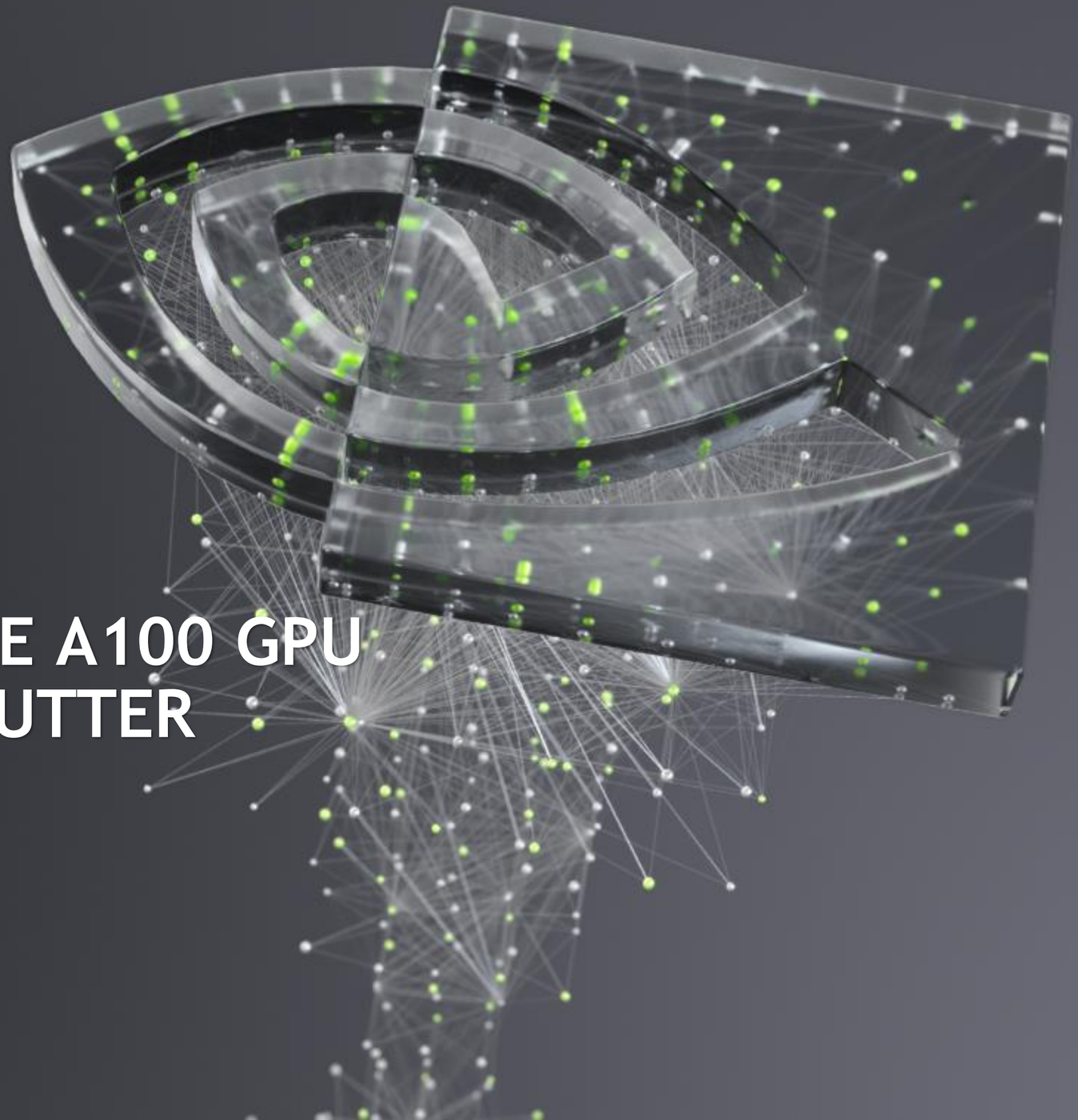
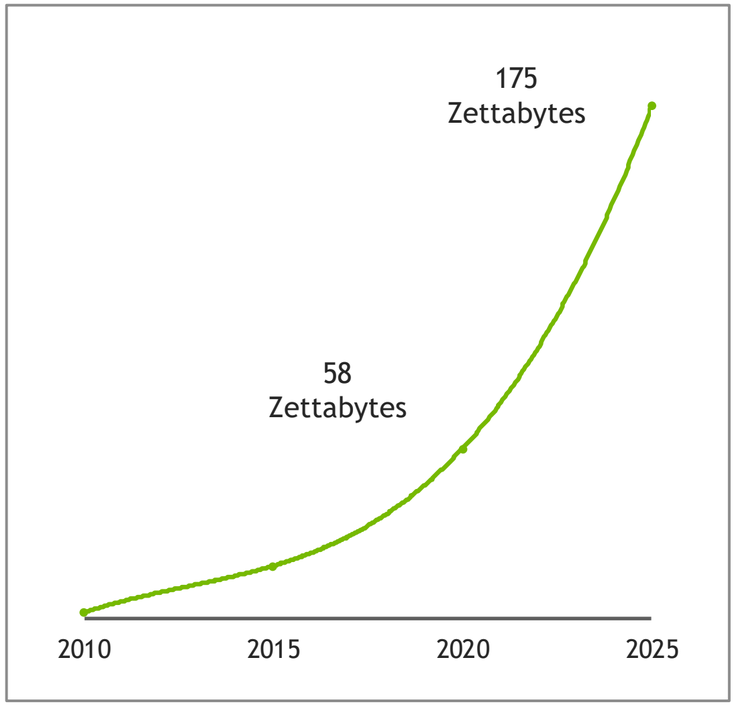


INSIDE THE NVIDIA AMPERE A100 GPU IN THETAGPU AND PERLMUTTER

JULY 28, 2021

EXPLODING DATA AND MODEL SIZE

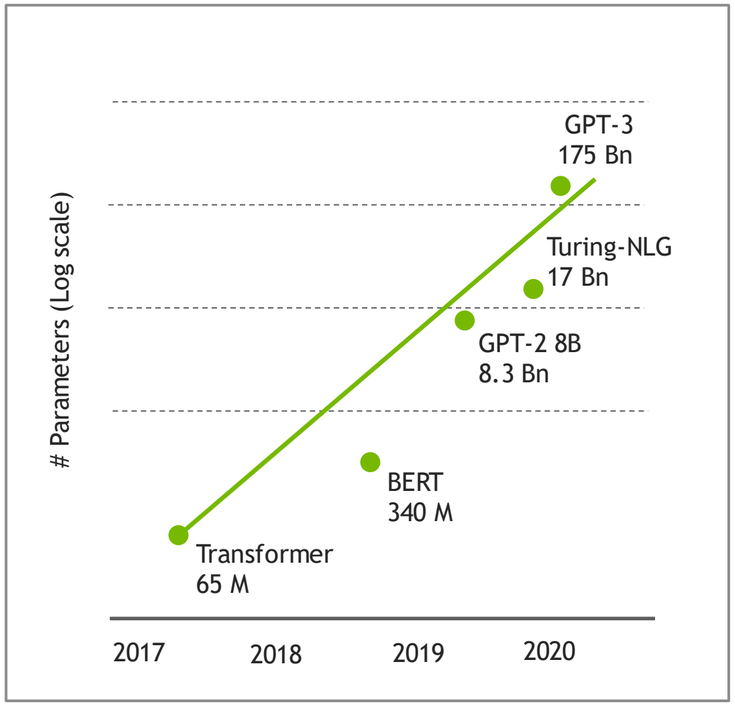


Big Data Growth

90% of the World's Data in last 2 Years

Growth In Scientific Data

Fueled by Accurate Sensors & Simulations



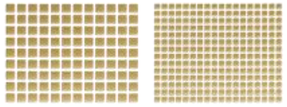
Exploding Model Size

Driving Accuracy

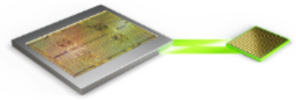
Source for Big Data Growth chart: IDC – The Digitization of the World (May, 2020)

NVIDIA A100 40GB

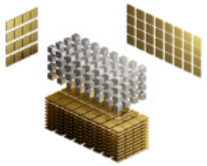
The World's Highest Performing AI Supercomputing GPU



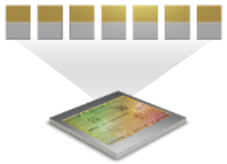
40GB HBM2e
For large datasets
and models



1.5 TB/s +
World's highest memory bandwidth
to feed the world's fastest GPU

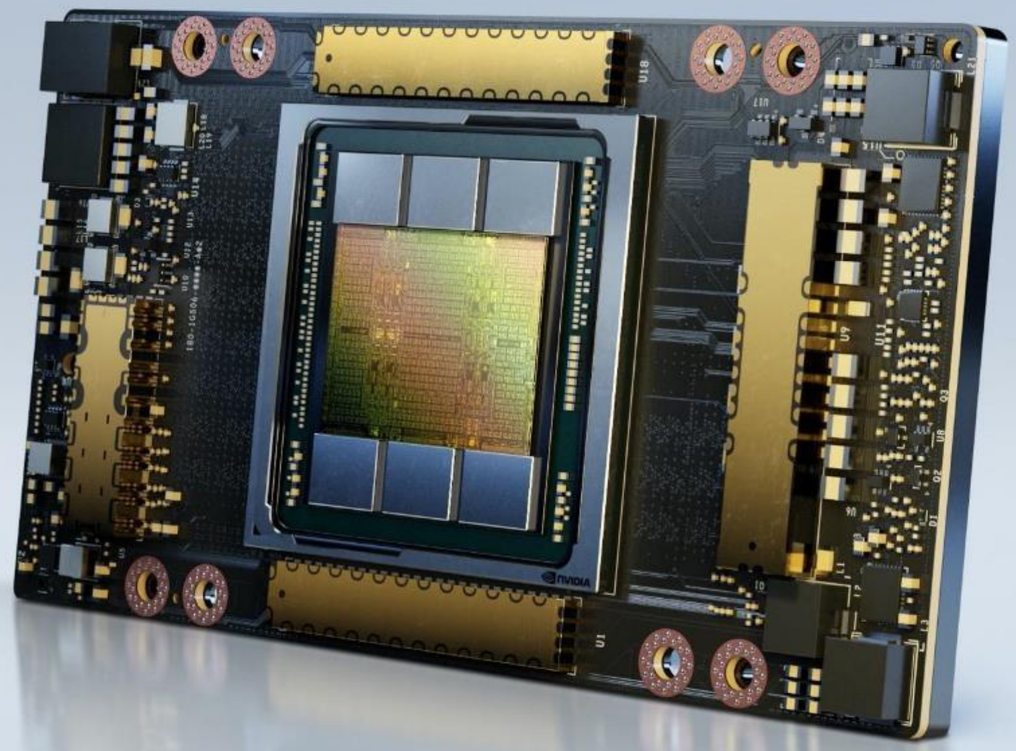


3rd Gen Tensor Core



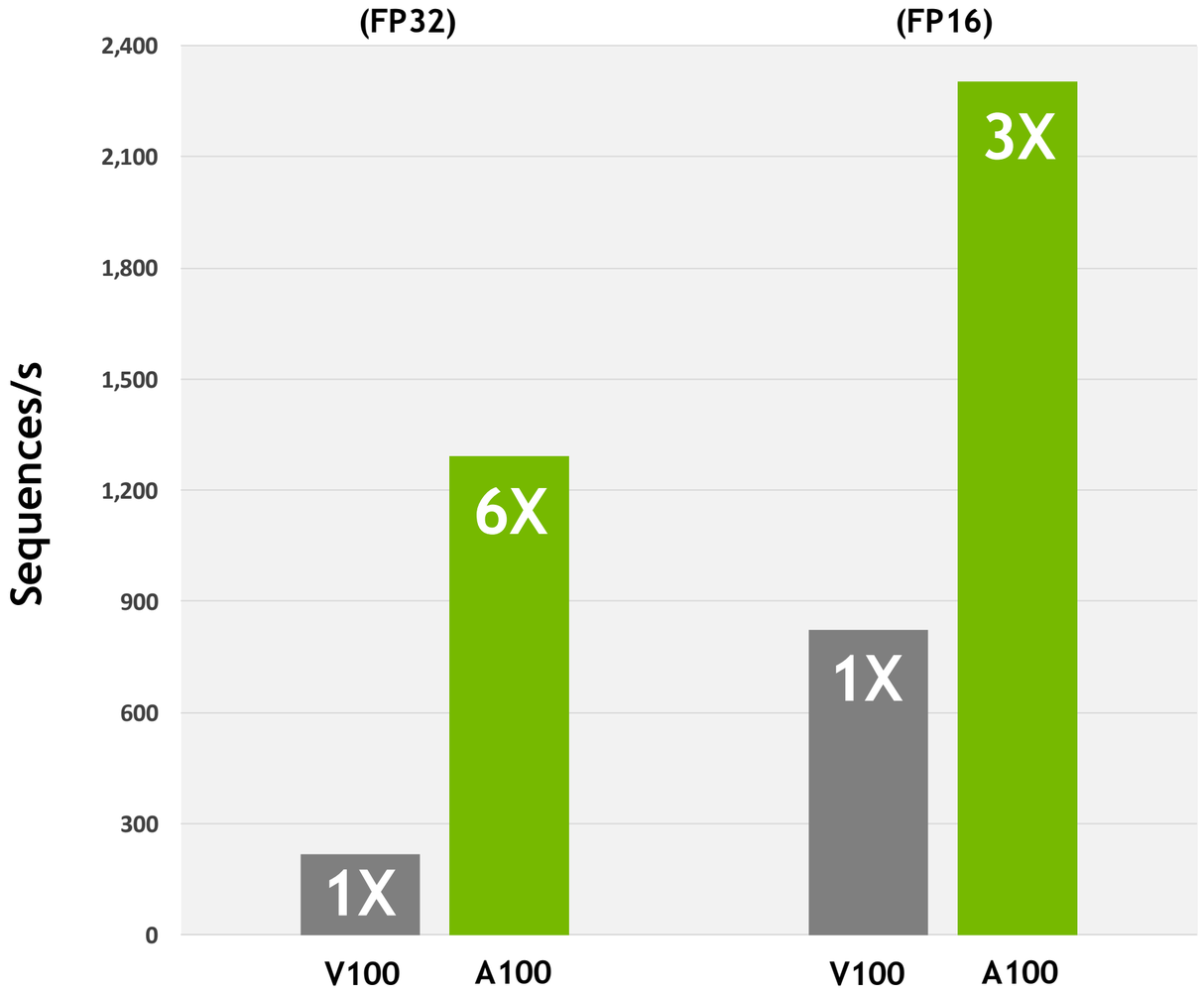
Multi-Instance GPU

3rd Gen NVLink

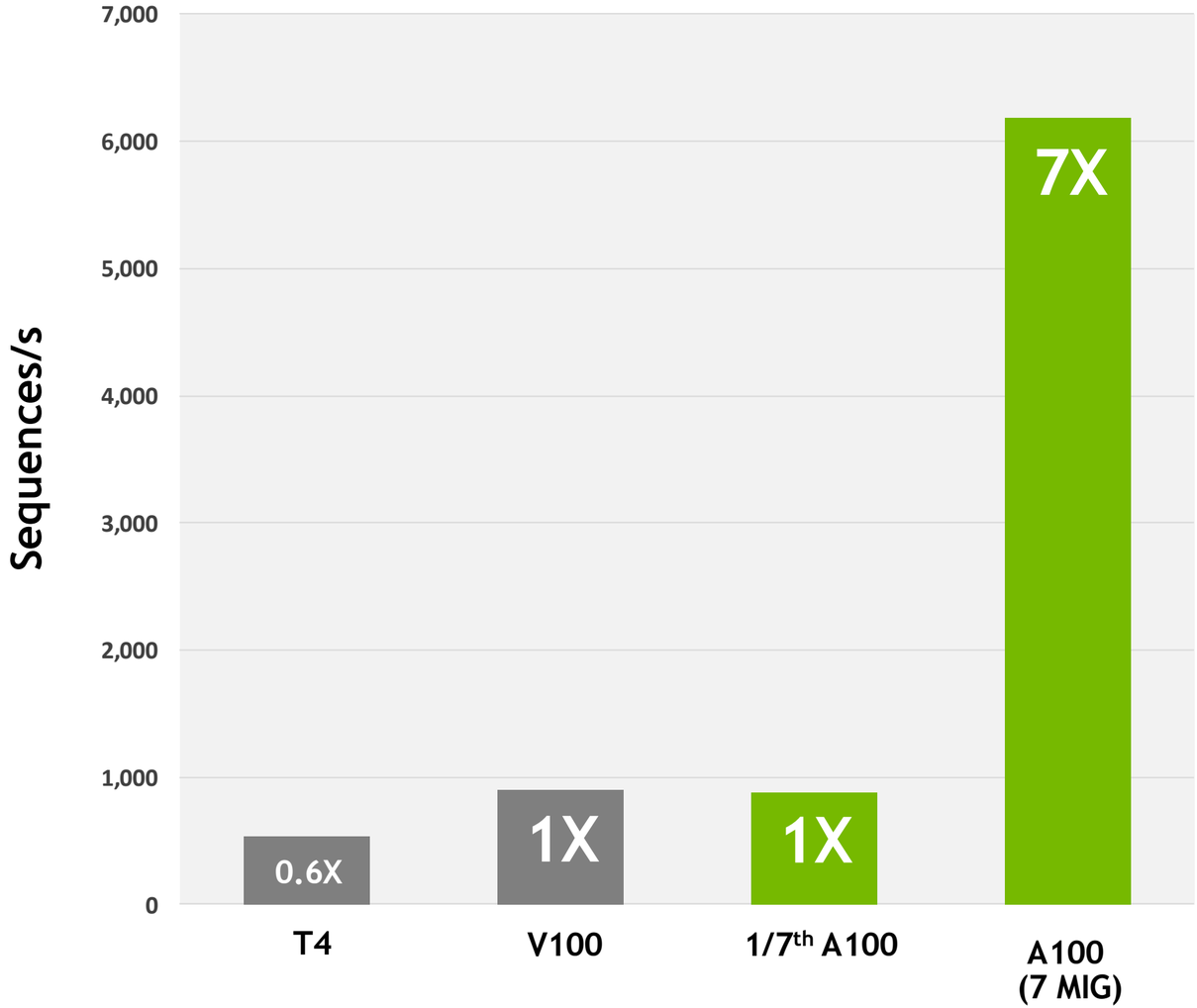


UNIFIED AI ACCELERATION

BERT-LARGE TRAINING

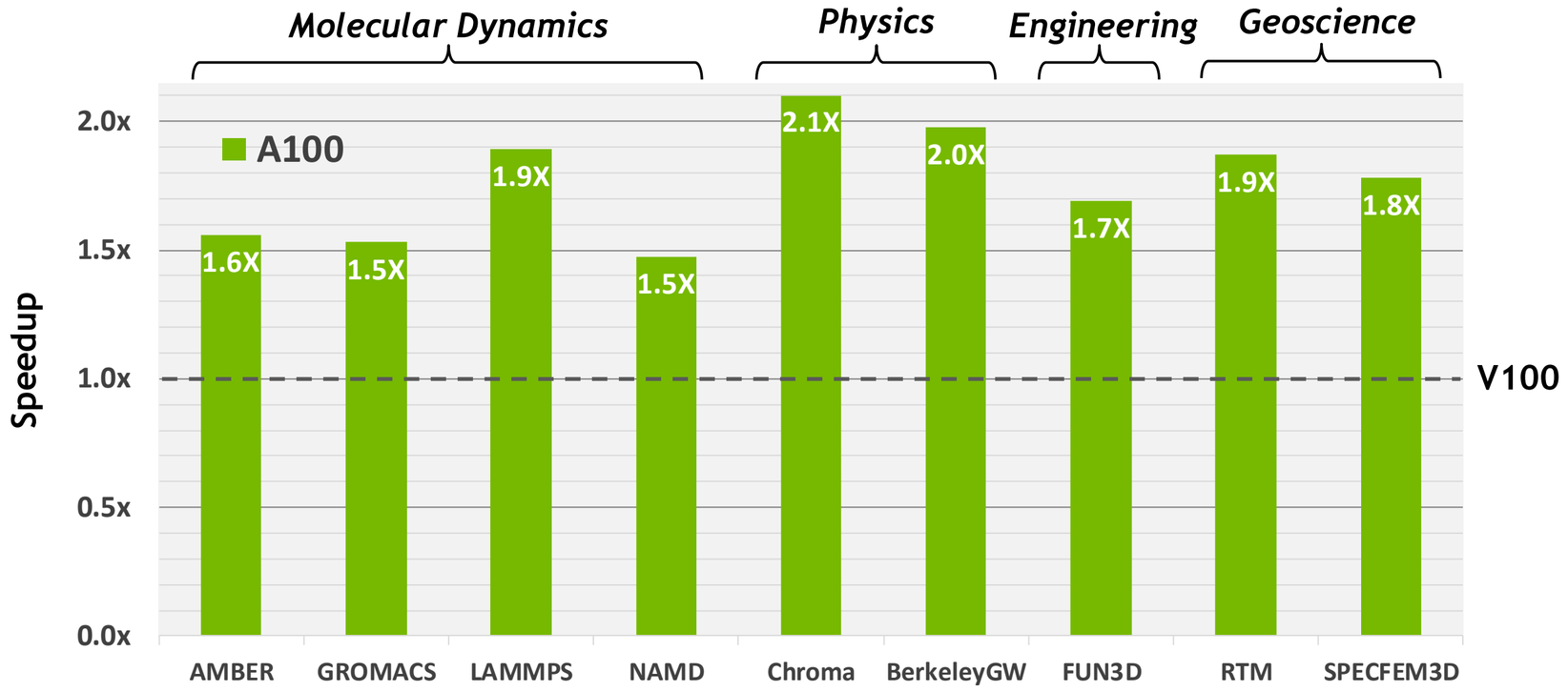


BERT-LARGE INFERENCE



All results are measured
BERT Large Training (FP32 & FP16) measures Pre-Training phase, uses PyTorch including (2/3) Phase1 with Seq Len 128 and (1/3) Phase 2 with Seq Len 512, V100 is DGX1 Server with 8xV100, A100 is DGX A100 Server with 8xA100, A100 uses TF32 Tensor Core for FP32 training
BERT Large Inference uses TRT 7.1 for T4/V100, with INT8/FP16 at batch size 256. Pre-production TRT for A100, uses batch size 94 and INT8 with sparsity

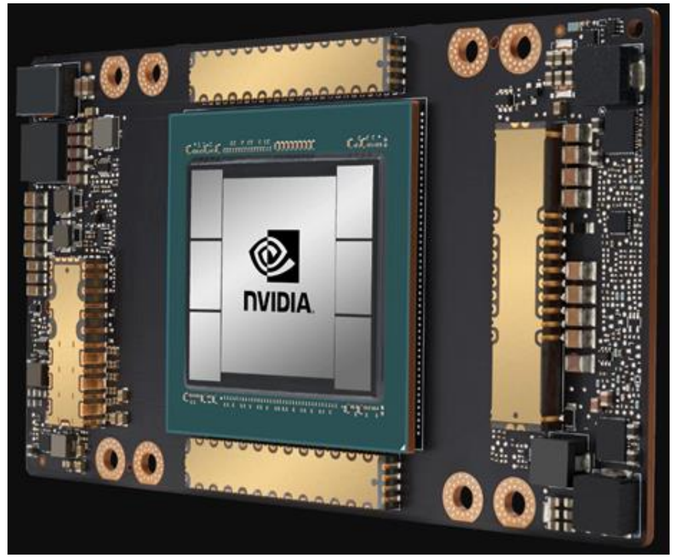
ACCELERATING HPC WITH UP TO 2X PERF OVER V100



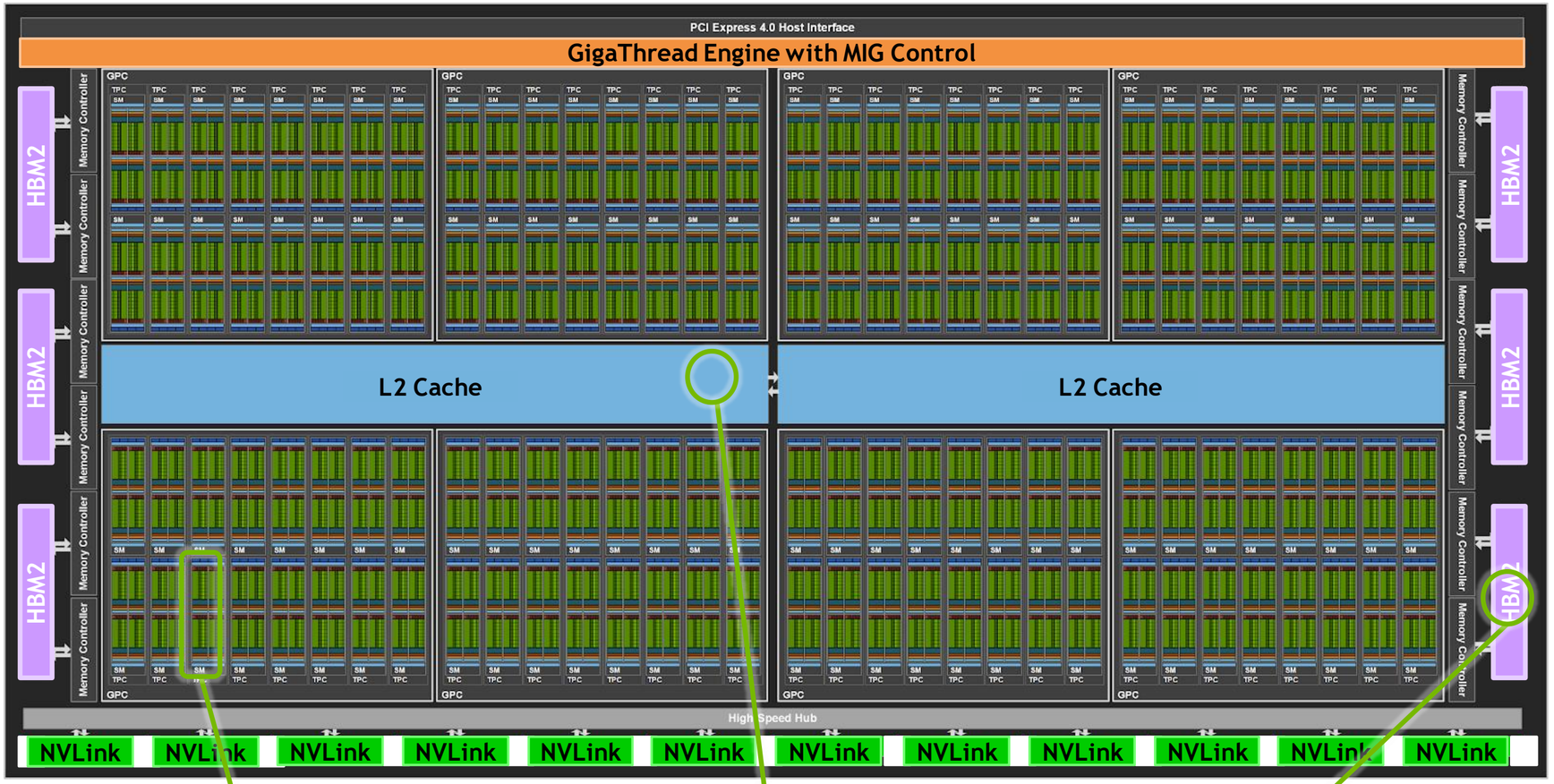
All results are measured
Except BerkeleyGW, V100 used is single V100 SXM2. A100 used is single A100 SXM4
More apps detail: AMBER based on PME-Cellulose, GROMACS with STMV (h-bond),
LAMMPS with Atomic Fluid LJ-2.5, NAMD with v3.0a1 STMV_NVE
Chroma with szscl21_24_128, FUN3D with dpw, RTM with Isotropic Radius 4 1024^3,
SPECFEM3D with Cartesian four material model
BerkeleyGW based on Chi Sum and uses 8xV100 in DGX-1, vs 8xA100 in DGX A100

NVIDIA A100: SPECS

	V100	A100
SMs	80	108
Tensor Core Precision	FP16	FP64, TF32, BF16, FP16, I8, I4, B1
Maximum Shared Memory per Block	96 kB	160 kB
Unified L1/SMEM per SM	128 kB	192 kB
L2 Cache Size	6144 kB	40960 kB
Memory Bandwidth	900 GB/sec	1555 GB/sec
NVLink Interconnect	300 GB/sec	600 GB/sec
FP64 Throughput	7.8 TFLOPS	9.7 (FMA) | 19.5 TFLOPS (MMA)
FP32 Throughput	15.7 TFLOPS	19.5 TFLOPS
TF32 Tensor Core Throughput	N/A	156 (Dense) | 312 TFLOPS (Sparse)



A100 BLOCK DIAGRAM



108 SMs
6912 CUDA Cores

40MB L2
6.7x capacity

1.56 TB/s HBM2
1.7x bandwidth

A100 SM

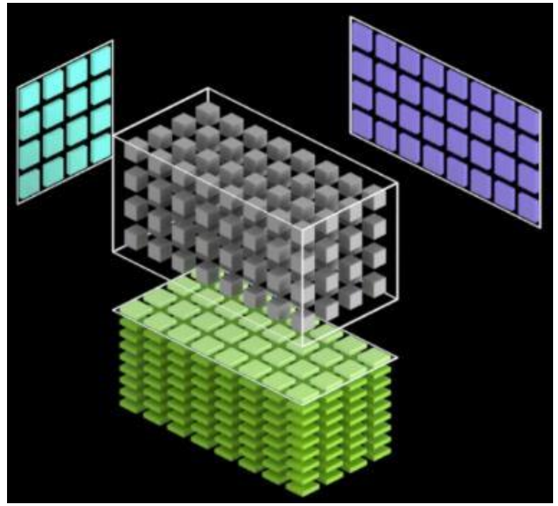
Third-generation Tensor Core

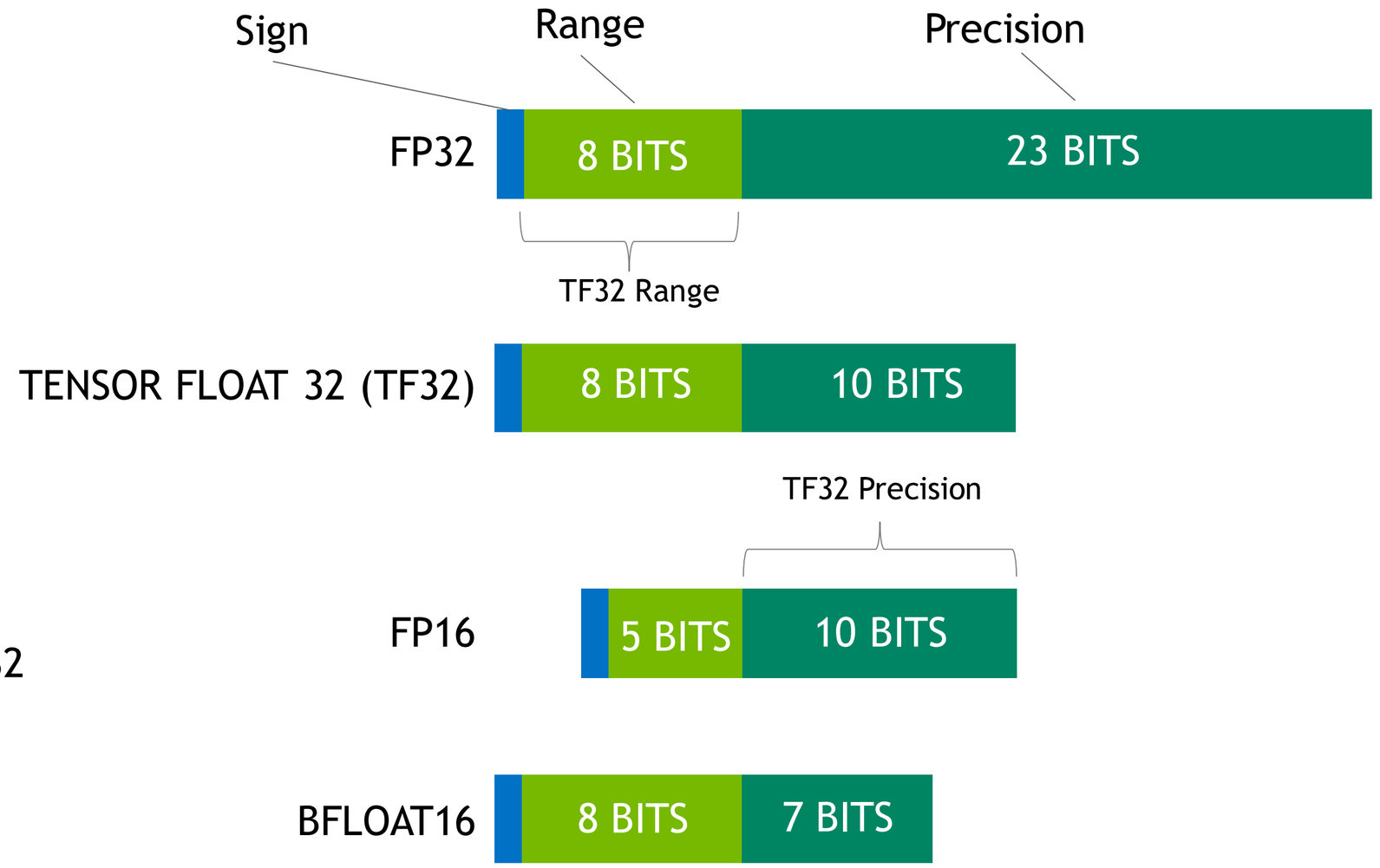
Faster and more efficient
Comprehensive data types
Sparsity acceleration

Asynchronous data movement and synchronization

Increased L1/SMEM capacity

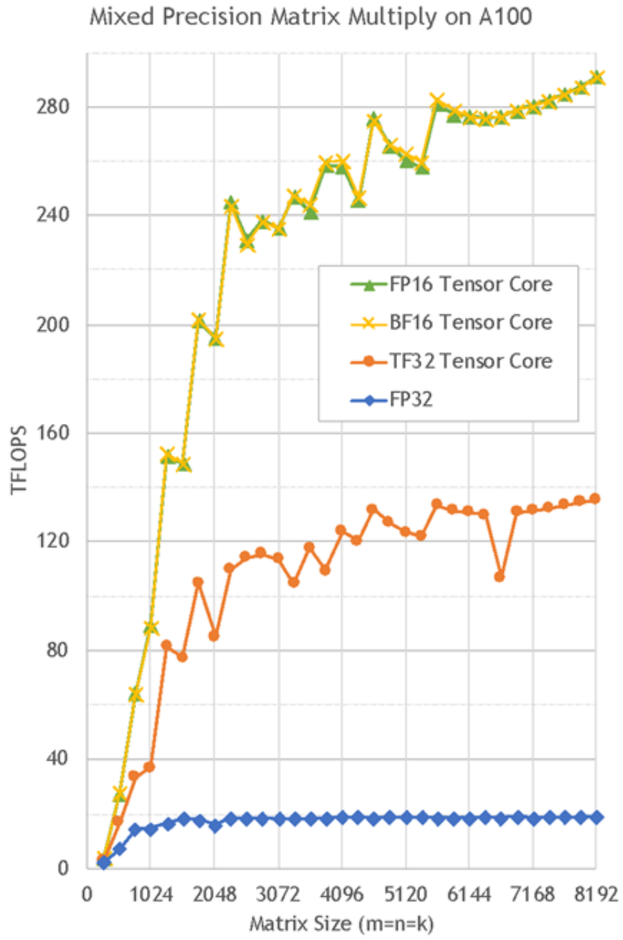
NEW TF32 TENSOR CORES

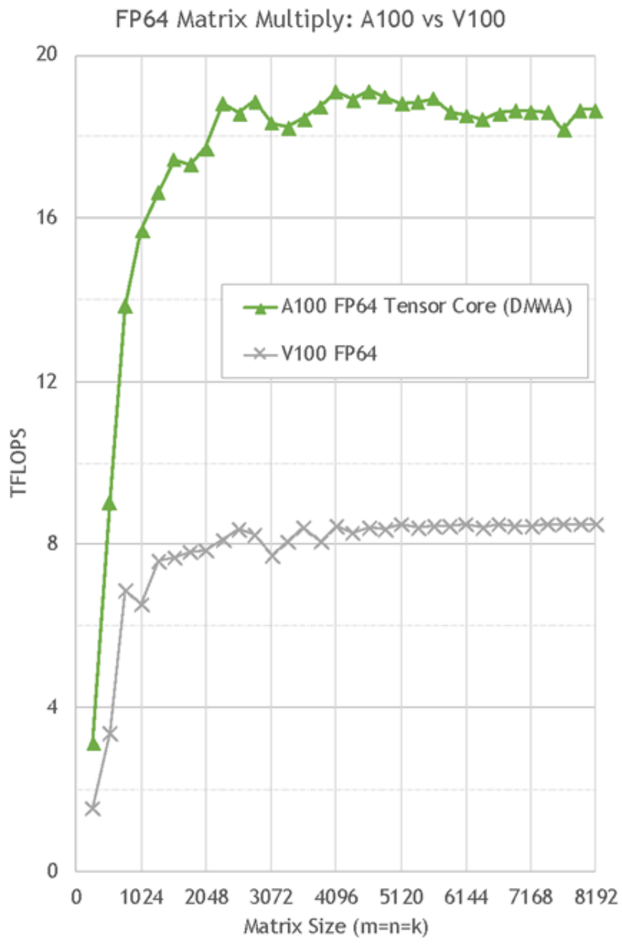




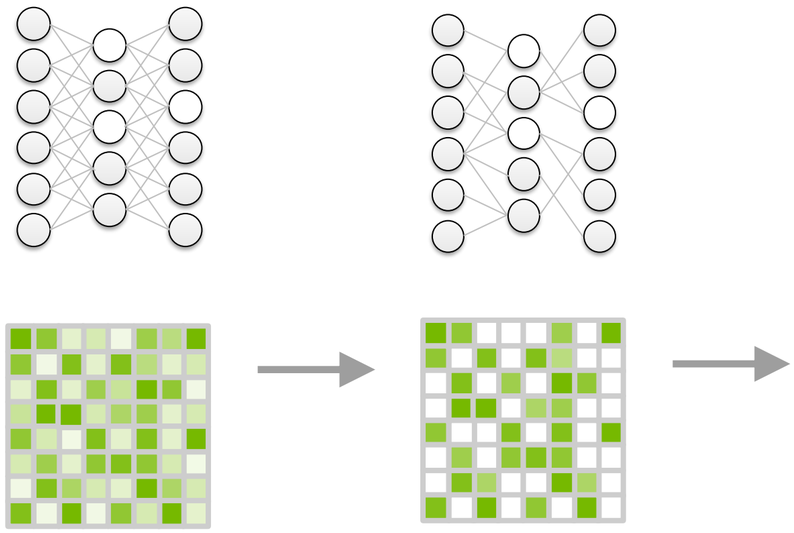
- Range of FP32 and Precision of FP16
- Input in FP32 and Accumulation in FP32
- No Code Change Speed-up for Training

MATRIX MULTIPLY THROUGHPUT WITH CUBLAS





STRUCTURAL SPARSITY BRINGS ADDITIONAL SPEEDUPS

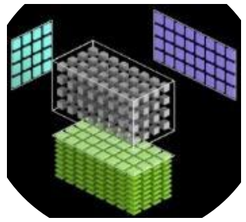


Dense Matrix

Sparse Matrix

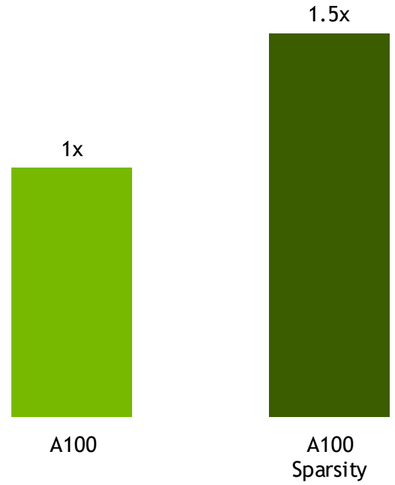
2X

Faster Execution



A100 Tensor Core

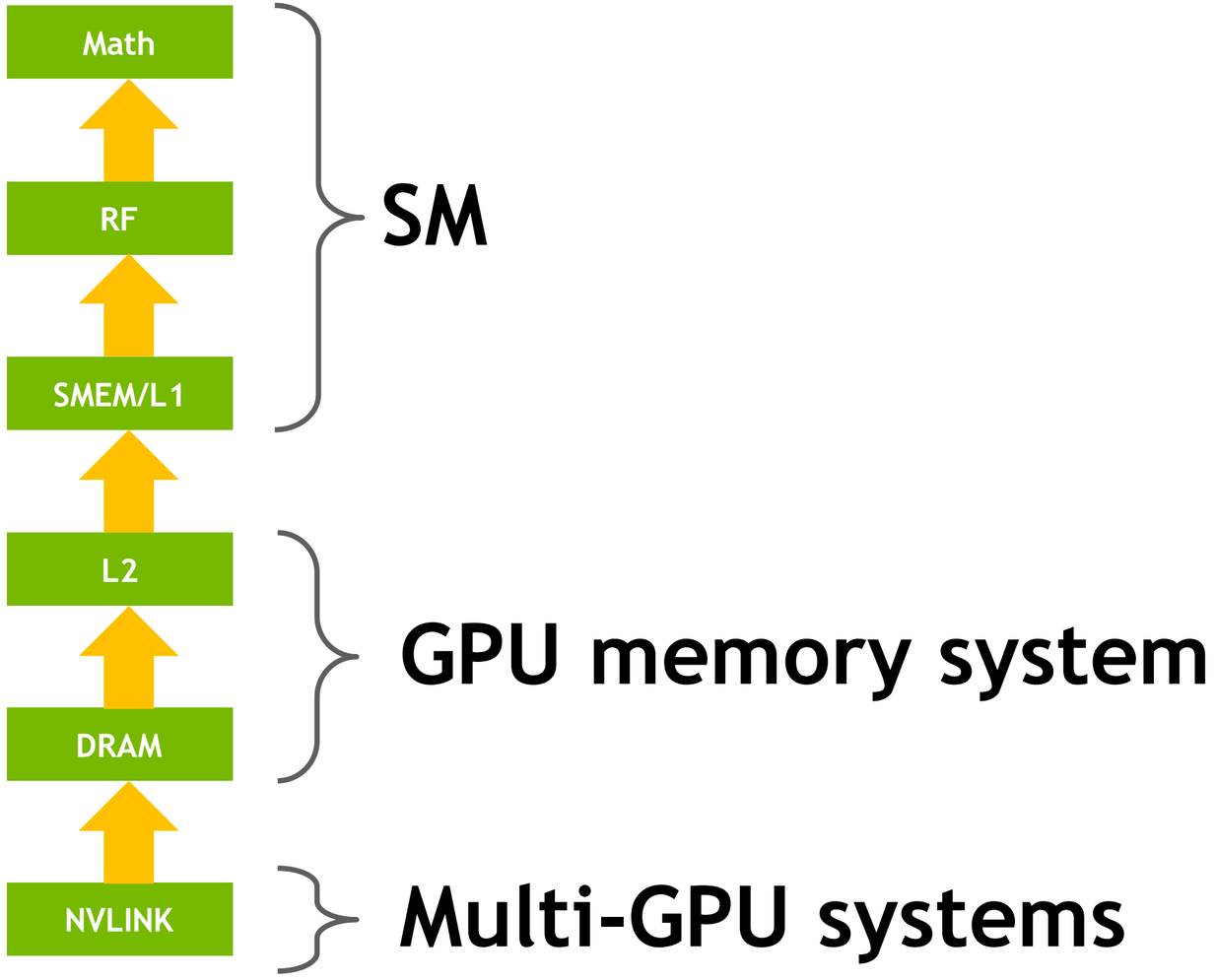
BERT Large Inference



- Structured sparsity: Half the values are zero
- Skip half of the compute and mem fetches
- Compute up to 2x rate vs non-sparse

BERT Large Inference | precision = INT8 with and without sparsity | Batch sizes - no sparsity: bs256, with sparsity: bs49, A100 with 7 MIGS

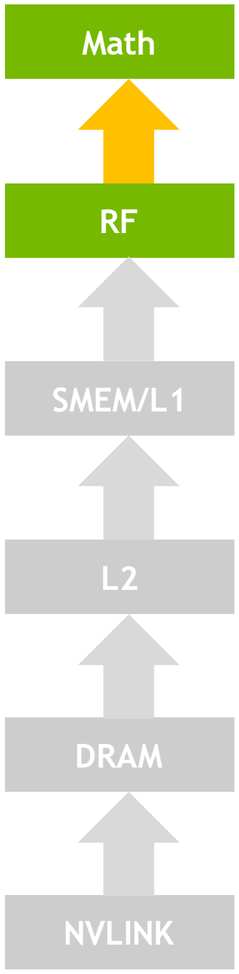
A100 STRONG SCALING INNOVATIONS



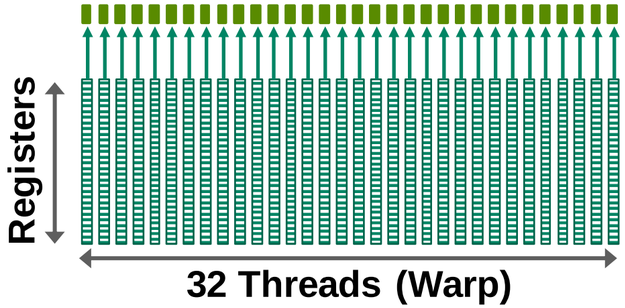
Improve speeds & feeds and efficiency across all levels of compute and memory hierarchy

A100 TENSOR CORE

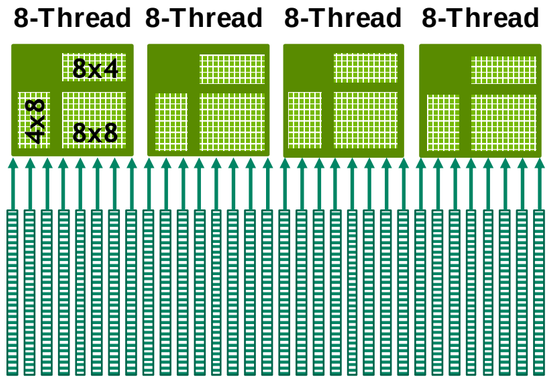
2x throughput vs. V100, >2x efficiency



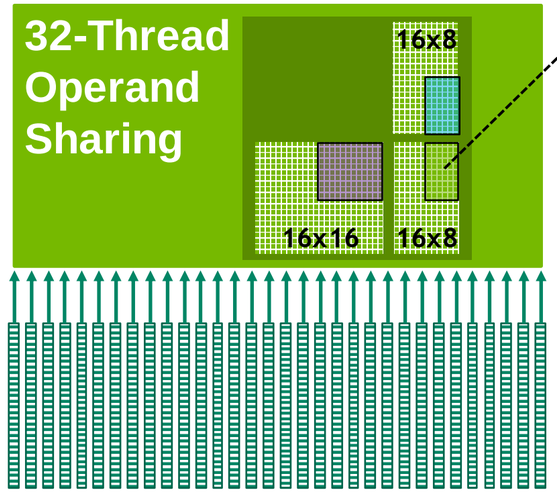
FFMA
(32 MACs, 2 cycles)



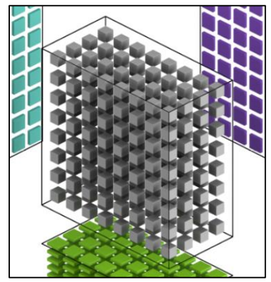
V100 TC Instruction
(1024 MACs, 8 cycles)



A100 TC Instruction
(2048 MACs, 8 cycles)



A100 TC (1 cycle)

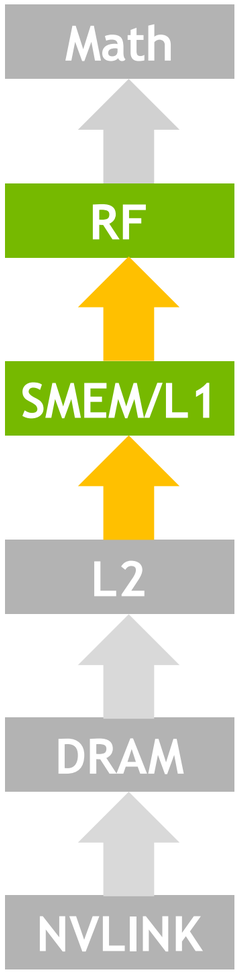


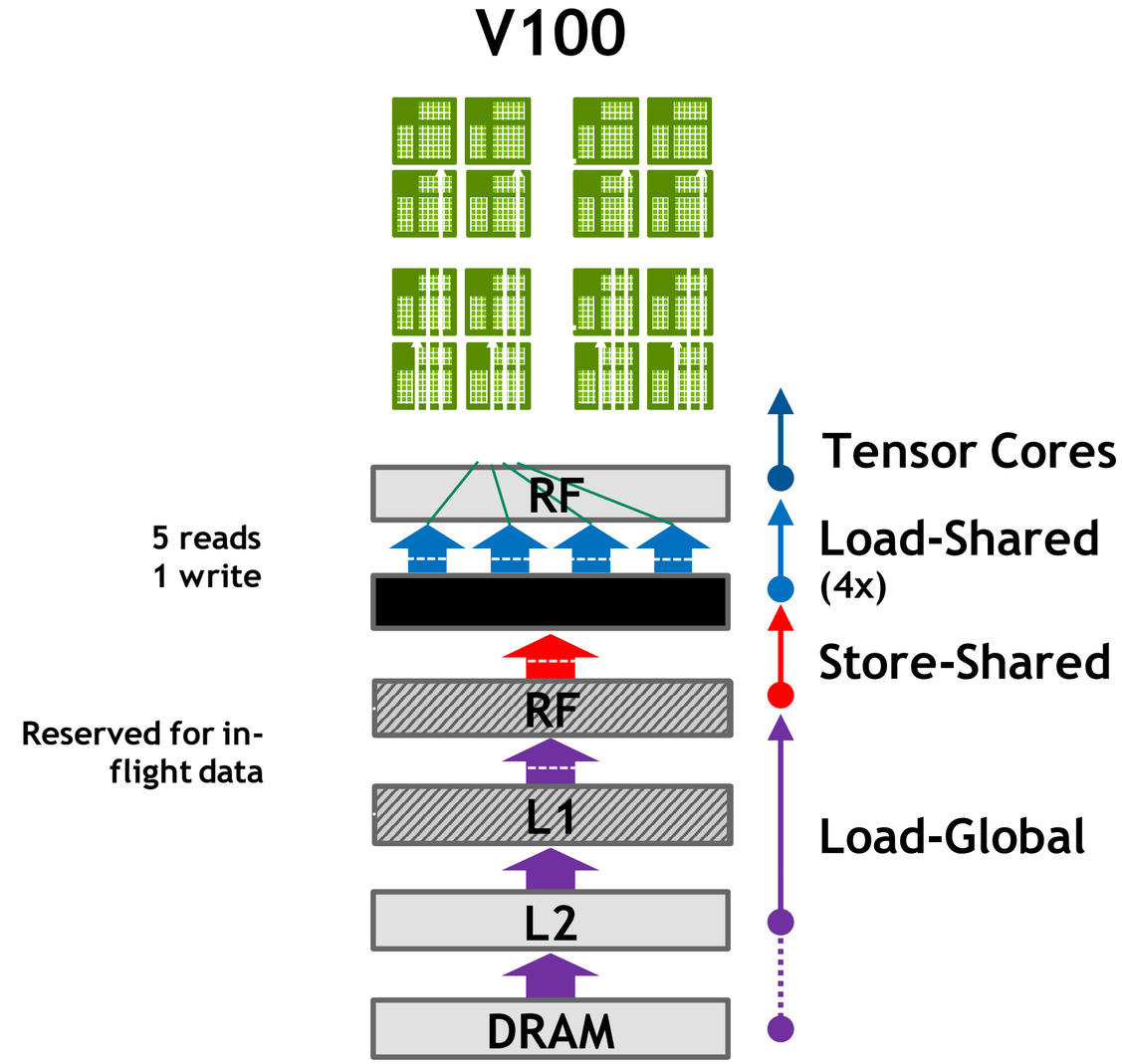
16x16x16 matrix multiply	FFMA	V100 TC	A100 TC	A100 vs. V100 (improvement)	A100 vs. FFMA (improvement)
Thread sharing	1	8	32	4x	32x
Hardware instructions	128	16	2	8x	64x
Register reads+writes (warp)	512	80	28	2.9x	18x
Cycles	256	32	16	2x	16x

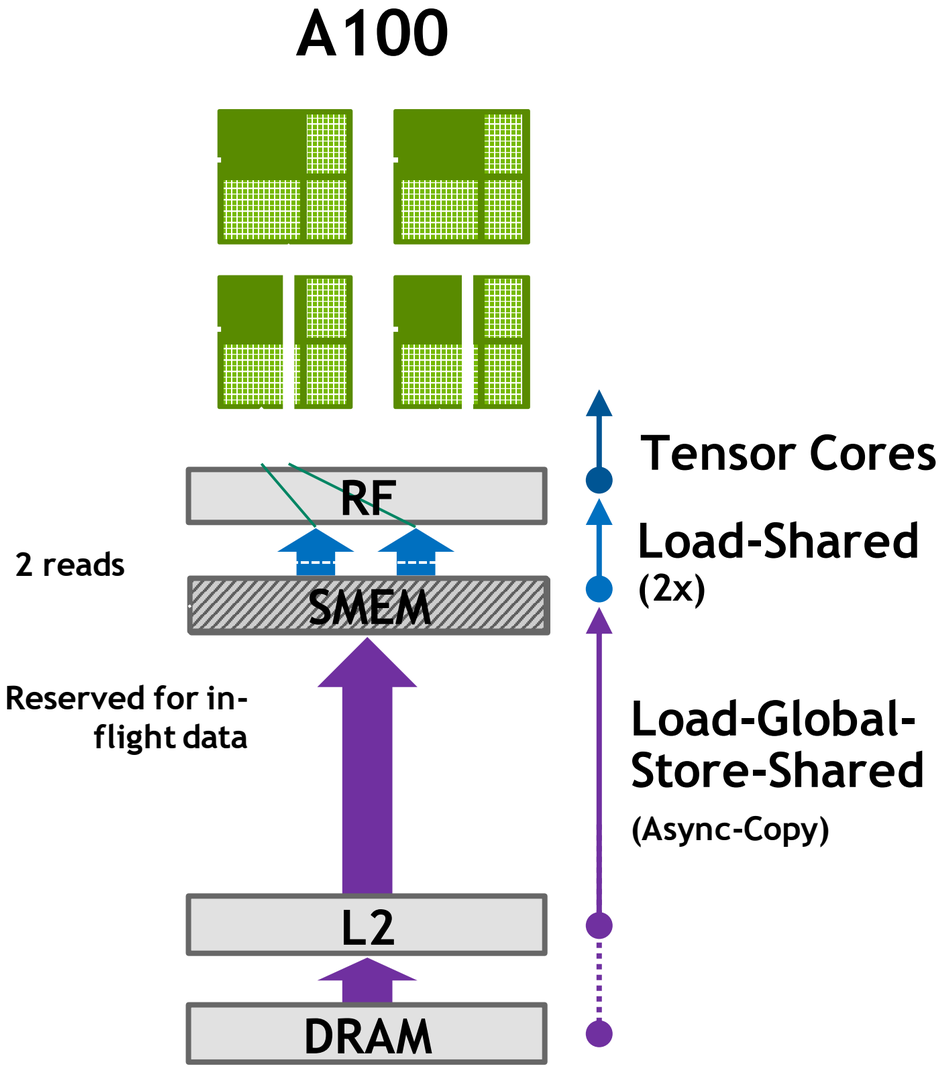
Tensor Cores assume FP16 inputs with FP32 accumulator, V100 Tensor Core instruction uses 4 hardware instructions

A100 SM DATA MOVEMENT EFFICIENCY

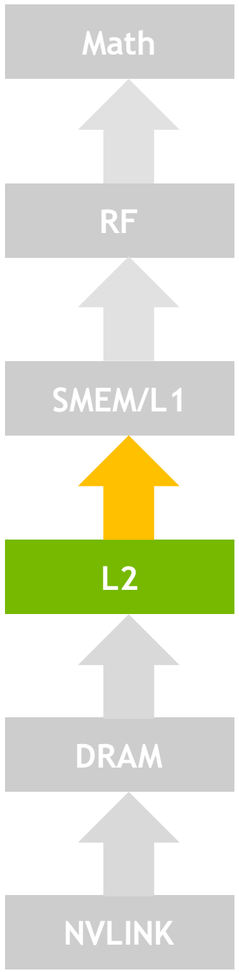
3x SMEM/L1 bandwidth, 2x in-flight capacity

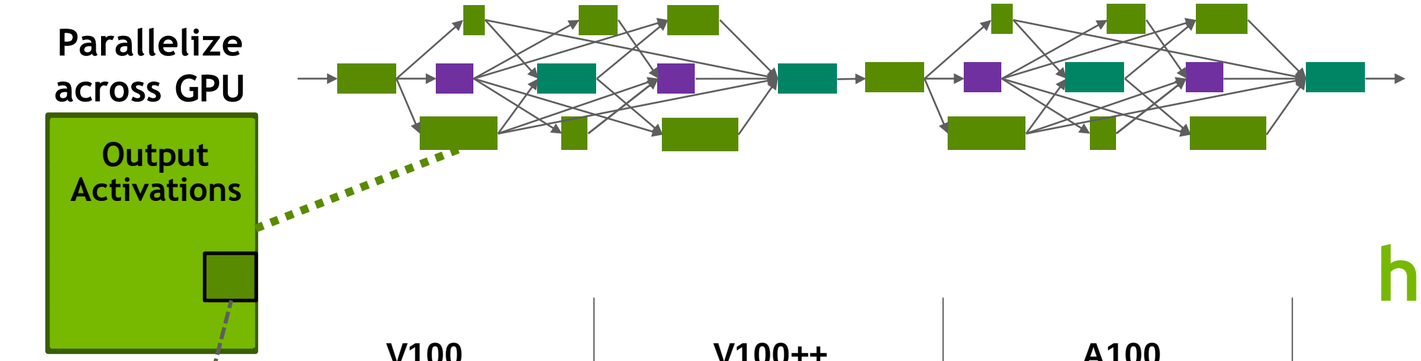


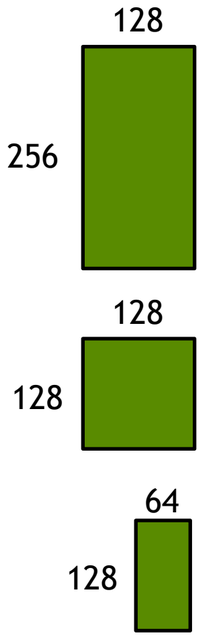




A100 L2 BANDWIDTH

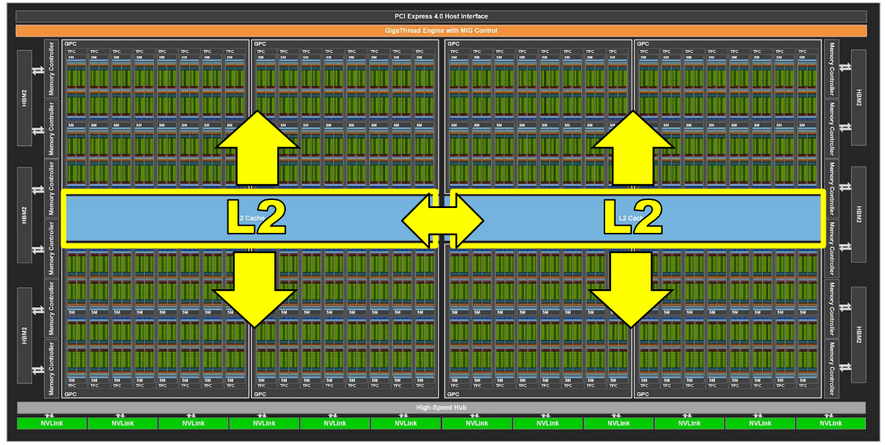






	V100	V100++ (hypothetical)	A100
	80 SMs V100 TC 64 L2 slices 32 B/clk/slice	108 SMs A100 TC 64 L2 slices 32 B/clk/slice	108 SMs A100 TC 80 L2 slices 64 B/clk/slice
	12 B/clk/SM 47%	24 B/clk/SM 127%	24 B/clk/SM 51%
	16 B/clk/SM 63%	32 B/clk/SM 169%	32 B/clk/SM 68%
	24 B/clk/SM 94%	48 B/clk/SM 253%	48 B/clk/SM 101%

Split L2 with hierarchical crossbar - 2.3x increase in bandwidth over V100, lower latency



A100 DRAM BANDWIDTH

Faster HBM2

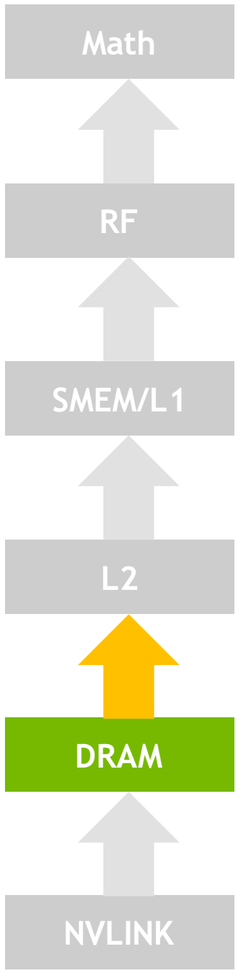
25% more pins, 38% faster clocks

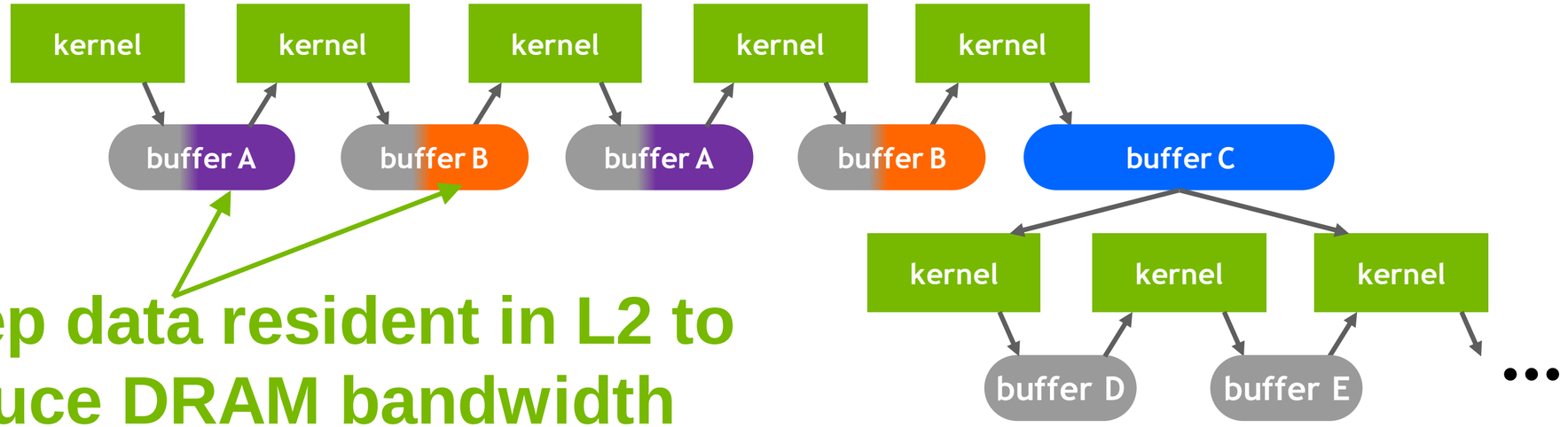
→ 1.6 TB/s, 1.7x vs. V100

Larger and smarter L2

40MB L2, 6.7x vs. V100

L2 residency controls





Keep data resident in L2 to reduce DRAM bandwidth

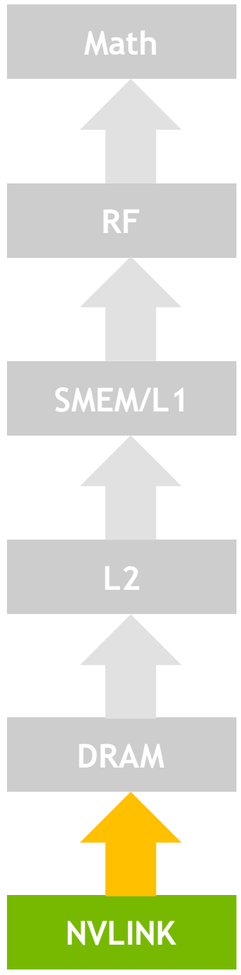
A100 NVLINK BANDWIDTH

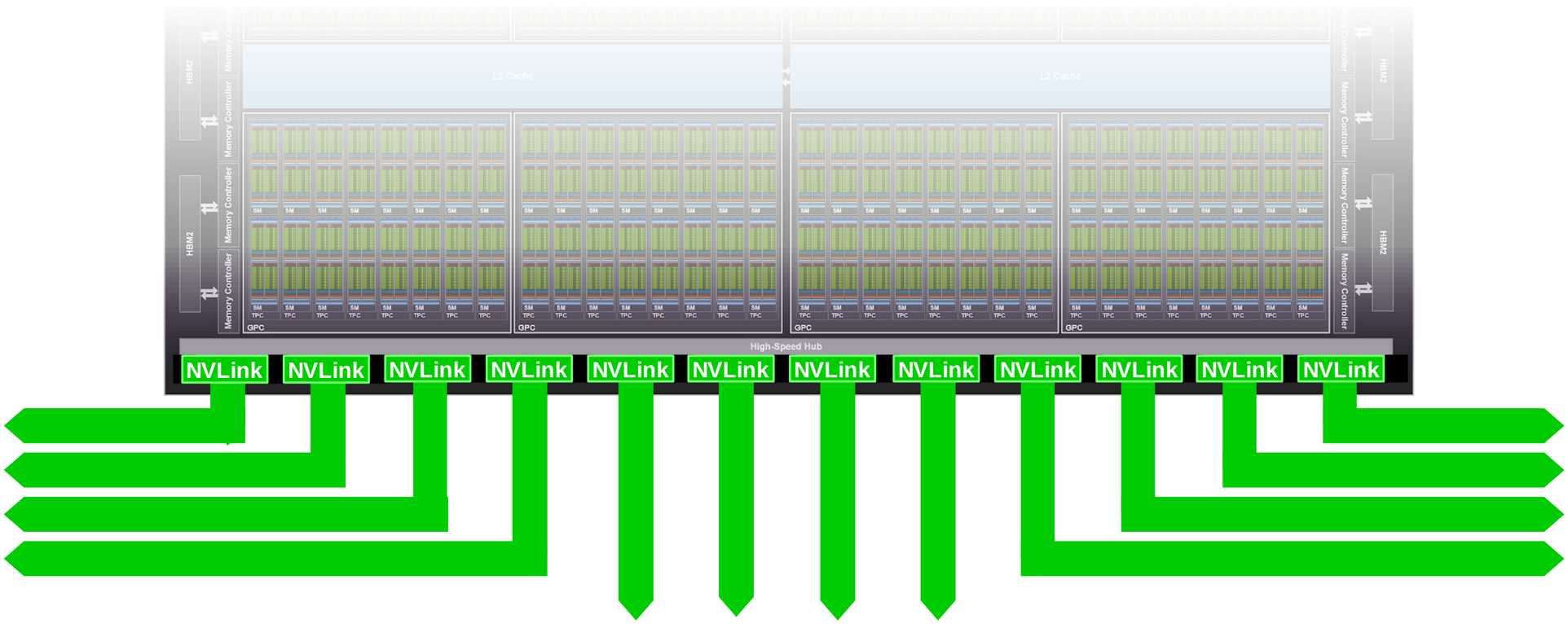
Third Generation NVLink

50 Gbit/sec per signal pair

12 links, 25 GB/s in/out, 600 GB/s total

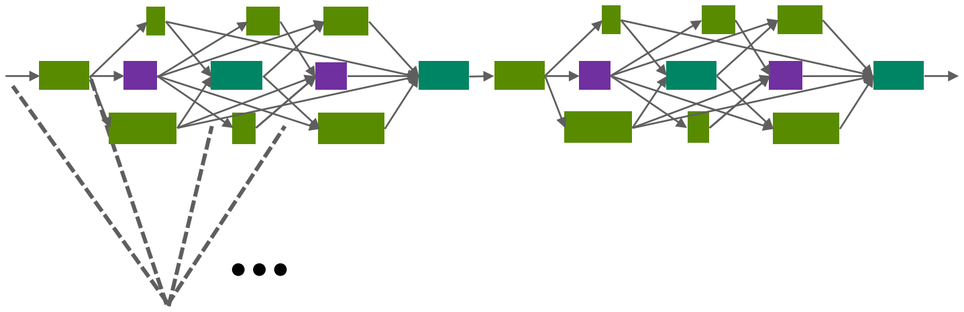
2x vs. V100





NVIDIA

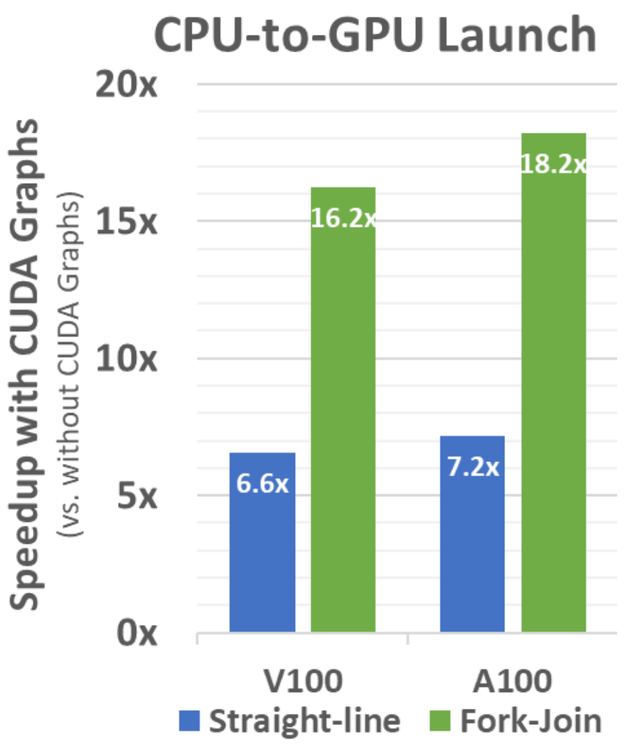
A100 ACCELERATES CUDA GRAPHS

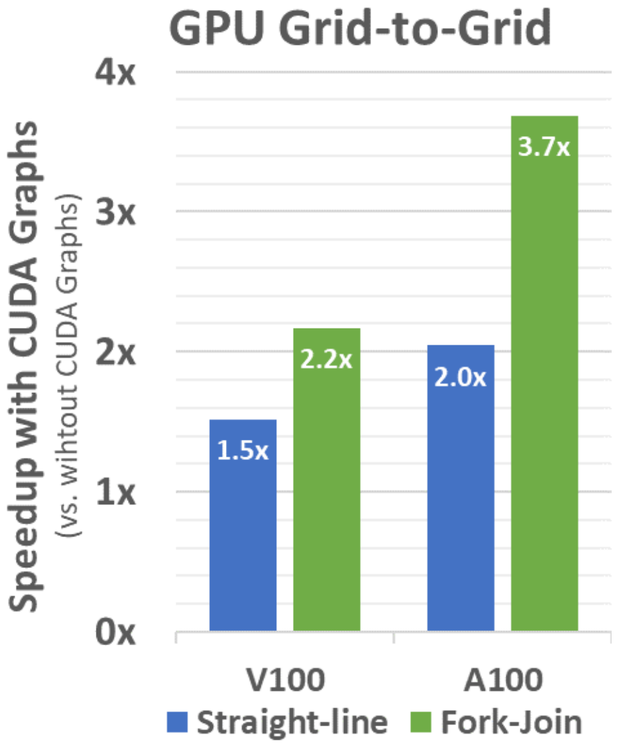


Grid launches:

- *CPU-to-GPU*
- *GPU grid-to-grid*

With strong scaling CPU and grid launch overheads become increasingly important (Amdahl's law)





32-node graphs of empty grids, DGX1-V, DGX-A100

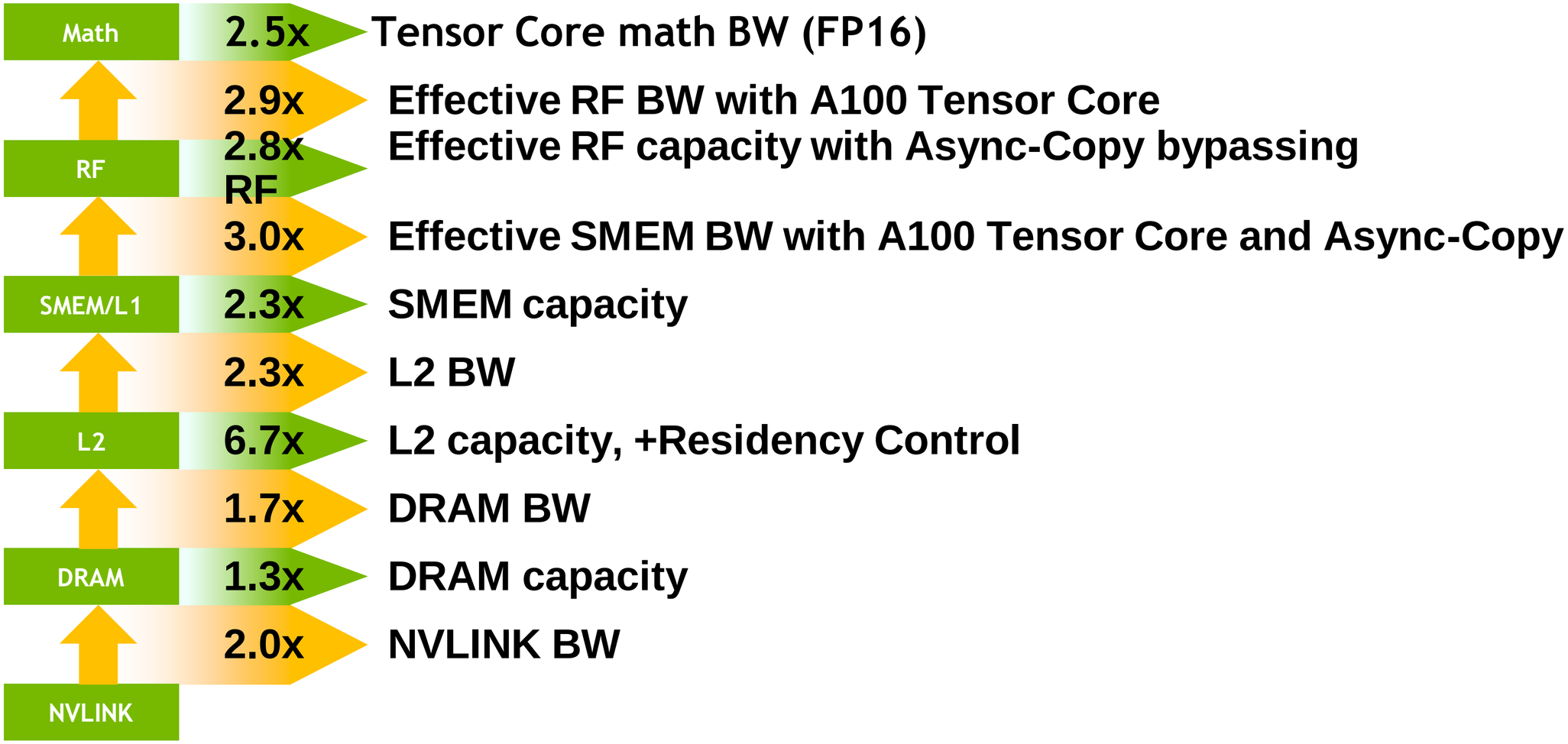
One-shot CPU-to-GPU graph submission and graph reuse

Microarchitecture improvements for grid-to-grid latencies

A100 STRONG SCALING INNOVATIONS

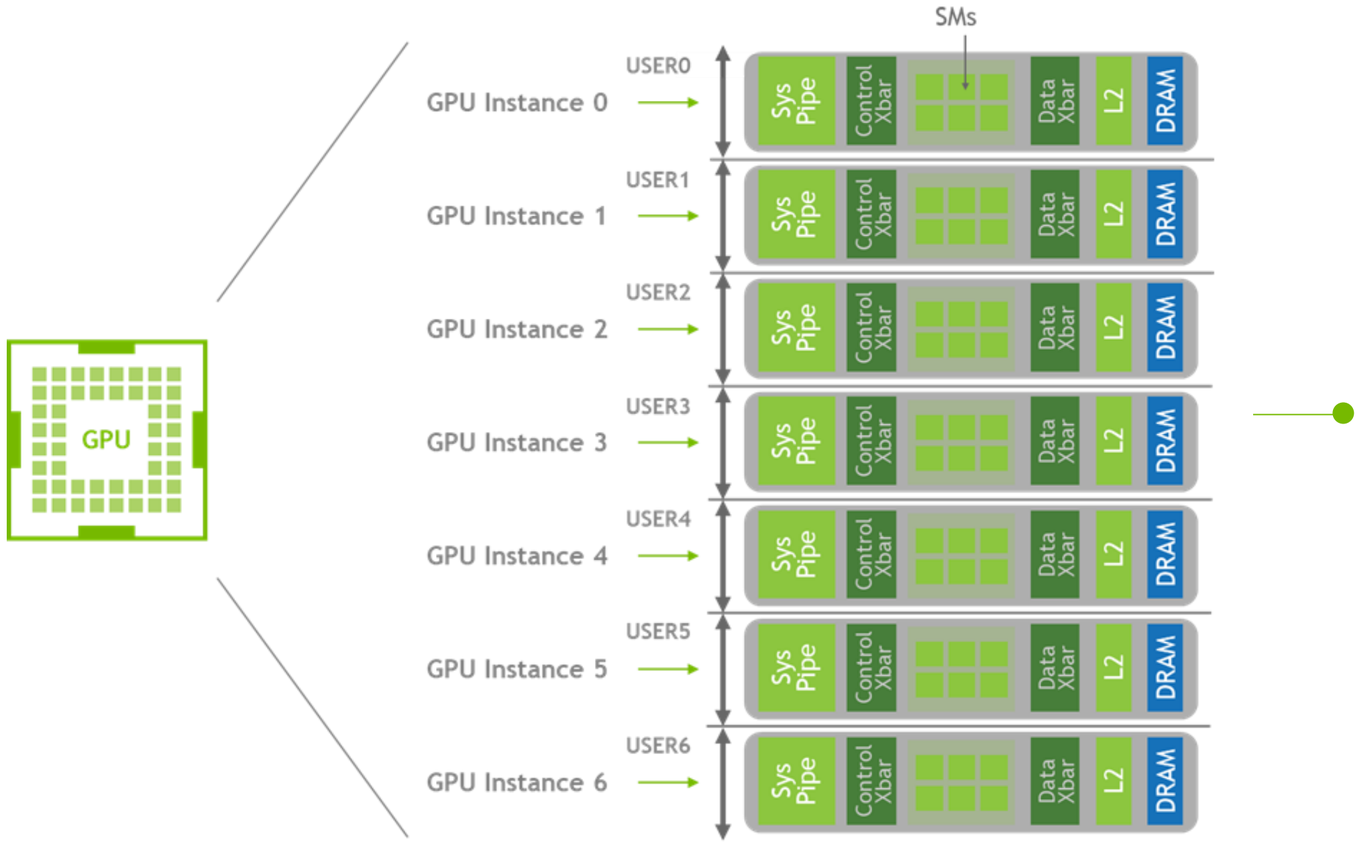
Delivering unprecedented levels of performance

A100 improvements over V100



MULTI-INSTANCE GPU (MIG)

Optimize GPU Utilization, Expand Access to More Users with Guaranteed Quality of Service



Up To 7 GPU Instances In a Single A100:
Dedicated SM, Memory, L2 cache, Bandwidth for hardware QoS & isolation

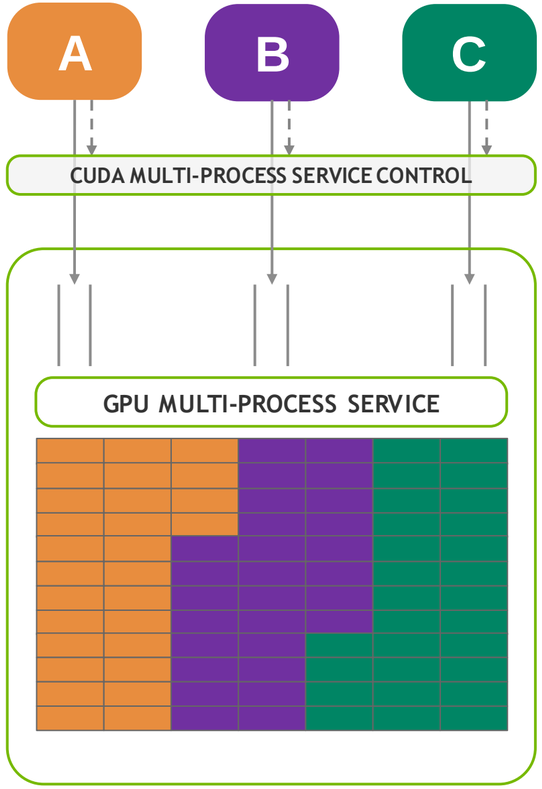
Simultaneous Workload Execution With Guaranteed Quality Of Service:
All MIG instances run in parallel with predictable throughput & latency

Right Sized GPU Allocation:
Different sized MIG instances based on target workloads

Diverse Deployment Environments:
Supported with Bare metal, Docker, Kubernetes, Virtualized Env.

