Computing Data Transfer \gg Types of Data

Computing Data Transfer

Types of Data

Vertex Attributes

Buffer objects.

Uniforms. (User declared, automatic, such as transformation matrices)

Texture Objects.

Commands. (Assume amount of data is small per command.)

-1

Computing Data Transfer \gg Data Transfer Boundaries

Data Transfer Boundaries

CPU Memory to GPU Memory (or CPU \rightarrow GPU for short)

If separate video card, crosses PCIe bus in typical current systems.

If CPU and GPU use same memory, no transfer at all.

Memory to GPU

If separate video card, Memory refers to GPU memory.

If CPU and GPU use same memory then Memory refers to that memory.

Computing Data Transfer \gg Data Transfer Boundaries

Illustration:

Consider a buffer object to hold sphere coördinates.

It is transferred from CPU \rightarrow GPU Memory only once.

It is transferred to GPU each time a sphere rendered \dots

... which can be many times per frame.

-3

Computing Data Transfer \gg Data Origination

Data Origination

How to Find:

Look for OpenGL calls that result in data transfer.

Vertex Attributes:

glVertex, glNormal,

If these are in an array, work out number of iterations.

For glDrawArrays, account for number of elements.

If data is sourced from a buffer object, only account for transfer once for client arrays for each rendering pass. Computing Data Transfer \gg Data Origination

Amount of Data
Tally the sizes.
glVertex3f(): 3 * sizeof(float) = 3 * 4 = 12 B.
Etc.
Also tally buffer objects used by the code.
Assume that a buffer object data sent from CPU to GPU only once per load.

Uniforms:

Sent just once per rendering pass.

Example: 2014 Homework 4

Example: 2014 Homework 4

Render Spiral 1:

n: chain length.

s: opt_segments.

Vertex attributes emitted in a loop nest. Each iteration:

Total: 64 s(n-1) B

Render Spiral 2: