LA-UR-14-24432
Approved for public release; distribution is unlimited.

Title: Simulating Floods in Virtual Reality

Author(s): Tasseff, Byron A.
Judi, David R.

Intended for: 7th International Congress on Environmental Modelling and Software, 2014-06-15/2014-06-20 (San Diego, California, United States)

Issued: 2014-06-16

Simulating Floods in Virtual Reality

Byron Tasseff & David Judi

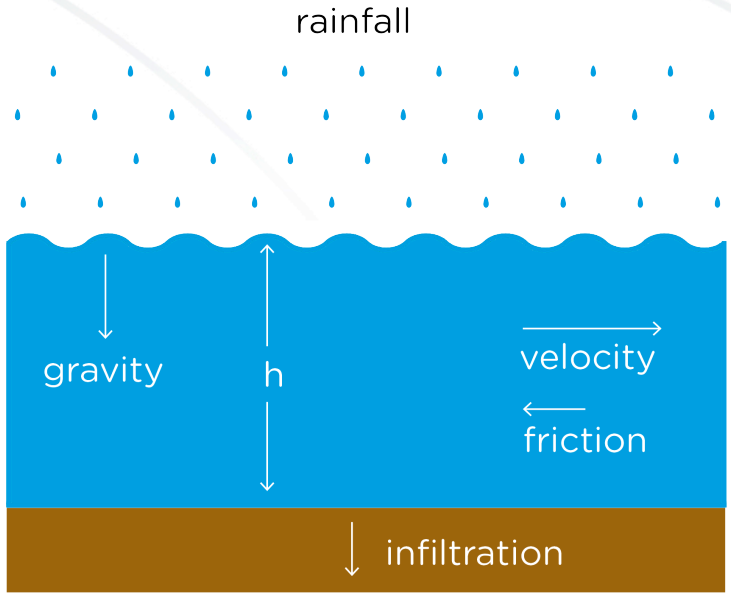
June 17, 2014

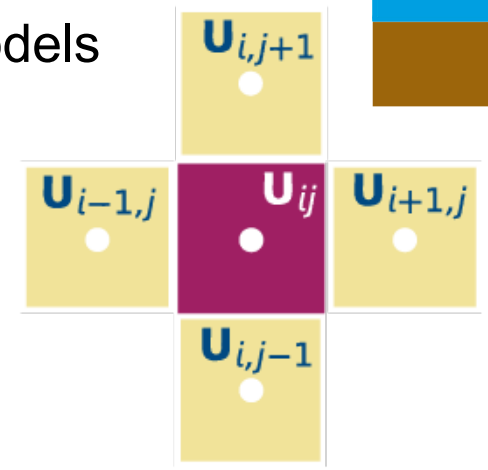
Energy and Infrastructure Analysis @ LANL

- Increase security of the nation's infrastructure
  - Electric power
  - Transportation (e.g. roads)
  - Telecommunications
  - Water
    - Water resource planning
    - Urban water and wastewater
    - Hydroelectric power
    - Water hazards and fast response

