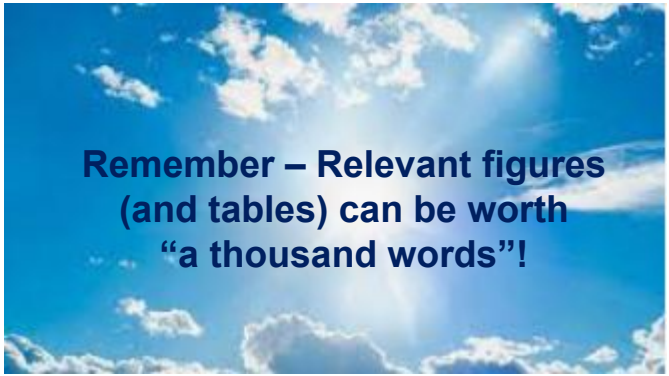
# Team E1 - FPGA Accelerated Fluid Simulation

---

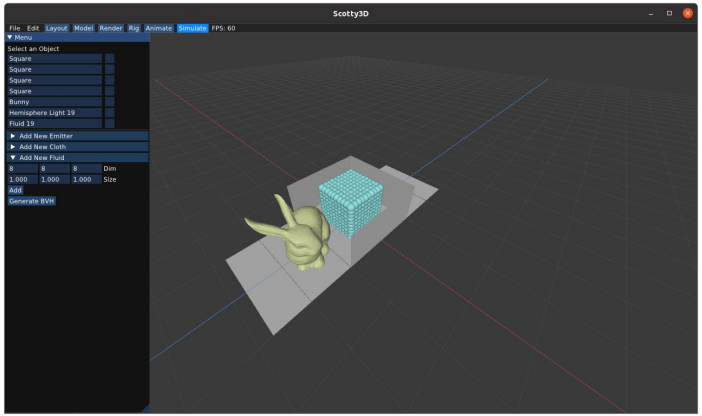
Jeremy Dropkin, Alice Lai, Ziyi Zuo

Add your 12 slides after this slide… [remember, 12 min talk + 3 min Q/A]



# Motivation

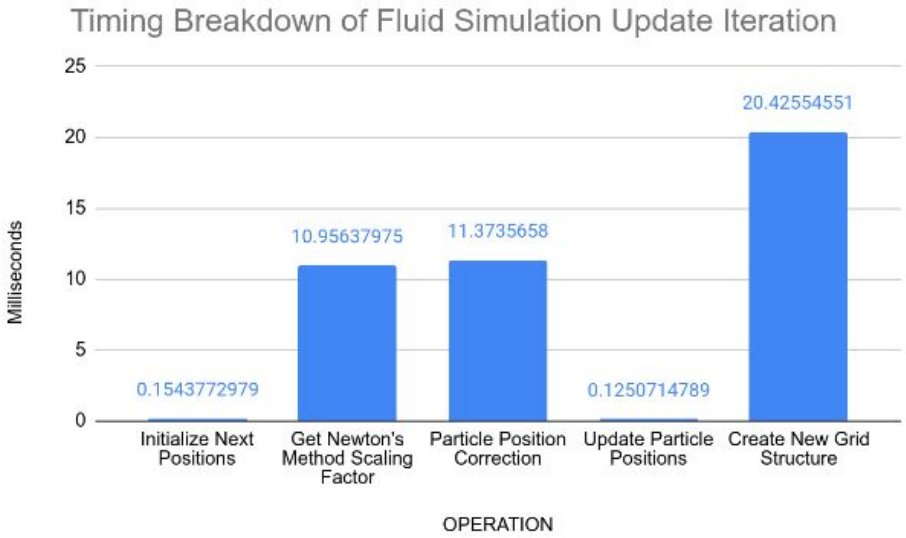
- Scotty3D is a 3D graphics software package that runs only as a multithreaded CPU Program
- Fluid Simulation Library runs poorly on CPU, throttling the FPS to fractional values
- Our goal is to create an **FPGA-based application that can accelerate Scotty3D's Fluid Simulation Library**




Scotty3D Interface

# Performance Requirements

- Baseline
  - ~3 second render time (per frame) on an i7-8665U @ 1.90 GH x 8
- Our goal is to make at leas a **10x speedup**
  - ~300ms to render per frame

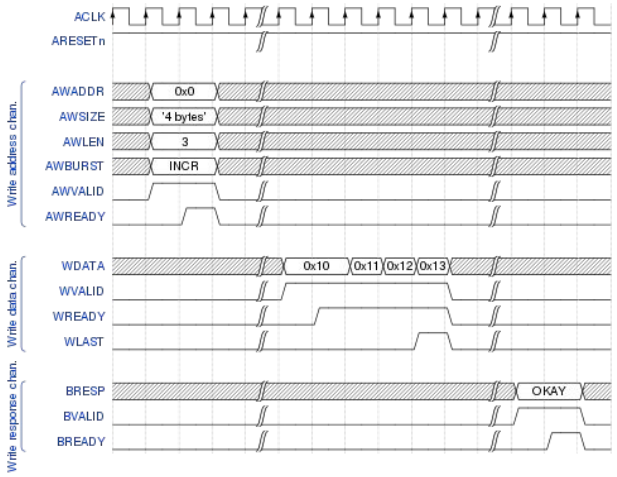


# Data Communication Requirements

---

- Data transfer between CPU and fabric should not bottleneck the actual rendering task

- Minimize interpretation overhead between CPU and FPGA format
  - CPU/Fabric communication use AXI interface




AXI Interface Waveform Diagram

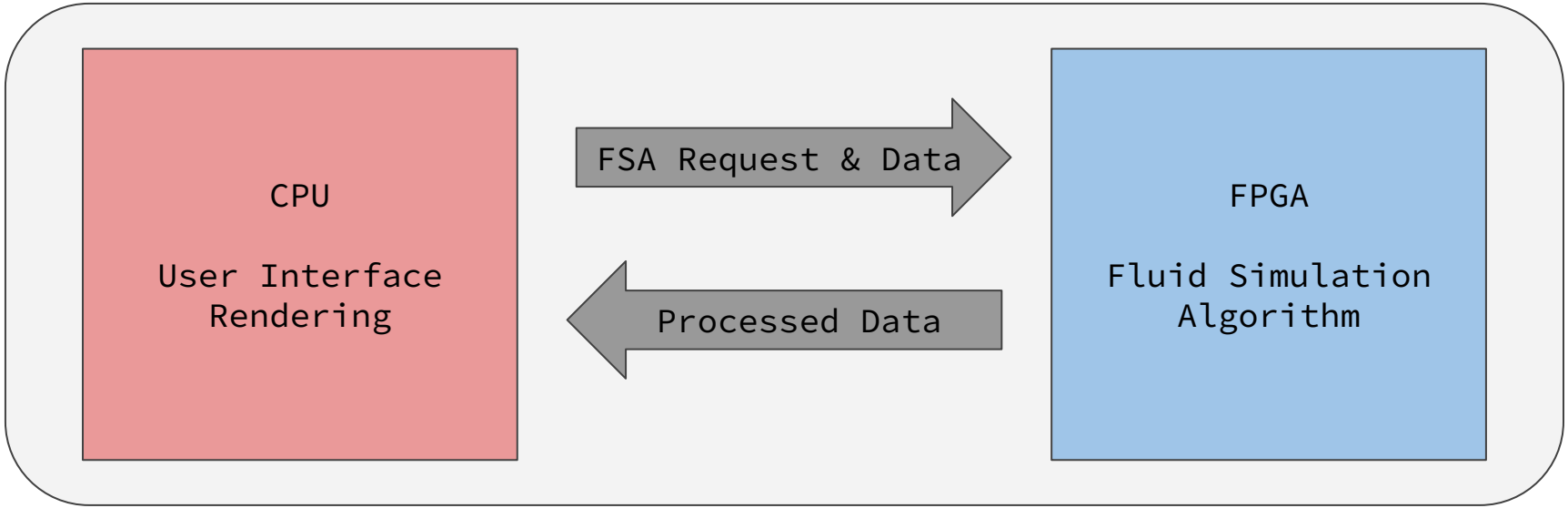
# Use Case Requirements

---

- Accelerated implementation does not add to overhead or complications to the user experience
  - We will collect user feedback through user testing to ensure there is no additional overhead
- Perform computation within a 10W TBP envelope
  - Ensure low-power compute
  - Desktops typically run between 50W and 100W

