



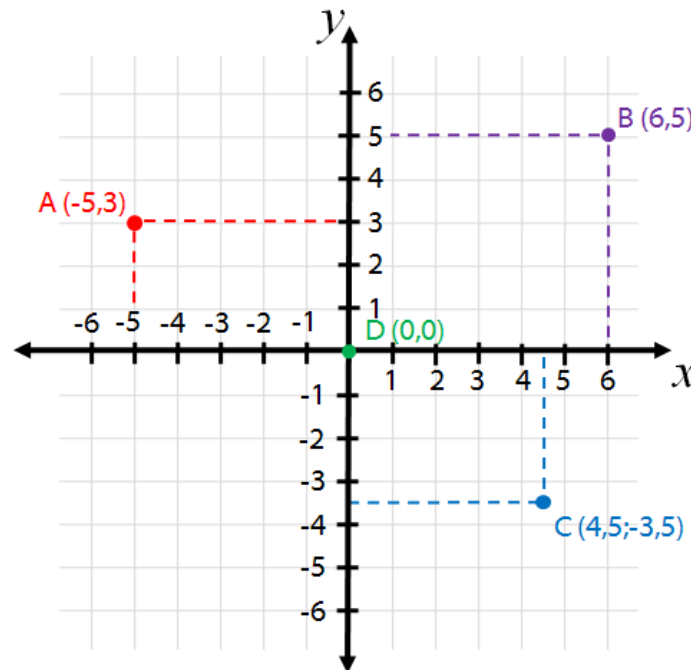
Visualização Bidimensional – Parte 1

Rossana Baptista Queiroz

Sistema de Referência do Universo (SRU)

2

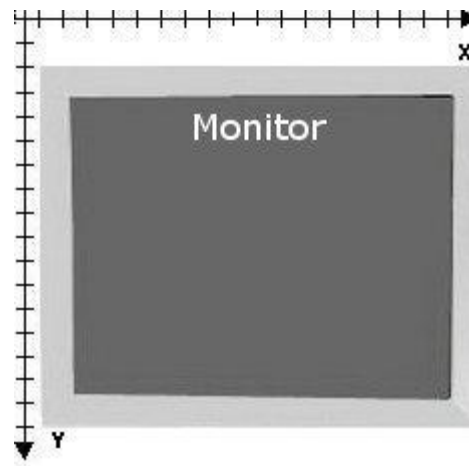
- Região do espaço utilizada na aplicação
- Normalmente é adotado o sistema cartesiano
 - Todos os modelos e comandos OpenGL são definidos em relação a este sistema de referência



Sistema de Coordenadas de Tela (SRT)

3

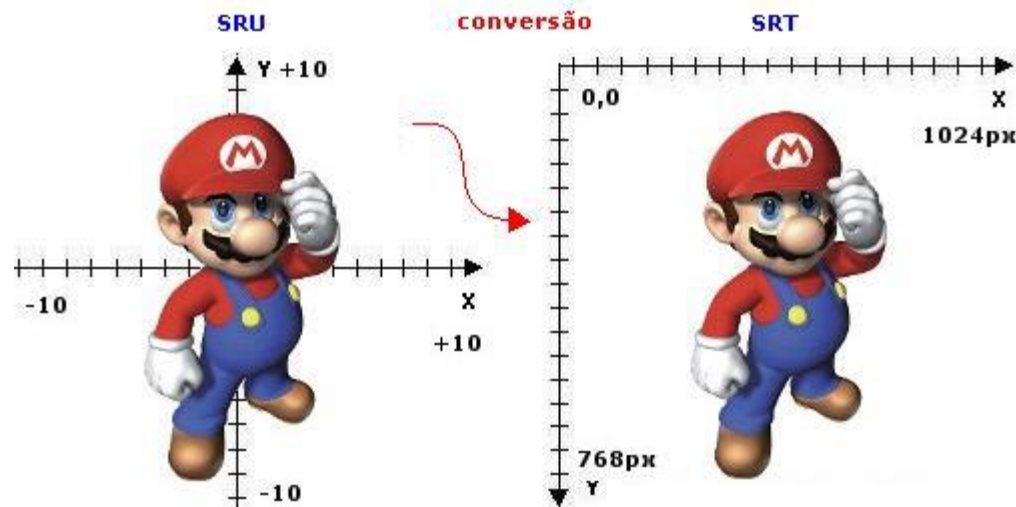
- No monitor do computador é adotado o SRT
- No SRT, a origem fica no canto superior esquerdo do monitor



SRU vs. SRT

4

- Para visualizar corretamente modelos definidos no SRU, é necessário fazer uma conversão ou mapeamento



http://www.bdjogos.com/conteudo.php?link=capitulo_45.php

SRU vs. SRT

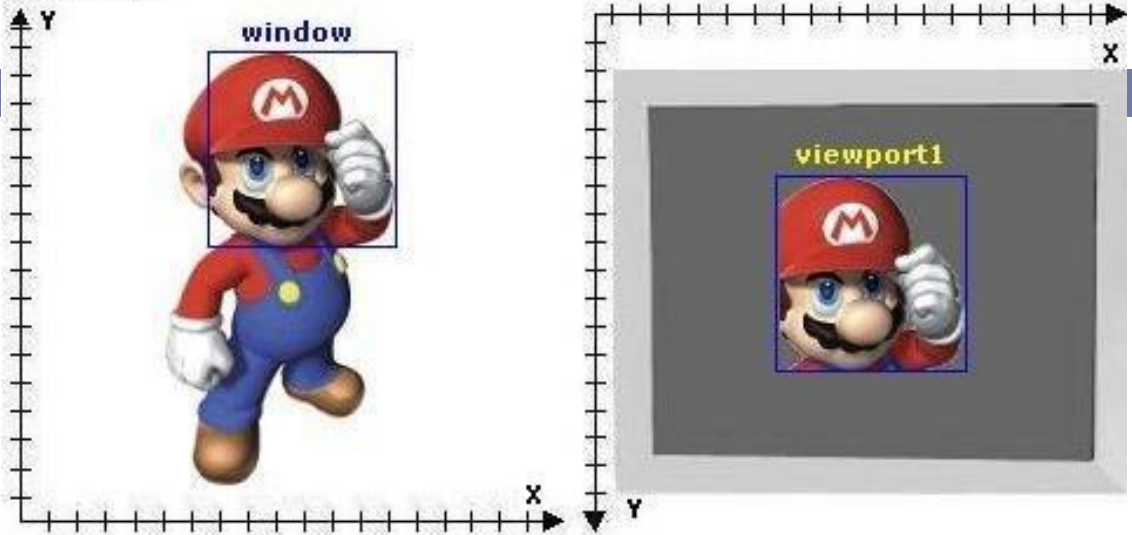
5

- Como o universo é infinito é preciso especificar qual porção queremos mapear na tela
- A essa área que delimita a área de interesse do usuário chamamos de **window** ou janela de seleção.
- Uma **window** é delimitada através das coordenadas de seus cantos (esquerdo, direito, superior, inferior) no **sistema SRU**
- De forma análoga também é necessário definir **em que parte desejamos exibir o conteúdo na window no monitor**
- Essa região é chamada de **viewport** ou janela de exibição, uma **viewport**. Em OpenGL, uma *viewport* normalmente é delimitada pelo tamanho da janela GLUT correspondente ao tamanho da resolução do monitor.

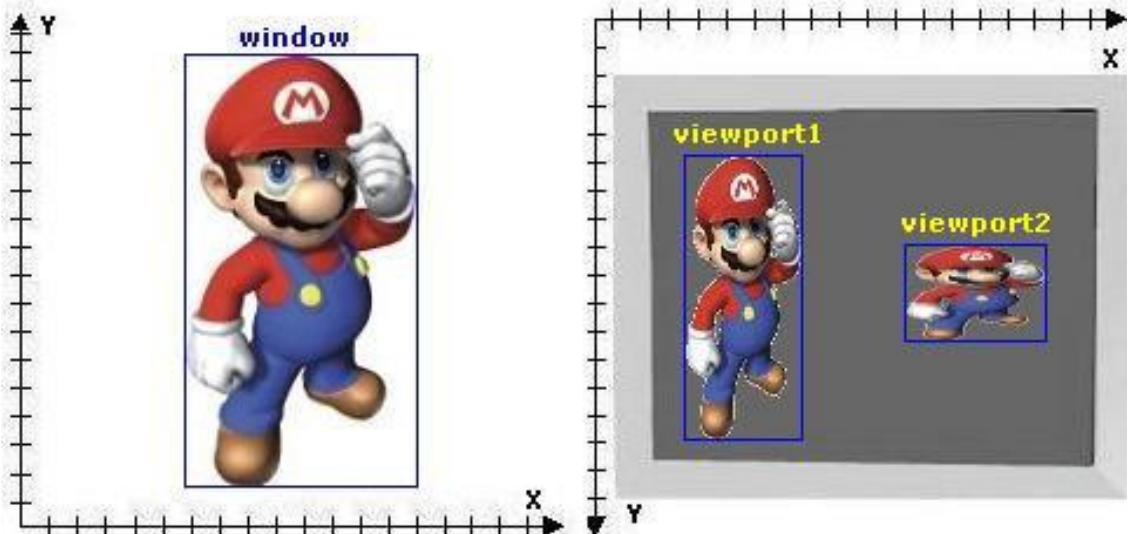
SRU vs. SRT

6

EXEMPLO 1



EXEMPLO 2



SRU vs. SRT

7

- Para realizar o mapeamento do SRU para o SRT em OpenGL, é necessário especificar a **viewport** e a **window**.

```
gluOrtho2D (Gldouble left, Gldouble right,  
Gldouble bottom, Gldouble top);
```

Essa função serve para definir a **window** quando se está trabalhando com desenhos 2D. Seus parâmetros correspondem especificamente cada borda da window. Isto é: x mínimo (borda esquerda - left) – x máximo (borda direita - right) – y mínimo (borda inferior - bottom) – y máximo (borda superior - top).

SRU vs. SRT

8

□ Exemplo: casa.cpp

```
// Estabelece a janela de seleção (esquerda, direita, inferior,  
// superior) mantendo a proporção com a janela de visualização  
if (largura <= altura)  
    gluOrtho2D (-40.0f, 40.0f, -40.0f*altura/largura,  
40.0f*altura/largura);  
else  
    gluOrtho2D (-40.0f*largura/altura, 40.0f*largura/altura,  
-40.0f, 40.0f);
```

Altura e largura é o tamanho da janela da GLUT em pixels

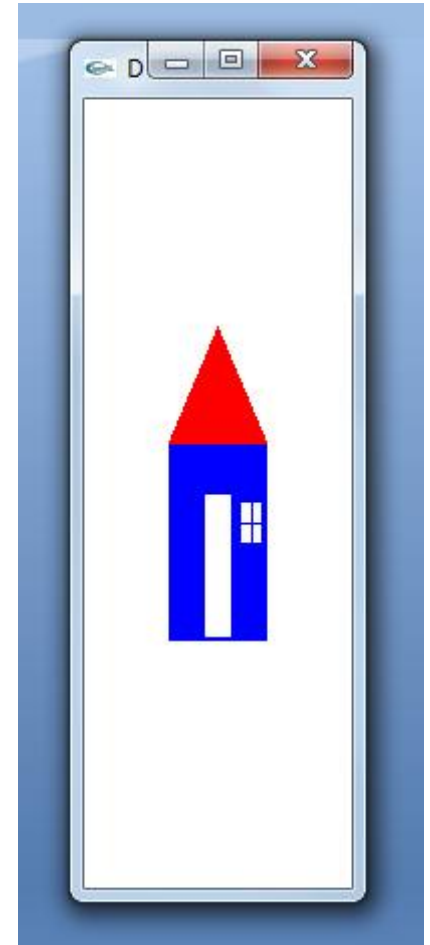
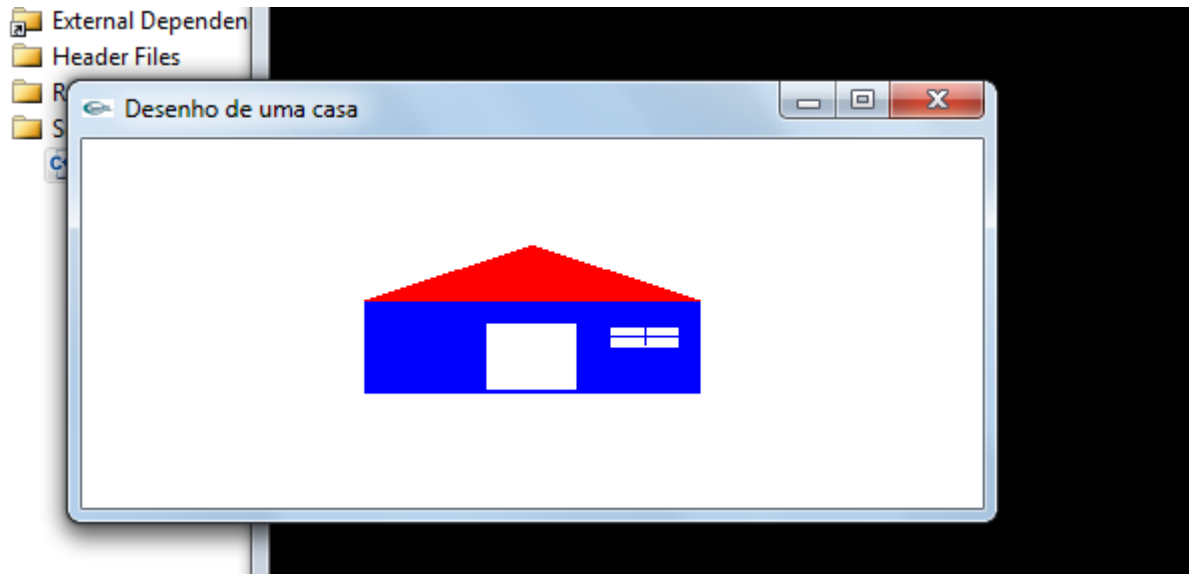
A relação largura/altura (chamada de aspecto) garante que a *window* será definida de maneira que a imagem final não se deforme

SRU vs. SRT

9

□ Exemplo: casa.cpp

Se não fizéssemos a correção do aspecto...



SRU vs. SRT

10

```
void glViewport( GLint x, GLint y, GLsizei  
width, GLsizei height);
```

Serve para definir a **viewport**. Os valores especificados nos dois primeiros parâmetros x e y definem a posição da janela na tela (canto inferior esquerdo) enquanto os dois últimos definem a largura e a altura dessa tela.

Se você não especificar viewports no seu programa, o OpenGL assume o viewport default e cria uma para trabalhar com toda a tela.

SRU vs. SRT

11

□ Exemplo: casa.cpp

```
// Especifica as dimensões da Viewport  
glViewport(0, 0, largura, altura);
```

SRU vs. SRT

12

□ Exemplo: DuasViewports.cpp

Em AlteraTamanhoJanela(GLsizei w, GLsizei h):

```
// Atualiza as variáveis  
largura = w/2;  
altura = h;
```

Em Desenha()

```
// Define a Viewport 1 na metade esquerda da janela  
glViewport(0, 0, largura, altura);  
// Desenha a casa na Viewport 1  
DesenhaCasa();  
  
// Define a Viewport 2 na metade direita da janela  
glViewport(largura, 0, largura, altura);  
// Desenha a casa na Viewport 2  
DesenhaCasa();
```

Referências

14

- COHEN, Marcelo, MANSSOUR, Isabel. OpenGL : uma abordagem prática e objetiva. São Paulo : Novatec, 2006. 478 p.
 - Capítulo 7
- Materiais do professor Leandro Tonietto e João Bittencourt