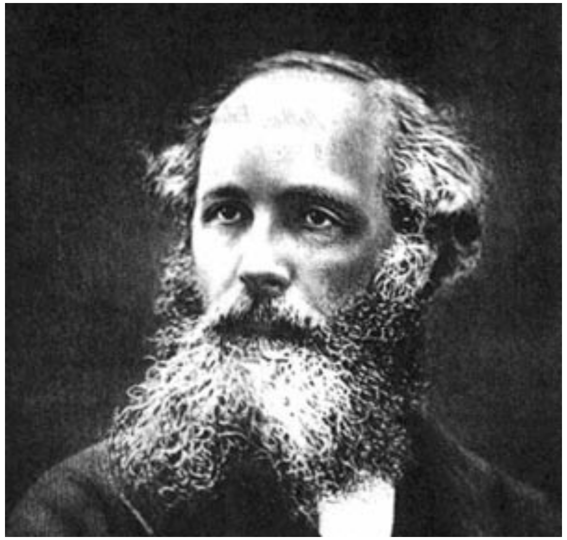
Johannes Kepler



James Maxwell

Blaise Pascal

Alessandro Volta

Who would win a 100 Meter Sprint?

Is Data Placement Optimization Still Relevant On Newer GPUs?

Md Abdullah Shahneous Bari[1], Larisa Stoltzfus[2],

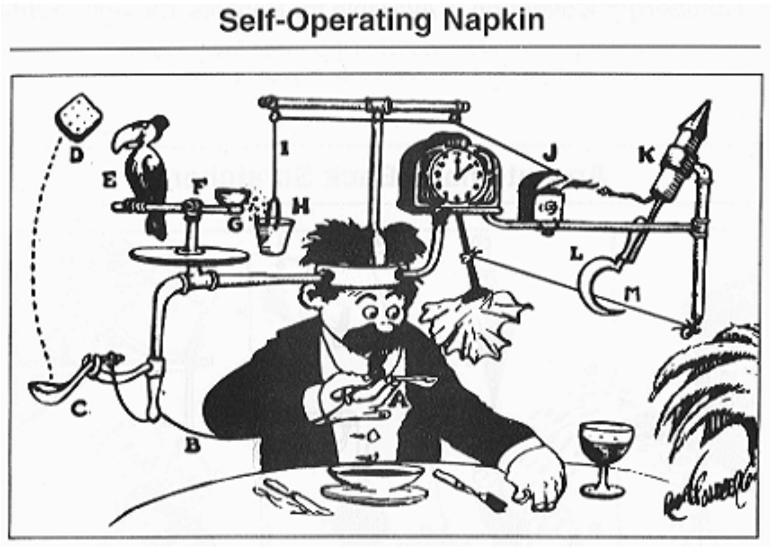
Pei-Hung Lin[3], Chunhua Liao[3], Murali Emani[3], Barbara Chapman[1,4]

1. Stony Brook University 2. University of Edinburgh
3. Lawrence Livermore National Laboratory 4. Brookhaven National Laboratory

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344 and was supported by the LLNL-LDRD Program under Project No. 18-ERD-006. LLNL-PRES-761109

What is a GPU? Should I Really Care?

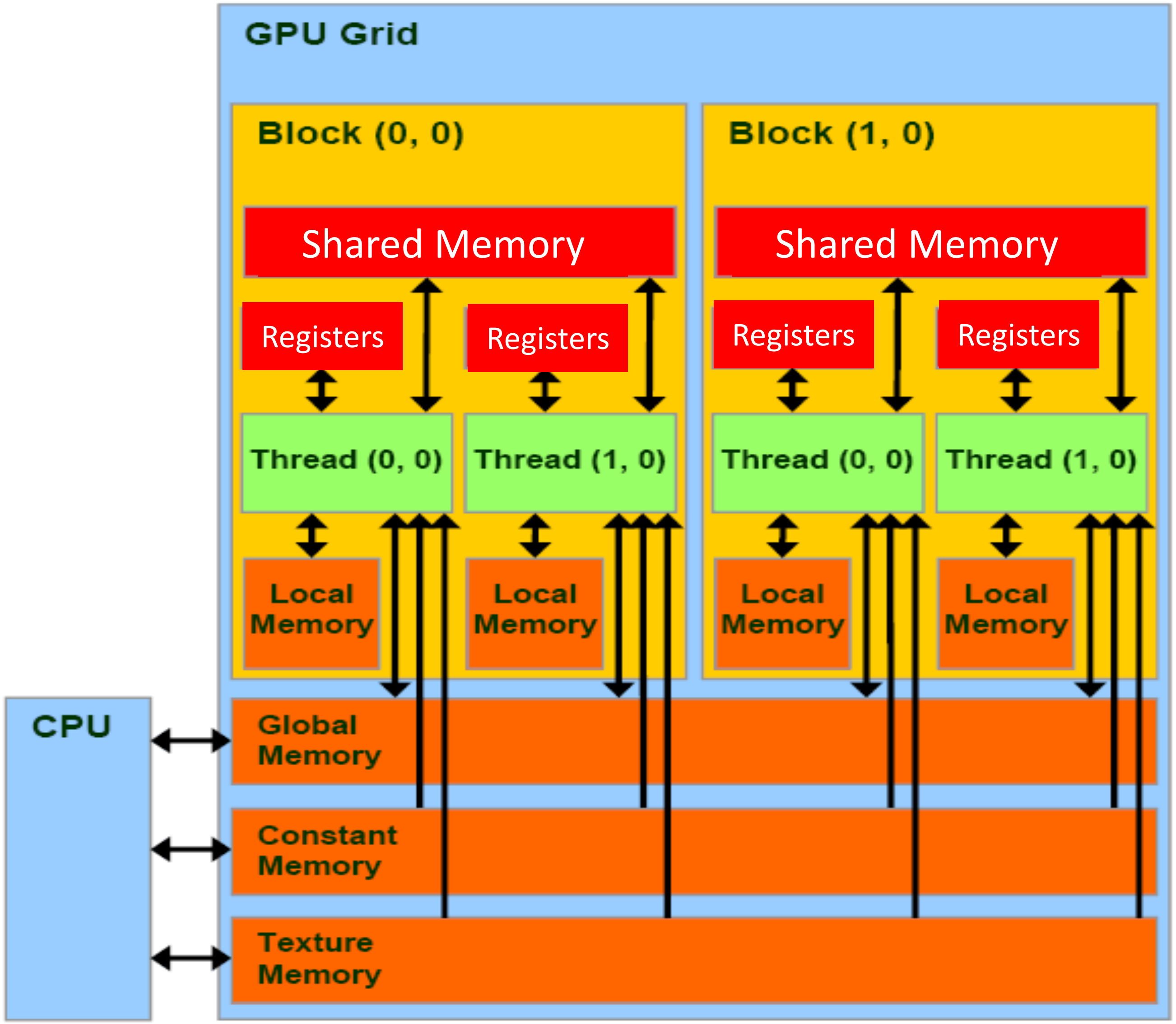
- Graphics Processing Unit
- Massively parallel
- Thousands of cores
- Now-a-days used for general purpose computing
- 5 of the top 10 supercomputers uses NVIDIA GPU
- Aaand…… Deep Learning



Easy to Use???