# Solution Approach

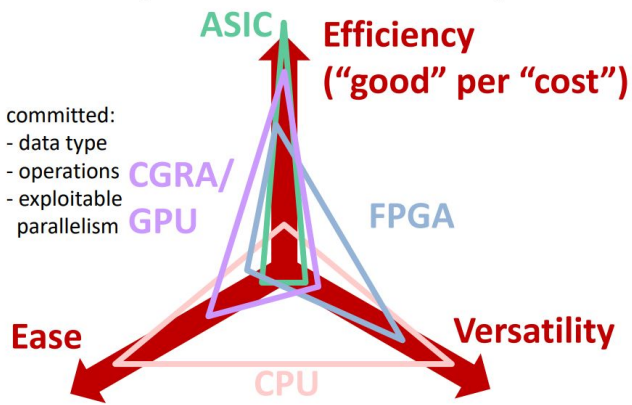
- Accelerate Fluid Simulation Algorithm on the FPGA fabric
- CPU handles the rest of the Scotty3D rendering stack



# Solution Platform - Motivation

Why FPGA?

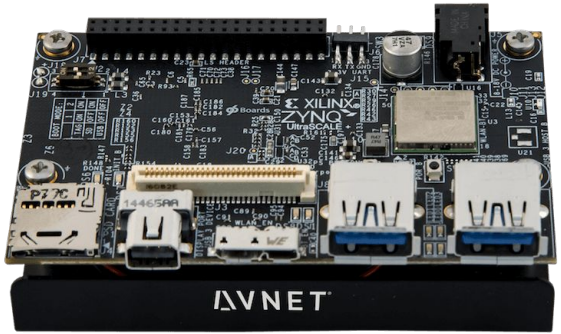
- CPU multithreading is not enough to take advantage of high thread-level parallelism
- GPUs are capable of exploiting thread-level parallelism using vectorized operators
- FPGA!
  - Extract high thread-level parallelism
  - Handle irregular code by instantiating extra control structures
  - Highly flexible memory architectures




From 18643 course slides

# Solution Platform

- Xilinx Vivado Design Suite
  - Development and synthesis platform for Xilinx FPGA platforms
- Vitis HLS
  - High Level Synthesis
    - Generate hardware from C/C++ code
    - Control structures through use of pragmas and compiler directives
    - Result is transpiled to HDL
- Xilinx Ultra96 FPGA
  - Arm Core + Reconfigurable Fabric




XILINX VITIS™

# Solution Approach - Algorithm

- Exploit thread-level parallelism
  - Unroll loops and instantiate multiple computational units

- Exploit FPGA BRAMS
  - Block RAM - high speed versatile memories
  - Large BRAM capacity means we can make multiple copies of data
  - Use this to accelerate neighbor-finding tasks

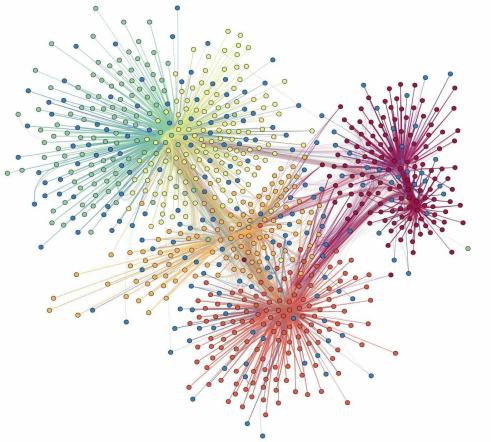
**Algorithm 1** Simulation Loop

**for all** particles $i$ **do**
  apply forces $\mathbf{v}_i \Leftarrow \mathbf{v}_i + \Delta t \mathbf{f}_{ext}(\mathbf{x}_i)$
  predict position $\mathbf{x}_i^* \Leftarrow \mathbf{x}_i + \Delta t \mathbf{v}_i$
**end for**
**for all** particles $i$ **do**
  find neighboring particles $N_i(\mathbf{x}_i^*)$
**end for**
**while** $iter < solverIterations$ **do**
  **for all** particles $i$ **do**
    calculate $\lambda_i$
  **end for**
  **for all** particles $i$ **do**
    calculate $\Delta \mathbf{p}_i$
    perform collision detection and response
  **end for**
  **for all** particles $i$ **do**
    update position $\mathbf{x}_i^* \Leftarrow \mathbf{x}_i^* + \Delta \mathbf{p}_i$
  **end for**
**end while**
**for all** particles $i$ **do**
  update velocity $\mathbf{v}_i \Leftarrow \frac{1}{\Delta t}(\mathbf{x}_i^* - \mathbf{x}_i)$
  apply vorticity confinement and XSPH viscosity
  update position $\mathbf{x}_i \Leftarrow \mathbf{x}_i^*$
**end for**

# Technical Challenges

- Computation Irregularity
  - Computing collisions with neighboring particles is a task with a high level of irregularity

- Software/Hardware interfacing
  - Communication needs to happen over a seamless interface to minimize data serialization/unpacking

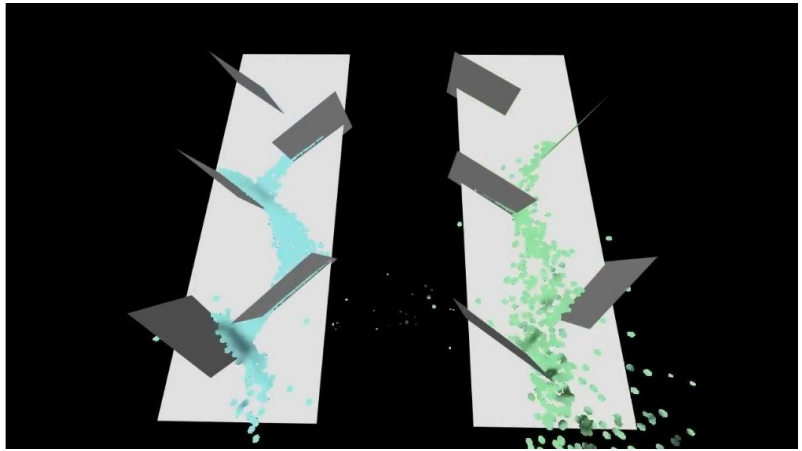


Computationally Irregular Structure

# Testing & Verification

- Quantitative evaluation:
  - Time for each code chunk
  - Time for each frame to render
  - Comparison of resulting data against a golden result

- Qualitative evaluation:
  - Visual inspection to ensure rendered animations still retain “fluid-like” quality



Demo of fluid simulation

# Tasks and Division of Labor

| Alice | Jeremy | Ziyi |
| --- | --- | --- |
| - Introduce new intuitive UI<br>- Refactor algorithm in a straightforward manner for FPGA translation<br>- Organize data transfer for new UI | - Manage data transfer between FPGA and CPU<br>- Set up HDMI Interface for Visualization<br>- Optimize fluid simulation algorithm on FPGA<br>- Implement Request scheduler | - Analyze computational kernel for exploitable parallelism and acceleration opportunities<br>- Set up Vitis HLS workspace and build configurations<br>- Optimize fluid simulation algorithm on FPGA |

# Schedule

| TASK TITLE | TASK OWNER | WEEK 4 | | | | | WEEK 5 | | | | | WEEK 6 | | | | | WEEK 7 | | | | | WEEK 8 | | | | | WEEK 9 | | | | | WEEK 10 | | | | | WEEK 11 | | | | | WEEK 12 | | | | | WEEK 13 | | | | | WEEK 14 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F | M | T | W | R | F |
| Platform | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initial Workflow Setup | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure out memory transfer | Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure out HDMI | Jeremy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Integrate data in / data out | Jeremy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Identify code chunks that complicate FPGA synthesis | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Redesign C++ code to reduce complexity | Alice | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reimplement C++ code for complexity | Alice/Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Slack | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| FPGA | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Analyze Hardware Resources | Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Set up Performance Benchmarking | Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Get data from interface | Jeremy/Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Run base kernel on FPGA | Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Integration | Jeremy/Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Analyze Performance of first iteration | Ziyi | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Slack | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Acceleration | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Initialize Next Positions | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Get Newton's Method Scaling Factors | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Get Particle Position Corrections | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Update Particle Positions | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Reconstruct Grid Structure | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Slack | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| CPU | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Set up Performance Benchmarking | Jeremy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Command-Line Interface | Alice | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Data out from interface | Alice/Jeremy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Request Scheduler | Jeremy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Request Deployment | Jeremy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Slack | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Misc | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Final Presentation Slides | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Slack | All | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |