

nVISION 08
THE WORLD OF VISUAL COMPUTING

High Performance Computing

David B. Kirk, Chief Scientist, NVIDIA

A History of Innovation

- Invented the Graphics Processing Unit (GPU)
- Pioneered programmable shading
- Over 2000 patents*



1995
NV1
1 Million
Transistors



1999
GeForce 256
22 Million
Transistors



2002
GeForce 4
63 Million
Transistors



2003
GeForce FX
130 Million
Transistors



2004
GeForce 6
222 Million
Transistors



2005
GeForce 7
302 Million
Transistors



2006-2007
GeForce 8
754 Million
Transistors



2008
GeForce GTX 200
1.4 Billion
Transistors

Real-time Ray Tracing Demo

- Real system
- NVSG-driven animation and interaction
- Programmable shading
- Modeled in Maya, imported through COLLADA
- Fully ray traced

2 million polygons

Bump-mapping

Movable light source

5 bounce reflection/refraction

Adaptive antialiasing



Poznaj świadectwa z dokumentacji procesu
beatyfikacyjnego i kanonizacyjnego Jana Pawła II

CUDA



już od 7 marca



Książka wraz z filmem VCD „Cud” dostępna w księgarniach,
parafiach, salonach EMPIK, w księgarniach internetowych
i na stronie www.stanislawbm.pl tel. 012 429 52 17

PARTNER MEDIALNY
TVP 1 TVP KRAKÓW

SPONSORZY
Nasz Dziennik

GAZETA
Krajkowa

VOX

niedziela

GOŚC

WISŁA

CELESTYNA

WARTA

FIAT

EUROPEJSKI

WYDZIAŁ

CUDA™ Uses Kernels and Threads for Fast Parallel Execution

- Parallel portions of an application are executed on the GPU as **kernels**
 - One **kernel** is executed at a time
 - Many threads execute each **kernel**
- Differences between CUDA and CPU threads
 - CUDA threads are extremely lightweight
 - Very little creation overhead
 - Instant switching
 - CUDA uses 1000s of threads to achieve efficiency

Simple “C” Description For Parallelism

```
void saxpy_serial(int n, float a, float *x, float *y)
{
    for (int i = 0; i < n; ++i)
        y[i] = a*x[i] + y[i];
}
// Invoke serial SAXPY kernel
saxpy_serial(n, 2.0, x, y);
```

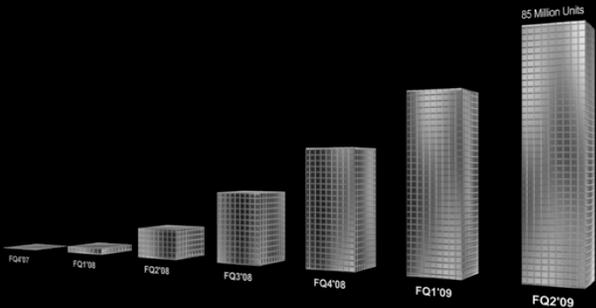
Standard C Code

```
__global__ void saxpy_parallel(int n, float a, float *x, float *y)
{
    int i = blockIdx.x*blockDim.x + threadIdx.x;
    if (i < n) y[i] = a*x[i] + y[i];
}
// Invoke parallel SAXPY kernel with 256 threads/block
int nblocks = (n + 255) / 256;
saxpy_parallel<<<nblocks, 256>>>(n, 2.0, x, y);
```

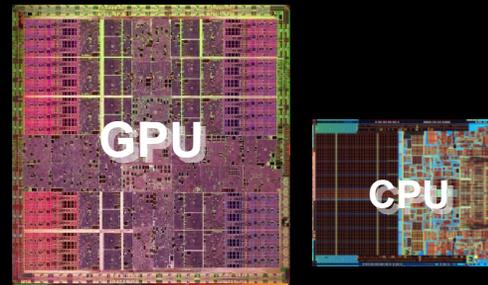
Parallel C Code

The Key to Computing on the GPU

- **Standard high level language support**
 - C, soon C++ and Fortran
 - Standard and domain specific libraries
- **Hardware Thread Management**
 - No switching overhead
 - Hide instruction and memory latency
- **Shared memory**
 - User-managed data cache
 - Thread communication / cooperation within blocks
- **Runtime and tool support**
 - Loader, Memory Allocation
 - C stdlib

