

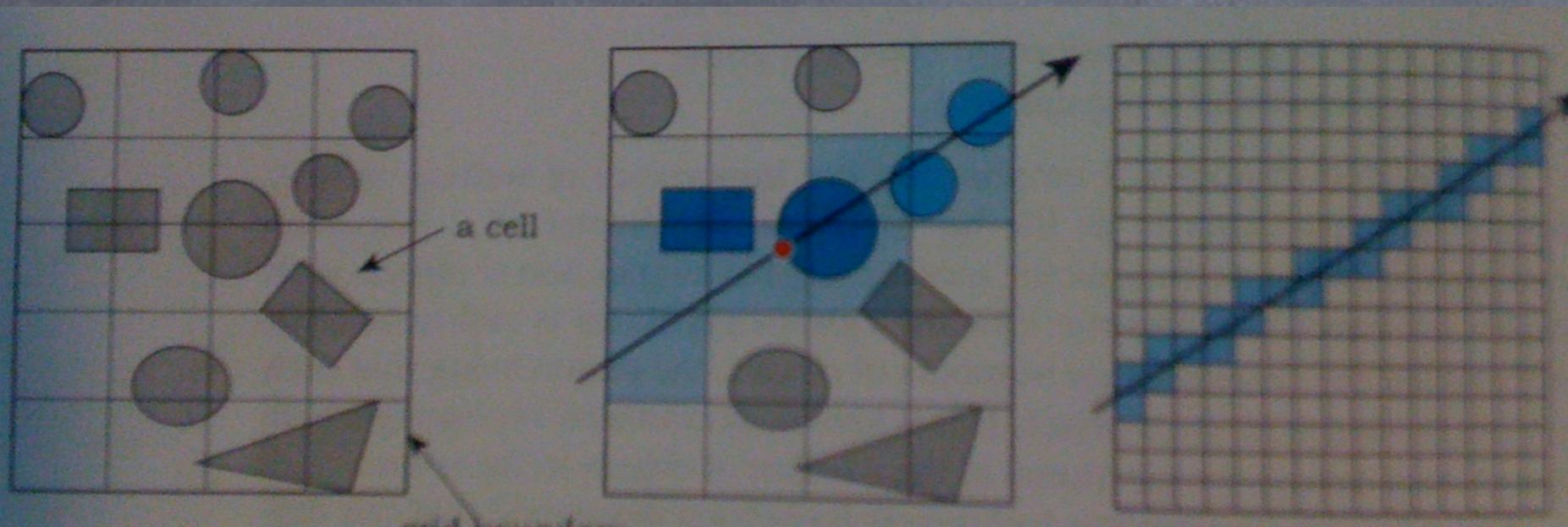
Regular-Grid Accelerated Ray-Tracing Using CUDA

By: Krunal Patel

Simple Ray-Tracer

- ⦿ Spheres and Rectangles
- ⦿ Triangle Meshes via PLY format
- ⦿ Phong Shading with Shadows
- ⦿ Multiple light sources
- ⦿ Anti-Aliasing via regular sampling
- ⦿ CPU vs GPU references
- ⦿ Brute vs Grid acceleration schemes

