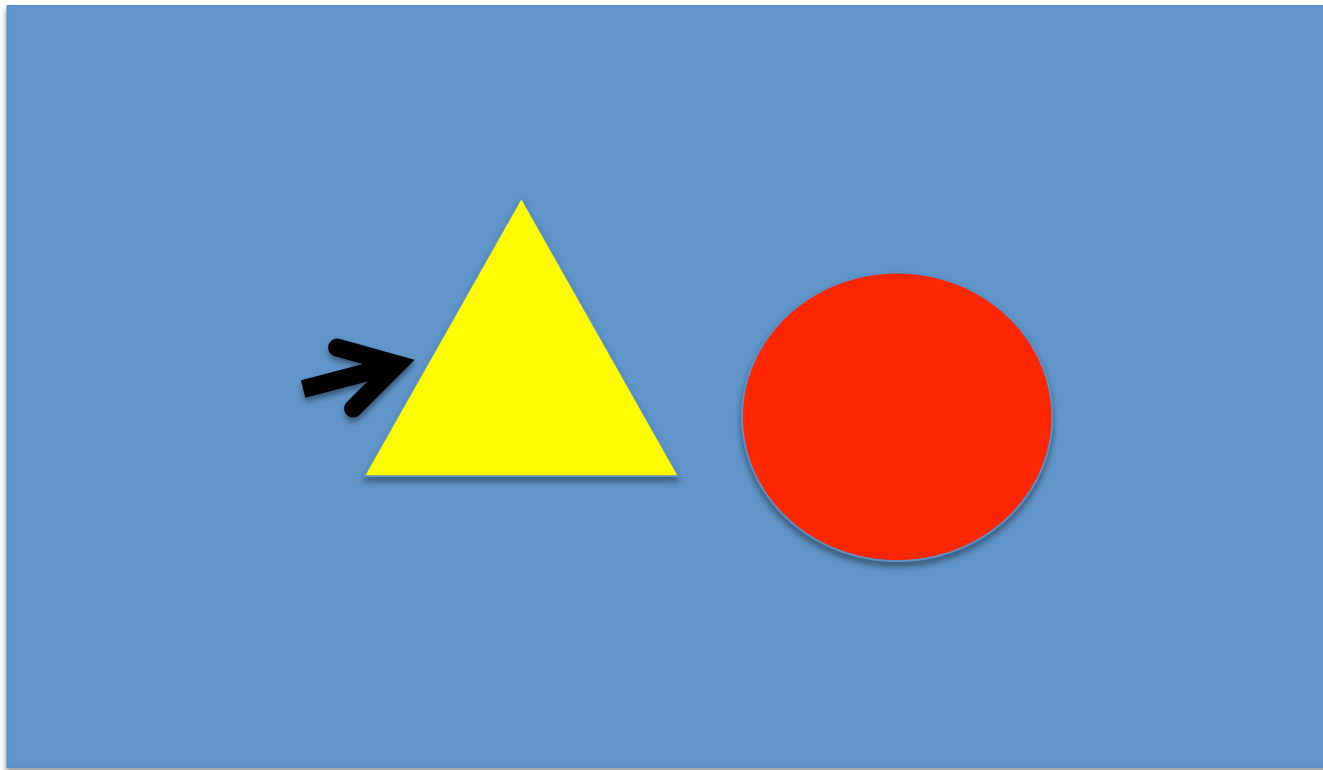


# OpenGL Picking

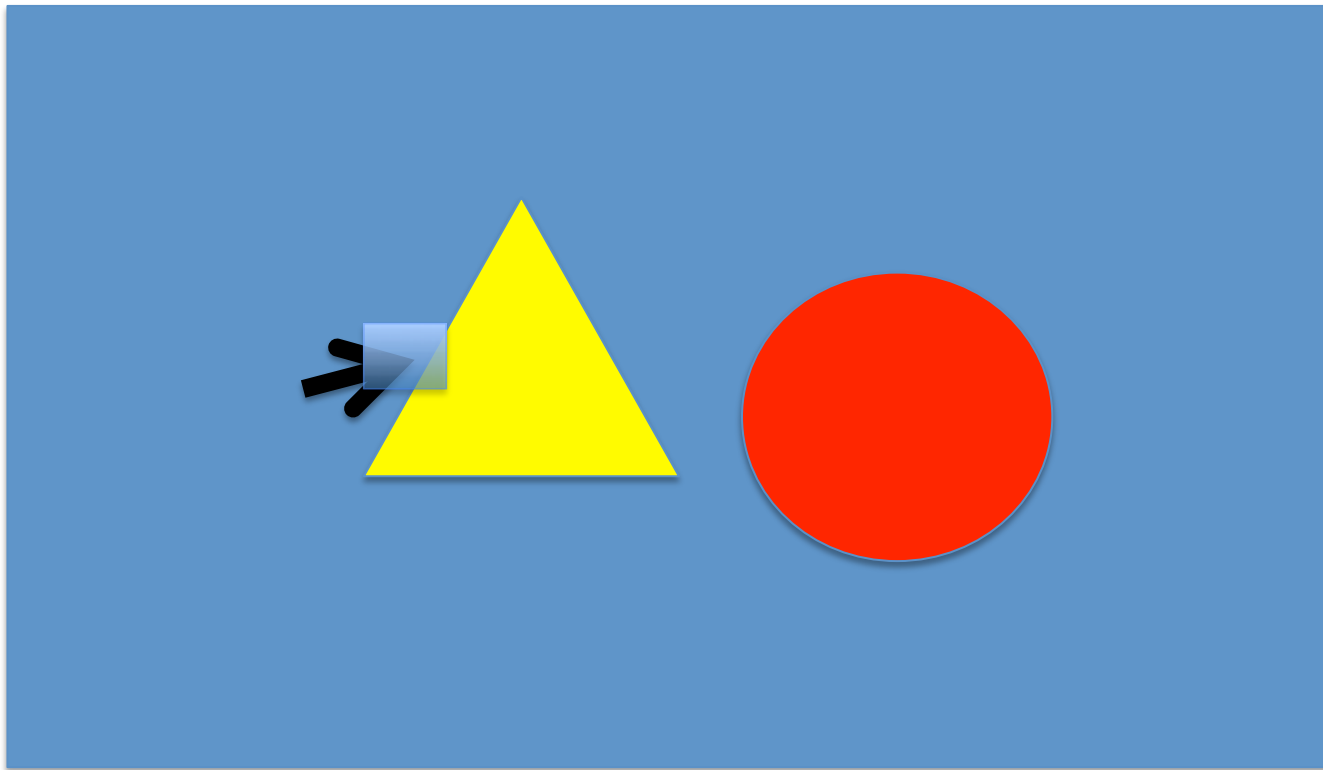
# The Basic Concept

- OpenGL can tell you what objects are being rendered to the viewport
  - Remember if objects fall outside the world window, they are not displayed
  - Another name for the world window is 'clipping volume'
- How is this related to 'picking'?

Create a tiny little clipping volume ...



Create a tiny little clipping volume ...



Create a tiny little clipping volume ...



# What does OpenGL Tell You?

- If you give each object a name, then OpenGL will tell you the names of the objects that are 'hit'
  - Here 'hit' means 'being rendered to' the window
- You need to supply a buffer for the OpenGL to store the hit objects. For example,

```
GLuint selectBuffer[100];  
...  
glSelectBuffer(100, selectBuffer);
```

# Processing Hits

Hits will be recorded in the Select Buffer

```
GLuint selectBuffer[100];  
glSelectBuffer(100, selectBuffer);
```

|              |                   |
|--------------|-------------------|
| 1            | ← number of names |
| 4.2822e+009  | ← min depth value |
| 4.28436e+009 | ← max depth value |
| 2            | ← first name      |
| 2            |                   |
| 4.2732e+009  |                   |
| 4.27334e+009 |                   |
| 6            |                   |
| 1            |                   |
| ...          |                   |

# Where are the names copied from?

- From OpenGL 'name stack'
- 'name stack' is an OpenGL internal data structure to store the names of the objects that are rendered to the display window
- What do I need to do to initialize the name stack?

```
glInitNames();
```

```
glPushName(0); // push 0 on stack
```



# How do I render the objects into that tiny little window??

- In OpenGL, you change the clipping volume by manipulating the 'Projection Matrix', i.e.,  
GL\_PROJECTION
- Normally, you would do:

```
glMatrixMode(GL_PROJECTION);  
glLoadIdentity();  
gluOrtho2D(xmin, ymin, xmax, ymax);  
.... // draw objects
```