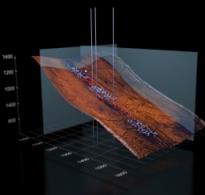


CUDA

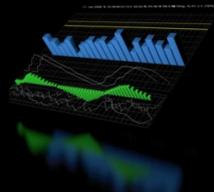


Heterogeneous Computing

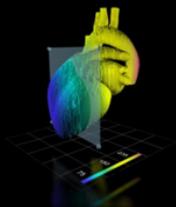
nVISION 08
THE WORLD OF VISUAL COMPUTING



Oil & Gas



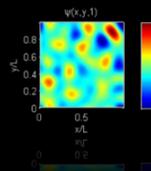
Finance



Medical



Biophysics



Numerics



Audio



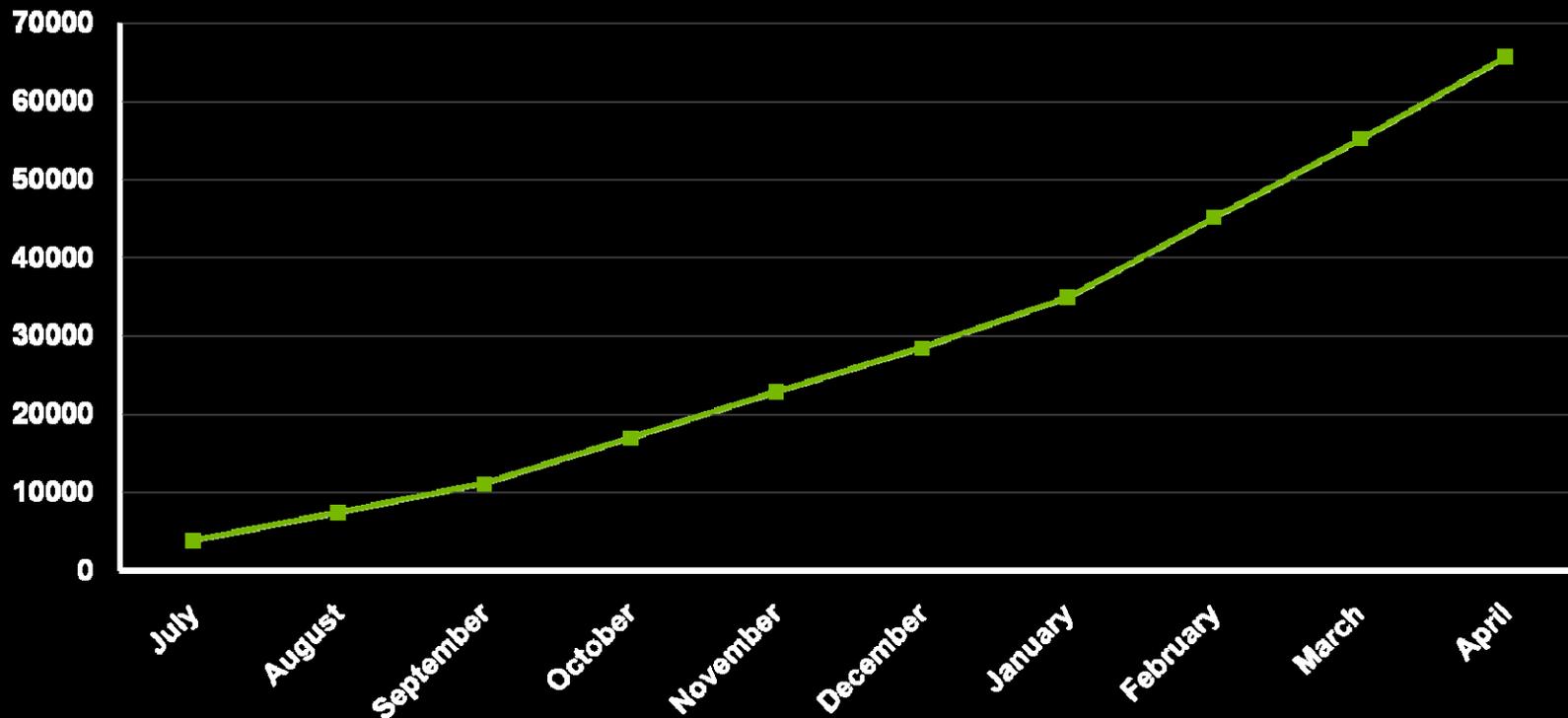
Video



Imaging

CUDA Compiler Downloads

© 2008 NVIDIA Corporation



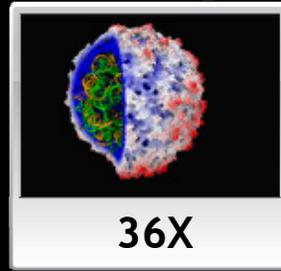
Universities Teaching Parallel Programming With CUDA

