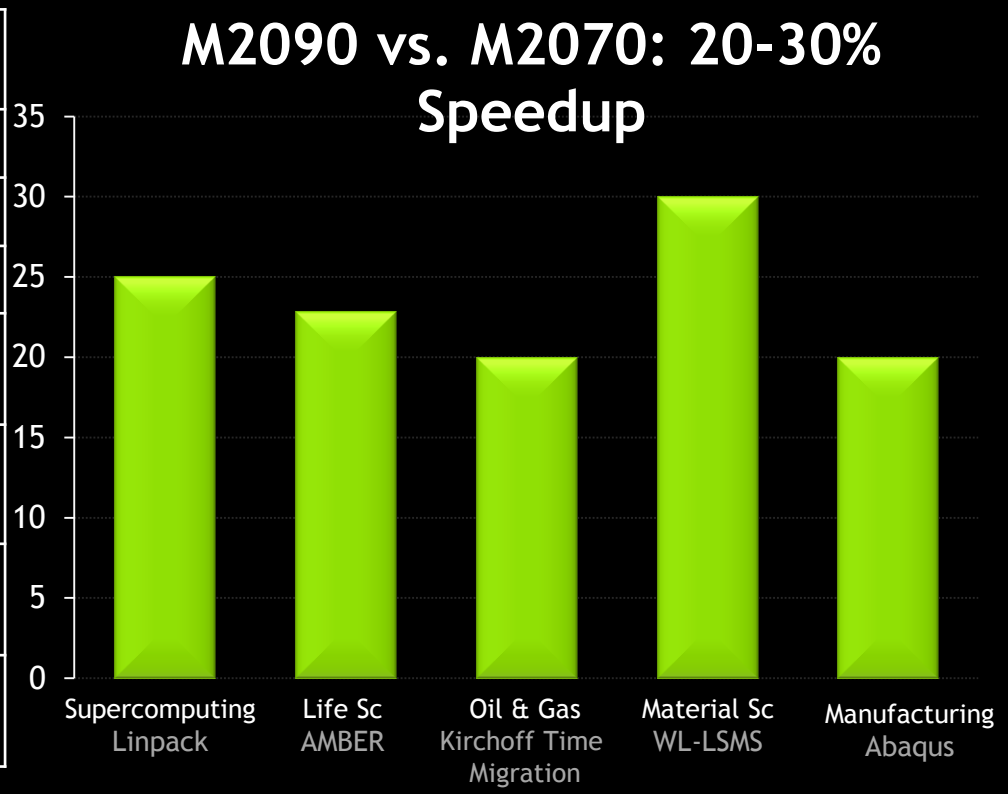
NVIDIA GPU コンピューティング最新情報
～Keplerアーキテクチャを採用した次世代Tesla製品のご紹介～

エヌビディア ジャパン
Tesla Quadro事業部
セールスマネージャー　吉田圭二

Tesla 現行製品 M/C20xx (Fermi)

Name		M2090	M/C2075	M2070
GPU Arch		Fermi	Fermi	Fermi
# of cores		512	448	448
Memory size		6 GB	6 GB	6 GB
Memory bandwidth (ECC off)		177.6 GB/s	150 GB/s	150 GB/s
Power Management		Yes	Yes	No
Peak Performance In GFlops	SP	1331	1030	1030
	DP	665	515	515



高い倍精度演算性能と高い信頼性を実現

GPU Technology Conference 2012

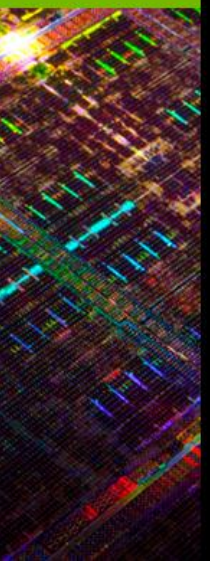
世界最大の GPU コンピューティングイベント！

- 2012年5月14～17日にサンノゼで開催
- 世界50ヵ国から3,000人が参加
- 300以上のセッションで、最先端の GPU コンピューティングの取組みを紹介
- チュートリアルセッション

ウェブサイト

http://www.gputechconf.com

Kepler アーキテクチャの Tesla 製品の発表

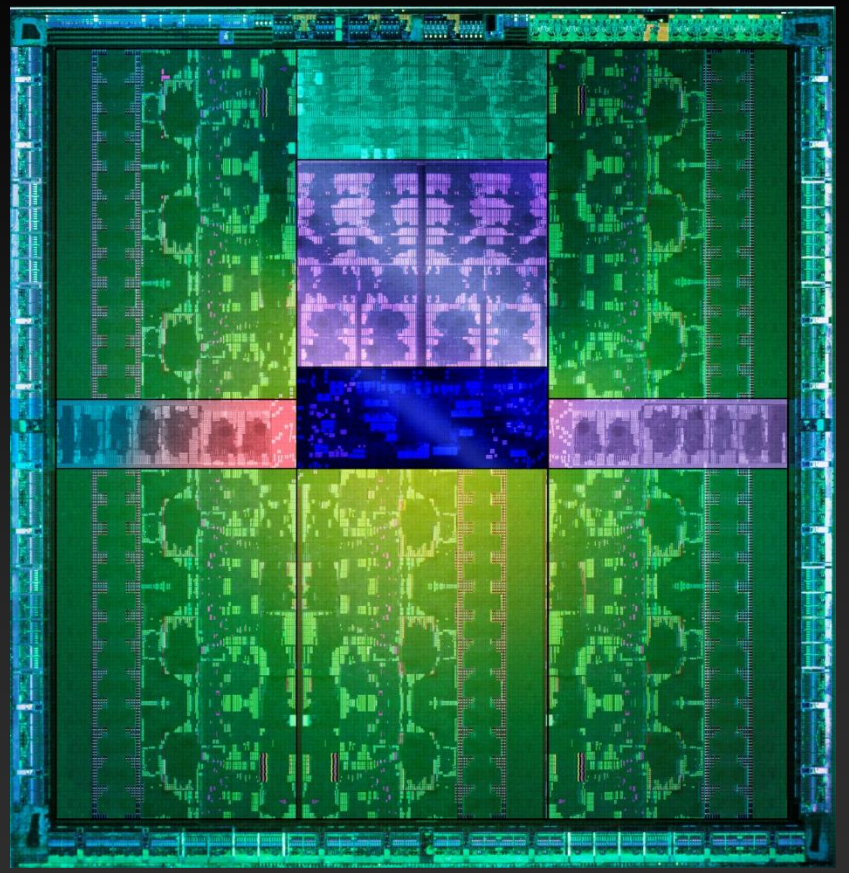


GTC Japan 2012

- ❖主催: エヌビディア ジャパン
- ❖共催: 東京工業大学 GPU コンピューティング研究会
- ❖日時: 2012年7月26日（木）午前10時から午後8時
- ❖会場: 東京ミッドタウンホール＆カンファレンス（六本木）
- ❖参加費: 無料
- ❖予定参加者: 1,500名
- ❖予定協賛会社：40社
- ❖イベントサイト：http://www.gputechconf.jp

Kepler

Fastest, Most Efficient HPC Architecture Ever



SMX

Hyper-Q

Dynamic Parallelism

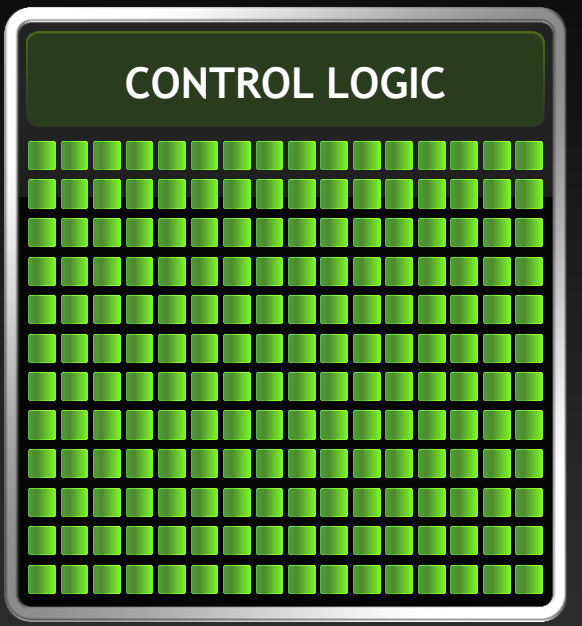
Kepler: Fast & Efficient

SM
Fermi

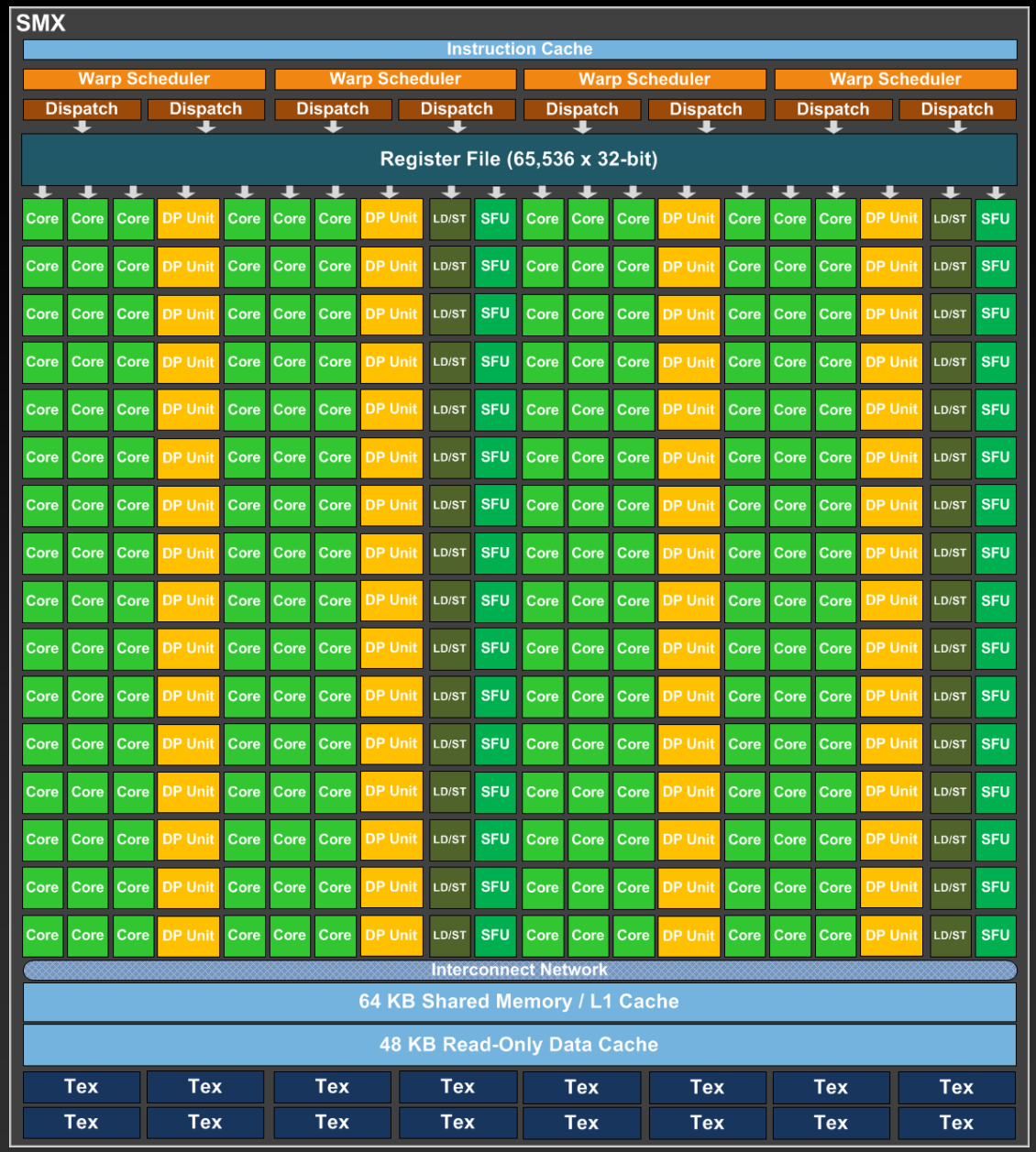
32 cores

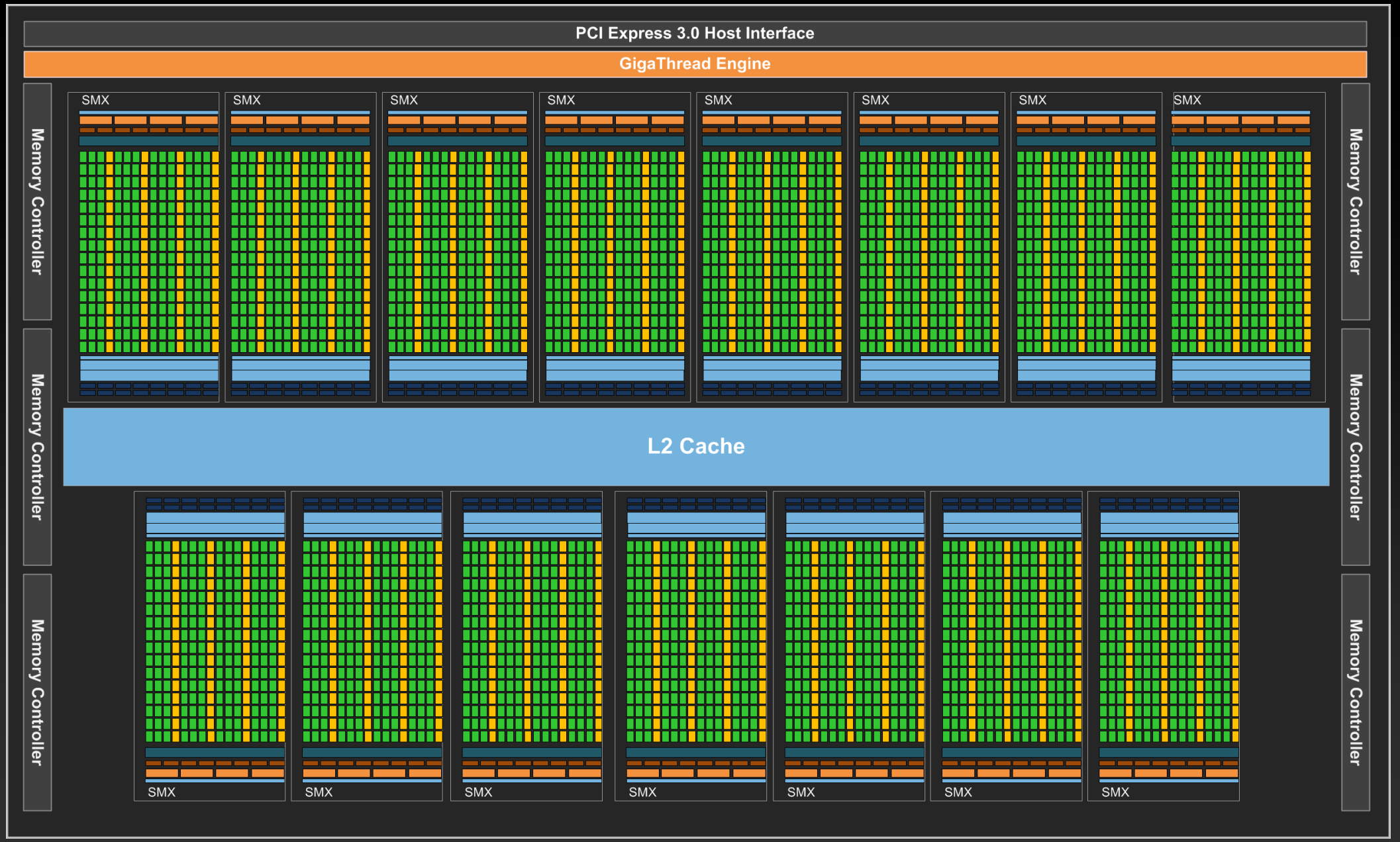
3x
Perf / Watt

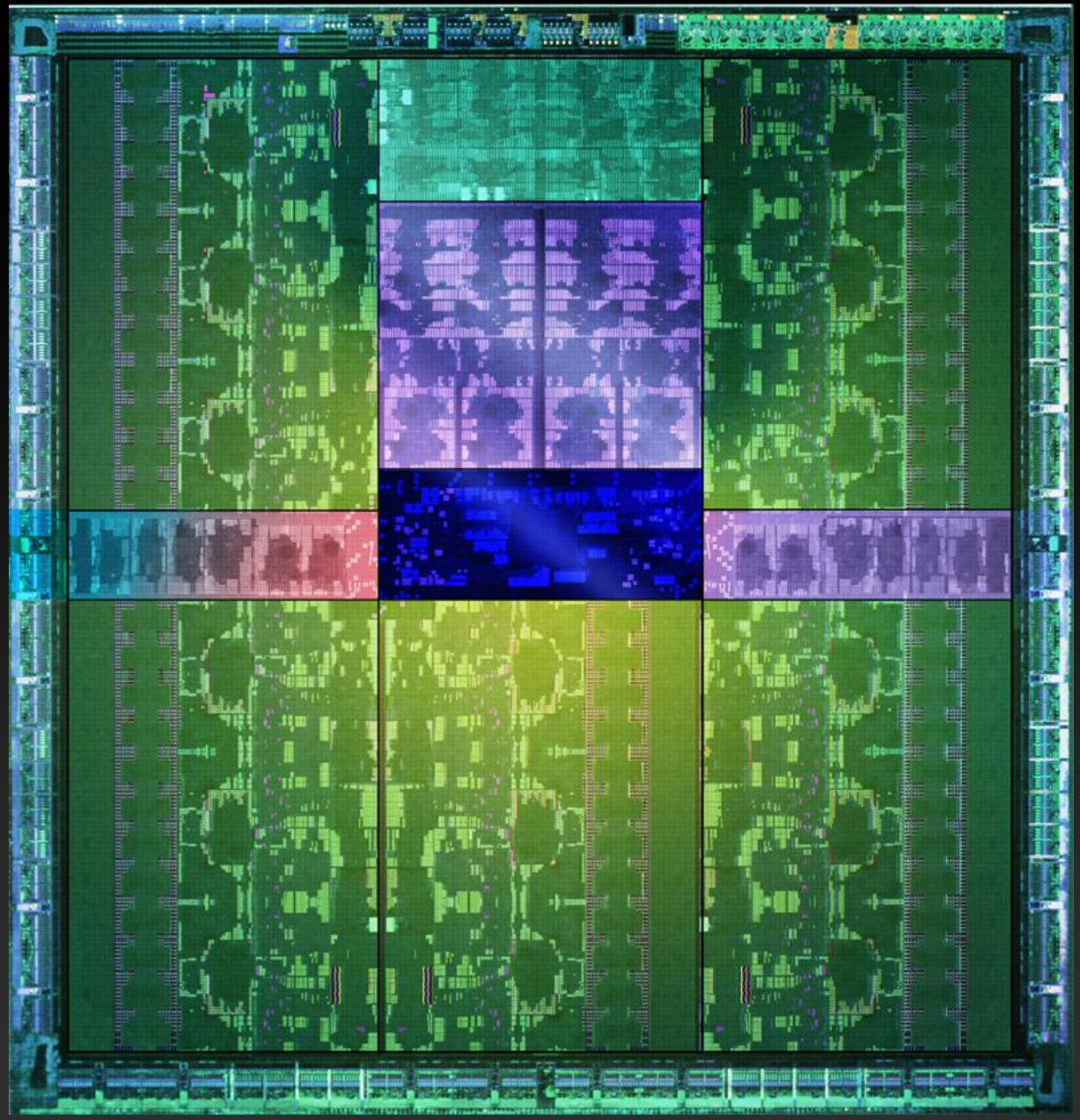
SMX
Kepler



192 cores


SMX
Instruction Cache
Warp Scheduler
Warp Scheduler
Warp Scheduler
Warp Scheduler
Dispatch
Register File (65,536 x 32-bit)
Core
DP Unit
LD/ST
SFU
Interconnect Network
64 KB Shared Memory / L1 Cache
48 KB Read-Only Data Cache
Tex


PCI Express 3.0 Host Interface
GigaThread Engine
Memory Controller
Memory Controller
Memory Controller
Memory Controller
Memory Controller
Memory Controller
SMX
SMX
SMX
SMX
SMX
SMX
SMX
SMX
L2 Cache
SMX
SMX
SMX
SMX
SMX
SMX
SMX



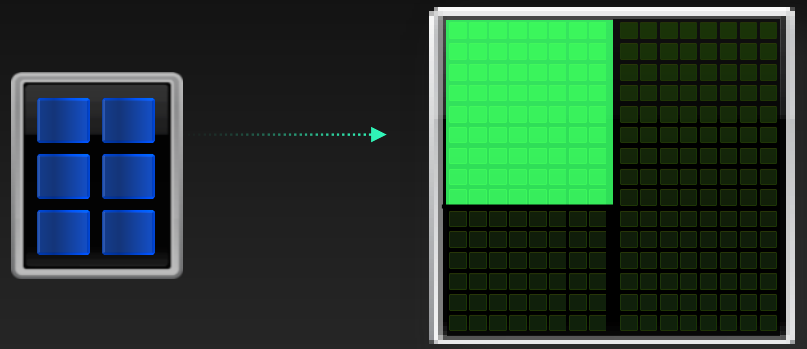

1 Petaflop
Just 10 Racks
400 kWatt

Hyper-Q

CPU Cores Simultaneously Run Tasks on Kepler

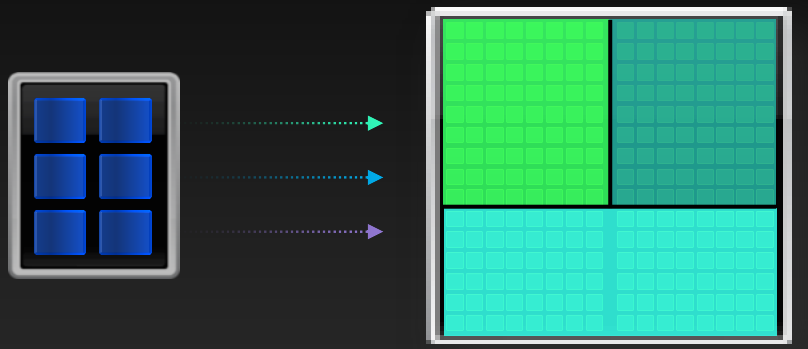
FERMI

1 MPI Task at a Time



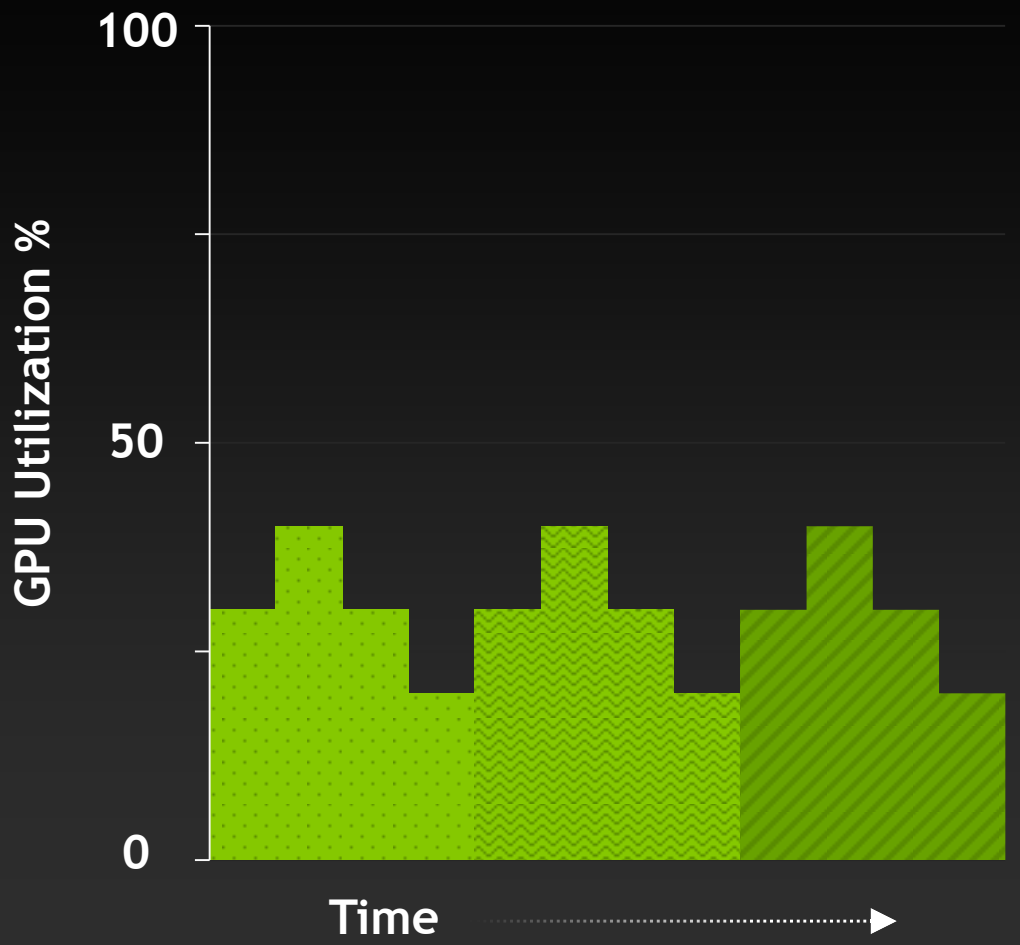
KEPLER

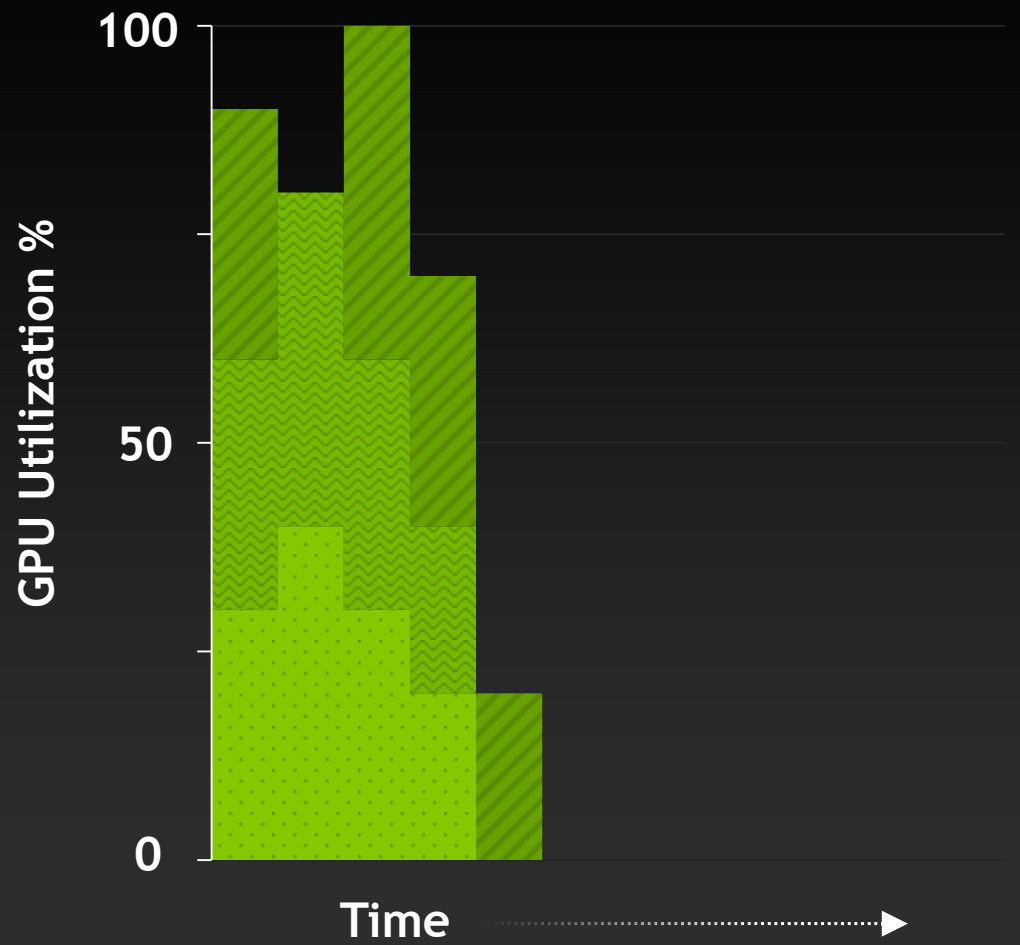
32 Simultaneous MPI Tasks



Hyper-Q

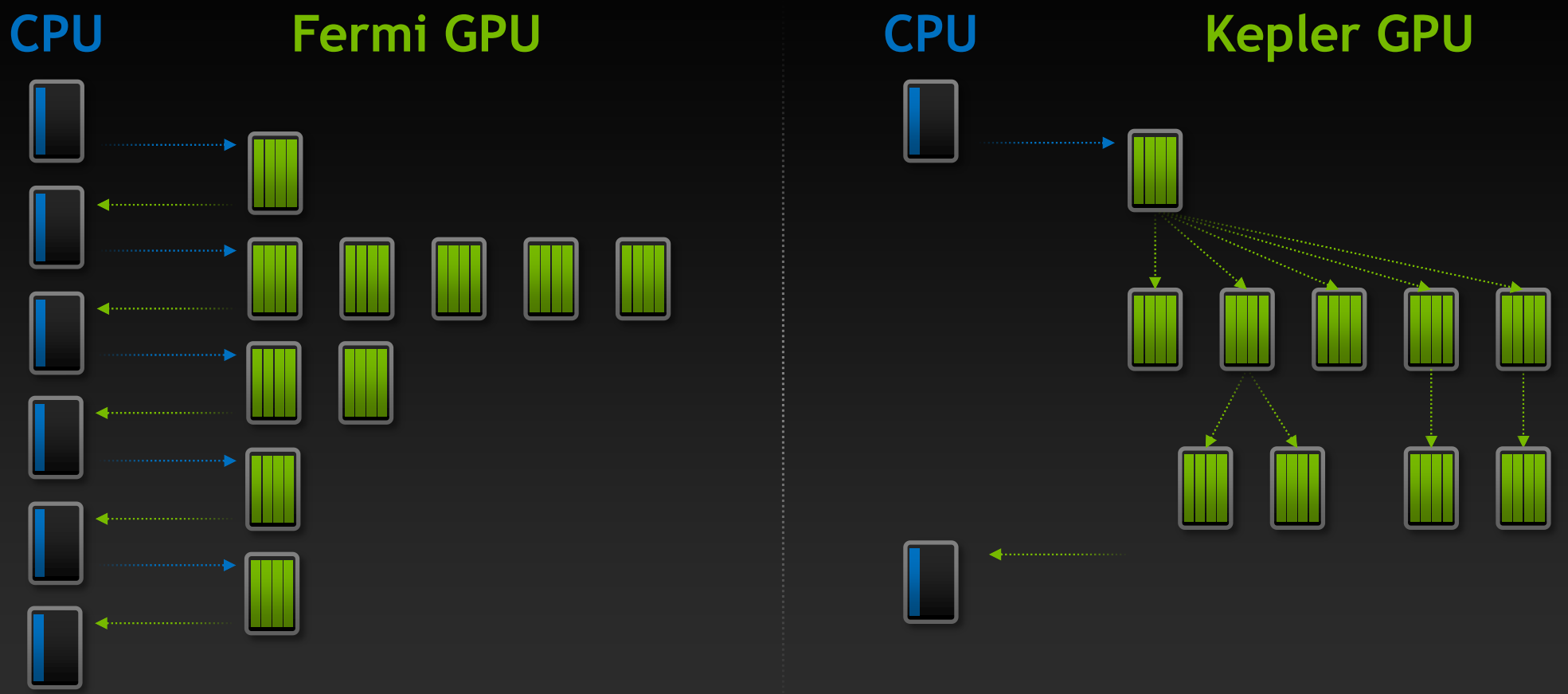
Max GPU Utilization, Slashes CPU Idle Time





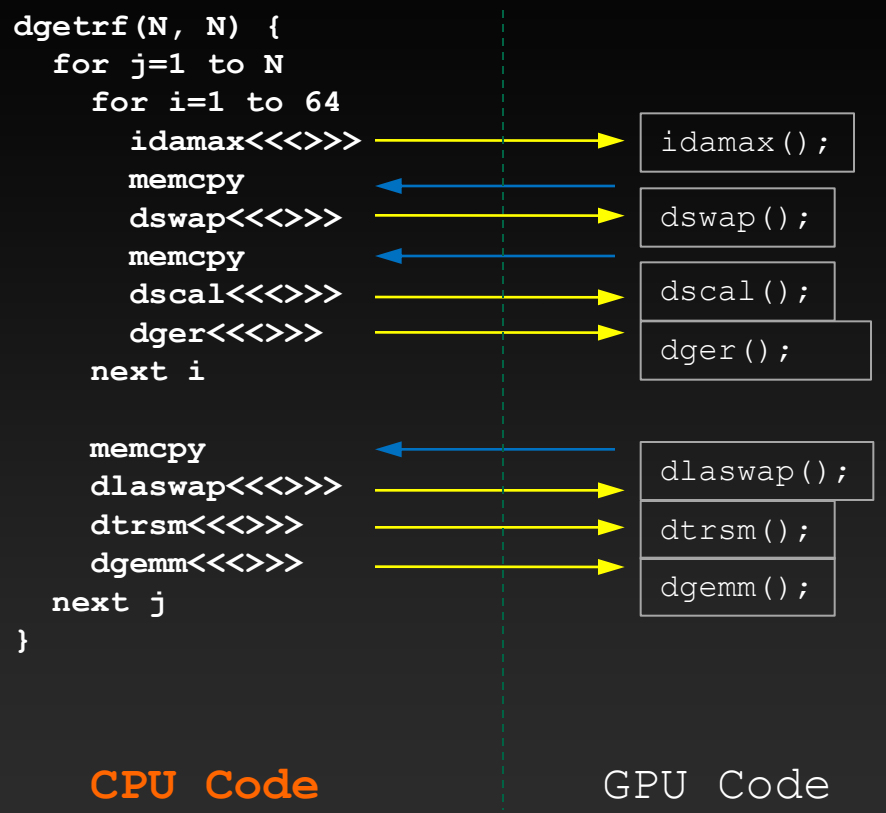
Dynamic Parallelism

GPU Adapts to Data, Dynamically Launches New Threads

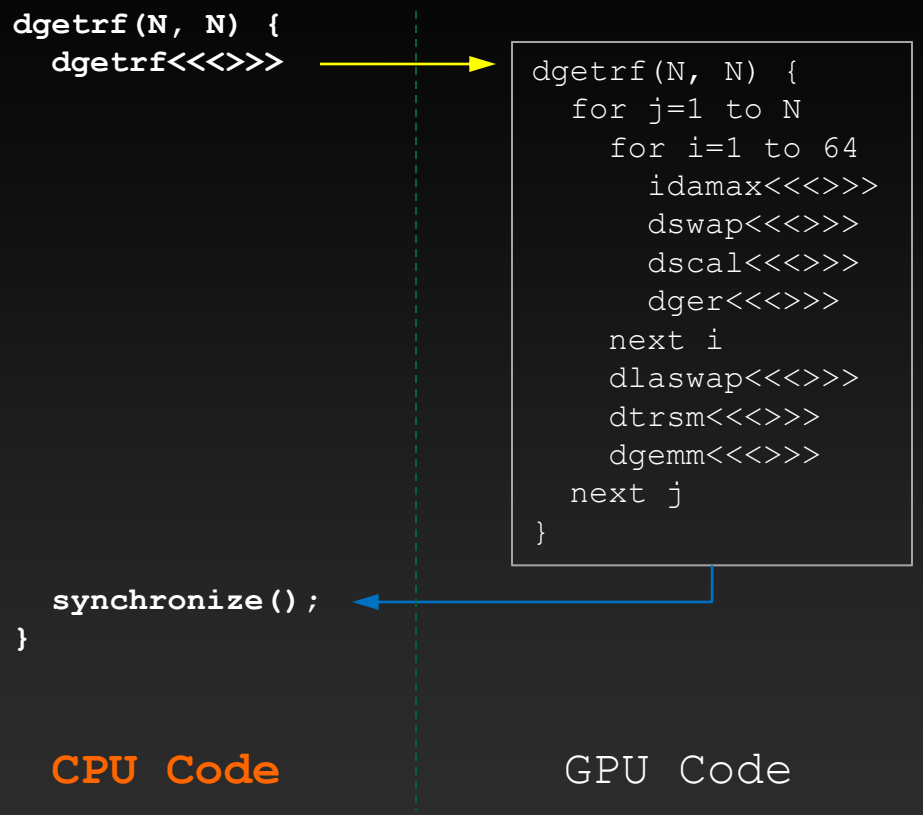


Simpler Code: LU Example

LU decomposition (Fermi)



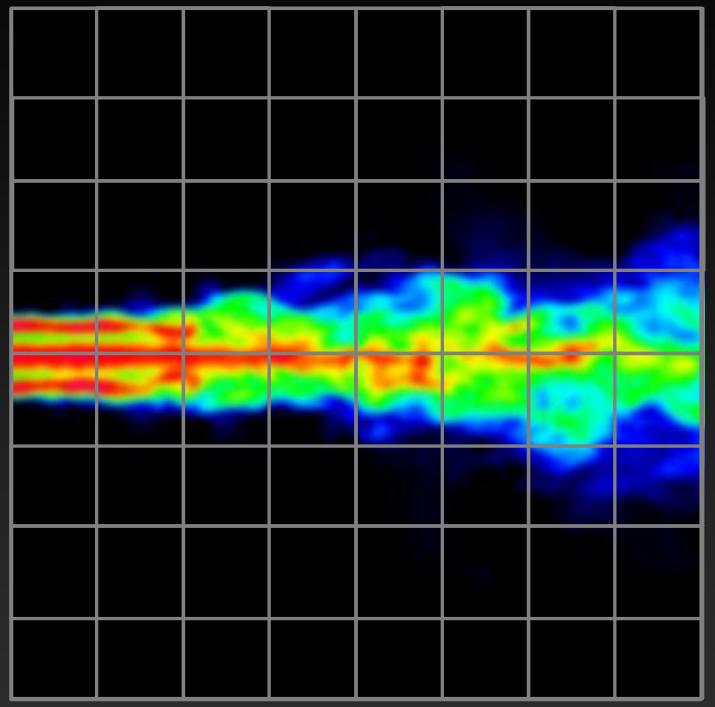
LU decomposition (Kepler)



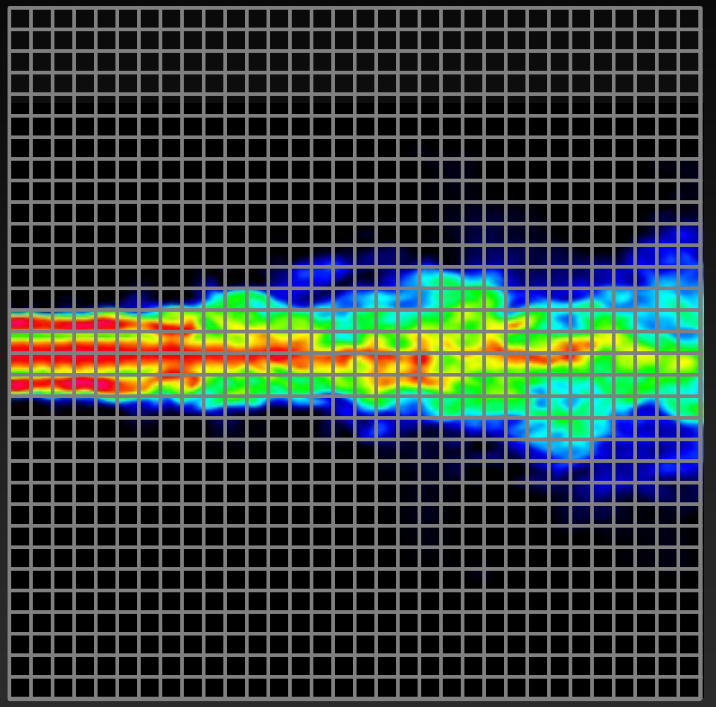
Dynamic Parallelism

Makes GPU Computing Easier & Broadens Reach

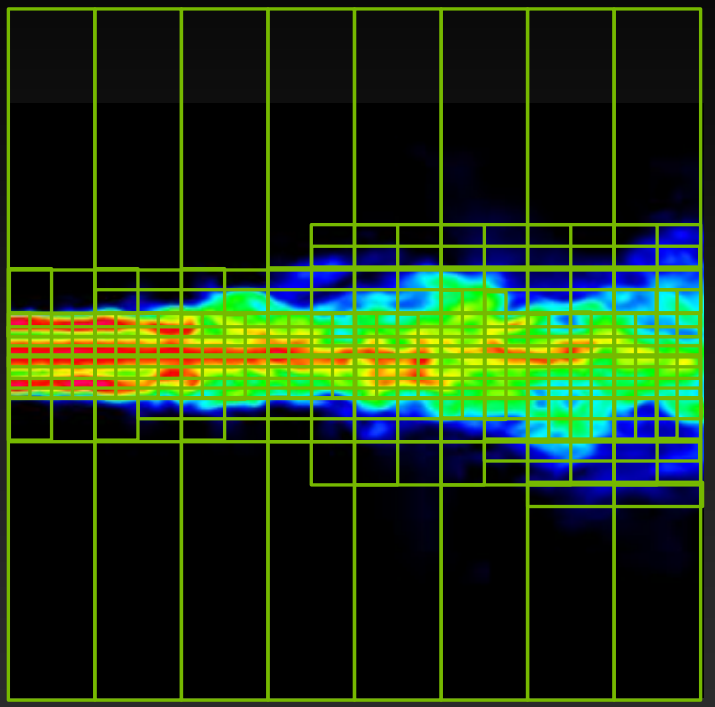
Too coarse



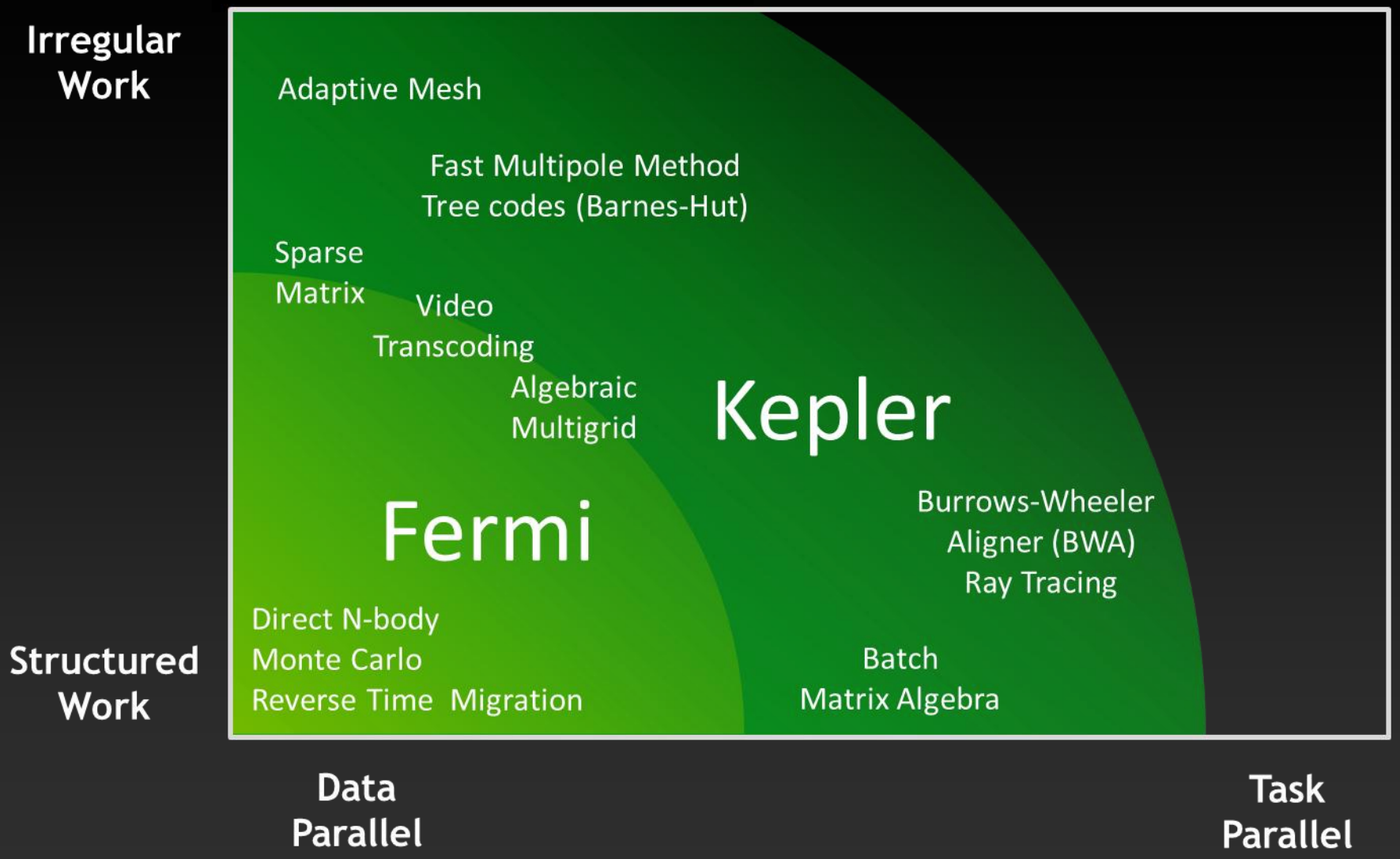
Too fine



Just right



Kepler Addresses Broader Set of Applications



Tesla K10

3x Single Precision

1.8x Memory Bandwidth

Image, Signal, Seismic

Available Now

Tesla K20

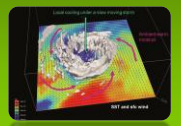
3x Double Precision

Hyper-Q, Dynamic Parallelism

CFD, FEA, Finance, Physics

Available Q4 2012

Supercomputing



Weather / Climate Modeling
Molecular Dynamics
Computational Physics

Life Sciences

Biochemistry
Bioinformatics
Material Science

Manufacturing

Structural Mechanics
Comp Fluid Dynamics (CFD)
Electromagnetics

Defense / Govt

Signal Processing
Image Processing
Video Analytics

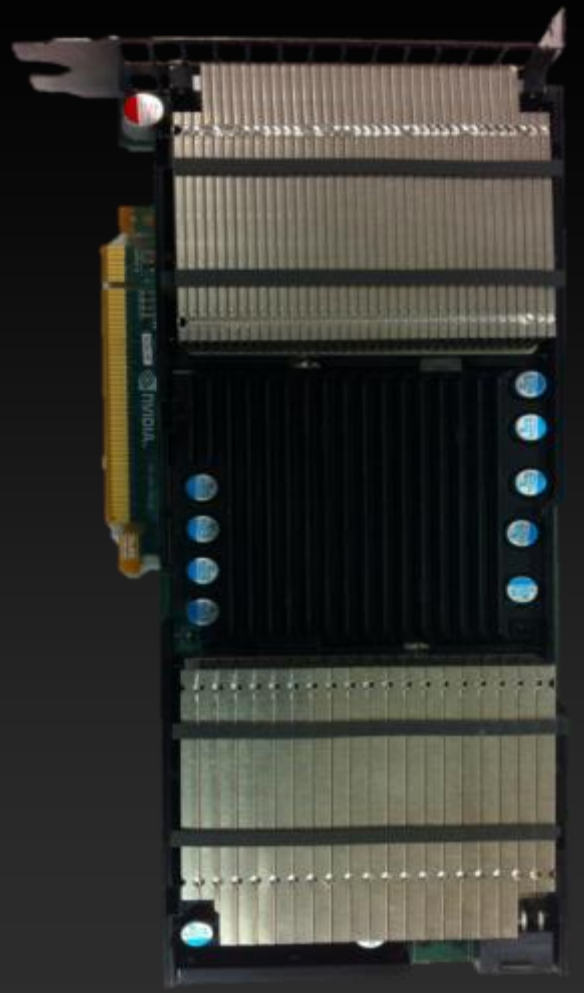
Oil and Gas

Reverse Time Migration
Kirchoff Time Migration

Q2	Q3	Q4
Tesla M2090	Fermi	Tesla K20
Tesla M2075		
Tesla K10		

Tesla K10 specification

Product Name	M2090	K10	
GPU Architecture	Fermi	Kepler GK104	
# of GPUs	1	2	
		Board	Per GPU
Single Precision Flops	1.3 TF	4.58 TF	2.29 TF
Double Precision Flops	0.66 TF	0.190 TF	0.095 TF
# CUDA Cores	512	3072	1536
Memory size	6 GB	8 GB	4GB
Memory BW (ECC off)	177.6 GB/s	320 GB/s	160GB/s
PCI-Express	Gen 2: 8 GB/s	Gen 3: 16 GB/s	



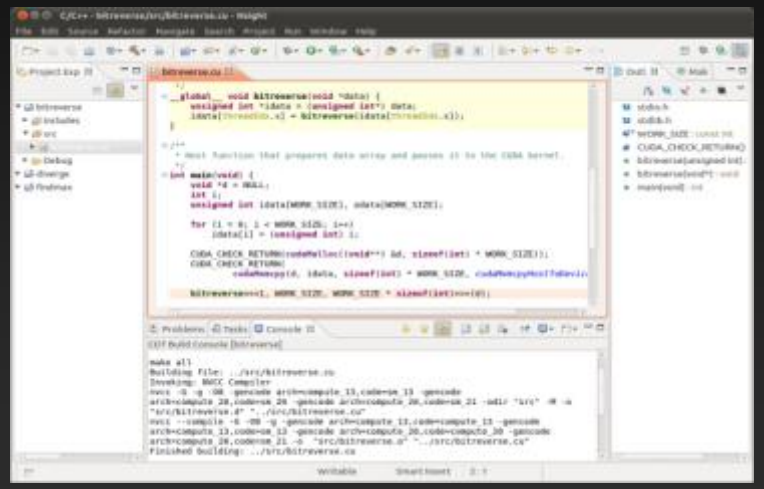
CUDA 5

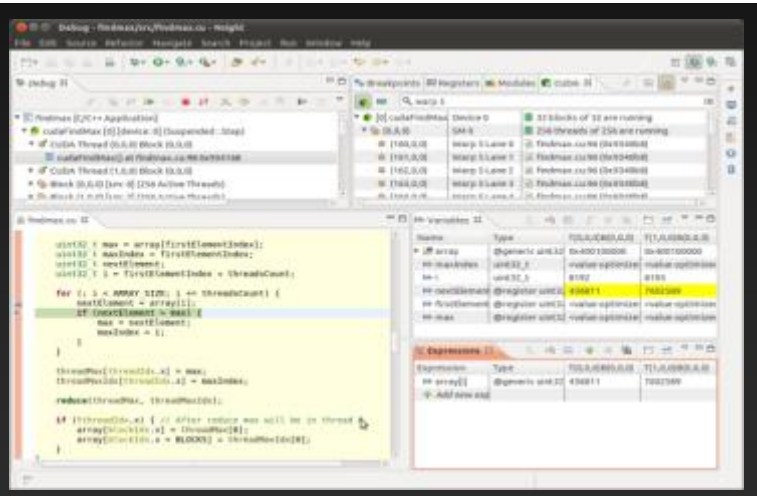
Nsight™ for Linux & Mac

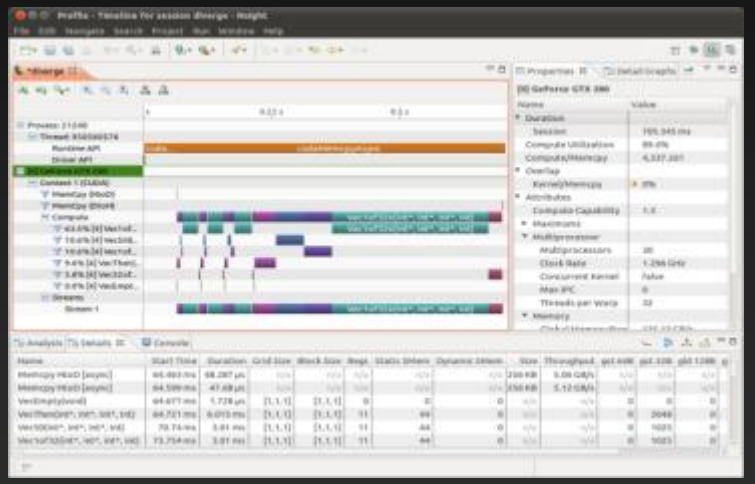
NVIDIA GPUDirect™

Library Object Linking

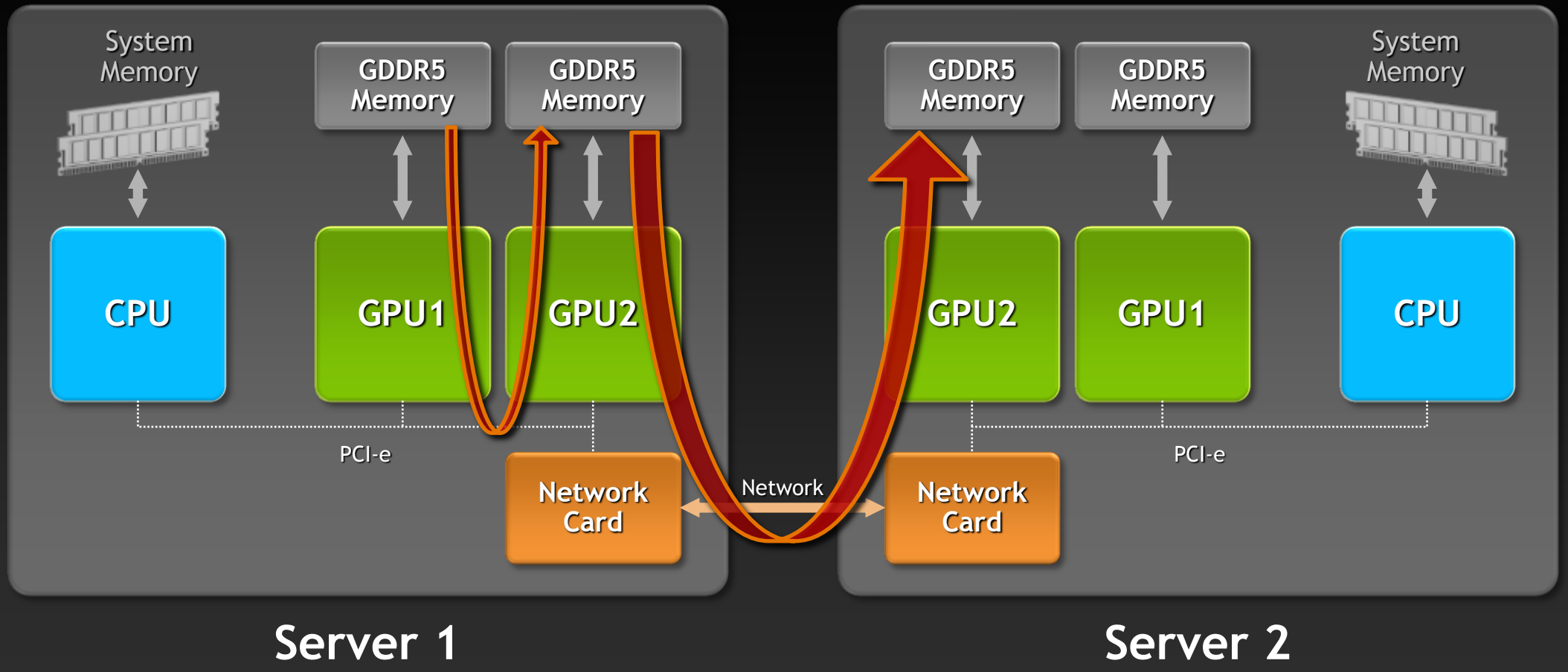
NVIDIA Nsight™ for Linux & Mac (and Windows of course)



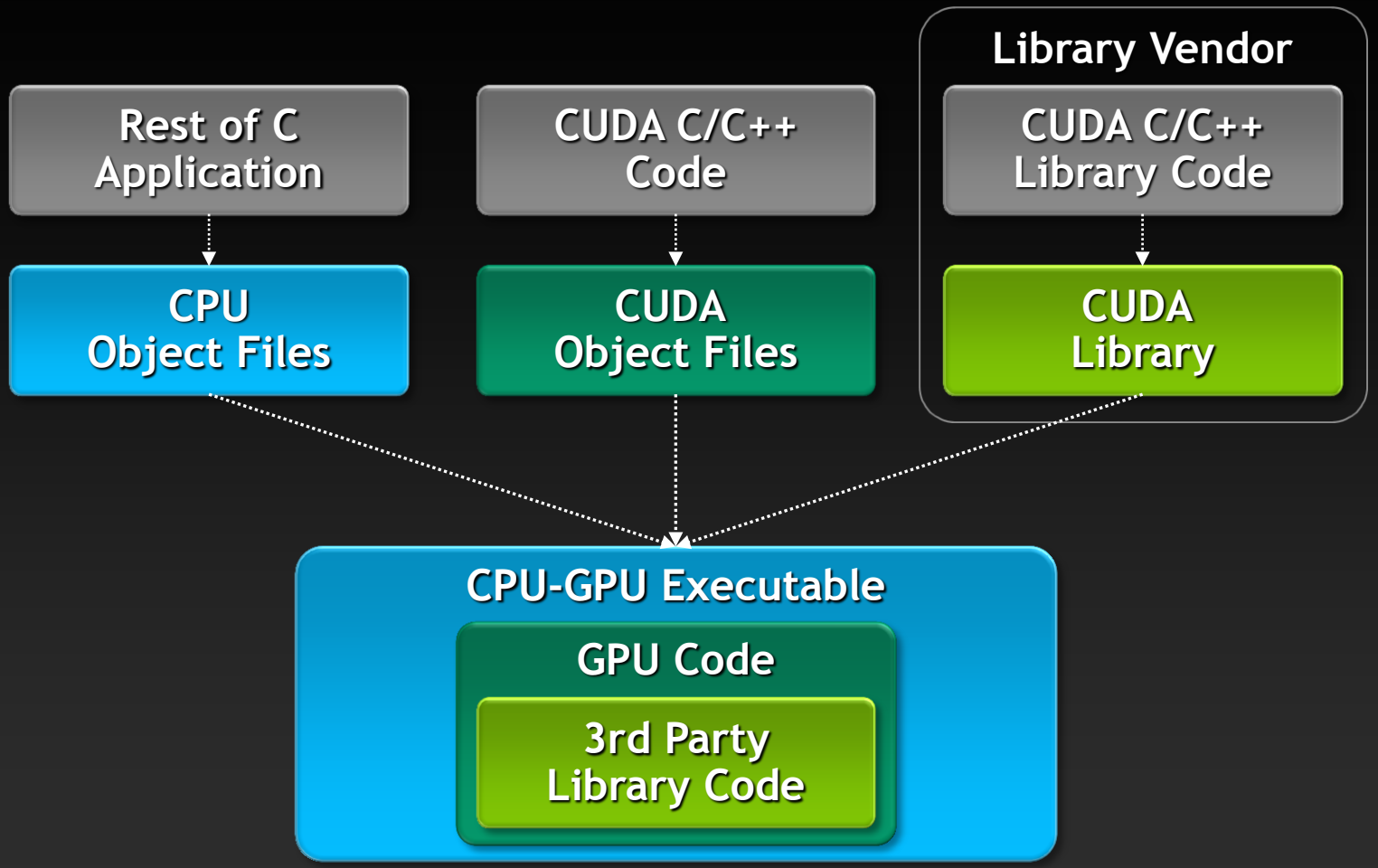




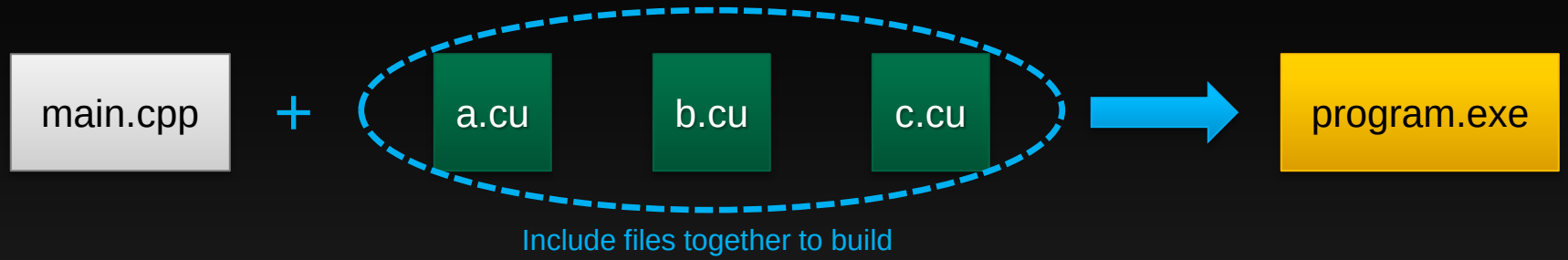
Kepler Enables Full NVIDIA GPUDirect™



3rd Party GPU Library Object Linking



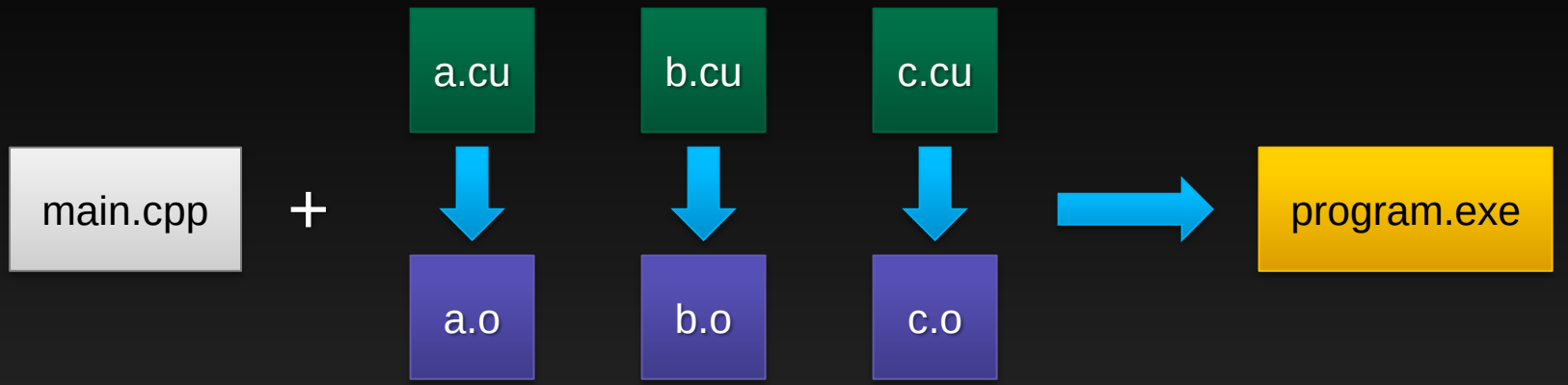
CUDA 4: Whole-Program Compilation & Linking



CUDA 4 required single source file for a single kernel

No linking external device code

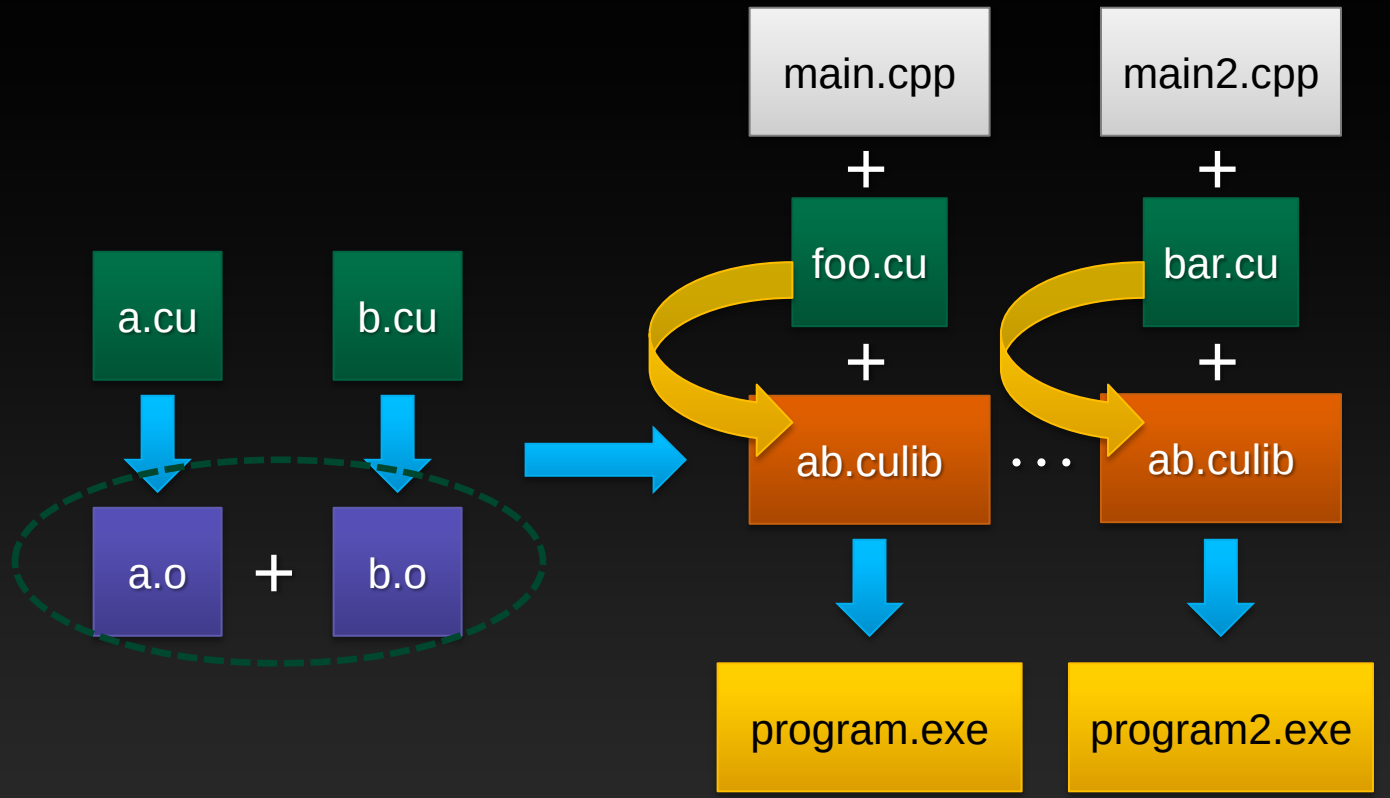
CUDA 5: Separate Compilation & Linking



Separate compilation allows building independent object files

CUDA 5 can link multiple object files into one program

CUDA 5: Separate Compilation & Linking



Can also combine object files into static libraries

Link and externally call *device* code

Facilitates code reuse, reduces compile time


Thank you