out
        settings->camera->MoveBackward(settings->camera->moveSpeed*10);
        zoomIn = false;
    }
```

How to use mouse scroll wheel:

Misc GLUT tips & tricks

For arrow keys, function keys, etc:

glutSpecialFunc(special);

```
void special (int key, int x, int y)
{
    if (key == GLUT_KEY_LEFT)
        //do left behavior
    if (key == GLUT_KEY_RIGHT)
        //do right behavior
    if (key == GLUT_KEY_UP)
        //do up behavior
    if (key == GLUT_KEY_DOWN)
        //do down behavior
    glutPostRedisplay();
}
```

Misc GLUT tips & tricks

Escape key = 27 in ASCII:

```
void keyboard (unsigned char key, int x, int y)
{
    if (key == 27) //escape key
        exit(0);
```

Controlling Redrawing

Do Not: Call display() directly

(redundant draws!)

Do: Call: glutPostRedisplay()

Sets a “window needs redrawn” flag

Multiple calls only redraws once

Call glutPostRedisplay():

- at the end of mouse & keyboard callbacks
- in your idle() function, or…
- in your timer() function……

Controlling Redrawing - Timer

glutIdleFunc():

- Hogs processor power (even when minimized!)
- Sometime a certain max frame rate limit desired
- Smoother animation possible

glutTimerFunc():

- Add small timer function, which just calls glutPostRedisplay():

```
void timer(int id)
{
    glutPostRedisplay();
}
```

- Then set the timer at the very end of your display function:

```
glutSwapBuffers();

glutTimerFunc( 33, timer, 0);


}//end display
```

Minimum # of milliseconds until timer called

ms	fps
10	100
16.6	60
33.3	30
50	20

Smooth Animations

Animating a set amount each frame is bad!

- Frame rate determines speed of object
- IdleFunc can lead to varying speeds, incompatibility between systems
- Even TimerFunc does not guarantee set frame rate

E.g. if you move ball 1 pixel every frame, what seems like a good speed on your laptop, might be unplayably fast on the grader's desktop.

Smooth Animations

Use system time to interpolate:

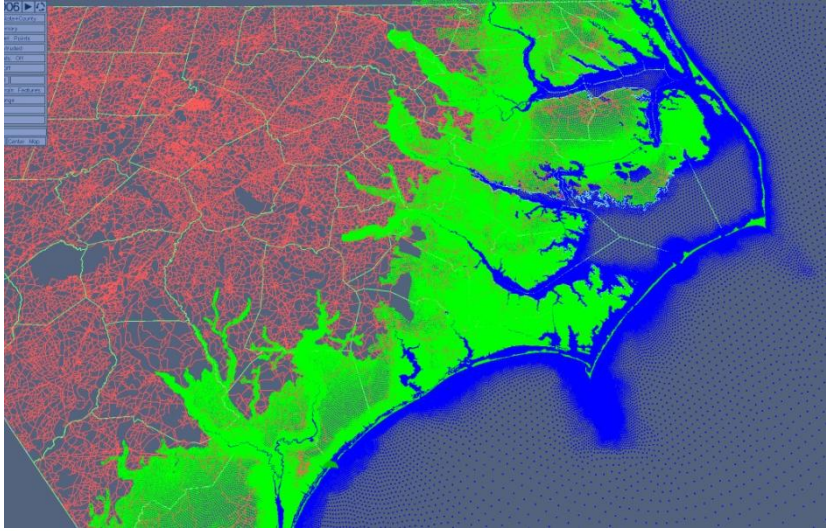
- Windows: GetTickCount() returns time in milliseconds since boot

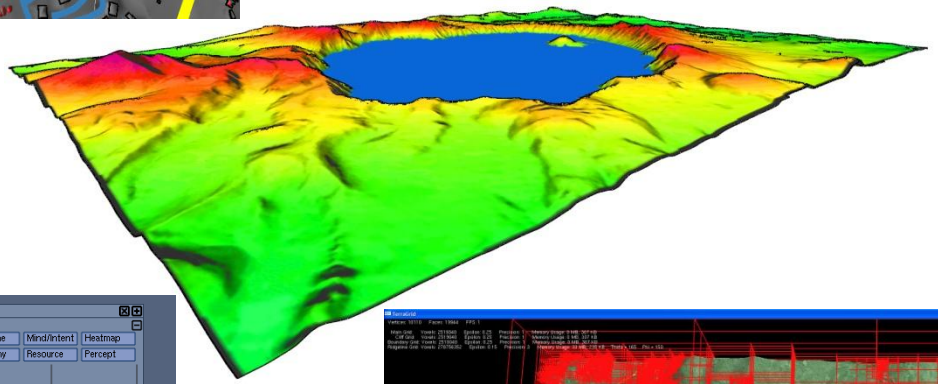
  #include “Windows.h”

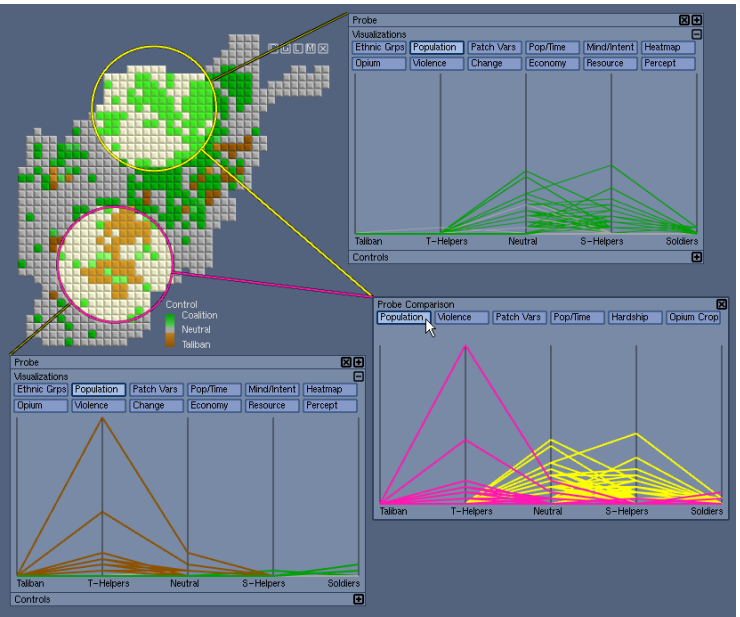
- Linux’s equivalent is: gettimeofday()

```
float lastUpdate;

void calcAnimation()
{
    float now = GetTickCount();
    float since = now - lastUpdate;
    if (since > 20) //optional update rate clamp, here it won't update more than 50Hz
    {
        object.x += object.dx * (since * 0.001); //if dx is units/sec, we convert ms to s
        object.y += object.dy * (since * 0.001);
        lastUpdate = now;
    }
}

void display()
{
    calcAnimation();

    //draw scene
}

void init()
{
    lastUpdate = GetTickCount();
}
```


Area Threshold:
500
# of Change Points Threshold:
3
None
2002
2003
3D Mode
Map Mode
Draw Interstates
Draw Thoroughfares
Draw Streets
Draw County Boundary
Plot 2002 Building Permits
Plot 2003 Building Permits
Plot 2004 Building Footprints
9288 Barnhardt Ln






Probe
Visualizations
Ethnic Grps
Population
Patch Vars
Pop/Time
Mind/Intent
Heatmap
Opium
Violence
Change
Economy
Resource
Percept
Taliban
T-Helpers
Neutral
S-Helpers
Soldiers
Controls
Control
Coalition
Neutral
Taliban
Probe Comparison
Population
Violence
Patch Vars
Pop/Time
Hardship
Opium Crop
Probe
Visualizations
Ethnic Grps
Population
Patch Vars
Pop/Time
Mind/Intent
Heatmap
Opium
Violence
Change
Economy
Resource
Percept
Taliban
T-Helpers
Neutral
S-Helpers
Soldiers
Controls