Surface Water Model

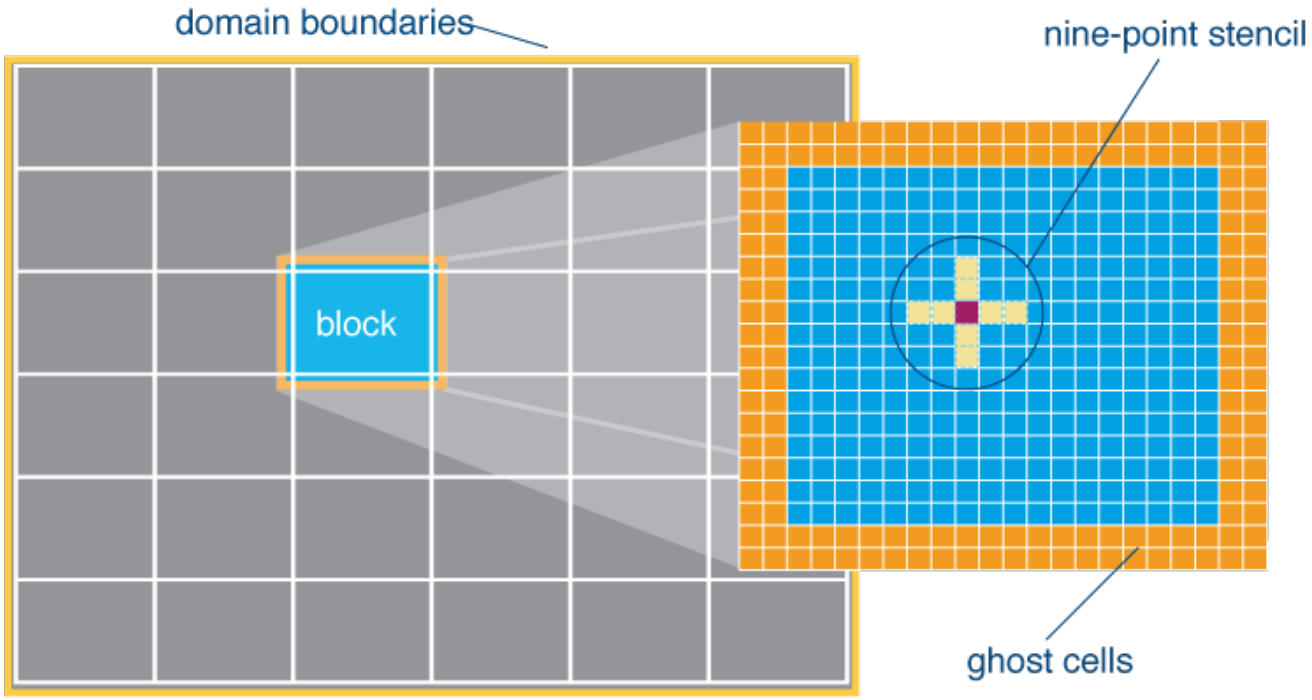
- 2D shallow water equations
  - Multiple scales (urban, regional)
  - Easily use with gridded datasets
- Distributed hydrologic models
  - Rainfall
  - Infiltration
  - Surface friction





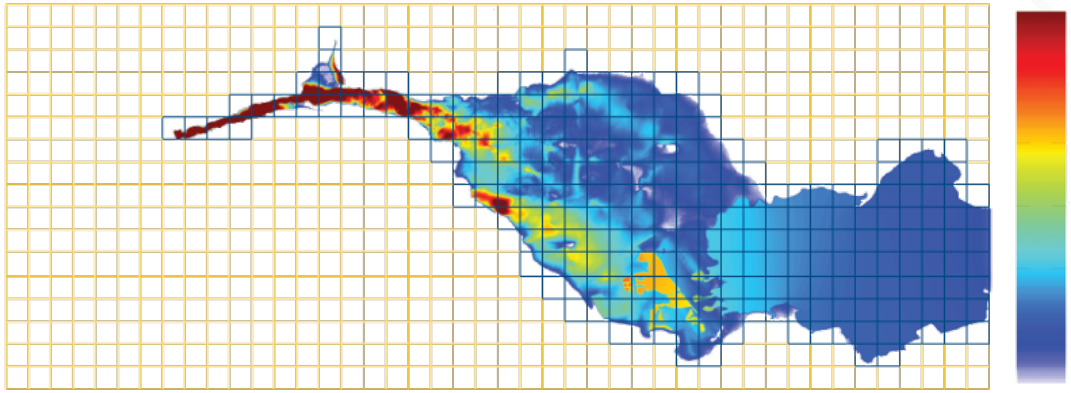
GPU Implementation

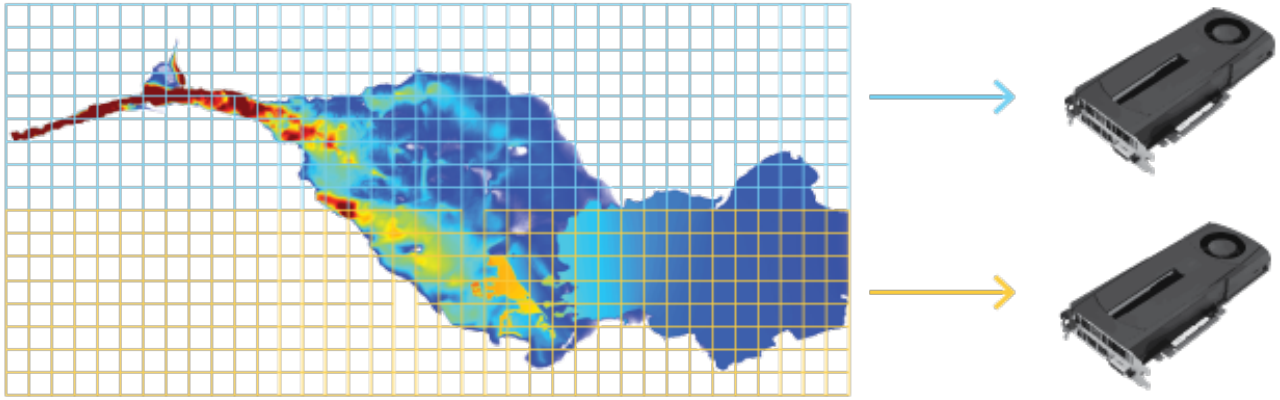
- Parallelize using graphics processing units (GPUs)
  - Hundreds of cores



Increasing Performance

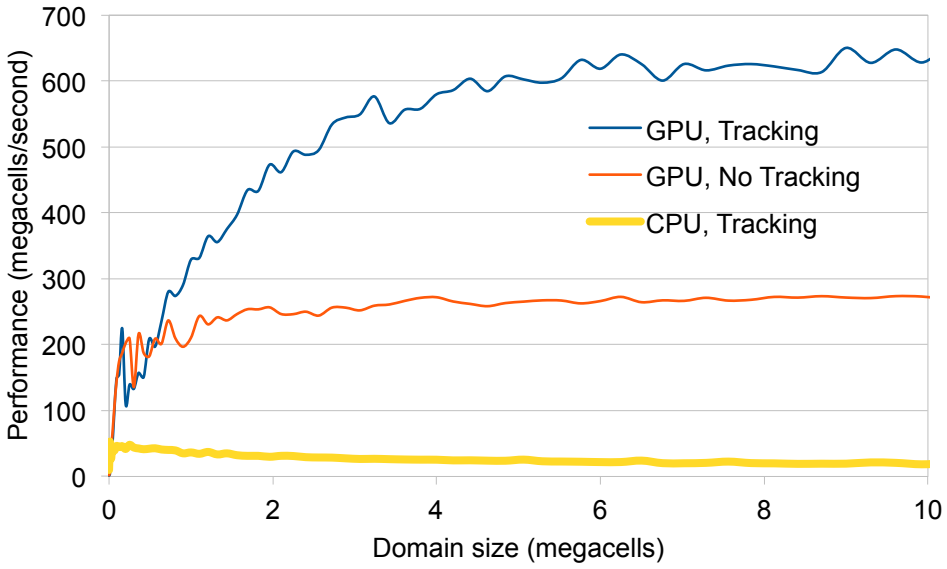
- Track wet cells/blocks
- Multiple GPUs