- Duke
- Erlangen
- ETH Zurich
- Georgia Tech
- Grove City College
- Harvard
- IIIT
- IIT
- Illinois Urbana-Champaign
- INRIA
- Iowa
- ITESM
- Johns Hopkins
- Kent State
- Kyoto
- Lund
- Maryland
- McGill
- MIT
- North Carolina - Chapel Hill
- North Carolina State
- Northeastern
- Oregon State
- Pennsylvania
- Polimi
- Purdue
- Santa Clara
- Stanford
- Stuttgart
- Suny
- Tokyo
- TU-Vienna
- USC
- Utah
- Virginia
- Washington
- Waterloo
- Western Australia
- Williams College
- Wisconsin

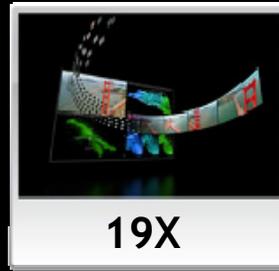
Wide Developer Acceptance



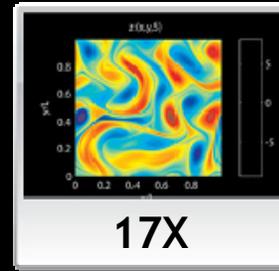
146X
Interactive visualization of volumetric white matter connectivity



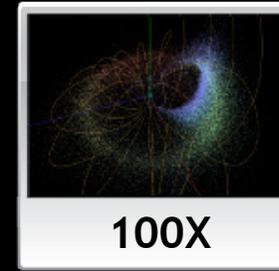
36X
Ionic placement for molecular dynamics simulation on GPU



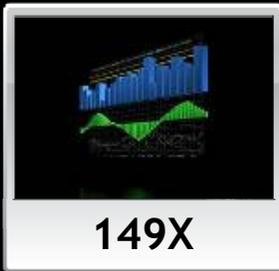
19X
Transcoding HD video stream to H.264



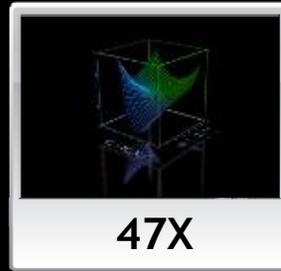
17X
Simulation in Matlab using .mex file CUDA function



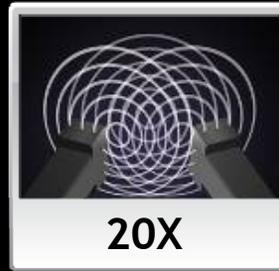
100X
Astrophysics N-body simulation



149X
Financial simulation of LIBOR model with swaptions



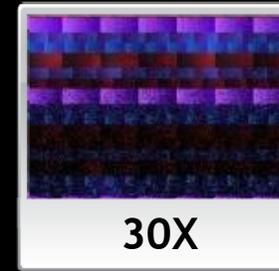
47X
GLAME@lab: An M-script API for linear Algebra operations on GPU



20X
Ultrasound medical imaging for cancer diagnostics

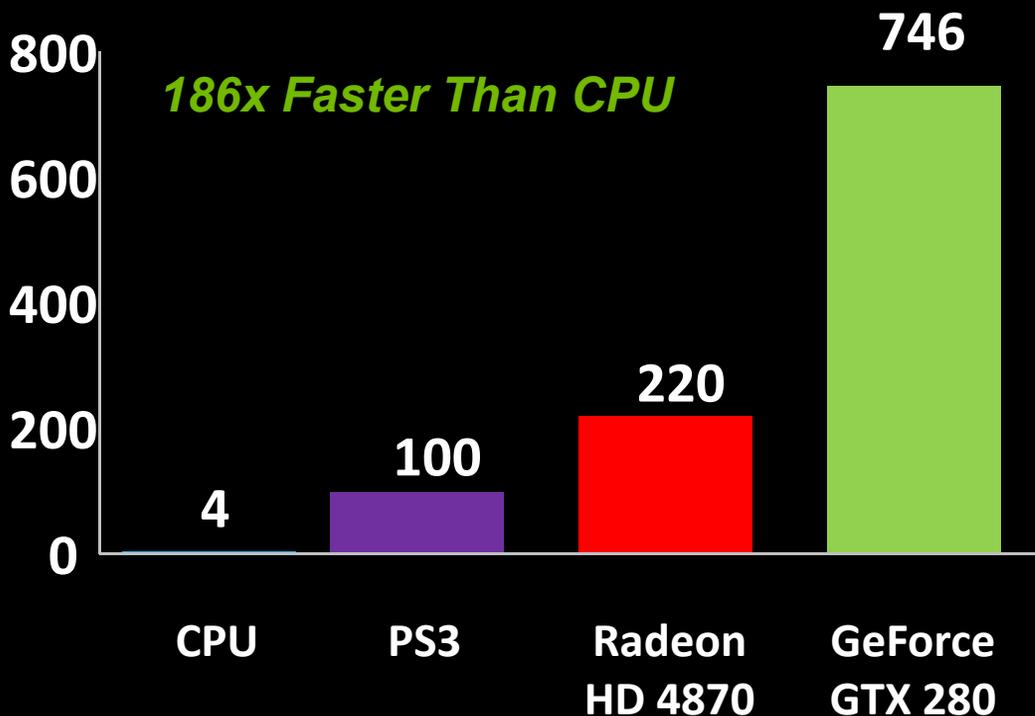


24X
Highly optimized object oriented molecular dynamics



30X
Cmatch exact string matching to find similar proteins and gene sequences

Folding@home on GeForce® / CUDA



Donor
Name: Jensen Huang
Team: Team WhoopAss
Hardware: Quadro FX 570M

Current Work Unit
Name: test protein A
Progress: 13824 / 5000000 = 0.276%
Performance: 350.18 Bar / sec
Time Left: 00h:09m:57m:18s

CUDA Zone

nVISION 08
 THE WORLD OF VISUAL COMPUTING

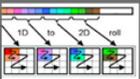

CUDA ZONE
USA - United States

DOWNLOAD CUDA WHAT IS CUDA CUDA U DEVELOPING WITH CUDA FORUMS NEWS AND EVENTS

LATEST CUDA NEWS Sign up for CUDA Developer Conference at NVISION 2008 and Learn How to Accelerate Your Application



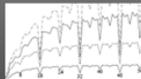
Programming Algorithms-by-Thread
10 x



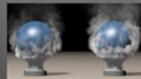
Concurrent Number Cruncher
10 x



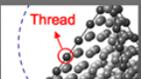
GPU4 VISION



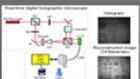
Efficient Computation of Sum Products on GPUs
270 x



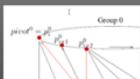
Low Viscosity Flow Simulations for Animation
55 x



MIDG
50 x



Real-time Digital Holographic Microscopy



Wait-free Programming for Computations on Graphics Processors



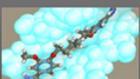
Real-time Visual Tracker by Stream Processing
10 x



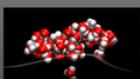
Ray Casting Deformable Models



Teraflops for Games and Derivatives Pricing
50 x



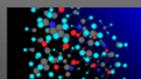
Molecular Dynamics of DNA and Liquids
18 x



GPUGRID.NET



Mixed Precision Linear Solvers
27 x



Accelerating Density Functional Calculations with GPU
40 x

Sort by Release Date
Share Your Work

Filter by Application Type

- Computational Fluid Dynamics
- Digital Content Creation
- Electronic Design Automation
- Finance
- Game Physics
- Graphics
- Imaging
- Numerics
- Life Sciences
- Libraries
- Oil & Gas
- Programming Tools
- Science
- Signal Processing
- Video & Audio

Filter by Content Type

- Application
- Code
- Multimedia
- Paper
- Presentation

Filter by Organization Type

- Academia
- Commercial
- Research

Faster is not “just Faster”

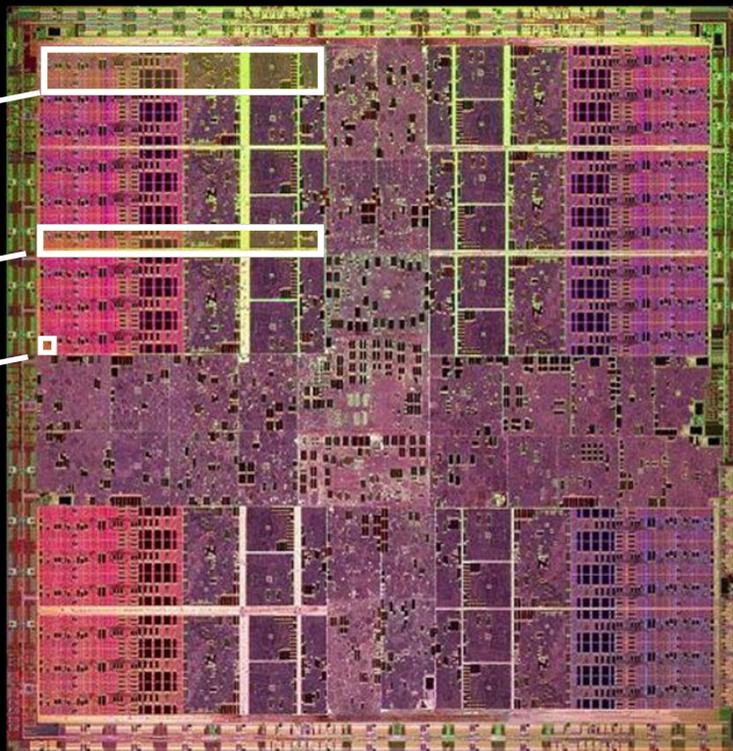
- 2-3X faster is “just faster”
 - Do a little more, wait a little less
 - Doesn't change how you work
- 5-10x faster is “significant”
 - Worth upgrading
 - Worth re-writing (parts of) the application
- 100x+ faster is “fundamentally different”
 - Worth considering a new platform
 - Worth re-architecting the application
 - Makes new applications possible
 - Drives “time to discovery” and creates fundamental changes in Science

Tesla™ T10: 1.4 Billion Transistors

Thread Processor
Cluster (TPC)

Thread Processor
Array (TPA)

Thread Processor

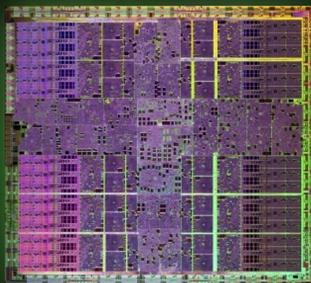
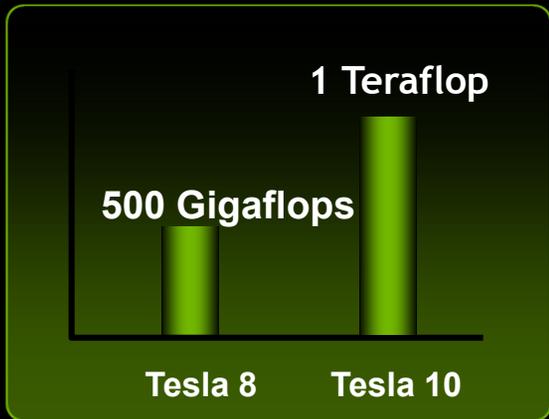


