

Initial Experiences of OpenCL

Sam Proctor, PhD Student, University
of Manchester

My Path to Using OpenCL

Image Processing on a Mac

Core Image

Multiple Threads

Grand Central Dispatch

OpenCL

Prototyping Tools

Quartz Composer

Sandbox Environment

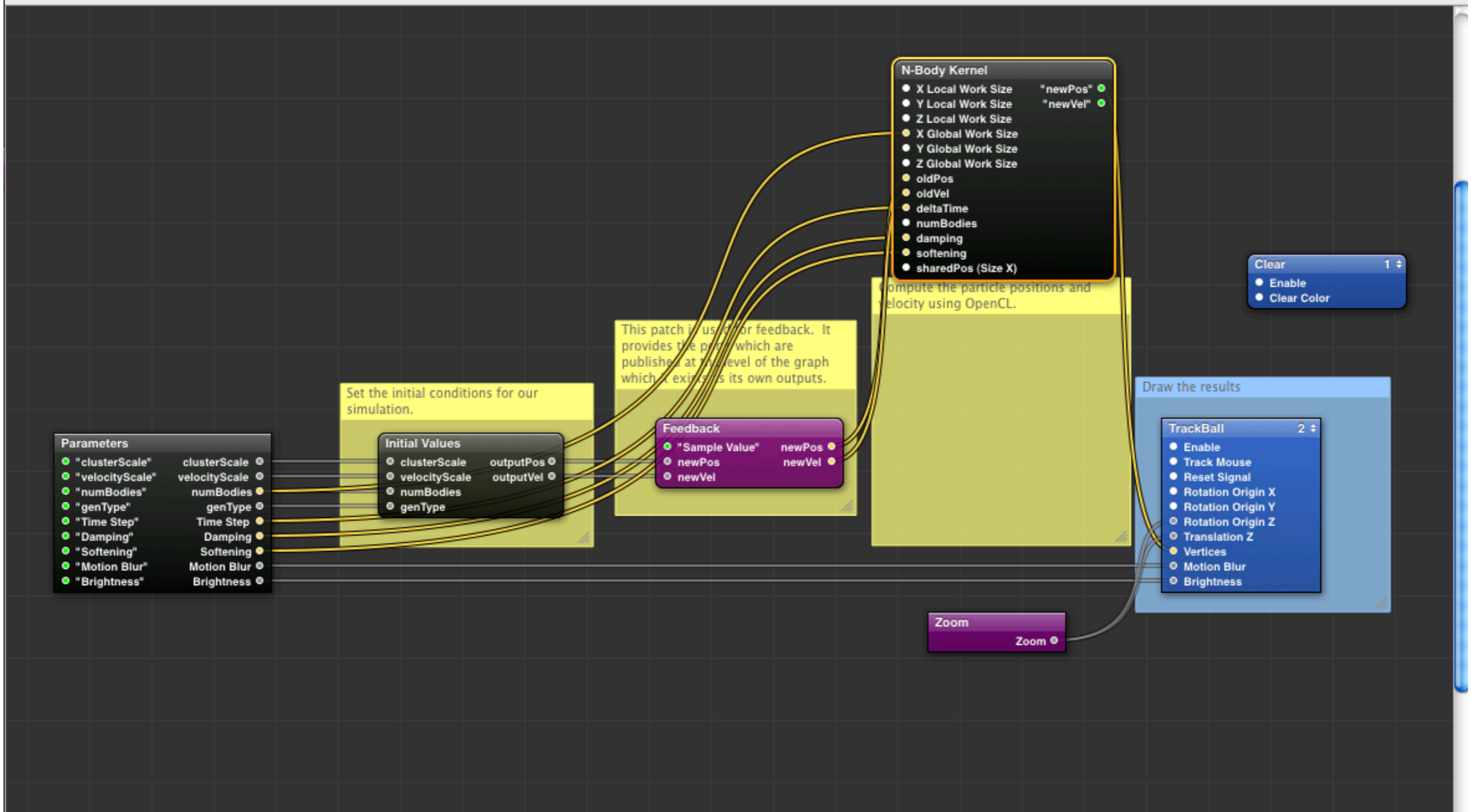
Create and test OpenCL Kernels

Wide range of options to import data

Easy setup of GPU

Quite Stable

Root Macro Patch



```

Inspector
Settings
Show Advanced Kernel Settings
main
1  /*
2  The following table illustrates how to declare inputs and outputs to the kernel:
3
4  Inputs:
5  Scalar   : <type> <name>           : e.g. float myInputValue
6  Array    : __global const <type> *<name> : e.g. __global const float4 *myInputValues
7  2D Image : __rd image2d_t <name>       : e.g. __rd image2d_t srcimg
8  3D Image : __rd image3d_t <name>       : e.g. __rd image3d_t srcimg
9
10 Outputs:
11 Scalar   : not supported
12 Array    : __global <type> *<name>     : e.g. __global float4 *myOutputValues
13 2D Image : __wr image2d_t <name>       : e.g. __wr image2d_t dstimg
14 3D Image : not supported
15
16 Declaring a local thread group array will result in an input of the form "<name> Size X" which is used to set the size of the array.
17 Local    : __local <type> *<name>      : e.g. __local float4 *sharedValues
18 */
19
20 /* A simple OpenCL Kernel that returns an input float4 array multiplied by a constant. */
21 /*
22 __kernel void main(__global const float4 *src, float x, __global float4 *dst)
23 {
24     int tid = get_global_id(0);
25     dst[tid] = src[tid] * (float4)(x, x, x, 1.);
26 }
27 */
28 /* The OpenCL kernel below does the same constant multiplication on an image. */
29
30 __kernel void main(__rd image2d_t srcimg, float x, __wr image2d_t dstimg)
31 {
32     int2 pos = (int2)(get_global_id(0), get_global_id(1));
33     float4 color = read_imagef(srcimg, CLK_ADDRESS_CLAMP_TO_EDGE | CLK_FILTER_NEAREST, pos);
34     write_imagef(dstimg, pos, color*x);
35 }
36
37
38
39
40

```

OpenCL Context Info

- Devices
 - Device Count
 - Device Type
 - Name
 - Vendor
 - Version
 - Max Compute Units
 - Max Work Item Dimensions
 - Max Work Group Size
 - Max Read Image Args
 - Max Write Image Args
 - Max Samplers
 - Max Constant Args
 - Max Constant Buffer Size
 - Global Mem Size
 - Local Mem Size
 - Image2D Max Width
 - Image2D Max Height