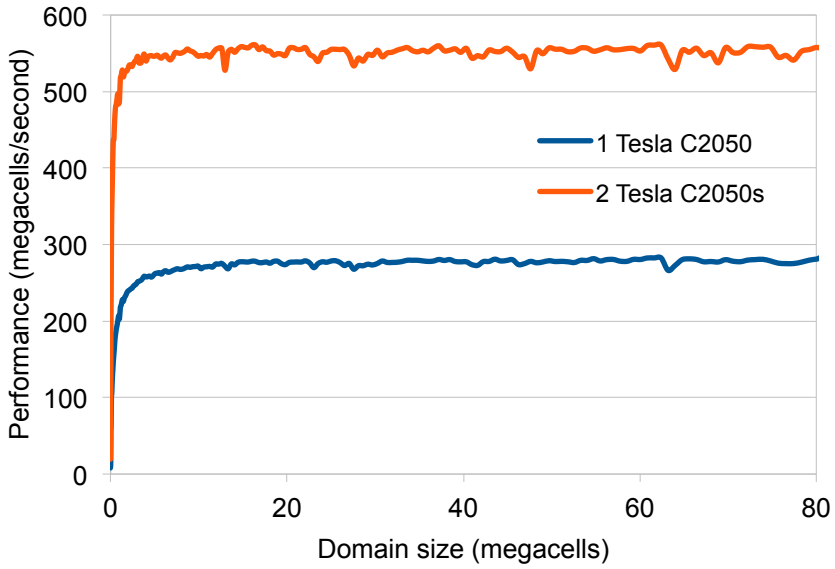




GPU Benchmarks

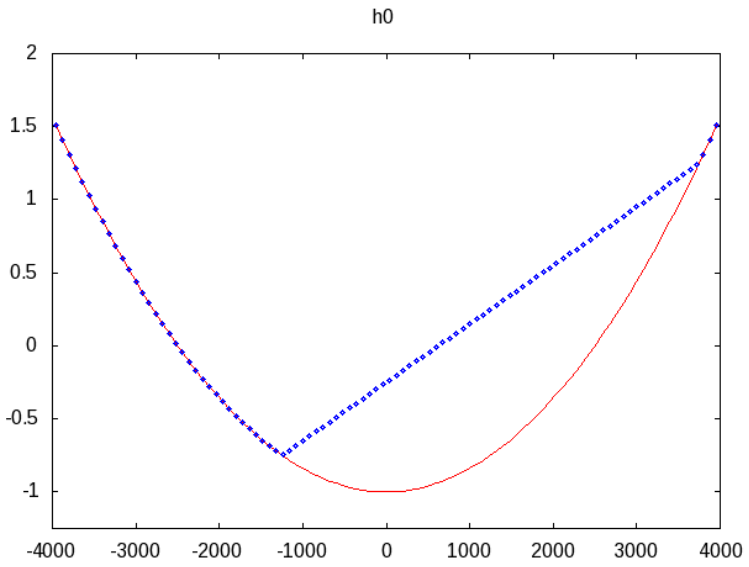
- GPU: Tesla C2050
  - 448 cores @ 1.15 GHz
- CPU: Intel Xeon x5670
  - 12 cores @ 2.93 GHz





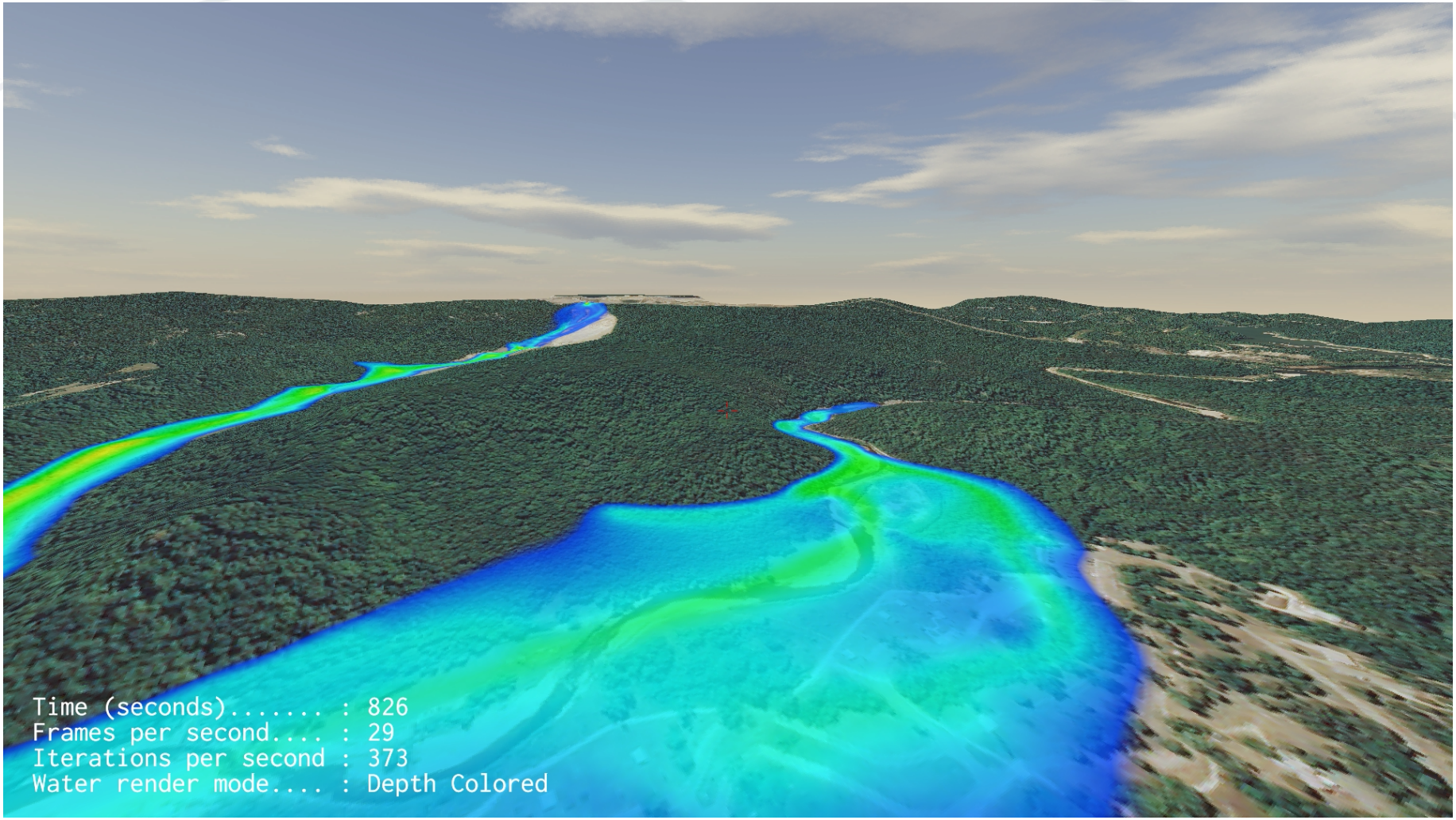
Implementation Uses

- More interesting problems
  - Larger domains
  - Longer timescales
  - Parameter sweeps
- Real time visualization
  - Increase model understanding
  - Immediate intuition for results
  - Emergency management



Visualization Integration

- OpenGL, GLFW
  - All data shared on GPU
- Render topography/water surface
- Simulate steps → Swap data → Render frame
- User interaction
  - Traverse the domain (WASD)
  - Interact with the simulation

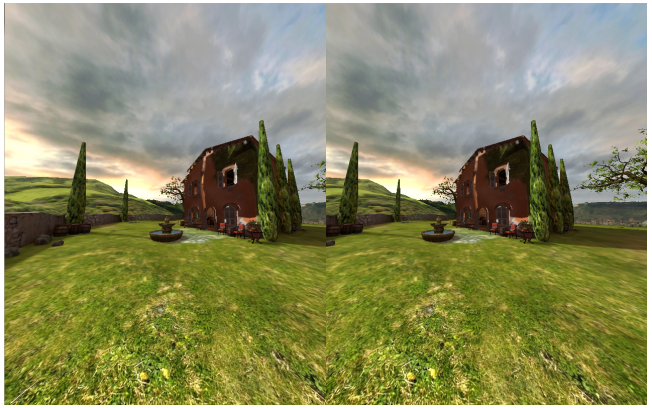

Time (seconds)..... : 826
Frames per second.... : 29
Iterations per second : 373
Water render mode.... : Depth Colored

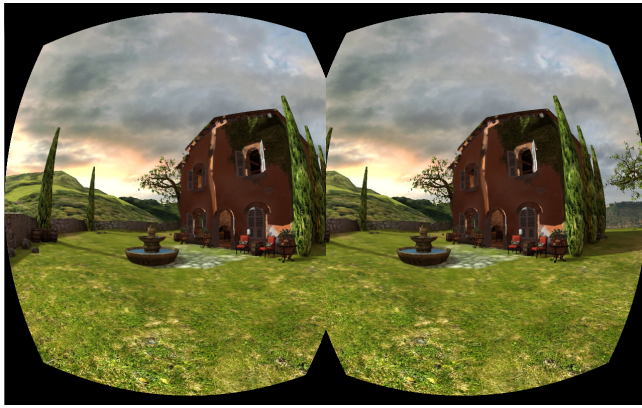
Virtual Reality

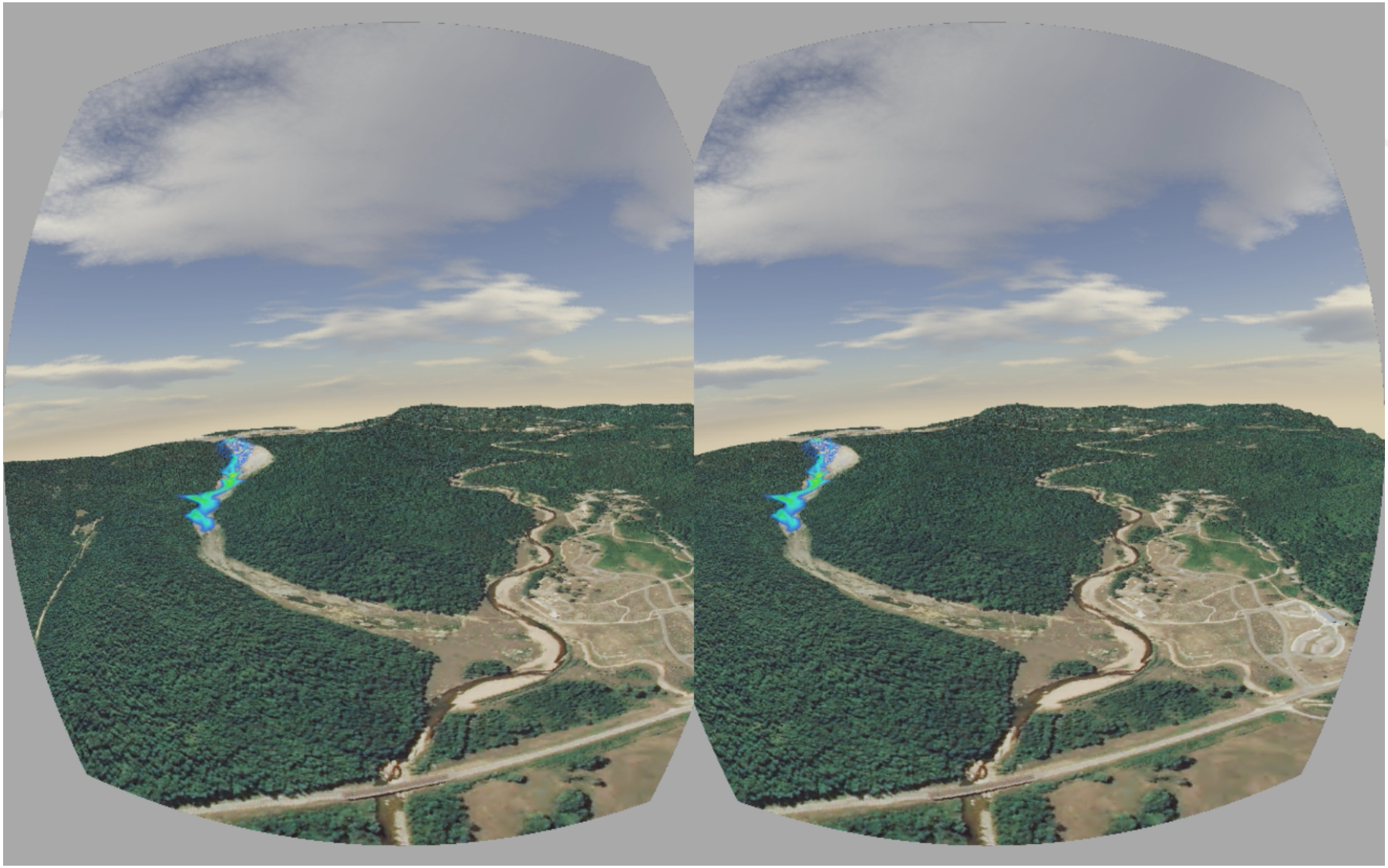
- Importance
  - Intuitive
  - Immersive
  - The future!
- Powerwall
- CAVE
- **Oculus Rift**

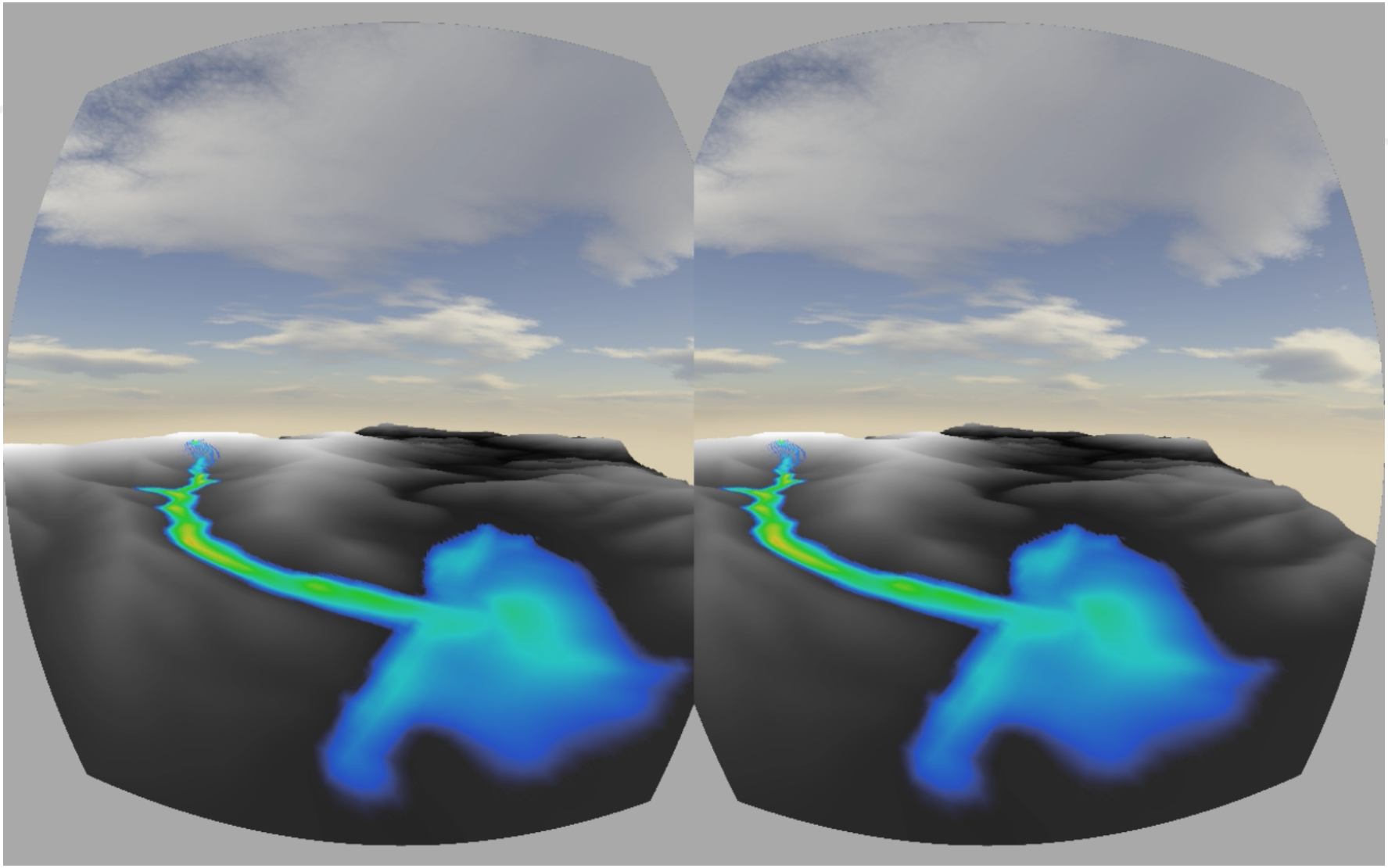
Oculus Rift implementation

- Extra steps
  - Get sensor information
  - Render two perspectives
  - Distort scene
  - 60 frames per second

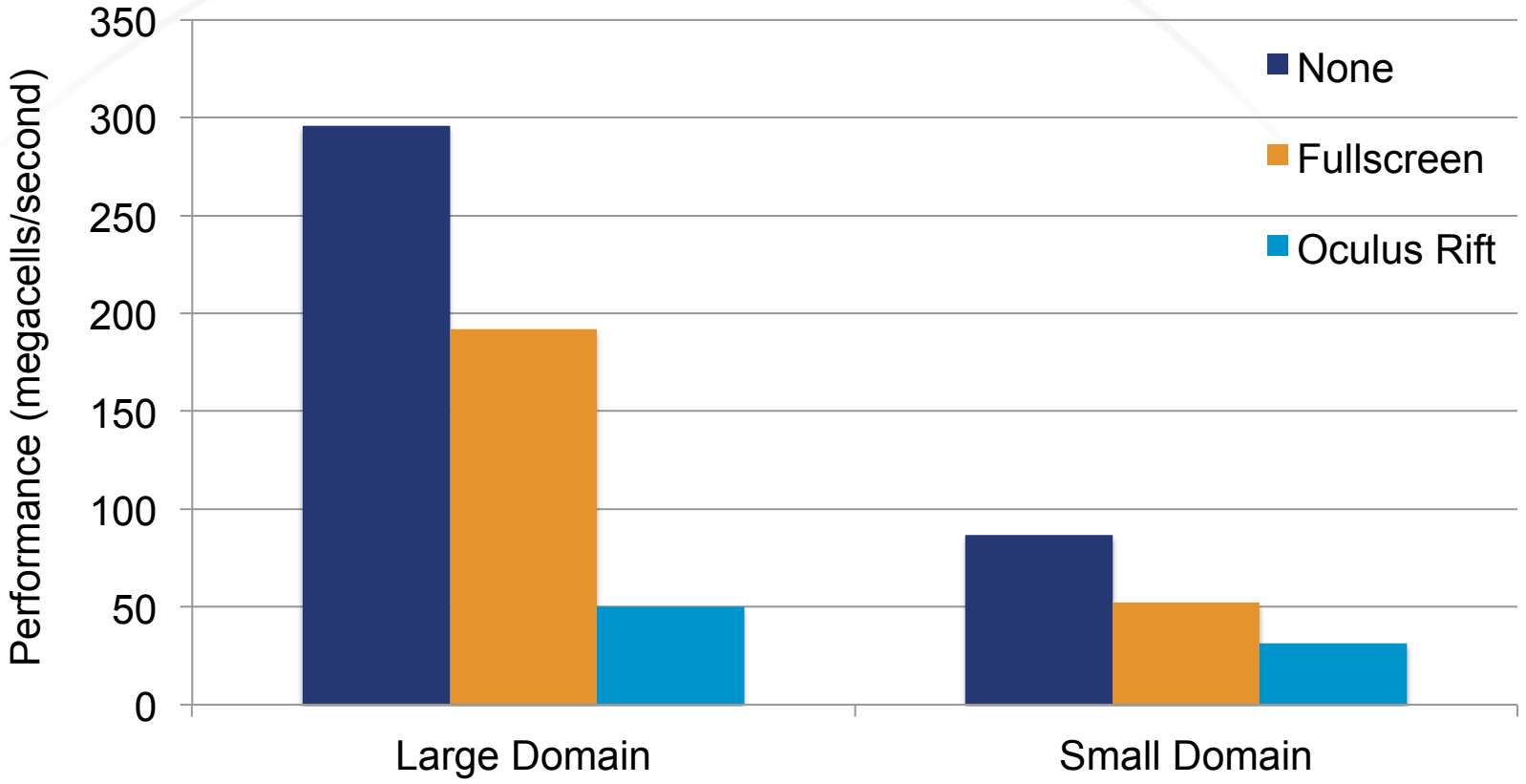








Visualization Benchmarking



Future Work

- Very short term
  - Add interaction to Rift
  - Controller usage
- Short Term
  - Increase performance of visualization
  - Multiple screens
- Long Term
  - Deploy to emergency planners

Acknowledgements

- Bradley Austin Davis (github.com/jherico)
- A. R. Brodtkorb, M. L. Sætra, M. Altinakar
- Sarah Hedeen
- James Arnold
- Hari Khalsa

Image Sources

- lanl.gov
- wikipedia.org

Thank you!

Questions