

Streaming Languages

Ian Buck

November 6

Brook

- **Stream** data type to C
 - Multidimensional streaming
 - Metacompile to variety of architectures.
 - No assumptions about # of parallel units
 - No assumptions about communication network
 - Compiler & RT library perform scheduling
 - Early in development, many questions unanswered

Types

- Standard C types
- Vector types
 - `vec3f`: 3 component floating point vector
 - `mat4i`: 4x4 matrix of 32-bit integers

Stream Declaration

```
stream Vertex {  
    vec4f pos;  
    vec4f color;  
    vec3f normal;  
    vec2f texture;  
}
```

```
Vertex vtx;  
Vertex triangles [][][3];  
vec3fs framebuffer[1024][1024];
```

Kernel Functions

kernel void

```
vtransform (Vertex vtx, Vertex out tvtx, mat4f matrix) {  
    tvtx.v = matrix * vtx.v  
}
```

Vertex v, tv;

mat4f m = {...};

LoadStream (v, "file.data");

vtransform(v, tv, m);

Multiple Outputs

kernel void

```
lineraster (Line l, Fragment outm f) {  
    while(...) {  
        f.pos = ...;  
        push(f);  
    }  
}
```

- Conditional Output supported
- Allow condition input?
 - Yes, user may need it.
 - No, hurts parallelism too much.

Stencils

kernel void

```
Flux (Flow grid_in[x:-1,1][y:-1,1], Flow out grid_out, float rate);
```

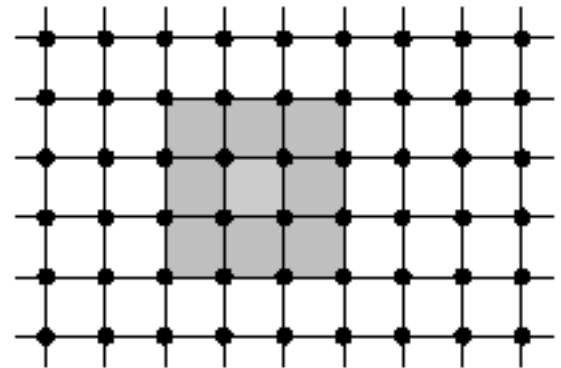
```
Flow grid[100][100];
```

```
Flux(grid, grid, 0.5f);
```

- Allow execution on a domain?

```
Flux(grid[1,98][1,98], ...);
```

Viscous fluxes



Reductions

- Function reduction

```
diverge >?= divergence(grid);
```

- Reduction variables

```
kernel void
```

```
maxmin(floats a, reduce float max, reduce float min) {
```

```
    max >?= a;
```

```
    min <?= a;
```

```
}
```

- User defined reductions?
 - Allow non-associative operations?
 - Examples?

Array Correlation

- Use array indices to indicate strides and dependencies.

kernel void

```
matrixmult(reduce c[i,j], a[k,j], b[i,k]) {  
    c += a*b;  
}
```

for i=0 to n-1

for j=0 to n-1

for k=0 to n-1

c[i,j] += a[k,j]*b[i,k]

Issues

- Need for large streaming examples.
- Stream management
- Stream Transformations or Sorting