

Other Useful Classes in the Ray Tracing Framework

class Vector

A 3-vector implementing various operations (+, -, *, /, dot, cross)

class Point

A 3D point implementing various operations (+, -, *, /, dot)

class Image

A very fragile class! Used to store image output from batch renders

class Matrix

Implementing 3x3, 4x3, or 4x4 matrix operations (depending on your ray tracers needs)

class Array1D

A simple dynamic 1D array class. Be very careful where you use this, due to the dynamic memory allocation

class Array2D

You may or may not want to define a dynamic 2D array class. This can be *very* detrimental to speed if used poorly.

class RGBColor

Class implementing operations on a RGB color triple (+, -, *, /, etc)

class Spectrum

Class implementing a spectral color class and its operations

class Color

A typedef to another class describing the color space used

class Thread

Class storing thread-specific data. E.g., each thread needs a unique random num. generator, separate ray storage, etc.

class Random

Class implementing a pseudo-random number generator.

class Ray

Really a container structure for ray data needed by multiple other methods during execution (*object intersection, hitpoint shading, normal computation*)

class Semaphore

A basic control mechanism for controlling access to data needed by multiple CPUs.

class Barrier

A basic control mechanism to synchronize multiple threads after completing their tasks.