Regular Grid Acceleration



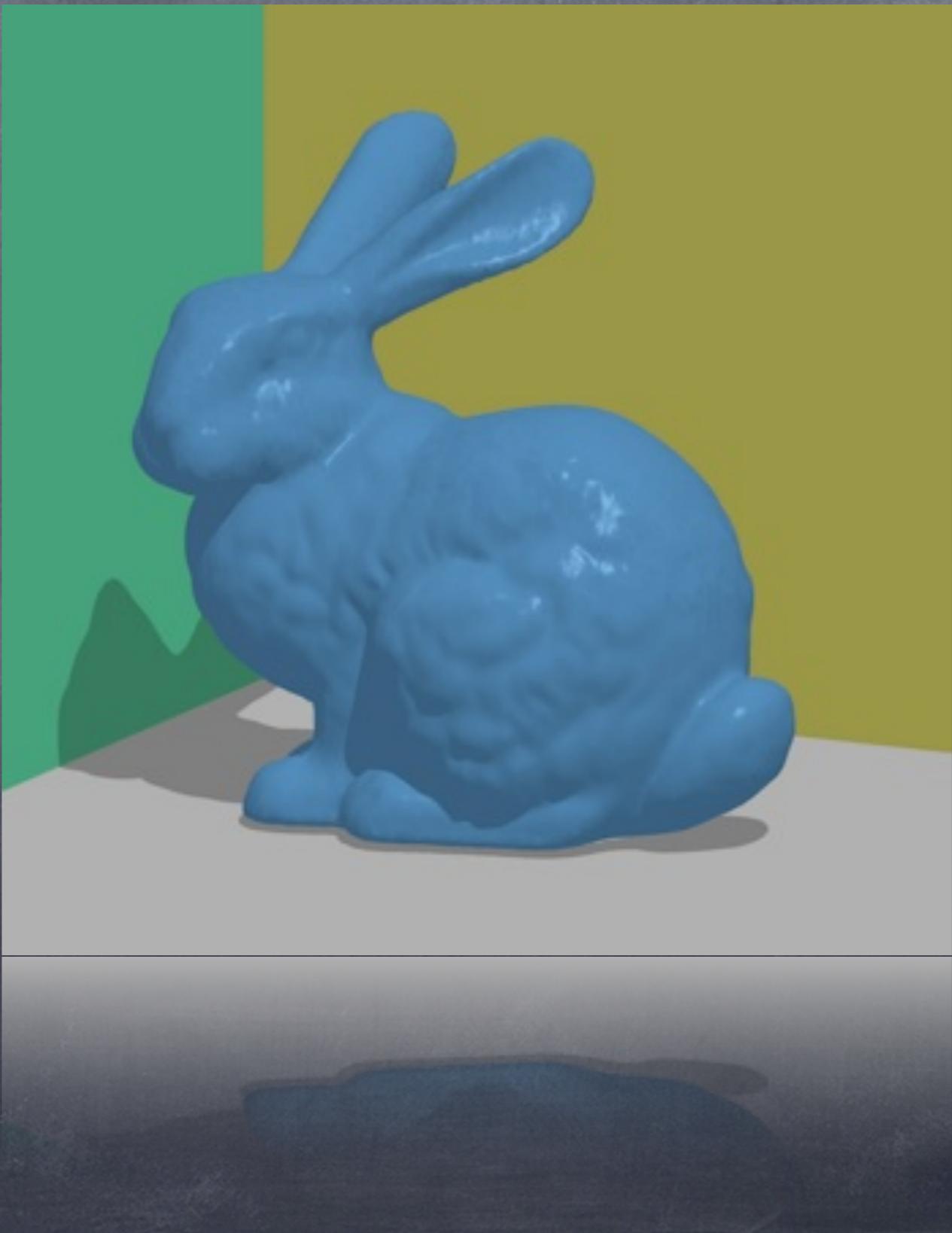
- ⦿ Objects placed in cells according to their bounding-box
- ⦿ Creation done on CPU; Traversal on GPU
- ⦿ Many early opt-out scenarios results in less branchy code to execute
- ⦿ First intersection found is the final one

Data

- ⦿ lots of spheres!
- ⦿ Megabytes of data transferred to device
- ⦿ Stanford Bunny 69K polygon model
- ⦿ Different test scenarios to highlight CUDA and regular grid acceleration scheme
- ⦿ Not using pinned memory

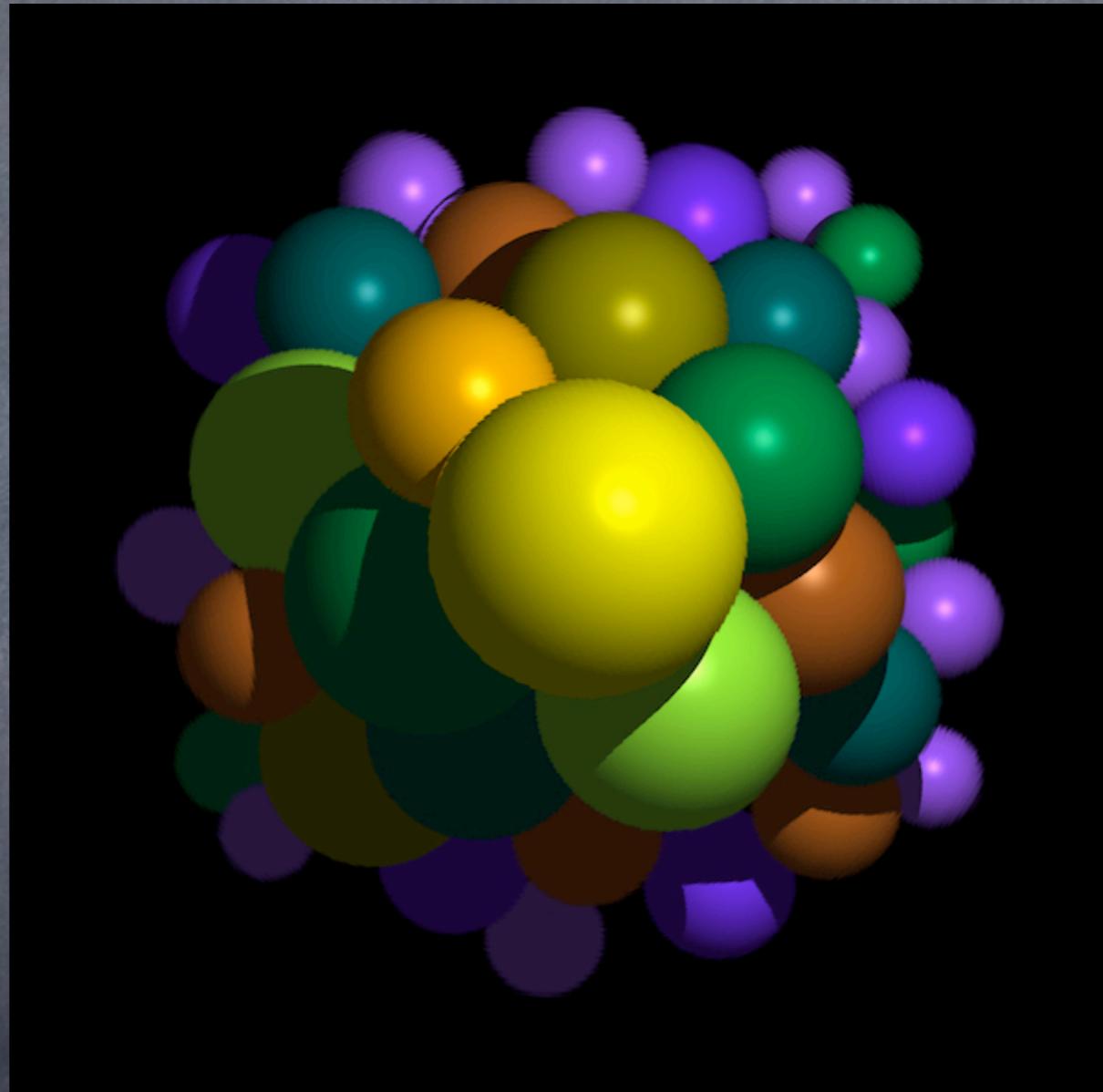
Performance (Grid Only)