MIG User Guide: https://docs.nvidia.com/datacenter/tesla/mig-user-guide/index.html

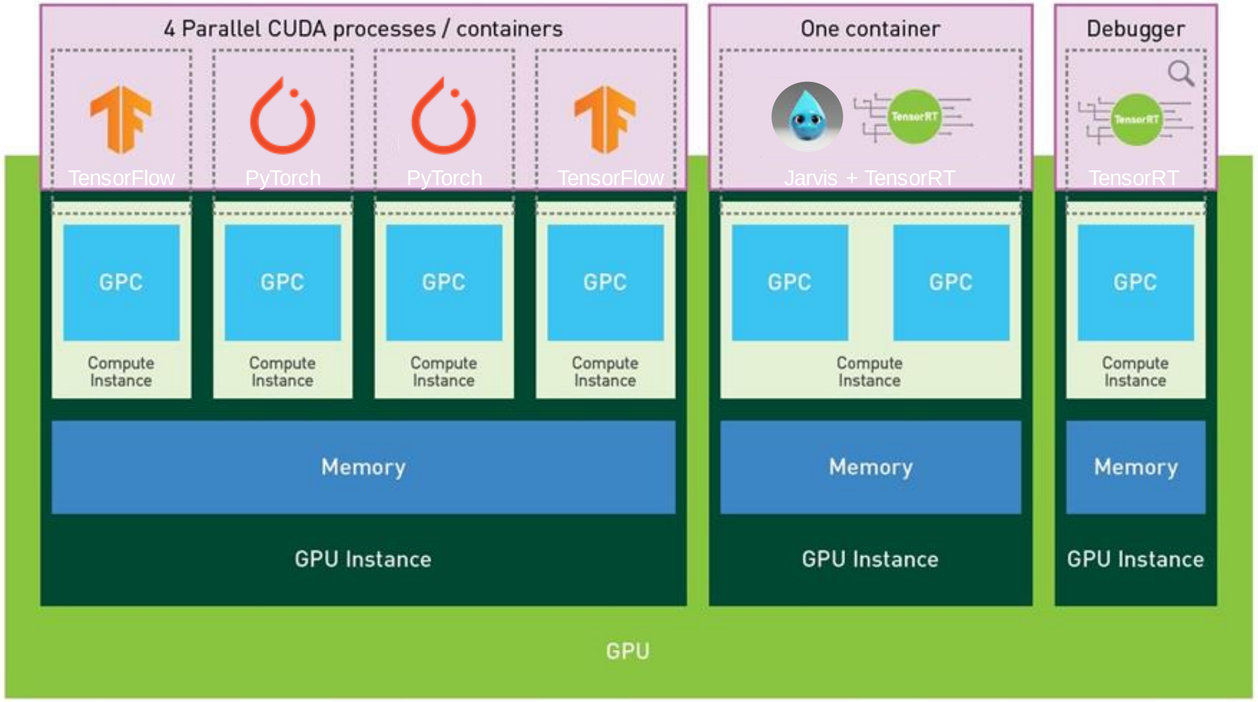
LOGICAL VS. PHYSICAL PARTITIONING



Multi-Process Service

Dynamic contention for GPU resources

Single tenant



Multi-Instance GPU

Hierarchy of instances with guaranteed resource allocation

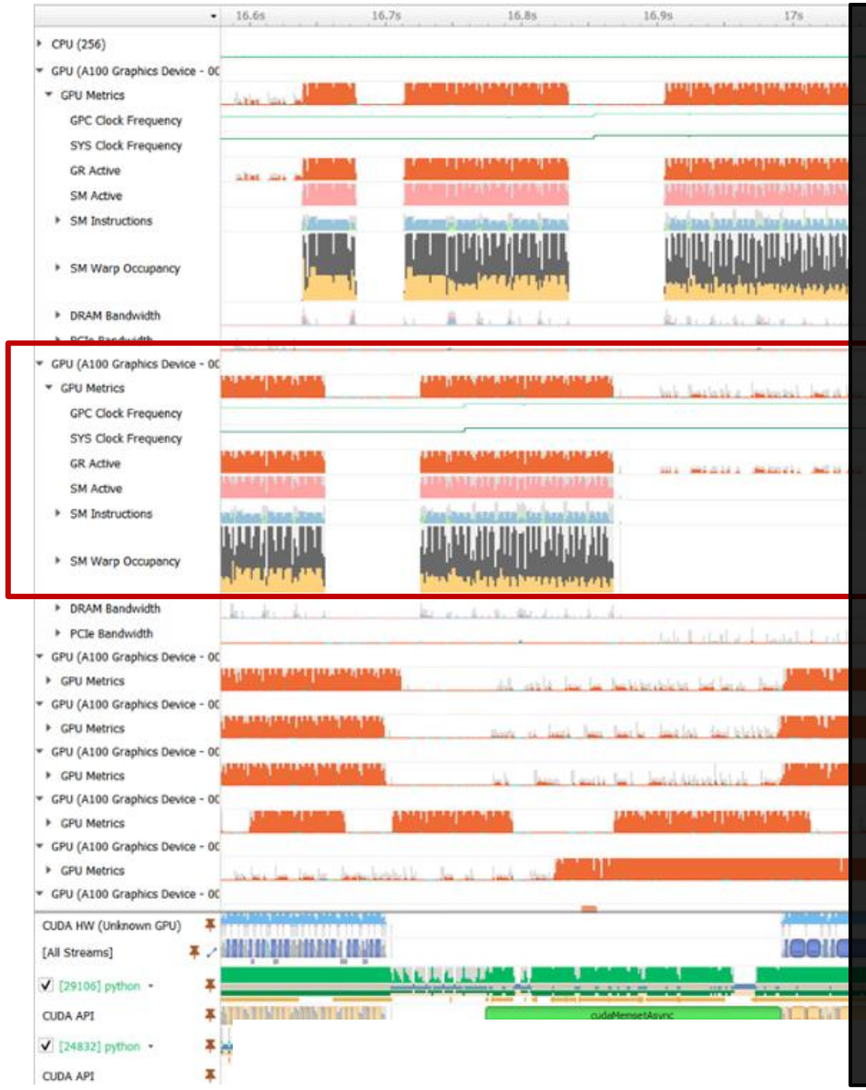
Multiple tenants

CUDA CONCURRENCY MECHANISMS

	Streams	MPS	MIG
Partition Type	Single process	Logical	Physical
Max Partitions	Unlimited	48	7
SM Performance Isolation	No	Yes (by percentage, not partitioning)	Yes
Memory Protection	No	Yes	Yes
Memory Bandwidth QoS	No	No	Yes
Error Isolation	No	No	Yes
Cross-Partition Interop	Always	IPC	Limited IPC
Reconfigure	Dynamic	Process launch	When idle

GPU METRICS SAMPLING IN NSIGHT SYSTEMS

GPU METRICS SAMPLING IN NSIGHT SYSTEMS



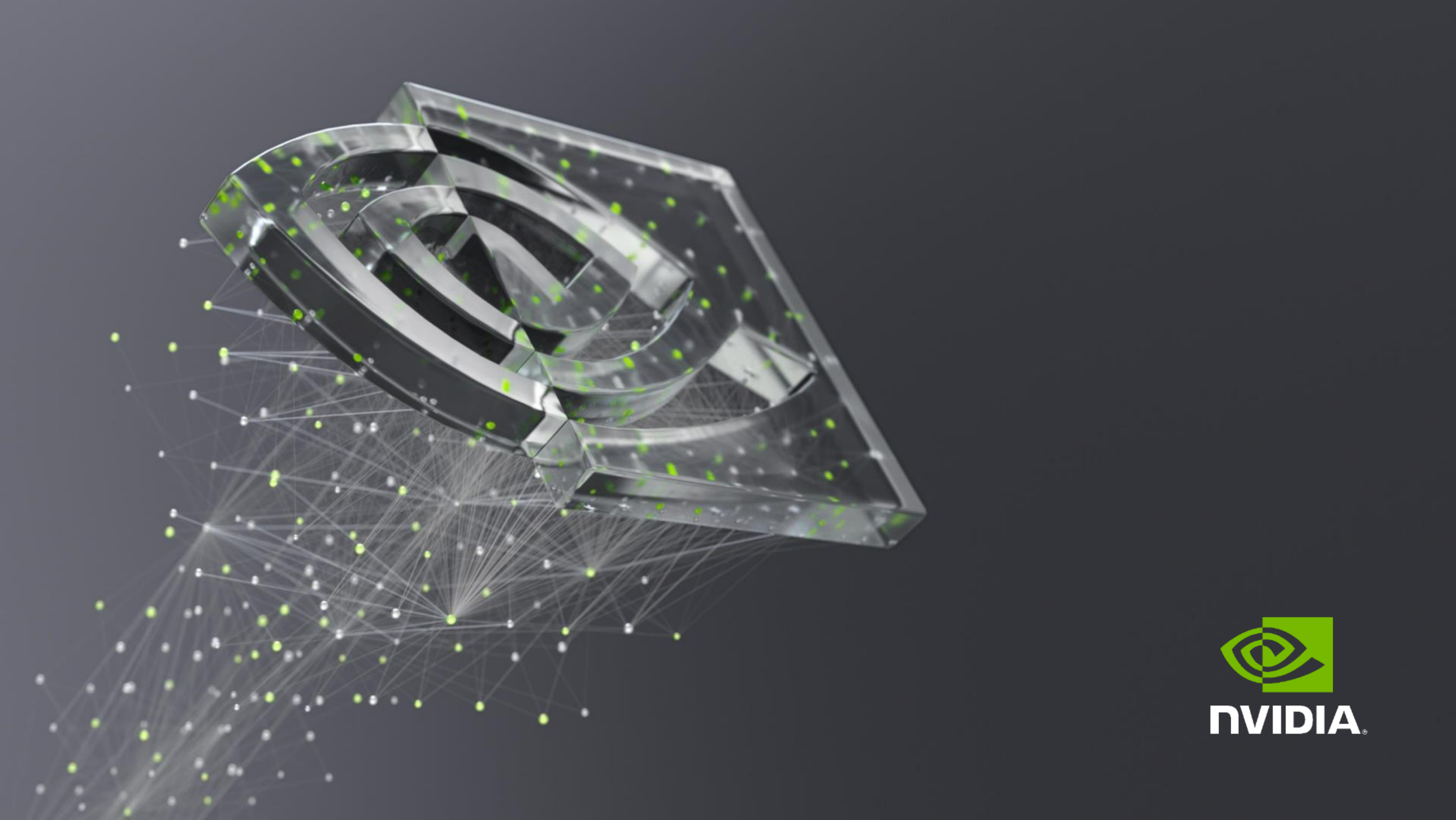
Hardware metrics collected at runtime to answer

- Is my GPU full? Sufficient grid size & streams?
- Is my instruction rate low (possibly I/O bound)?
- Am I using tensor cores?
- Can I see GPU Direct RDMA|Storage or other transfers?

System-wide GPU observation

10khz default can be increased depending on GPU

SM utilization metrics	I/O throughput metrics
SMs active	PCIe
Instructions	NVLink
Tensor cores	DRAM
Warp occupancy	


NVIDIA