Development of Applications Using OpenCL

Application Development

Troublesome to Setup

Easily crash the device

Kernels stored as plain text files or
simple strings

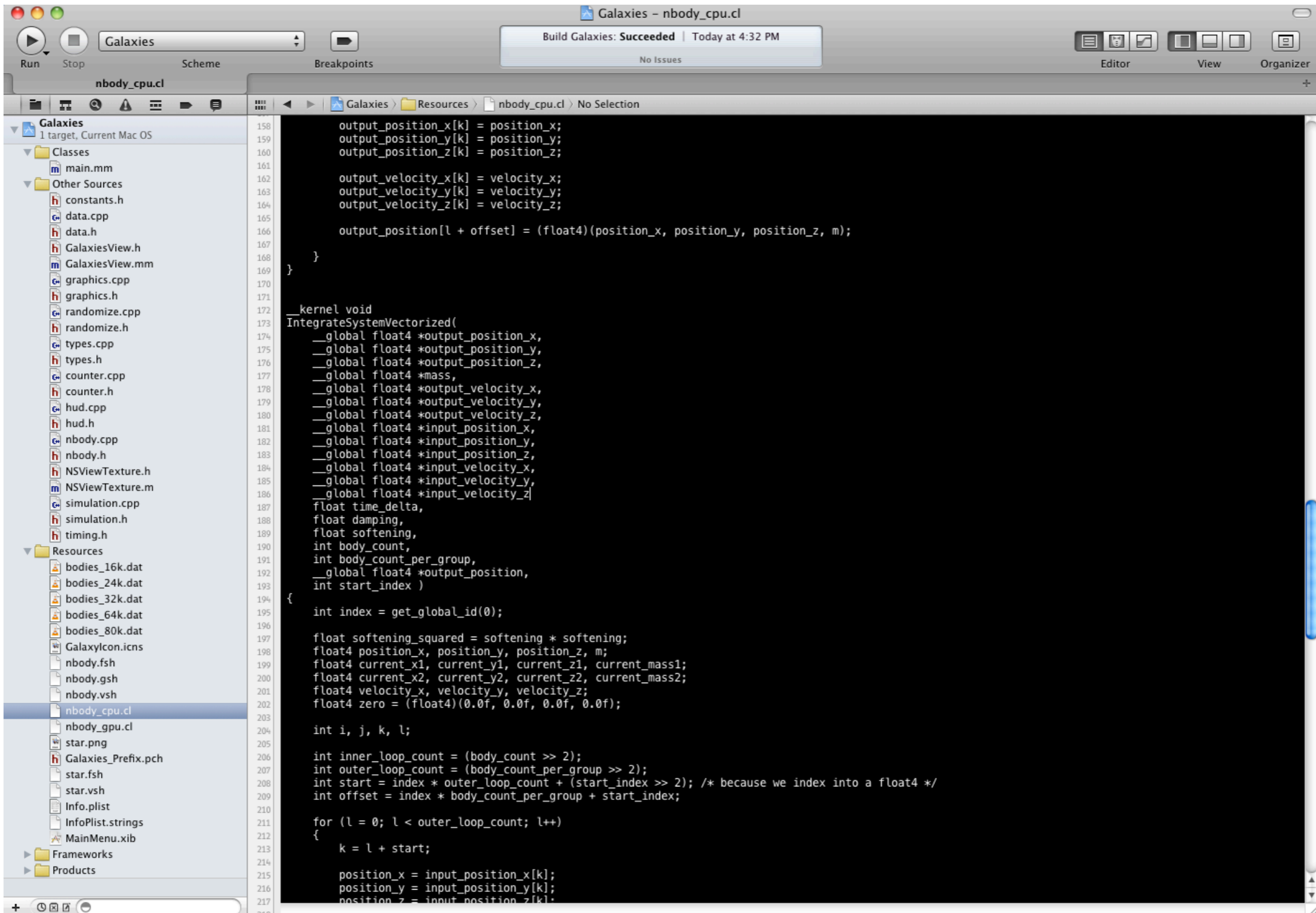
Kernels compiled at runtime

Debugging support not great


```

78
79     [[NSUserDefaults standardUserDefaults] setPersistentDomain:prefs
80         forKey:kBundleIdentifier];
81
82     [prefs release];
83 }
84
85 - (void)prepareOpenGL
86 {
87     NSDictionary *overrides = [[NSUserDefaults standardUserDefaults] dictionaryRepresentation];
88     NSDictionary *prefs = [[NSUserDefaults standardUserDefaults] persistentDomainForName:kBundleIdentifier];
89     id obj = nil;
90
91     if(nil == (obj = [overrides objectForKey:@"fullScreen"]))
92         obj = [prefs objectForKey:@"fullScreen"];
93     fullscreen = obj ? [obj boolValue] : YES;
94
95     if(nil == (obj = [overrides objectForKey:@"initMode"]))
96         obj = [prefs objectForKey:@"initMode"];
97     init_mode = obj ? [obj intValue] : 0;
98
99     if(nil == (obj = [overrides objectForKey:@"initDemo"]))
100        obj = [prefs objectForKey:@"initDemo"];
101    init_demo = obj ? [obj intValue] : 1;
102
103    if(nil == (obj = [overrides objectForKey:@"starScale"]))
104        obj = [prefs objectForKey:@"starScale"];
105    star_scale = obj ? [obj floatValue] : 1.0f;
106
107    if(nil == (obj = [overrides objectForKey:@"showHUD"]))
108        obj = [prefs objectForKey:@"showHUD"];