- res: upto 3000x3000
- samples: 32
- pinned: No



Build01

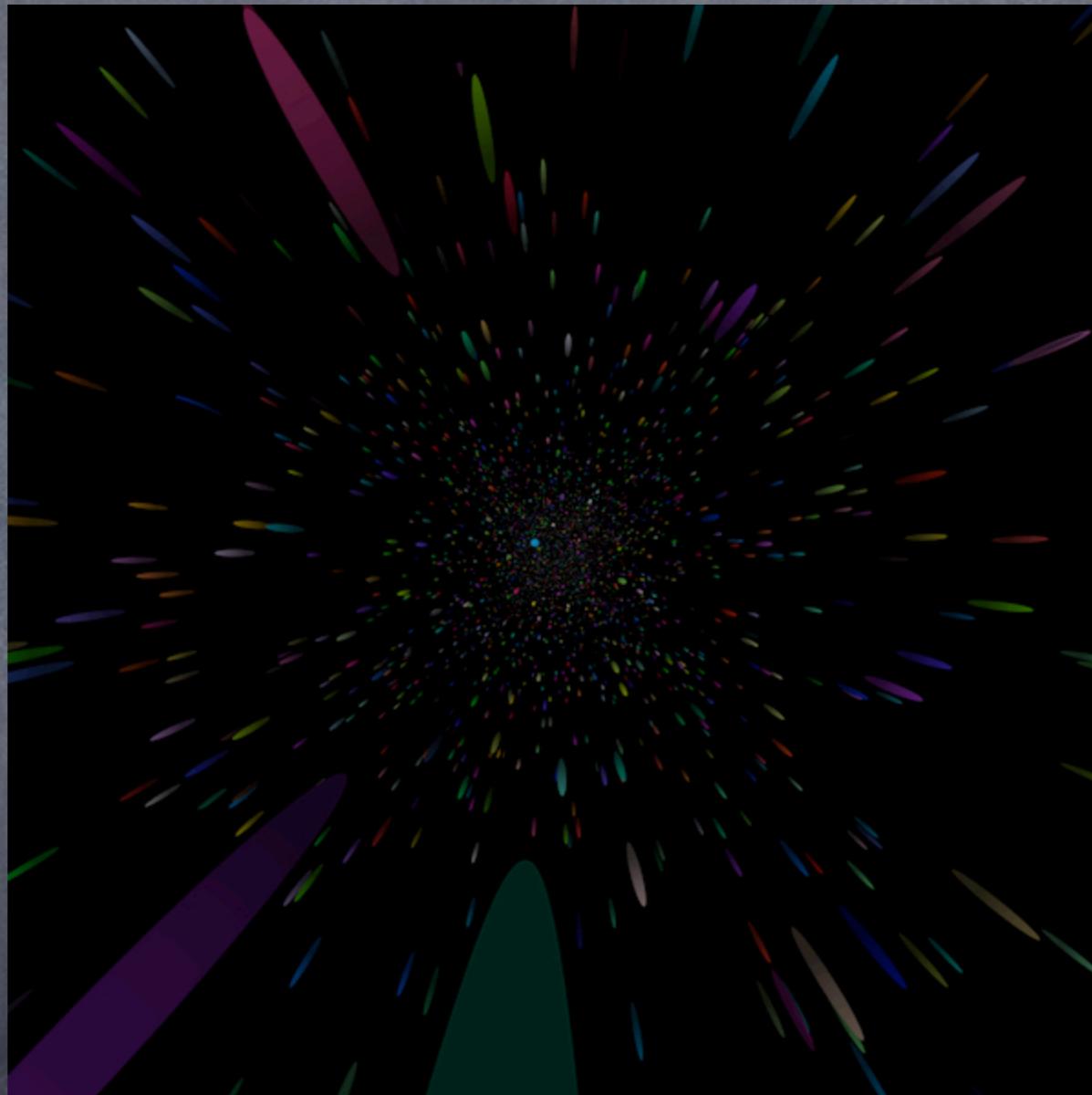
- 307 Objects
- 161s CPU
- 5s GPU



50x speedup

Build02

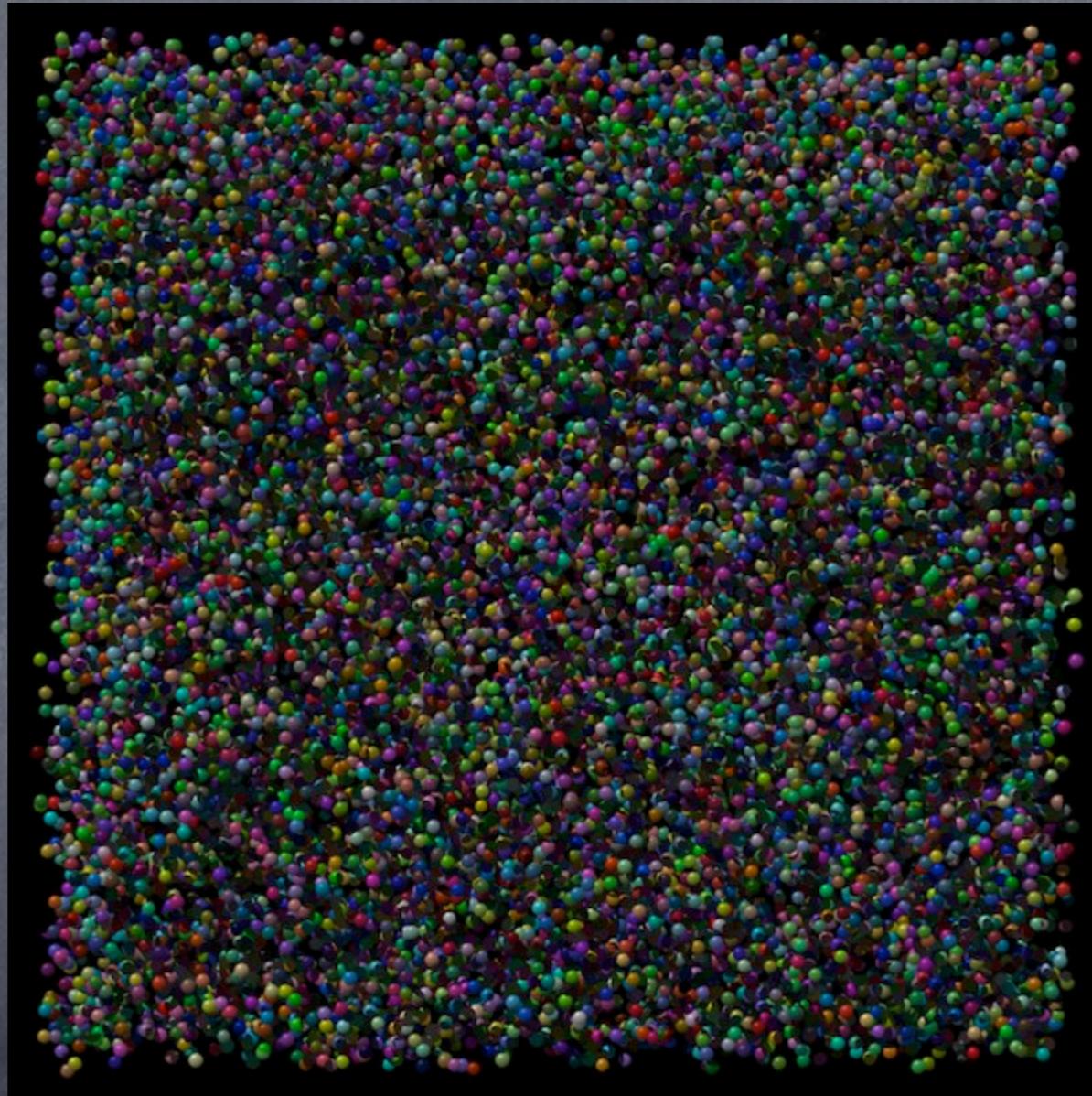
- 24K Spheres
- 200s CPU
- 8s GPU



25x speedup

Build03

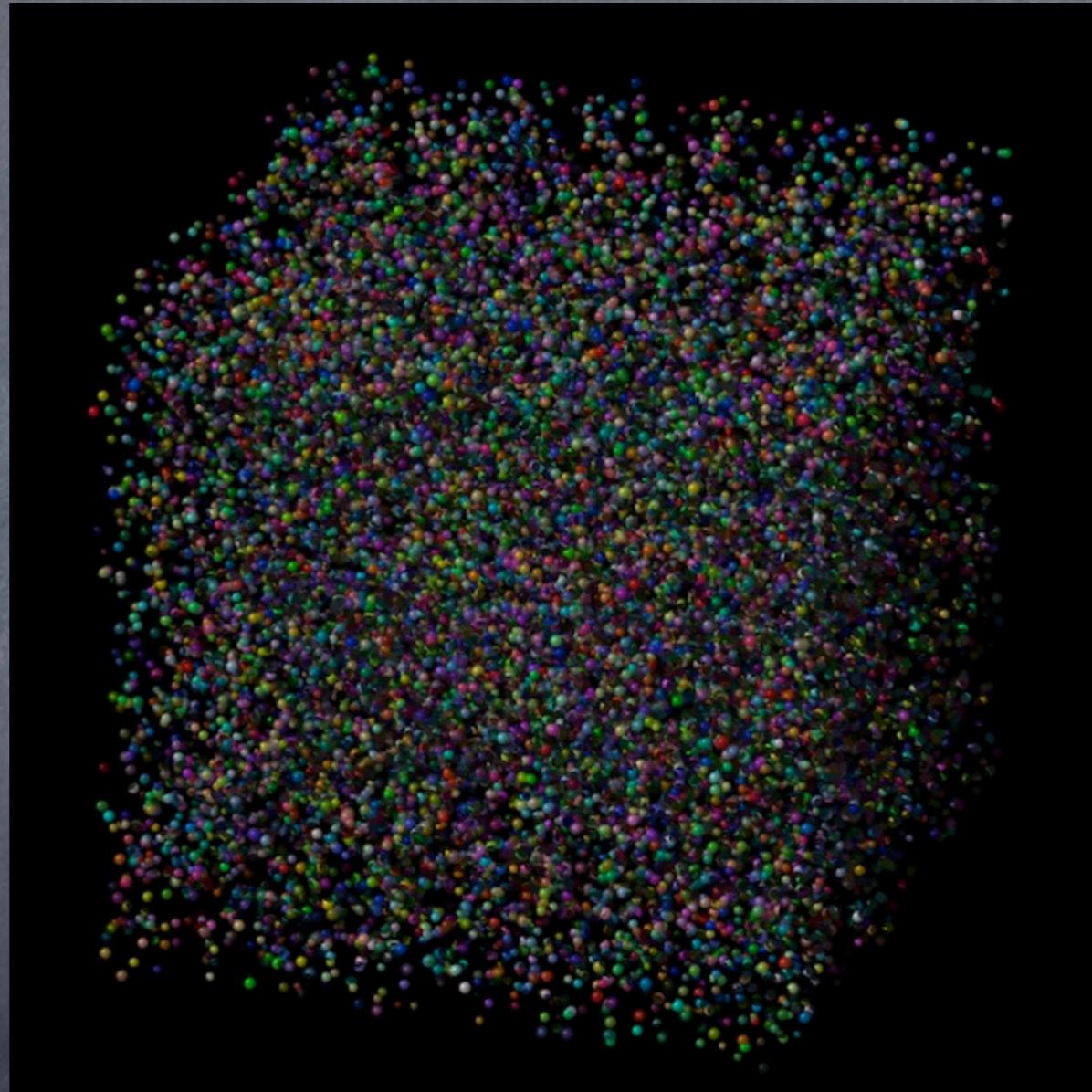
- 216K Spheres
- 60s CPU
- 15s GPU



4x speedup

Build04

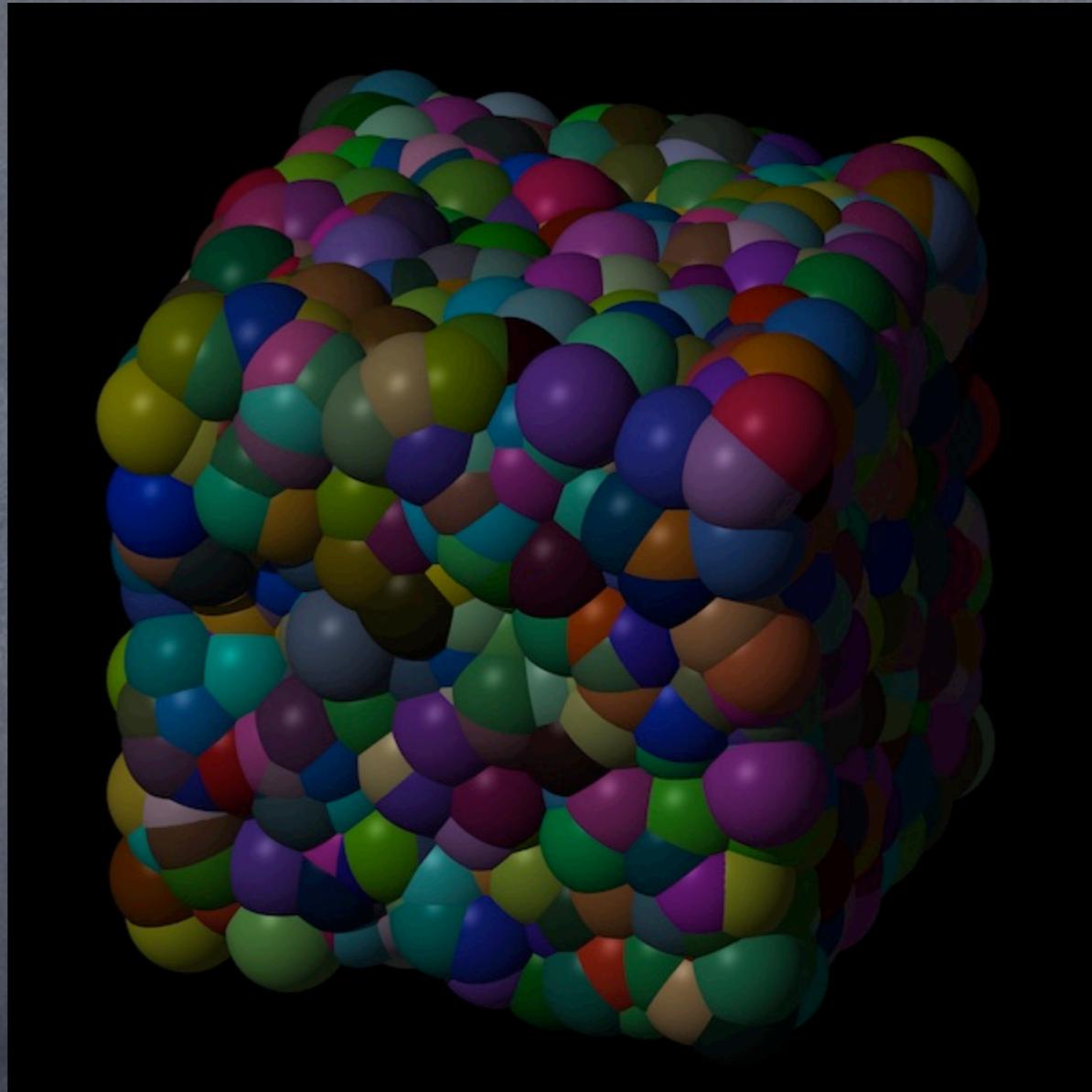
- 216K Spheres
- 270s CPU
- 45s GPU



6x speedup

Build05

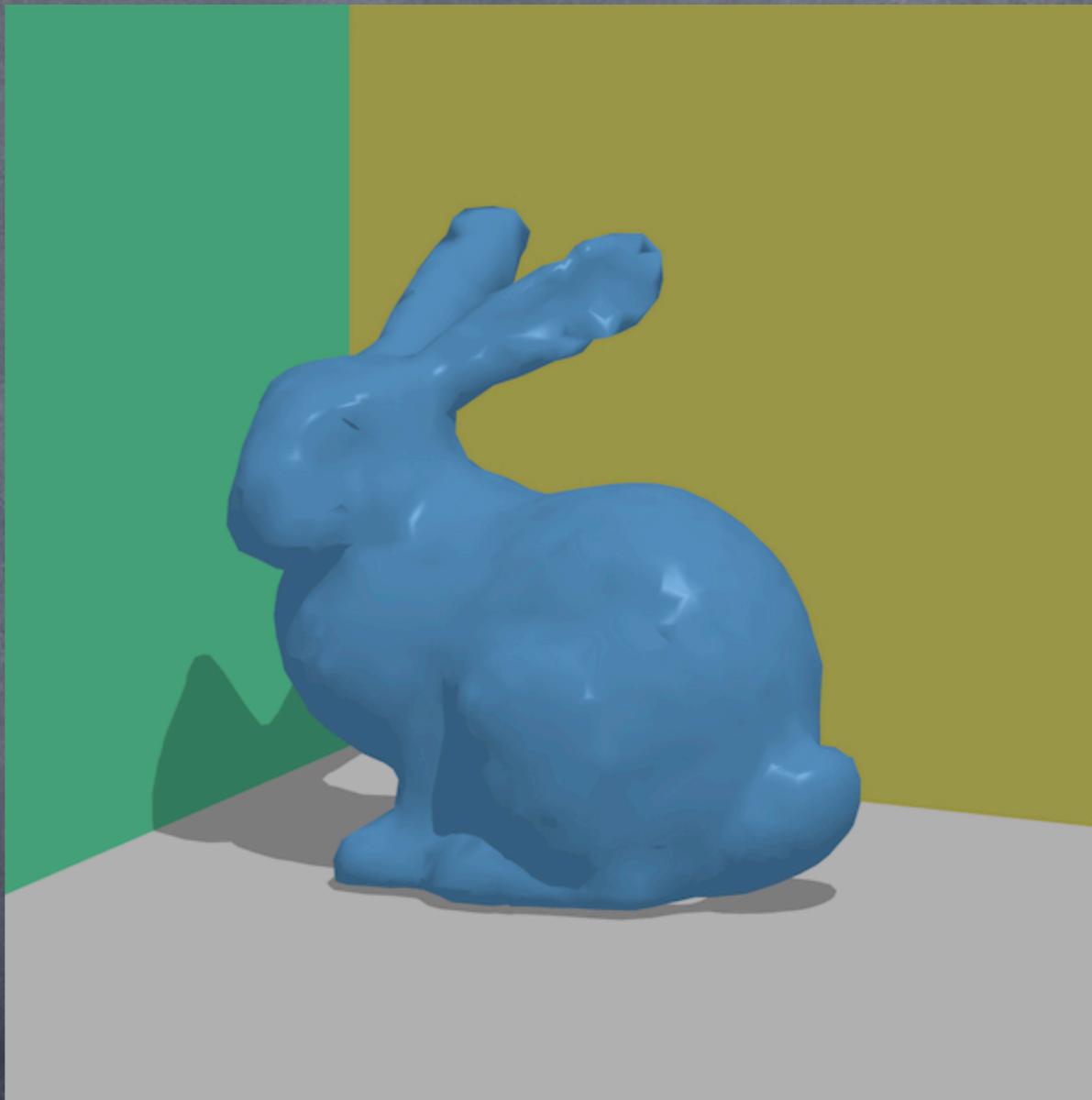
- 440K Spheres
- 70MB of Data
- 141s CPU
- 8s GPU



18x speedup

Build06

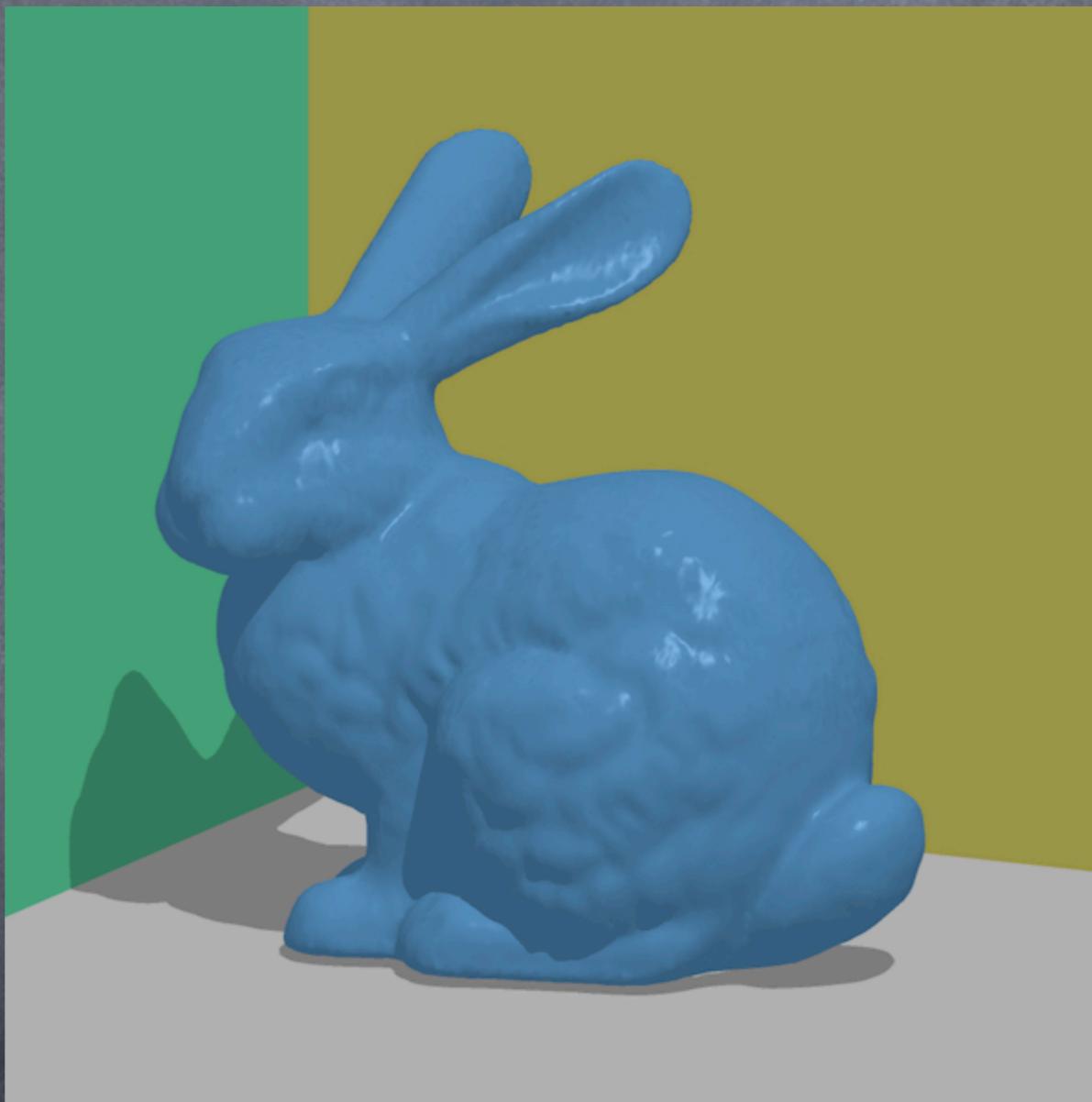
- 18K Objects
- 4K tris
- 30MB of Data
- CPU N/A
- 60s GPU



Over 5x speedup

Build07

- 314K Objects
 - 70K tris
 - 50MB of Data
 - CPU 407s
 - 310s Total
- GPU
- 261s Memory
alone!