*Die Photo
of Tesla T10*

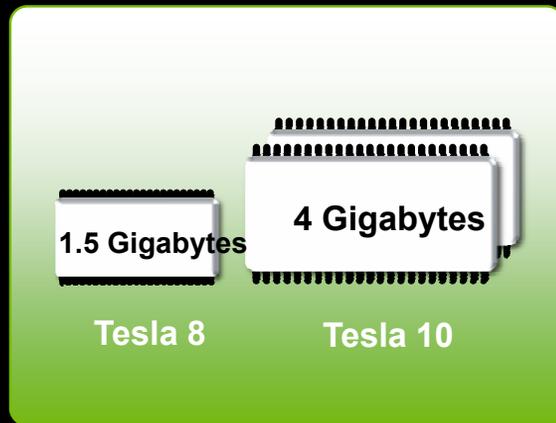


Tesla 10-Series

Double the Performance



Double the Memory



Double Precision

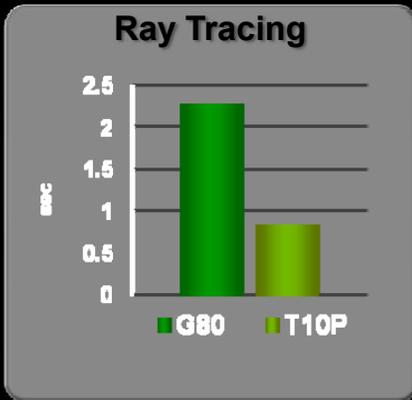
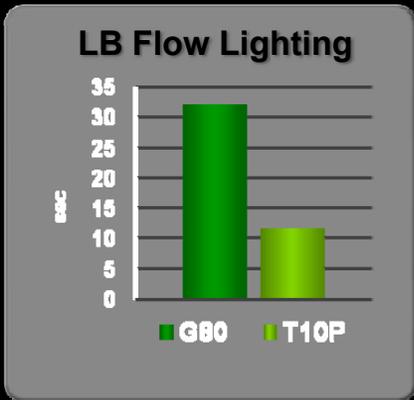
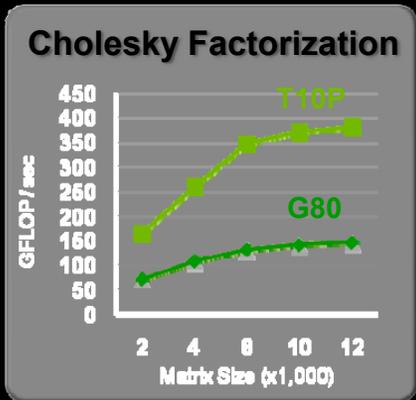
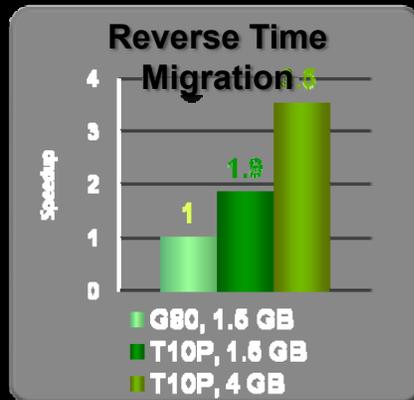
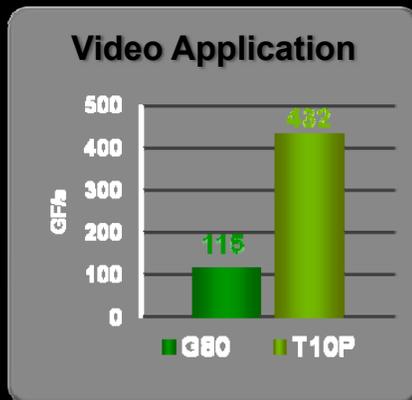
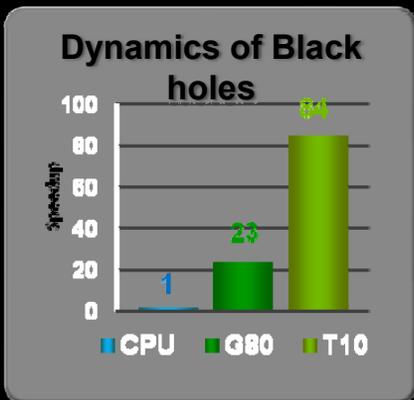
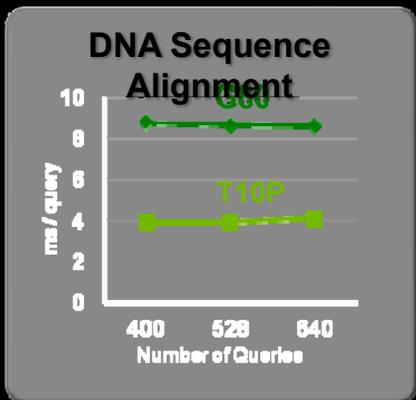
Three panels illustrating applications of Double Precision: Finance (a 3D bar chart), Science (a network diagram), and Design (a 3D model of a turbine).

Finance Science Design

Tesla T10 Double Precision Floating Point

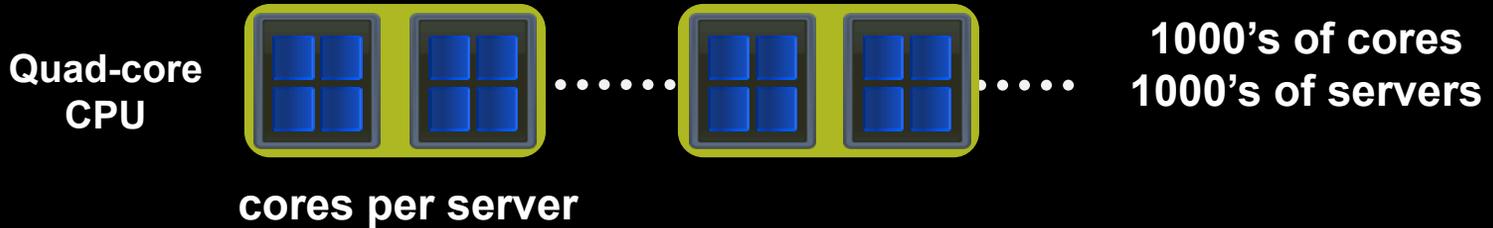
Precision	IEEE 754
Rounding modes for FADD	All 4 IEEE, round to nearest, zero, inf, -inf
Denormal handling	Full speed
NaN support	Yes
Overflow and Infinity support	Yes
Flags	No
FMA	Yes
Square root	Software with low-latency FMA-based convergence
Division	Software with low-latency FMA-based convergence
Reciprocal estimate accuracy	24 bit
Reciprocal sqrt estimate accuracy	23 bit
$\log_2(x)$ and 2^x estimates accuracy	23 bit

Double the Performance Using T10



How to Get to 100x?

Traditional Data Center Cluster

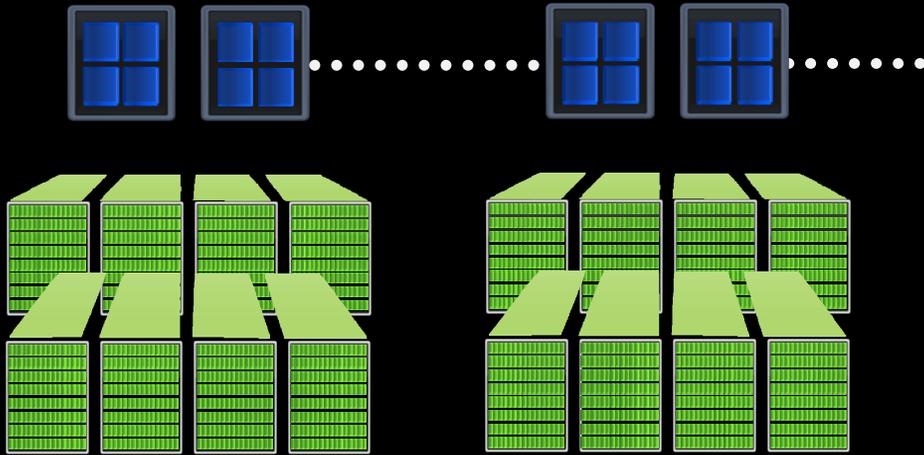


More Servers To Get More Performance



Heterogeneous Computing Cluster

10,000's processors per cluster



1928 processors

1928 processors

- Hess
- NCSA / UIUC
- JFCOM
- SAIC
- University of North Carolina
- Max Plank Institute
- Rice University
- University of Maryland
- GusGus
- Eotvas University
- University of Wuppertal
- IPE/Chinese Academy of Sciences
- Cell phone manufacturers