109    show_hud = obj ? [obj boolValue] : YES;
110
111    if(nil == (obj = [overrides objectForKey:@"showUpdates"]))
112        obj = [prefs objectForKey:@"showUpdates"];
113    show_updates_meter = obj ? [obj boolValue] : NO;
114
115    if(nil == (obj = [overrides objectForKey:@"showFramerate"]))
116        obj = [prefs objectForKey:@"showFramerate"];
117    show_fps_meter = obj ? [obj boolValue] : NO;
118
119    if(nil == (obj = [overrides objectForKey:@"showGigaflops"]))
120        obj = [prefs objectForKey:@"showGigaflops"];
121    show_gflops_meter = obj ? [obj boolValue] : YES;
122
123    if(nil == (obj = [overrides objectForKey:@"showDock"]))
124        obj = [prefs objectForKey:@"showDock"];
125    show_dock = obj ? [obj boolValue] : YES;
126
127    [self savePrefs];
128
129    if(init_mode < 0) init_mode = 0;
130    if(init_mode > 6) init_mode = 6;
131
132    if(init_demo < 0) init_demo = 0;
133    if(init_demo > 6) init_demo = 6;
134
135    InitDefaults(init_demo, star_scale, show_hud, show_updates_meter, show_fps_meter, show_gflops_meter, show_dock);
136    InitGalaxies(init_mode);
137    ResizeCallback([self bounds].size.width, [self bounds].size.height);
138

```



How did I find experience?

Prototyping tools make the experience
much better

Moving onto using it an application
requires a lot of patience

Fits in very well with the rest of the
parallel technologies on this platform

Overall: Positive