1x speedup

Build08

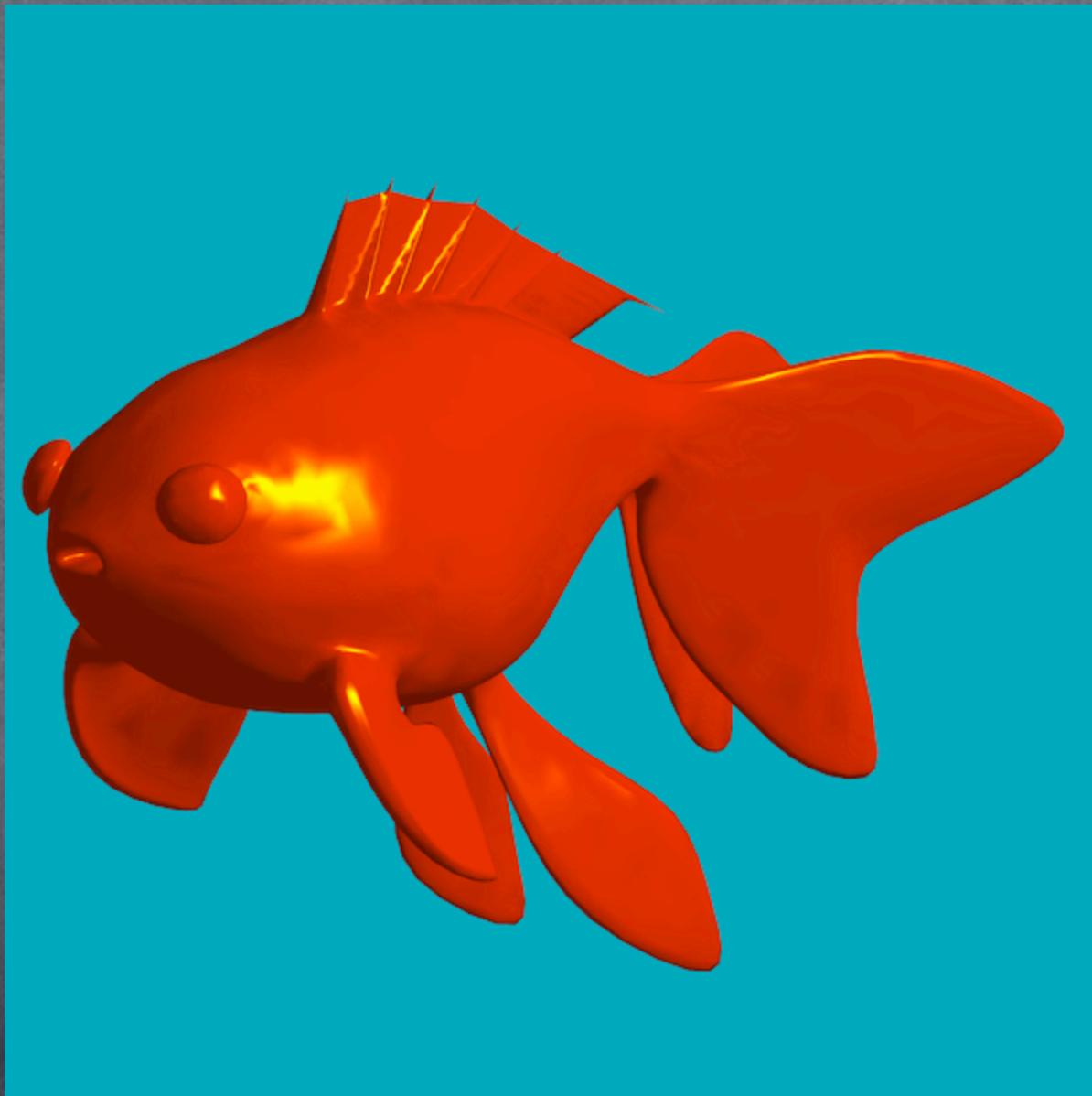
- 400K Objects
- 100K tris
- 60MB of Data
- 77s CPU
- 15s GPU



Over 5x speedup

Build09

- 80K Objects
- 22K tris
- 13MB of Data
- 114s CPU
- 19s GPU



Over 6x speedup

Improvements

- ⦿ Better use of memory in the grid structure
- ⦿ Thread per sample instead of pixel and sum
- ⦿ Double-precision!

Conclusion

- ⦿ Saw overall gains using CUDA on a ray-tracer
- ⦿ CUDA adapted nicely with branchy code
- ⦿ Grid acceleration helped further reduce memory operations
- ⦿ Looking forward to V3.0 of the SDK!!!

Acknowledgements

- ⦿ Kevin Suffern the author of "Ray Tracing from the Ground Up"
- ⦿ <http://www.raytracegroundup.com/>
- ⦿ Our TF Kevin Dale for setting me on the right track for my project