Building a 100TF datacenter

CPU 1U Server



4 CPU cores

0.07 Teraflop

\$ 2000

400 W

1429 CPU servers

\$ 3.1 M

571 KW

4 GPUs: 960 cores

4 Teraflops

\$ 8000

800 W

25 CPU servers

25 Tesla systems

\$ 0.31 M

27 KW

Tesla 1U System



10x lower cost

21x lower power

CPU

Tesla S1070 1U System



4 Teraflops¹

800 watts²

¹ single precision

² typical power

Tesla C1060 Computing Processor



957 Gigaflops¹

160 watts²

¹ single precision

² typical power

Application Software

Industry Standard C Language

Libraries

cuFFT

cuBLAS

cuDPP

System

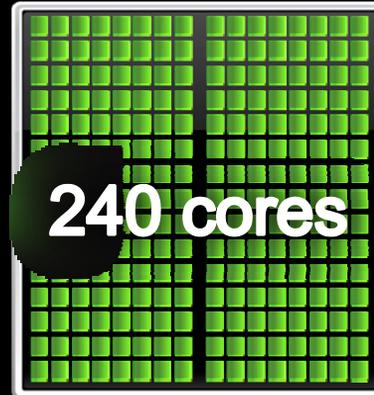
1U PCI-E Switch

CUDA Compiler

C Fortran Multi-core

CUDA Tools

Debugger Profiler



CUDA Source Code

Industry Standard C Language

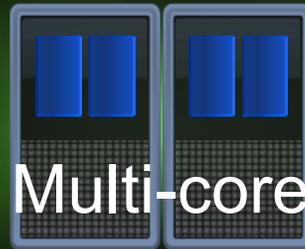
Industry Standard Libraries

CUDA Compiler

C Fortran

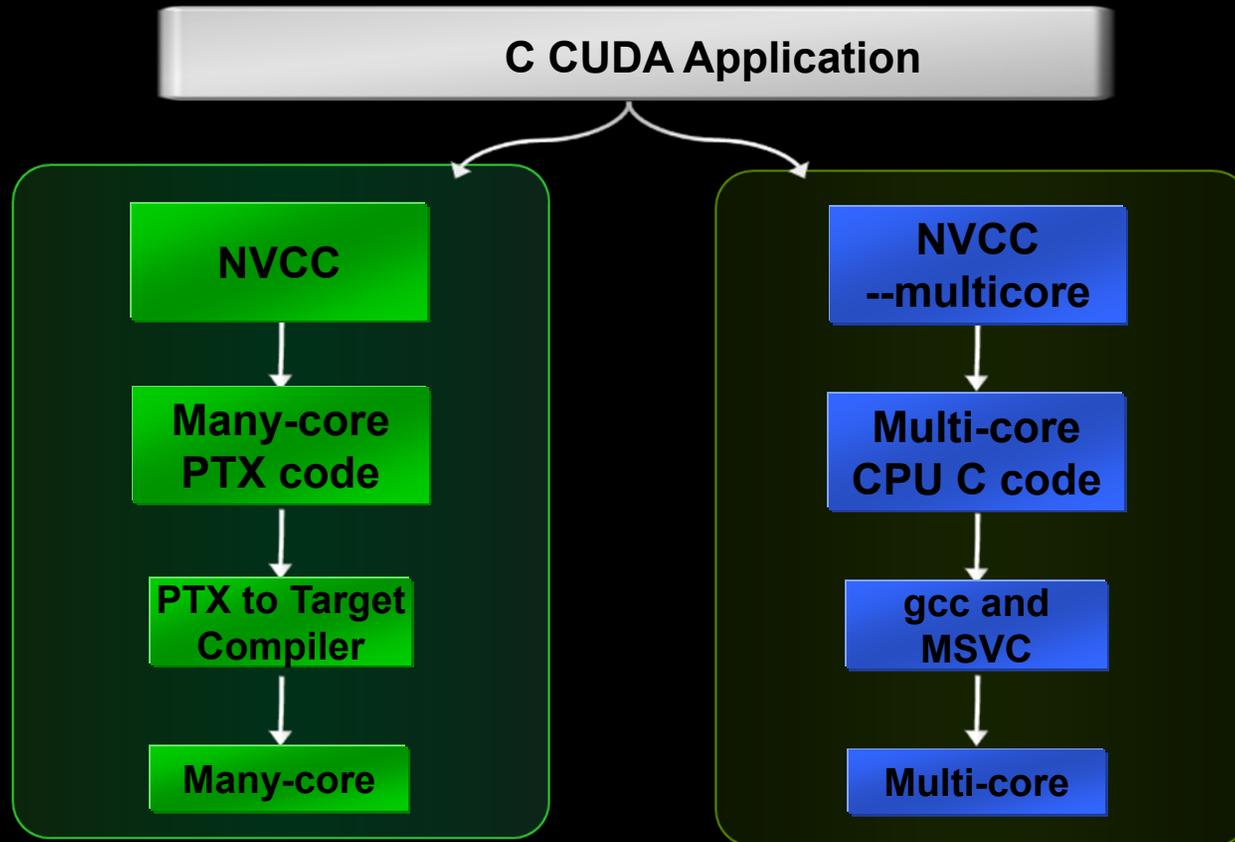
Standard

Debugger Profiler



Multi-core

CUDA 2.1: Many-core + Multi-core support





nVISION 08
THE WORLD OF VISUAL COMPUTING

CUDA Everywhere!