GPU Progress and Directions for Earth System Modeling

Stan Posey, sPosey@nvidia.com
NVIDIA, Santa Clara, CA, USA
Agenda: GPU Progress and Directions for Earth System Modeling

- Introduction of GPUs in HPC
- NVIDIA Application Strategy
- GPU Progress in ES Modeling
- NVIDIA Technology Directions
GPU Growth as Mainstream HPC Technology

The buyer plans for including accelerators in their next technical computing server purchase has more than doubled from 29% to over 65% in last 20 months.

IDC Market Research
April, 2013
All Major OEM Servers Offer NVIDIA GPUs
Schematic of GPU Accelerated Computing

**CPU**
Optimized for Serial Tasks

**GPU Accelerator**
Optimized for Parallel Tasks

- 10x Peak Performance
- 5x Energy Efficiency
How Applications Are Usually Accelerated

Application Software

Read input, decomposition

Heavy Parallel Operations

50% - 75% Hot Spot of Profile in Small % LoC

Parallel Operations

(Investigating OpenACC for more tasks on GPU)

Global solution, write output

GPU

- Hand-CUDA Parallel
- GPU Libraries, CUBLAS
- OpenACC Directives

CPU

Parallel Operations

- Read input, decomposition
- 50% - 75% Hot Spot of Profile in Small % LoC
- (Investigating OpenACC for more tasks on GPU)
- Global solution, write output
Real Application Speedups

- 146X: Medical Imaging, U of Utah
- 36X: Molecular Dynamics, U of Illinois, Urbana
- 18X: Video Transcoding, Elemental Tech
- 50X: Matlab Computing, AccelerEyes
- 100X: Astrophysics, RIKEN

- 149X: Financial Simulation, Oxford
- 47X: Linear Algebra, Universidad Jaime
- 20X: 3D Ultrasound, Techniscan
- 130X: Quantum Chemistry, U of Illinois, Urbana
- 30X: Gene Sequencing, U of Maryland
- Full application? *Often kernel only without data transfer* . . .

- What is the reference CPU? *Often old and dusty x86* . . .

- How many CPU cores in the comparison? *Often 1 core . . . but who uses only 1 core nowadays?*

**NOTE:** Missing context often fault of NVIDIA and not the organizations referenced.
Next Migration Underway: Accelerated HPC

- Titan: 18,688 Tesla K20X GPUs
- 27 Petaflops Peak, 17.59 Petaflops on Linpack
- 90% of Performance from GPUs
GPU-Driven Fast and Energy Efficient HPC

**TITAN at ORNL**
World’s Fastest Open Science Supercomputer
18,688 Tesla K20X GPU Accelerators
27 Petaflops Peak
90% of Performance from GPUs

**Eurora at CINECA**
World’s Most Energy Efficient Supercomputer
128 Tesla K20 GPU Accelerators
3150 MFLOPS/Watt
$100k Energy & 300 Tons of CO$_2$ Saving Per Year
GPU Acceleration at Leadership HPC Sites

United States
- Oak Ridge National Labs
- Lawrence Livermore National Labs
- Sandia National Labs
- NOAA Gaea (ORNL)
- NCSA Blue Waters
- NCAR Yellowstone (Geyser & Caldera)
- NASA Pledies; Discover

Germany
- Juelich
- HLRS
- Max Planck
- TU Dresden

UK
- Cambridge
- EPCC
- Oxford
- STFC

Rest of Europe
- BSC, Spain
- CINECA, Italy
- CEA, France
- CSCS, Switzerland

Japan
- Tokyo Tech
- RIKEN
- Tsukuba

China
- NSC, Shenzhen
- NSC, Tianjin
- CAS IPE

Rest of World
- MSU, Russia
- RAS, Russia
- IITs, India
Important OEM Collaborations in ES Modeling

Collaboration on large deployments; OpenACC development
- TITAN — ORNL; Blue Waters — NCSA; Gaea — NOAA/ORNL; Piz Daint — CSCS

Collaboration on strategic deployments; Member Openpower
- Yellowstone — NCAR; Discover — NASA GSFC
- Openpower Consortium: IBM, NVIDIA, Google, Mellanox, Tyan, others

Strategic large deployments including Tsubame – Tokyo Inst of Tech

Strategic deployments including Pleadies – NASA ARC
Remote Visualization With GPU-Driven VDI

VIRTUAL MACHINE
- Windows 7
- Applications
- NVIDIA GRID Enabled Virtual Desktop
- NVIDIA Driver

VIRTUAL DESKTOPS

VDI
POWERED BY NVIDIA GRID™
Beyond HPC

GPU-Driven
Big Data

GPU Adoption Underway in Data Analytics

- Analyzing Twitter
  - salesforce
- Searching Audio
  - shazam
- Visual Shopping
  - cortexica
- Real-time Video Delivery
  - elemental
Agenda: GPU Progress and Directions for Earth System Modeling

- Introduction of GPUs in HPC
- NVIDIA Application Strategy
- GPU Progress in ES Modeling
- NVIDIA Technology Directions
GPU Motivation: Higher Resolution = High Cost

Resolution is of key importance to increase simulation quality
2x resolution $\approx 10x$ computational cost

Source:

CSCS
Swiss National Supercomputing Centre

Reality
Example: NASA and Global Cloud Resolving GEOS-6

Programming weather, climate, and earth-system models on heterogeneous multi-core platforms
September 7-8, 2011 at the National Center for Atmospheric Research in Boulder, Colorado

The Finite-Volume Dynamical Core on GPUs within GEOS-5
- Dr. William Putman, Global Modeling and Assimilation Office, NASA GSFC

NASA targeting GEOS global model resolution at sub-10-km to 1-km range

<table>
<thead>
<tr>
<th>Grid resolution</th>
<th>Westmere CPU cores</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 KM</td>
<td>12,000</td>
<td>Possible today</td>
</tr>
<tr>
<td>3 KM</td>
<td>300,000</td>
<td>Reasonable but not available</td>
</tr>
<tr>
<td>1 KM</td>
<td>10,000,000</td>
<td>Impractical, need accelerators</td>
</tr>
</tbody>
</table>

NVIDIA Application Strategy Since 2010

- Initial focus on climate and atmospheric research
  - Opportunities to refactor code and use of CUDA

- Later focus on operational models with directives
  - User community imposed Fortran-only requirements

- Direct investments in applications engineering
  - Current collaborations in 6 models and growing

- Continued development in libraries and OpenACC
  - CUBLAS, CUSPARSE, collaborations with PGI, CAPS, Cray

- Collaborations with OEMs (development, benchmarks, etc.)
  - Cray, IBM, HP, SGI, etc.
Programming Strategies for GPU Acceleration

Applications

- **GPU Libraries**
  - Provides Fast “Drop-In” Acceleration

- **OpenACC Directives**
  - GPU-acceleration in Standard Language (Fortran, C, C++)

- **Programming Languages**
  - Maximum GPU Architecture Flexibility

Increasing Development Effort
## NVIDIA Application Engineering Investments

<table>
<thead>
<tr>
<th>Model</th>
<th>Focus</th>
<th>GPU Approach</th>
<th>Collaboration</th>
</tr>
</thead>
<tbody>
<tr>
<td>WRF</td>
<td>NWP/Climate</td>
<td>CUDA C, OpenACC</td>
<td>NCAR, Cray, NVIDIA</td>
</tr>
<tr>
<td>COSMO</td>
<td>NWP/Climate</td>
<td>CUDA C, OpenACC</td>
<td>CSCS, SCS, MeteoSwiss, NVIDIA</td>
</tr>
<tr>
<td>CAM-SE</td>
<td>Climate</td>
<td>CUDA Ftn, OpenACC</td>
<td>ORNL, Cray, NVIDIA</td>
</tr>
<tr>
<td>NIM/FIM</td>
<td>NWP/Climate</td>
<td>F2C-ACC, OpenACC</td>
<td>NOAA, OACC Vendors, NVIDIA</td>
</tr>
<tr>
<td>GEOS-5</td>
<td>Climate</td>
<td>CUDA Ftn, OpenACC</td>
<td>NASA, NVIDIA</td>
</tr>
<tr>
<td>NEMO</td>
<td>Ocean Model</td>
<td>OpenACC</td>
<td>NVIDIA, STFC (future)</td>
</tr>
</tbody>
</table>

### Other Evaluations:
- GFS, COAMPS, MPAS-A, ROMS; ICON, UKMO GungHo; GRAPES (CN), OLAM (BR)

### Other Investments:
- Government and Research Institutes without direct NVIDIA collaboration
Example: NOPP/ONR Funding in Accelerated HPC

Total of $3.75M funding distributed among 4 – 8 awards (closed Apr 2013)

Models: GFS, HIRAM, NIM, MOM, CESM, HYCOM, CICE, Wavewatch3, NUMA

ONR BAA13-011:

Advancing Air-Ocean-Land-Ice Global Coupled Prediction on Emerging Computational Architectures

Predictive simulations on heterogeneous architectures Central Processing Unit (CPU), MIC, GPU: identification of representative code patterns that either look particularly amenable or intractable to refactoring; establishment of pathways to maintain single source code compatible with multiple platforms; and determination of mechanisms to achieve optimal performance and portability.
Rapid OpenACC Growth Since 2011 Founding

26+ Community Applications

<table>
<thead>
<tr>
<th>2012</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cloverleaf</td>
<td>MiniMD</td>
</tr>
<tr>
<td>COSMO (Physics)</td>
<td>NICAM</td>
</tr>
<tr>
<td>DNS</td>
<td>NIM</td>
</tr>
<tr>
<td>EMGS ELAN</td>
<td>NEMO GYRE</td>
</tr>
<tr>
<td>GAMESS</td>
<td>PALM-GPU</td>
</tr>
<tr>
<td>GENE</td>
<td>Quantum Espresso</td>
</tr>
<tr>
<td>GEOS</td>
<td>RAMSES</td>
</tr>
<tr>
<td>GTC</td>
<td>ROMS</td>
</tr>
<tr>
<td>Harmonie</td>
<td>S3D</td>
</tr>
<tr>
<td>HBM</td>
<td>Seismic CPML</td>
</tr>
<tr>
<td>NIM</td>
<td>SPECFEM-3D</td>
</tr>
<tr>
<td>S3D</td>
<td>UPACS</td>
</tr>
<tr>
<td>WRF</td>
<td>WRF</td>
</tr>
</tbody>
</table>

HPC Industry Support Grows 2x

[List of companies and logos]
OpenACC in Practice for NWP and Climate

Examples: Use of directives with less effort; ease in maintenance and flexibility

- COSMO (Physics)
  - Goal: preserve physics code (22% of runtime), augmenting refactored dynamics in CUDA
  - Physics scheme speedup 4.2X vs. multi-core Xeon

- NICAM (Climate)
  - Hotspots using CUDA, then OpenACC
  - CUDA: 3.1x faster on GPU vs. CPU node
  - OpenACC: (preliminary) = 69-77% of CUDA
    - More portable, more maintainable
    - Full OpenACC port in progress

Results from MeteoSwiss/CSCS, and Tokyo Inst of Technology
TiTech NWP 2010 Achievement: ASUCA 145 TF

Tsubame 2.0
Tokyo Institute of Technology

- 1.19 Petaflops
- 4,224 Tesla M2050 GPUs

3990 Tesla M2050s
145.0 Tflops SP
76.1 Tflops DP

ASUCA and NWP Simulation on Tsubame 2.0, TiTech Supercomputer:
Dr. Takayuki Aoki, GSIC, Tokyo Institute of Technology, Tokyo Japan
TiTech Winner of 2011 Gordon Bell Prize

Special Achievement in Scalability and Time-to Solution

"Peta-scale Phase-Field Simulation for Dendritic Solidification on the TSUBAME 2.0 Supercomputer"
-- T. Shimokawabe, T. Aoki, et. al.

Tsubame 2.0
Tokyo Institute of Technology

4,224 Tesla GPUs + 2,816 x86 CPUs
Agenda: GPU Progress and Directions for Earth System Modeling

- Introduction of GPUs in HPC
- NVIDIA Application Strategy
- GPU Progress in ES Modeling
- NVIDIA Technology Directions
WRF

- [http://irina.eas.gatech.edu/EAS8802_Spring2011/Lecture7.pdf](http://irina.eas.gatech.edu/EAS8802_Spring2011/Lecture7.pdf)
- [http://www.mmm.ucar.edu/wrf/users/docs/user_guide_V3.1/users_guide_chap5.htm#_Installing_WRF](http://www.mmm.ucar.edu/wrf/users/docs/user_guide_V3.1/users_guide_chap5.htm#_Installing_WRF)
- [http://www.mmm.ucar.edu/wrf/WG2/GPU/WSM5.htm](http://www.mmm.ucar.edu/wrf/WG2/GPU/WSM5.htm)

WRF: Operational in 21 Countries; 153 Total

GPU Status of WRF Developments

Several non-trunk efforts at various stages:

- **2010**: Dynamics and some physics by Thomas Nipen at UBC – source at NVIDIA
- **2011**: Shortwave radiation model by NVIDIA (G. Ruetsch) and PGI (available)
- **2012**: C-DAC and HPC-FTE group working with NVIDIA India Developers
- **2012**: NOAA announced NIM dycore with WRF physics, but now GFS and YSU
- **2012**: Cray and OpenACC (Pete Johnsen) results at NCAR multi-core workshop
- **Ongoing**: KernelGen project: [www.kernelgen.org](http://www.kernelgen.org) update at NVIDIA GTC 2013
- **Ongoing**: 50% of physics schemes by Space Science Engineering Center, UW-M

Trunk efforts at various stages:

- WSM5 physics model (15% - 25%) in release 3.2 from 2009 (now dormant)
- WRF 3.5 with OpenACC – NVIDIA and NCAR (MMM – Dave Gill) collaboration
**OpenACC Developments for WRF 3.4/3.5**

**WRF Experiments on GPU Accelerators using OpenACC**
- Pete Johnsen, Cray, Inc.

**WRF routine **advance**_w**
- Dynamics routine to advance vertical velocity
- Standard Fortran use with OpenACC directives
- 2.1x speedup for 16 cores

Published WRF Speedups from SSEC

<table>
<thead>
<tr>
<th>WRF Module name</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single moment 6-class microphysics</td>
<td>500x</td>
</tr>
<tr>
<td>Eta microphysics</td>
<td>272x</td>
</tr>
<tr>
<td>Purdue Lin microphysics</td>
<td>692x</td>
</tr>
<tr>
<td>Stony-Brook University 5-class microphysics</td>
<td>896x</td>
</tr>
<tr>
<td>Betts-Miller-Janjic convection</td>
<td>105x</td>
</tr>
<tr>
<td>Kessler microphysics</td>
<td>816x</td>
</tr>
<tr>
<td>New Goddard shortwave radiance</td>
<td>134x</td>
</tr>
<tr>
<td>Single moment 3-class microphysics</td>
<td>331x</td>
</tr>
<tr>
<td>New Thompson microphysics</td>
<td>153x</td>
</tr>
<tr>
<td>Double moment 6-class microphysics</td>
<td>206x</td>
</tr>
<tr>
<td>Dudhia shortwave radiance</td>
<td>409x</td>
</tr>
<tr>
<td>Goddard microphysics</td>
<td>1311x</td>
</tr>
<tr>
<td>Double moment 5-class microphysics</td>
<td>206x</td>
</tr>
<tr>
<td>Total Energy Mass Flux surface layer</td>
<td>214x</td>
</tr>
<tr>
<td>Mellar-Yamada Nakanishi Niino surface layer</td>
<td>113x</td>
</tr>
<tr>
<td>Single moment 5-class microphysics</td>
<td>350x</td>
</tr>
<tr>
<td>Yonsei University planetary boundary layer scheme</td>
<td>108x</td>
</tr>
<tr>
<td>5-Layer Thermal diffusion land surface layer</td>
<td>211x</td>
</tr>
<tr>
<td>Pleim-Xiu surface layer</td>
<td>665x</td>
</tr>
</tbody>
</table>

Hardware: Core-i7 3930K, 1 core use; GTX 590 GeForce

Benchmark: CONUS 12 km for 24 Oct 24
433 x 308, 35 levels

WRF V3.2 and V3.3

Verification: WSM5 by NREL (J. Michalakes) and NVIDIA Applications Engr [Next 2 slides]

NOTE: All times without CPU data transfer

Source: Bormin Huang, Space Science and Engineering Center, UW-M
NVIDIA Verification of WSM5 from SSEC

**NOTE:** Times with no CPU data transfer

<table>
<thead>
<tr>
<th>2 x Core-i7 3930K, Total of 12 Cores; GPUs: 2 x Tesla K20X</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark: CONUS 12 km resolution for October 24, 2001;</td>
</tr>
<tr>
<td>433 x 308 grid points, 35 vertical levels</td>
</tr>
</tbody>
</table>

CONUS 12 km resolution domain for October 24, 2001, and is 433 x 308 horizontal grid points with 35 vertical levels.
NREL Verification of WSM5 from SSEC

High performance computing enhancements to WRFV3.5
- John Michalakes, National Renewable Energy Laboratory, USA


14th Annual WRF Users’ Workshop, 24 – 26 Jun 2013, Boulder, CO, USA

- GigaFlop ratings of WSM5 Thompson microphysics scheme (not a full model run)
- Data transfer times with CPU excluded in all results

Performance Results
K20X GPU vs. CPU = 3.34x
K20X GPU vs. Phi = 1.54x

**WSM5 Microphysics**

- Xeon Sandybridge (2x8 core)
- Xeon Phi (60C, 4T/C)
- NVIDIA Fermi GTX 590
- NVIDIA Kepler K20

Performance Results
K20X GPU vs. CPU = 3.34x
K20X GPU vs. Phi = 1.54x
GPU Accelerated Computing Growing Fast

"Intel is not taking share away from NVIDIA but rather both are expanding the use of accelerators."

Intersect360 Research
HPC User Site Census
July, 2013
# Accelerator Progress Reports at NCAR Workshop

*Programming weather, climate, and earth-system models on heterogeneous multi-core platforms*

*September 19-20, 2013 at the National Center for Atmospheric Research in Boulder, Colorado*

## Session:

<table>
<thead>
<tr>
<th>Session</th>
<th>Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Session 1 - Common Features and Challenges of Using Multi-core Systems</td>
<td>CPU</td>
</tr>
<tr>
<td>Session 2 - Optimization for Hybrid GPU systems</td>
<td>GPU</td>
</tr>
<tr>
<td>Session 3a - Porting and Optimizing for Intel Xeon and Xeon Phi</td>
<td>Phi</td>
</tr>
<tr>
<td>Session 3b - Porting and Optimizing for Intel Xeon and Xeon Phi</td>
<td>Phi</td>
</tr>
<tr>
<td>Session 4 - Common Features and Challenges of Using Multi-core Systems</td>
<td>CPU</td>
</tr>
<tr>
<td>Session 5 - Porting and Optimizing for Intel Xeon and Xeon Phi</td>
<td>Phi</td>
</tr>
<tr>
<td>Session 6a - Optimizing for GPUs using CUDA and OpenACC</td>
<td>GPU</td>
</tr>
<tr>
<td>Session 6b - Optimizing for GPUs using CUDA and OpenACC</td>
<td>GPU</td>
</tr>
</tbody>
</table>

## Organizers

- Jene Carpenter (nrel), Mark Govett (berl), Chris Kerr (gfdl), Rich Loft (ncar), Bill Putman (gsfc), William Sawyer (cscb)

Running the FIM and NIM Weather Models on GPUs
- Mark Govett (NOAA Earth System Research Laboratory)

NIM Serial Performance (2013)

- No changes to the source code
- Single Socket Performance
  - 10K horizontal points, 96 vertical levels
- Very efficient CPU performance
  - Measured 29% of peak performance (Intel Westmere)

<table>
<thead>
<tr>
<th>NIM</th>
<th>Opteron</th>
<th>Westmere</th>
<th>SandyBridge</th>
<th>Fermi</th>
<th>K20x</th>
</tr>
</thead>
<tbody>
<tr>
<td>runtime</td>
<td>143.0</td>
<td>86.8</td>
<td>60.0</td>
<td>25.0</td>
<td>20.7</td>
</tr>
</tbody>
</table>

- Parallel performance
  - Being run on up to 160 GPUs
  - Working on optimizing inter-GPU communications
NIM Parallel Performance

- Weak Scaling with Communications Optimization
  - Moved collective operation to the CPU instead of doing it on the GPU using GPU MappedMemory
  - Too many small writes across the PCIe bus
  - Resulted in a 5-17x speedup for the Pack Operation

<table>
<thead>
<tr>
<th>GPUs</th>
<th>GPU to GPU Comm Time</th>
<th>Total Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>232 (22%)</td>
<td>1034</td>
</tr>
<tr>
<td>40</td>
<td>247 (23%)</td>
<td>1054</td>
</tr>
<tr>
<td>160</td>
<td>266 (24%)</td>
<td>1076</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Operation</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialization</td>
<td>3 (1%)</td>
</tr>
<tr>
<td>Pack Data on GPU</td>
<td>45 (17%)</td>
</tr>
<tr>
<td>CPU – GPU Copy</td>
<td>59 (22%)</td>
</tr>
<tr>
<td>MPI Comms</td>
<td>82 (31%)</td>
</tr>
<tr>
<td>UnPack on GPU</td>
<td>77 (29%)</td>
</tr>
<tr>
<td>Total</td>
<td>266</td>
</tr>
</tbody>
</table>
3.5KM NIM on Titan in 2013

- Dynamics on GPU, Physics on CPU + OMP for now

- 10 day forecast, 10,262 horizontal points / GPU

<table>
<thead>
<tr>
<th>Resolution KM</th>
<th>Vertical Levels</th>
<th>GPUs</th>
<th>Dynamics Time</th>
<th>CPU-GPU Transfer</th>
<th>Physics Time</th>
<th>Total Time In Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>96</td>
<td>80</td>
<td>1700</td>
<td>400</td>
<td>1900</td>
<td>1.1</td>
</tr>
<tr>
<td>15</td>
<td>96</td>
<td>320</td>
<td>1700</td>
<td>400</td>
<td>1900</td>
<td>2.2</td>
</tr>
<tr>
<td>7.5</td>
<td>96</td>
<td>1280</td>
<td>1700</td>
<td>400</td>
<td>1900</td>
<td>4.4 (1.8%)</td>
</tr>
<tr>
<td>3.75</td>
<td>96</td>
<td>5120</td>
<td>1700</td>
<td>400</td>
<td>1900</td>
<td>8.8 (3.6%)</td>
</tr>
</tbody>
</table>
NEMO

Accelerating NEMO with OpenACC
- Maxim Milakov (NVIDIA)

NEMO Acceleration Using OpenACC

Background

- NEMO ocean modeling framework: http://www.nemo-ocean.eu/
- Used by 240 projects in 27 countries (14 in Europe, 13 elsewhere)

Approach

- Project based on NEMO 3.4, use of PGI Fortran compiler 12.9 preview
- Flat profile, 1st routine is 6%, many routines to accelerate for overall benefit
- OpenACC “present” clause keeps data on the device between subroutine calls
- Directives for 41 routines: rearranged loops in 12, temporary arrays in 13
- Other changes for improved MPI communication, other miscellaneous
NEMO Acceleration Using OpenACC

GYRE_50 Configuration, I/O disabled, OpenACC 1.0: Speedup ~5x

Source: Maxim Milakov, NVIDIA
Benchmarking - hardware

- “Sandy Bridge + Kepler” nodes, each having:
  - CPU: 2 sockets * Xeon E5-2670 (Sandybridge), 2.6GHz (3.3GHz Turbo Boost), 8 cores, 64 GB RAM
  - GPU: 2x Tesla K20X, ECC off, 6GB RAM each
  - 4x FDR Infiniband (56 Gb/s)

- Running configuration is GYRE_50 (1/4 degree), requires about 23GB of total RAM, fits 4 K20X

- The code is running on 2 nodes

- The performance is measured by running 1000 time steps, startup and shutdown overheads are not included in figures
Benchmarking - results

GPU vs. CPU - 3.1x speedup

Time steps per second

- Intel, CPU
- PGI, CPU
- GPU
Agenda: GPU Progress and Directions for Earth System Modeling

- Introduction of GPUs in HPC
- NVIDIA Application Strategy
- GPU Progress in ES Modeling
- NVIDIA Technology Directions
### Tesla Kepler Family

**World's Fastest and Most Efficient HPC Accelerators**

<table>
<thead>
<tr>
<th>GPUs</th>
<th>Single Precision Peak (SGEMM)</th>
<th>Double Precision Peak (DGEMM)</th>
<th>Memory Size</th>
<th>Memory Bandwidth (ECC off)</th>
<th>System Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>K20X</td>
<td>3.95 TF (2.90 TF)</td>
<td>1.32 TF (1.22 TF)</td>
<td>6 GB</td>
<td>250 GB/s</td>
<td>Server only</td>
</tr>
<tr>
<td>K20</td>
<td>3.52 TF (2.61 TF)</td>
<td>1.17 TF (1.10 TF)</td>
<td>5 GB</td>
<td>208 GB/s</td>
<td>Server + Workstation</td>
</tr>
<tr>
<td>K10</td>
<td>4.58 TF</td>
<td>0.19 TF</td>
<td>8 GB (4 GB ea.)</td>
<td>320 GB/s</td>
<td>Server only</td>
</tr>
</tbody>
</table>

**Weather & Climate, Physics, BioChemistry, CAE, Material Science**

**Image, Signal, Video, Seismic**
Kepler
Fastest, Most Efficient HPC Architecture Ever

SMX  3x Performance per Watt

Hyper-Q  Easy Speed-up for Legacy MPI Apps

Dynamic Parallelism  Parallel Programming Made Easier than Ever
NVIDIA GPU Roadmap (Details Require NDA)

NVIDIA Roadmap Trends:
* Increasing number of more flexible cores
* Larger and faster memories (6 GB today)
* Enhanced programming and standards
* Tighter integration with OEM systems
ARM Support Now Available Since CUDA 5.5

CUDA 5.5 Highlights

Full Support for ARM Platforms
- Native compilation on ARM

Optimized for MPI
- Faster Hyper-Q for all Linux distros
- MPI workload prioritization

Guided Performance Analysis
- Step-by-step optimization

Available Now
NVIDIA Quadro K6000: Kepler-Based 12GB GPU

Announced July 2013

Available Q3 2013

<table>
<thead>
<tr>
<th>QUADRO K6000 QUICK SPECS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CUDA Parallel-Processing Cores</td>
<td>2880</td>
</tr>
<tr>
<td>Frame Buffer Memory</td>
<td>12 GB GDDR5</td>
</tr>
<tr>
<td>Max Power Consumption</td>
<td>225 W</td>
</tr>
<tr>
<td>Graphics Bus</td>
<td>PCI Express 3.0 x16</td>
</tr>
<tr>
<td>Display Connectors</td>
<td>DVI-I (1), DVI-D (1) DP 1.2 (2), Optional Stereo (1)</td>
</tr>
<tr>
<td>Form Factor</td>
<td>4.376&quot; H x 10.5&quot; L Dual Slot</td>
</tr>
</tbody>
</table>
Summary For GPUs and ES Modeling

- Opportunities exist for GPUs to provide significant performance acceleration for ES Models
  - Improved simulation quality from higher resolutions
  - Faster time to predictions for operational forecasting
  - Cut down energy consumption in IT procedures

- Simulations recently considered intractable are now possible
  - Global models are cloud resolving scale
  - Parameter physics at higher resolutions and more frequent time steps
  - Expanded and more common use of ensembles
Thank you & Questions

Stan Posey, sposey@nvidia.com
NVIDIA, Santa Clara, CA, USA