- Programmability: OpenMP, OpenACC, **CUDA**, OpenCL
- Getting performance improvement
  - Easy if the algorithm is compliant
- Getting GOOD performance
  - Not so easy
- But, but, why??
  - **Complex memory hierarchy**

NVIDIA GPU Memory Hierarchy



Global/Device Memory

- ➢Largest off-chip memory
- ➢Serves as the main memory
- ➢Long latency
- ➢Limited bandwidth

Constant Memory

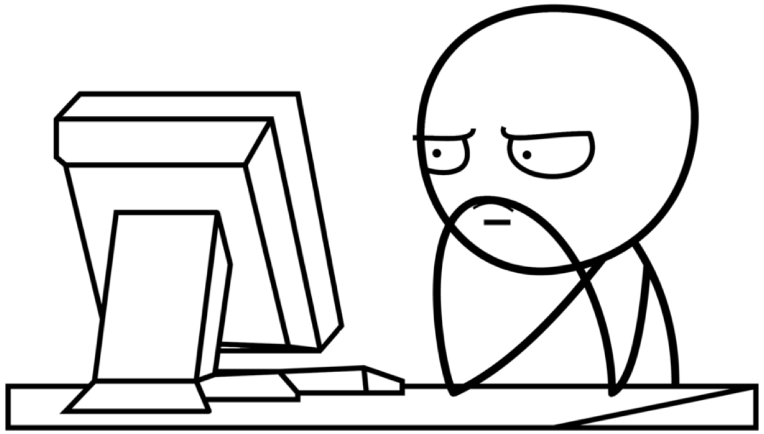
- Predefined part of global memory
- Cached and globally visible to all threads
- Read-only
- Can be as fast as cache
- Limited size
- Good for read-only data that needs to be repetitively broadcast to all GPU threads

Shared Memory

- ➢Software managed on-chip data cache
- ➢Per Streaming Multiprocessor (SM)
- ➢Limited size
- ➢Low latency and high bandwidth

Texture Memory

- Off-chip, cached and read-only
- Actual memory bound to device memory
- Can occupy the whole device memory bound to the texture unit
- Texture cache specially optimized for 2-D, 3-D spatial locality



How to Get GOOD Performance?

- Duh, use the memory hierarchy well
  - Optimize your code
  - Place your data in appropriate memories
- Not an easy job
- Not to mention, change in hierarchy could undo everything
- NVIDIA tend to change the hierarchy almost in every generation

Well, What Are the Engineers at NVIDIA doing???
Is It Any Better Now???

Let's Figure It Out

How has the impact of data placement changed over time?

Four kinds of memory

Global memory (GPU DRAM)

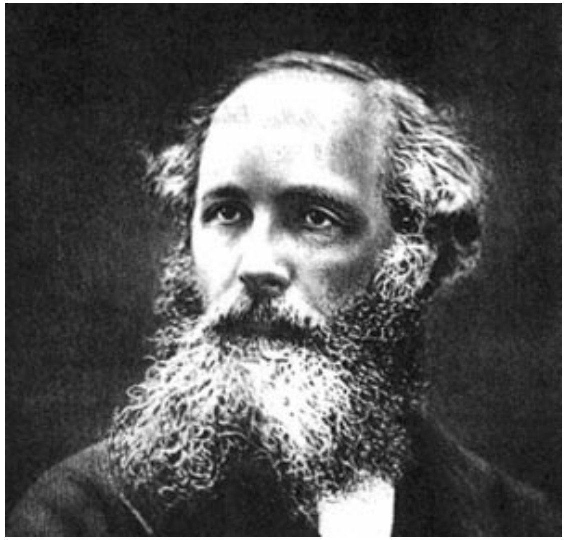
Constant memory

Shared memory

Texture memory

Designing The Experiments

Johannes Kepler

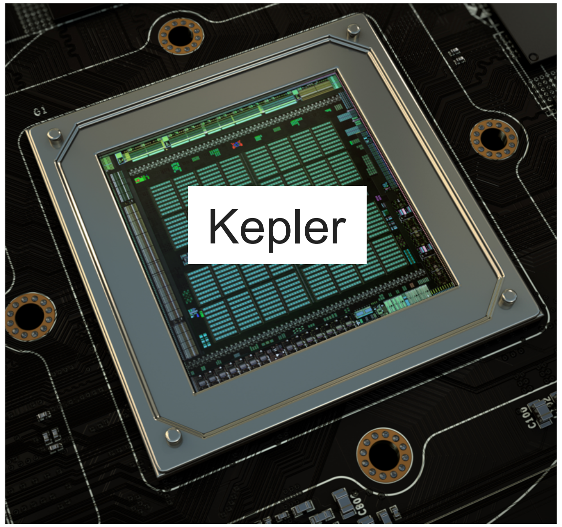


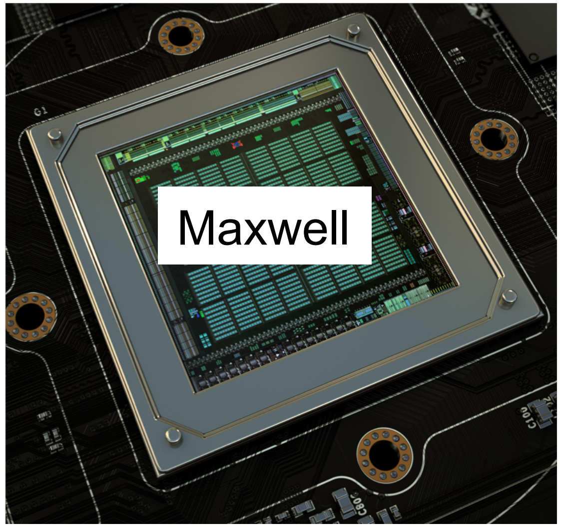
James Maxwell

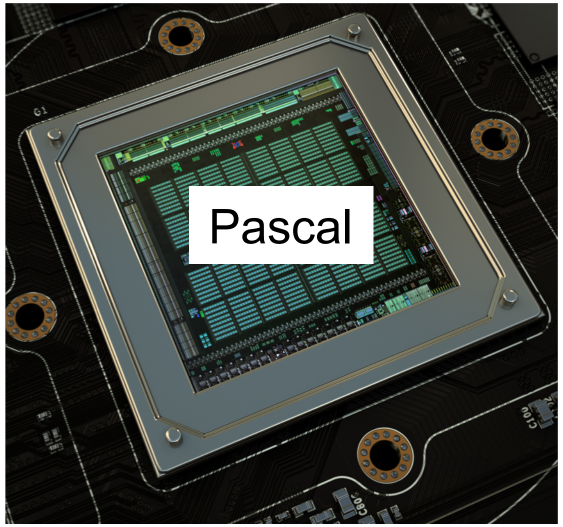
Blaise Pascal

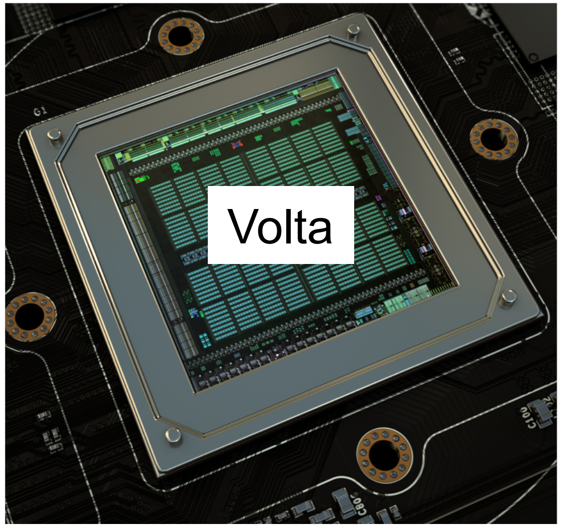
Alessandro Volta

Who would win a 100 Meter Sprint?









GPUs?

Applications

Microbenchmarks

CUDA Kernels

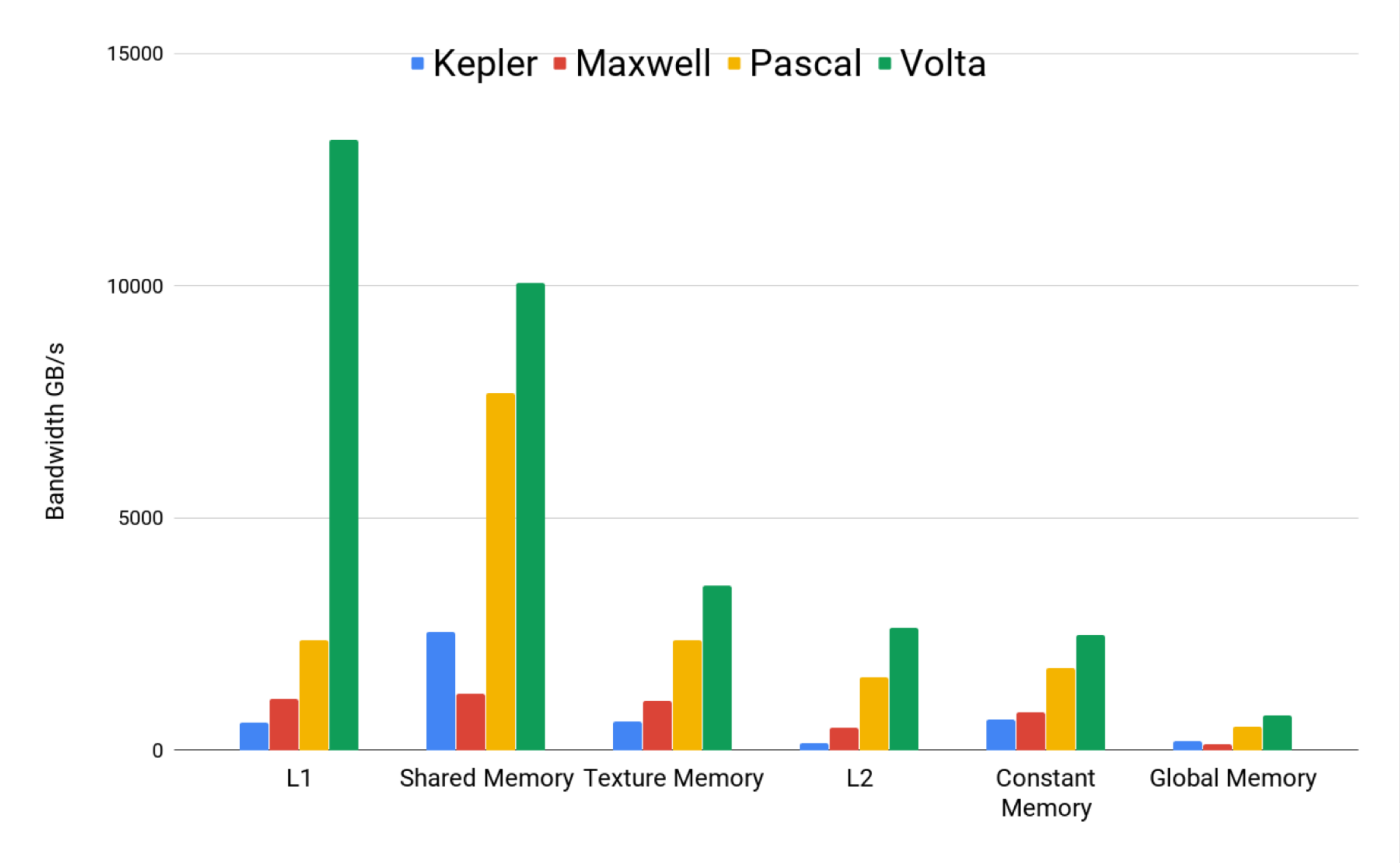
- Using only one type of special memory
- Using mixed types of special memories

Proxy App

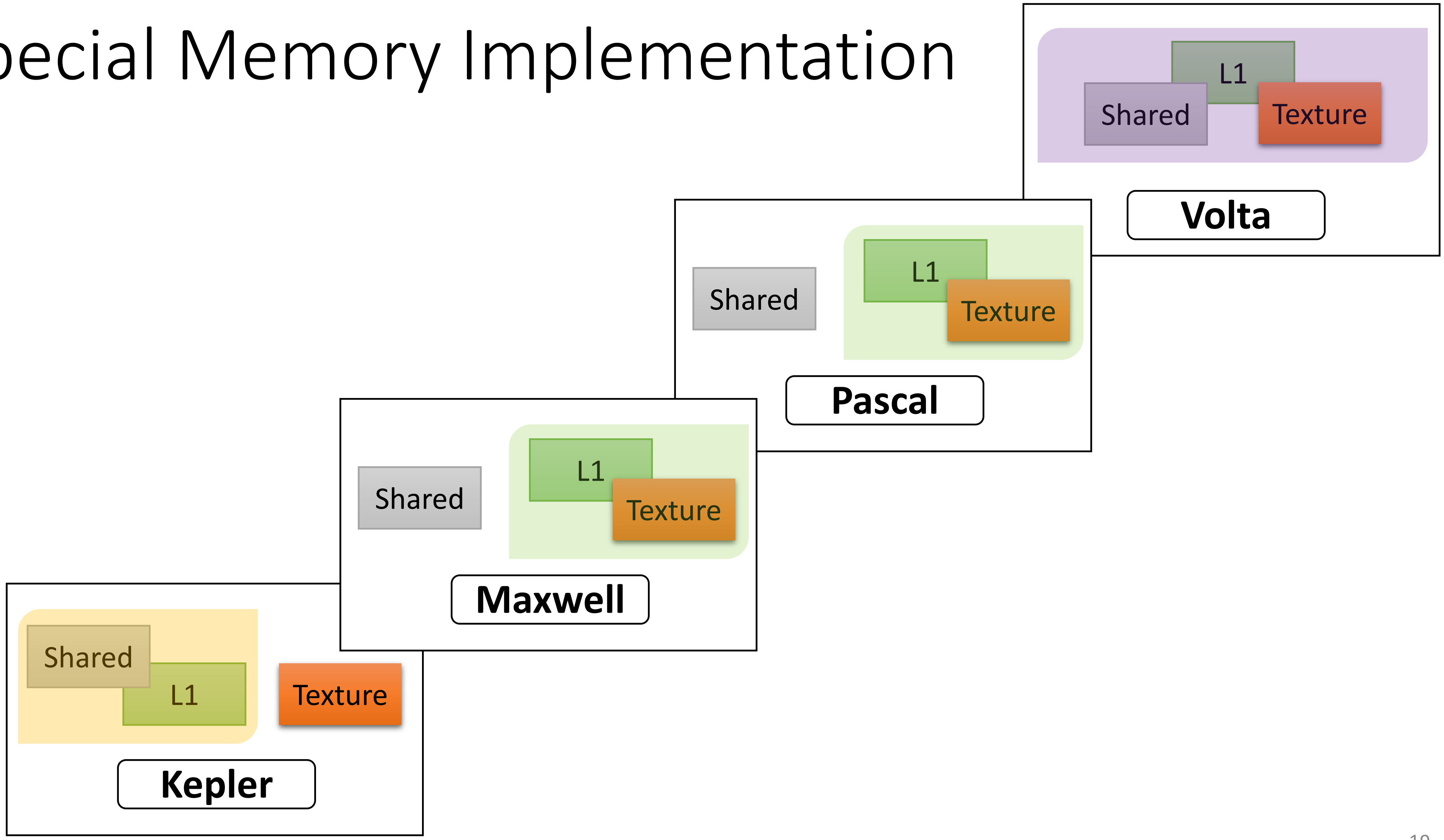
Microbenchmark

- Used several microbenchmarks
  - GPUmembench
  - Pointer chasing benchmark
- Measured metrics
  - Global, Constant, Shared, Texture memory and L1, L2 cache properties
  - Size, latency, bandwidth etc.

Bandwidth Across GPUs



Special Memory Implementation

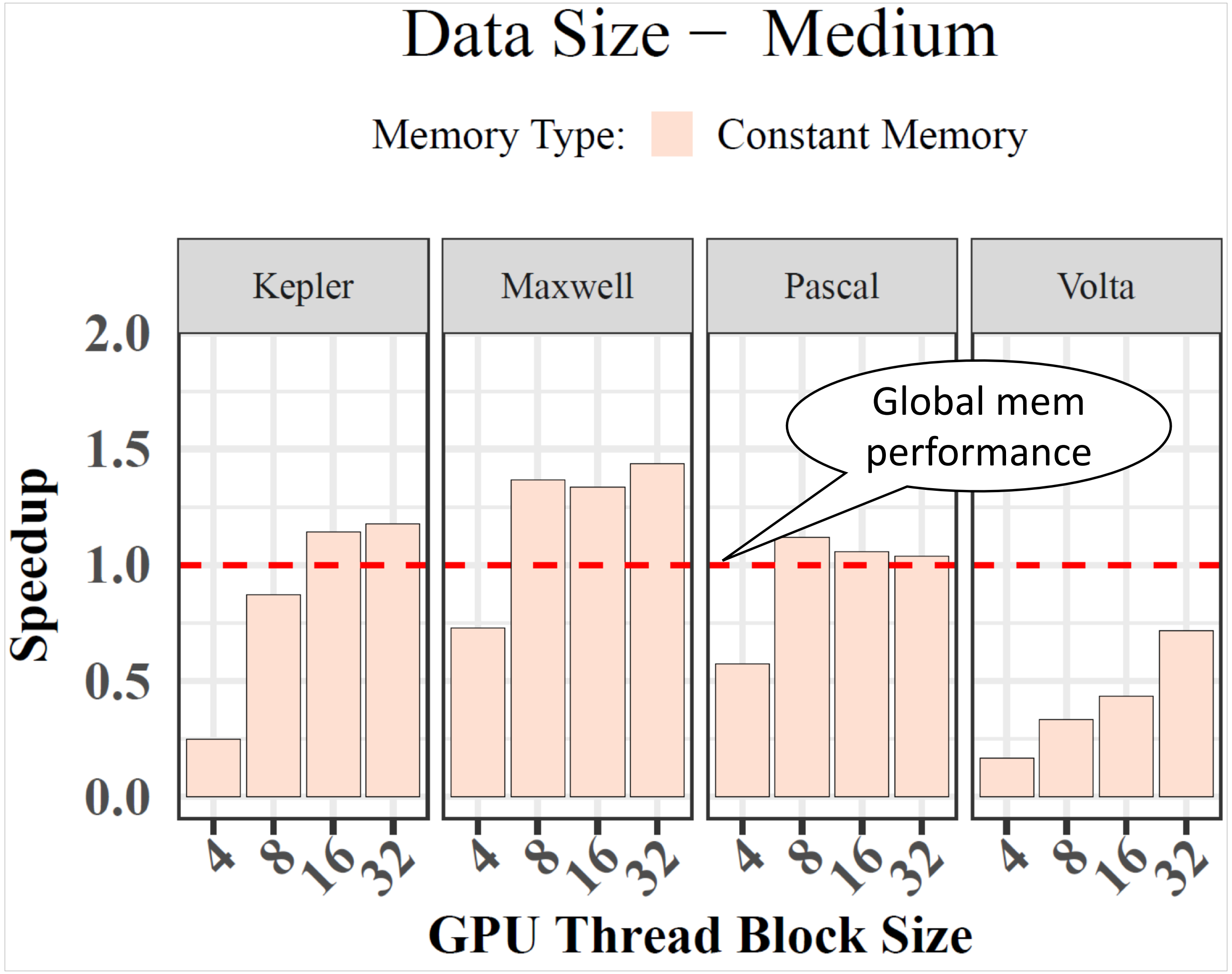


CUDA Kernels – Single Type of Memory

- Used 3 representative kernels that historically showed good performance with special memories
  - Ray Tracing – Constant memory
  - Matrix Matrix Multiplication – Shared memory
  - Heat Transfer Simulation – Texture memory
- Other configurable parameters
  - Different data sizes
  - GPU Thread block size

Constant Memory

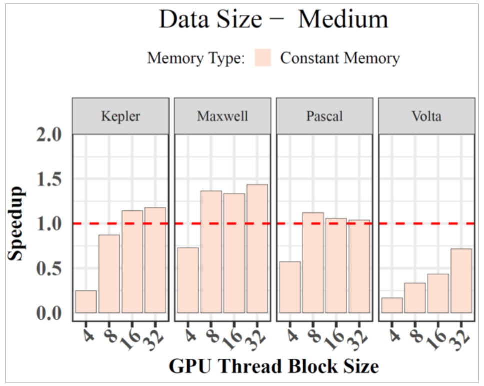
With newer generations, constant memory data placement increasingly insignificant



Analysis

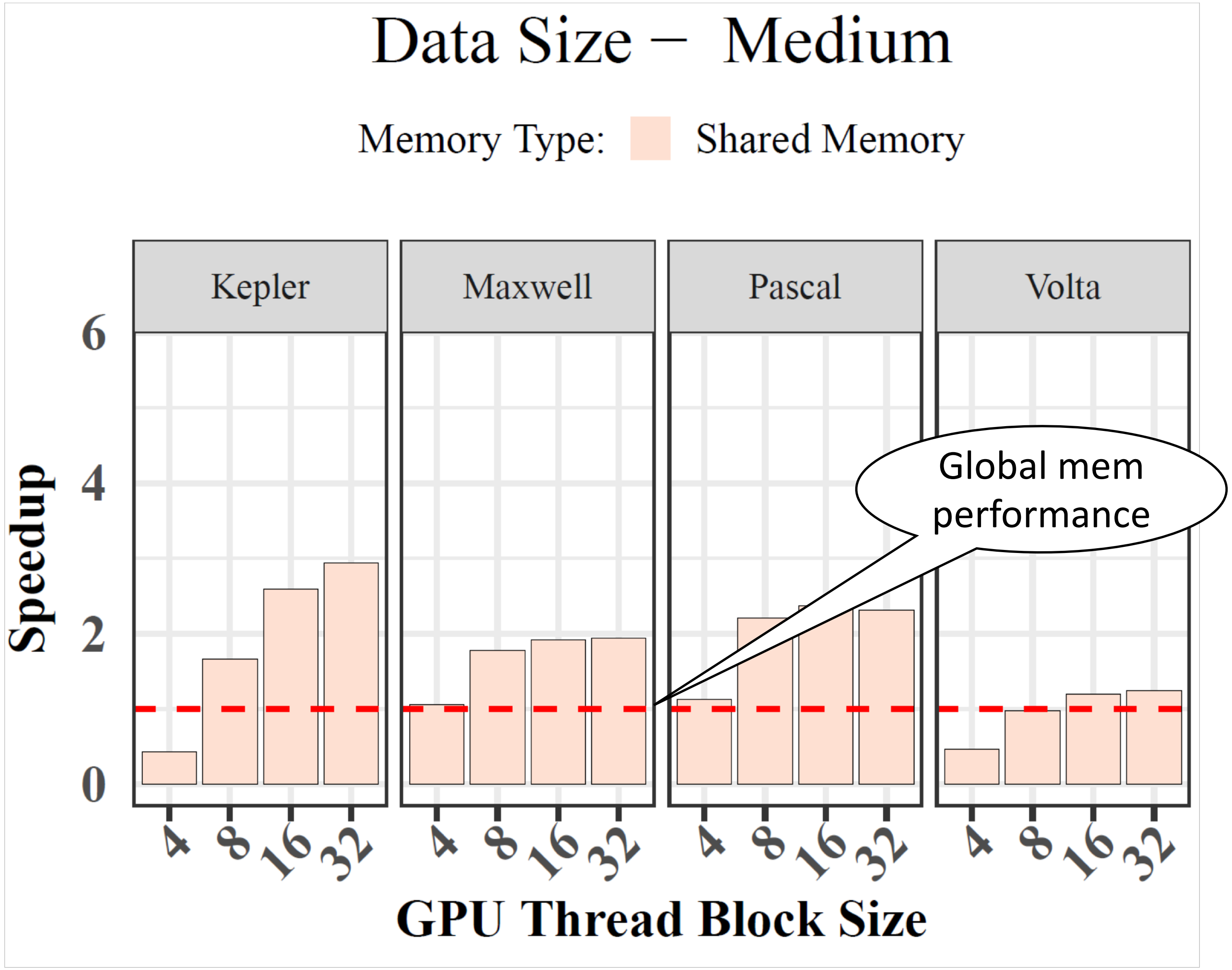
Improvement in Global memory bandwidth along with L1, L2 cache

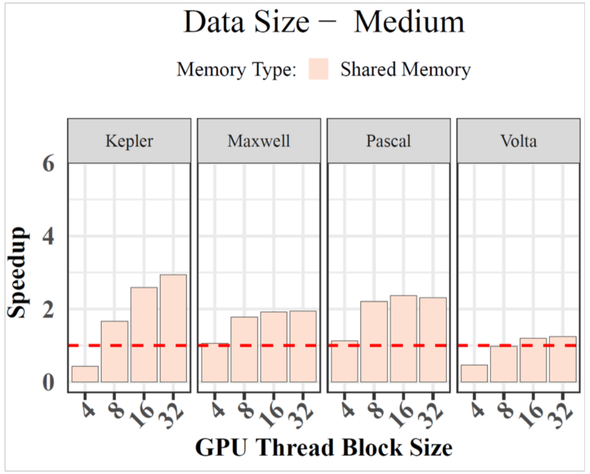
Increasing percentage of stalls due to pipeline busy



Shared Memory

With newer generations, data placement increasingly insignificant





HBM2 and unified memory design results in Volta global memory performance improvement

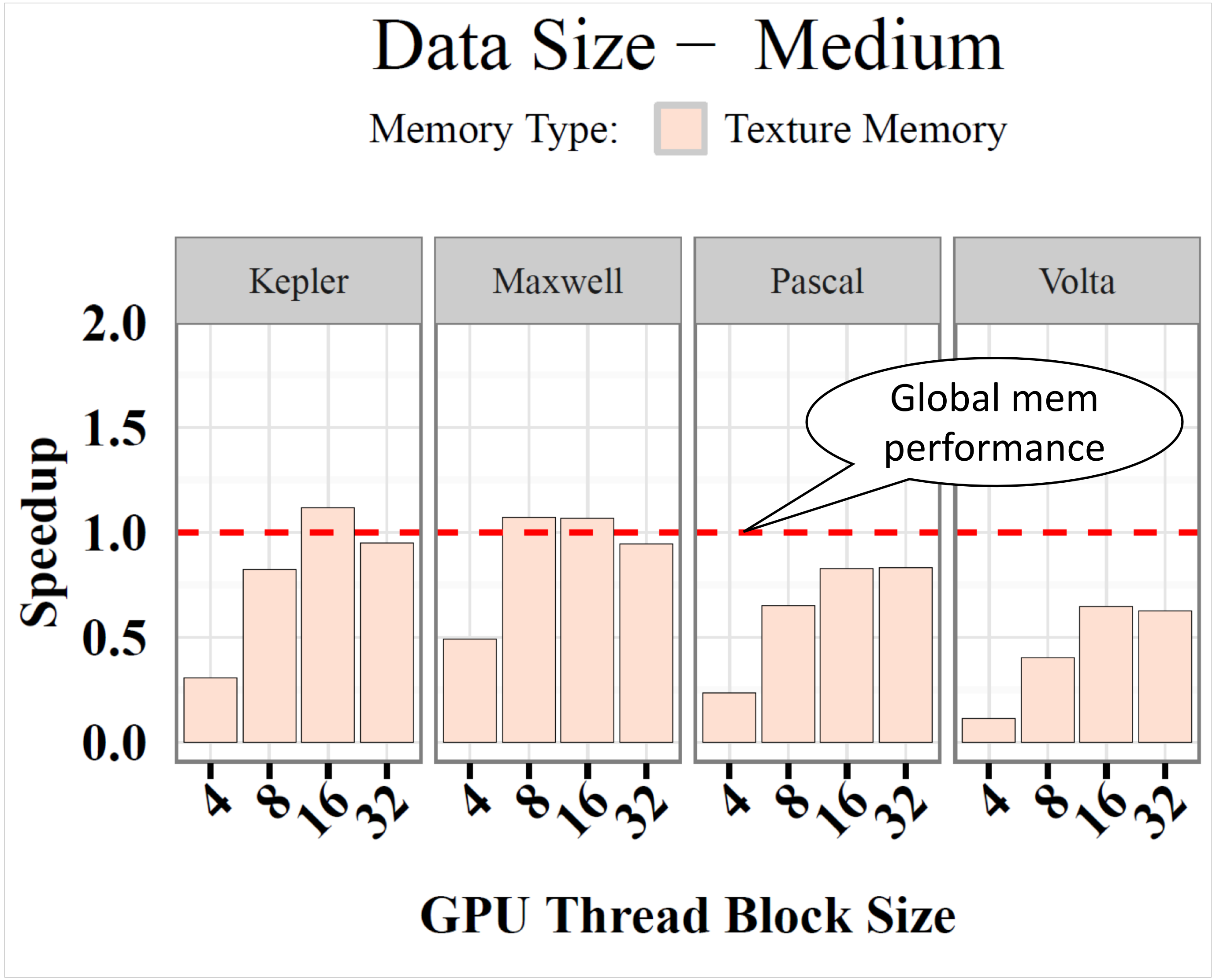
Why not Pascal then?

- Special memory not unified

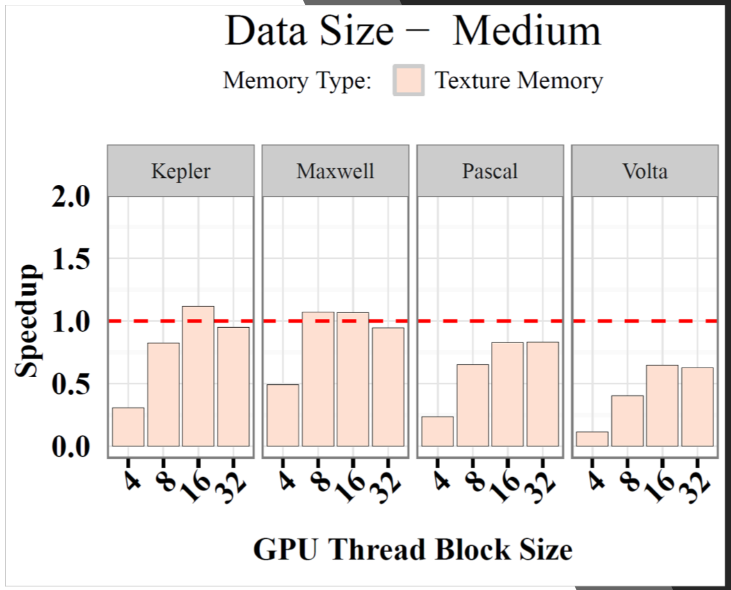
Analysis

Texture Memory

With newer generations, data placement increasingly insignificant



Analysis



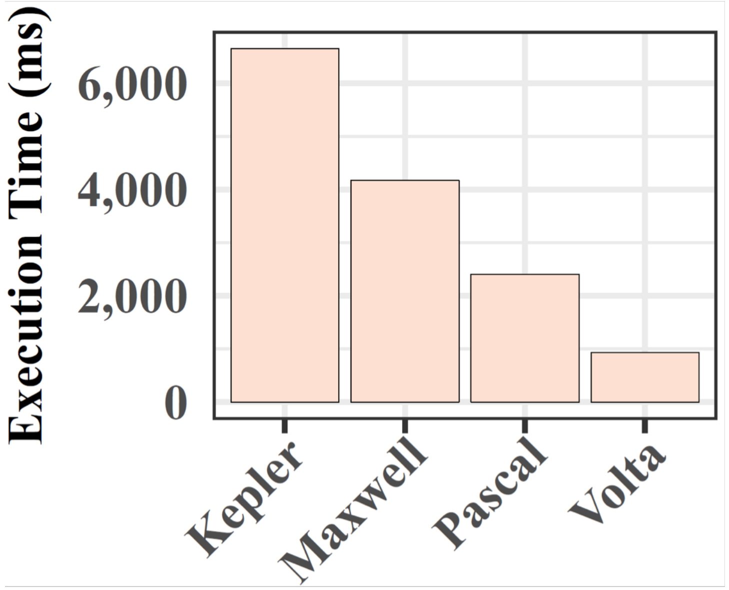
Memory design?

- L1 and Texture cache in the same unit in Maxwell, Pascal and Volta

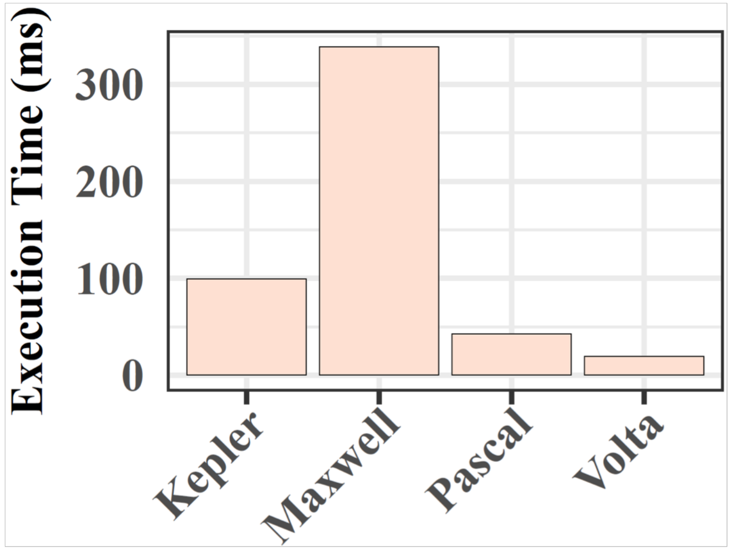
Bandwidth?

- Global memory, L1 out weighs Texture

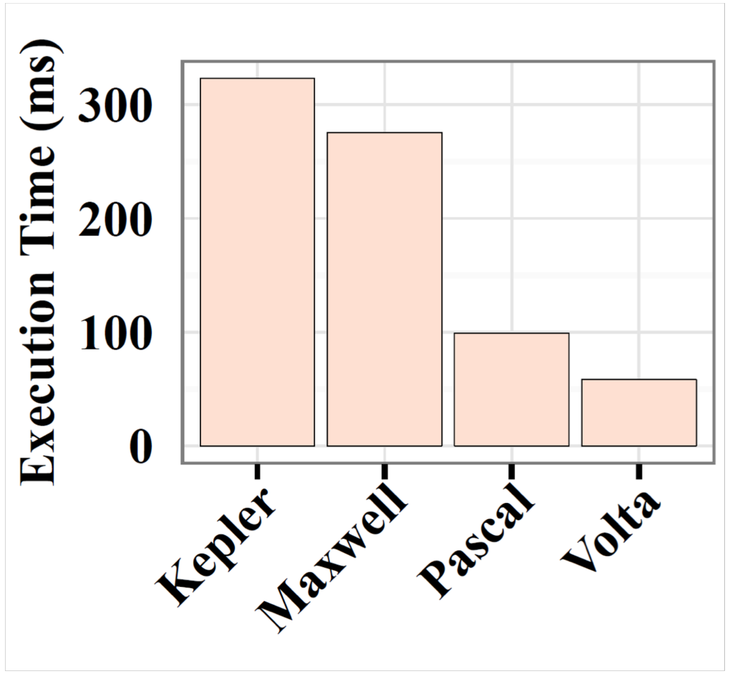
Constant memory



Shared memory



Texture memory



Special Memory Units' Performance Improvement Across GPUs

All types of memories on newer GPUs have improved performances

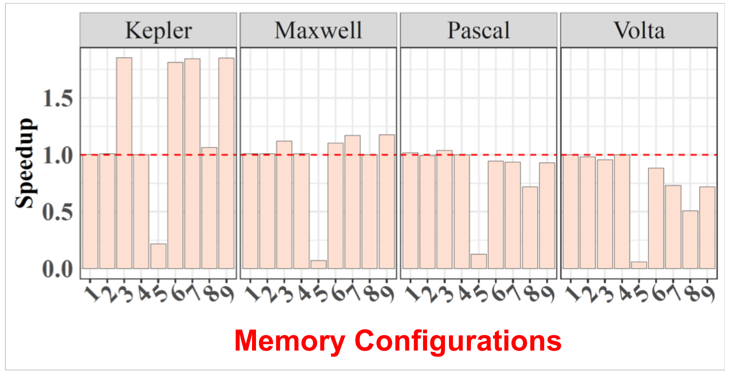
Special Memory Units’ Performance Improvement Across GPUs

CUDA Kernels – Mixed Types of Memory

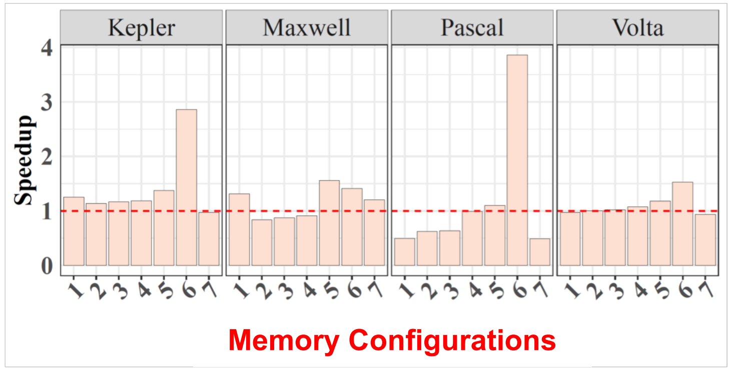
- Used 3 representative kernels
  - Sparse Matrix - Vector Multiplication (SPMV)
  - Matrix - Matrix Multiplication
  - Computational Fluid Dynamics

- Each data placement configuration uses multiple types of special memories

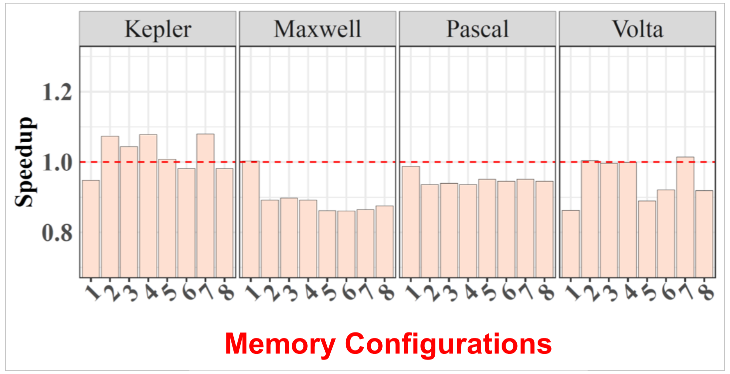
SPMV



MM



CFD



Speedup

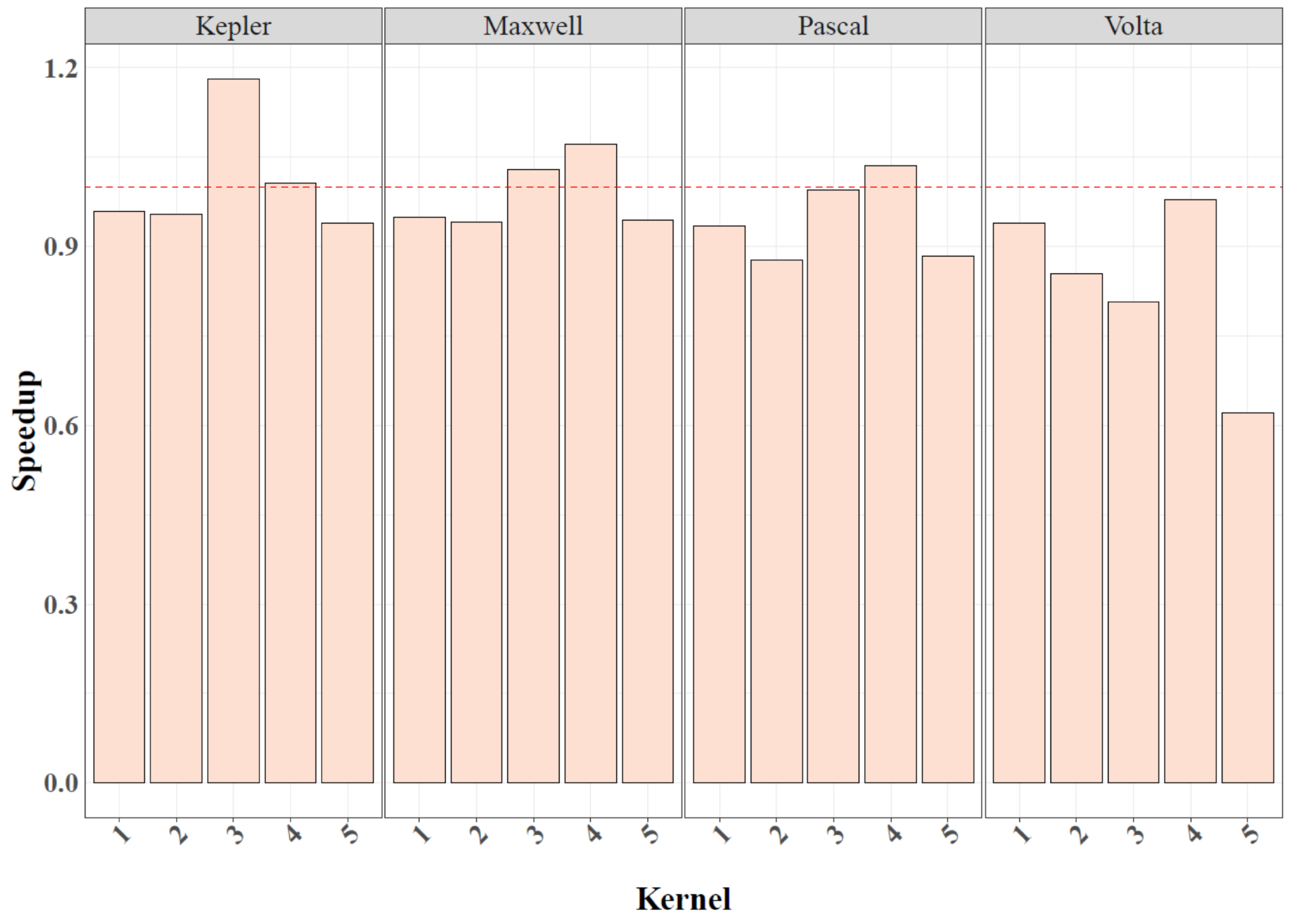
Proxy App - Lulesh

- ➢ Used mixed memory implementation
- ➢ Using special memory in real application is cumbersome
- ➢ Require a lot of modification
- ➢ Used 3 different data sizes (e.g., 4, 45, 90)
    - ➢ For larger data sizes (e.g., 45, 90) unable to use constant, shared memory

Speedup

Memory properties of special memories significantly limit their usage in real application

Data Size 45



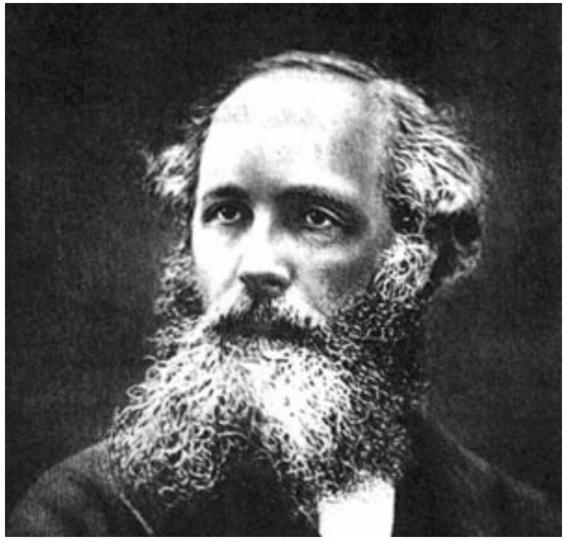
Key Takeaways

- ➢All types of memories on newer GPUs have improved performances
- ➢Global memory bandwidth, unified cache design helps narrow the performance gap between global and special memories
- ➢Memory properties of special memories significantly limit their usage in real application

Future Work

- Investigate the data placement optimization on energy consumption
- Automated code transformation to exploit special memories

Johannes Kepler



James Maxwell

Blaise Pascal

Alessandro Volta

So, Who Won?

Thank you ☺

- All types of memories on newer GPUs have improved performances
- Global memory bandwidth, unified cache design helps narrow the performance gap between global and special memories
- Memory properties of special memories significantly limit their usage in real application