# How do I render the objects into that tiny little window??

- Now, you need to specify your pick window before you specify your normal camera, i.e., gluOrtho2D()
- That is,

```
GLint vp[4];  
glMatrixMode(GL_PROJECTION);  
glLoadIdentity();  
glGetIntegerv(GL_VIEWPORT, vp);  
gluPickMatrix(x, y, w, h, vp);  
gluOrtho2D(xmin, ymin, xmax, ymax);  
.... // draw objects
```

# But you really want to see is not the tiny little window ...

- You have to render your objects twice
- First time render the tiny little window (not really shown to you), in GL\_SELECT mode
- Second time render the objects using your normal setting, in GL\_RENDER mode

```
glRenderMode(GL_SELECT);  
... // prepare name stack  
... // render tiny little window  
glRenderMode(GL_RENDER);  
... // render regular scene
```

# It is something like this ...

In your mouse call back function when you try to pick

```
glRenderMode(GL_SELECT);
glInitNames();
glPushName(0);
glSelectBuffer(100, selectBuffer); // assuming you have declared selectBuffer
glMatrixMode(GL_PROJECTION);
glPushMatrix();
glLoadIdentity();
gluPickMatrix(.....);
gluOrtho2D(...);
//draw your objects with names here ....
glMatrixMode(GL_PROJECTION);
glPopMatrix();
int hits = glRenderMode(GL_RENDER);
//resume normal drawing ....
```

# Wait, how do I give names to my objects

In your drawing routine

```
glLoadName(1);  
// draw whatever you want .....
```

```
glLoadName(2);  
// draw whatever you want .....
```

These names will be copied to your name stack

# Processing Hits

Hits will be recorded in the Select Buffer

```
GLuint selectBuffer[100];  
glSelectBuffer(100, selectBuffer);
```

|              |                   |
|--------------|-------------------|
| 1            | ← number of names |
| 4.2822e+009  | ← min depth value |
| 4.28436e+009 | ← max depth value |
| 2            | ← first name      |
| 2            |                   |
| 4.2732e+009  |                   |
| 4.27334e+009 |                   |
| 6            |                   |
| 1            |                   |
| ...          |                   |