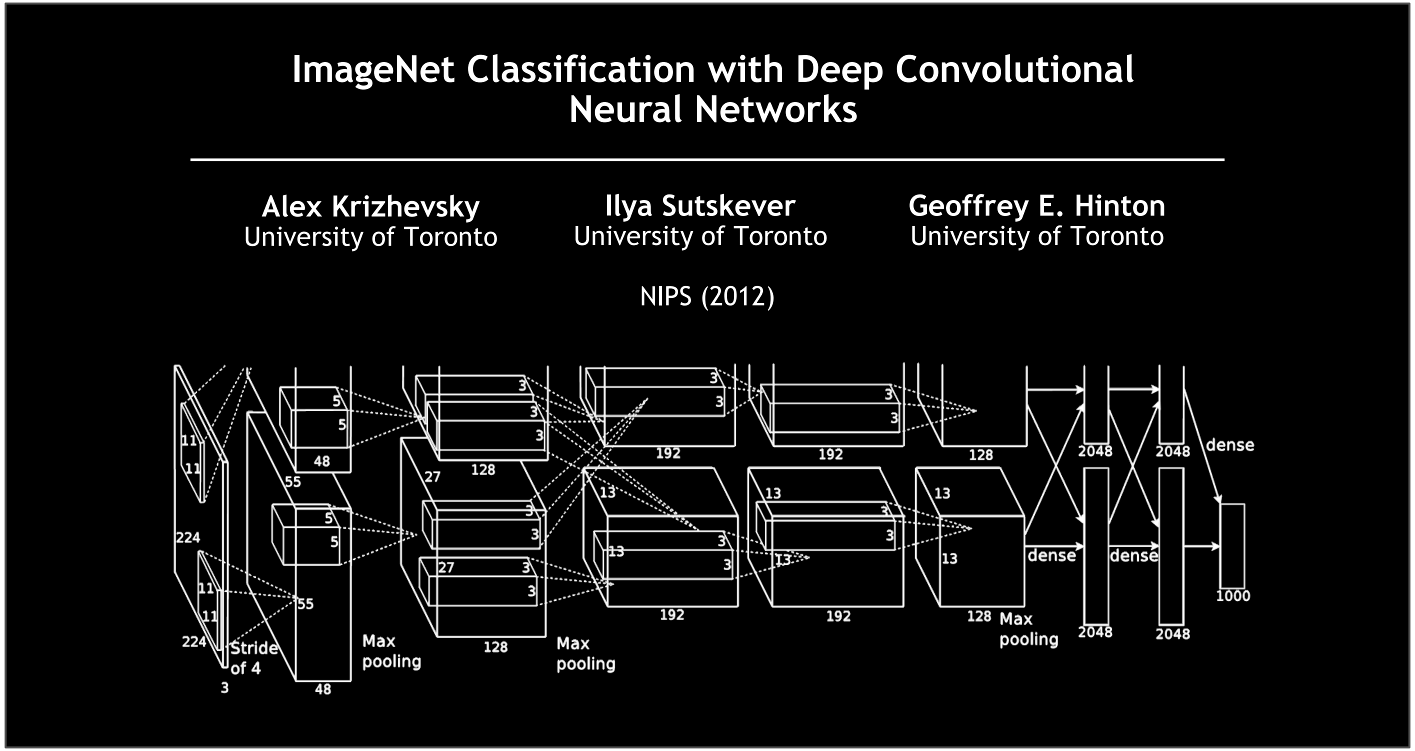
ACCELERATED COMPUTING FOR AI

Bryan Catanzaro, 28 October 2017

NVIDIA

DEEP LEARNING BIG BANG



Deep Learning

NVIDIA GPU

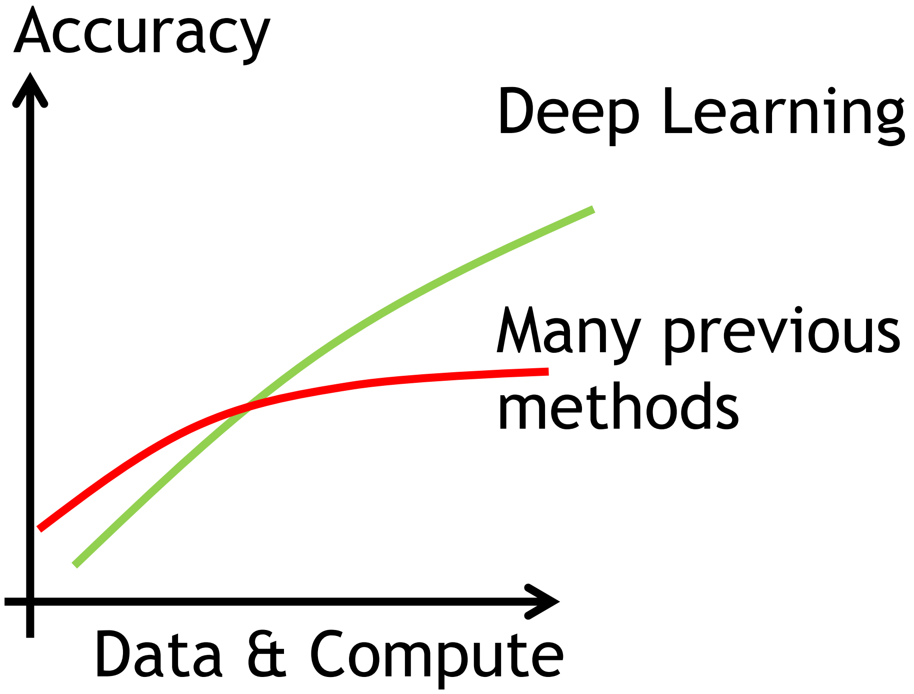
WHY IS DEEP LEARNING SUCCESSFUL

Big data sets

New algorithms

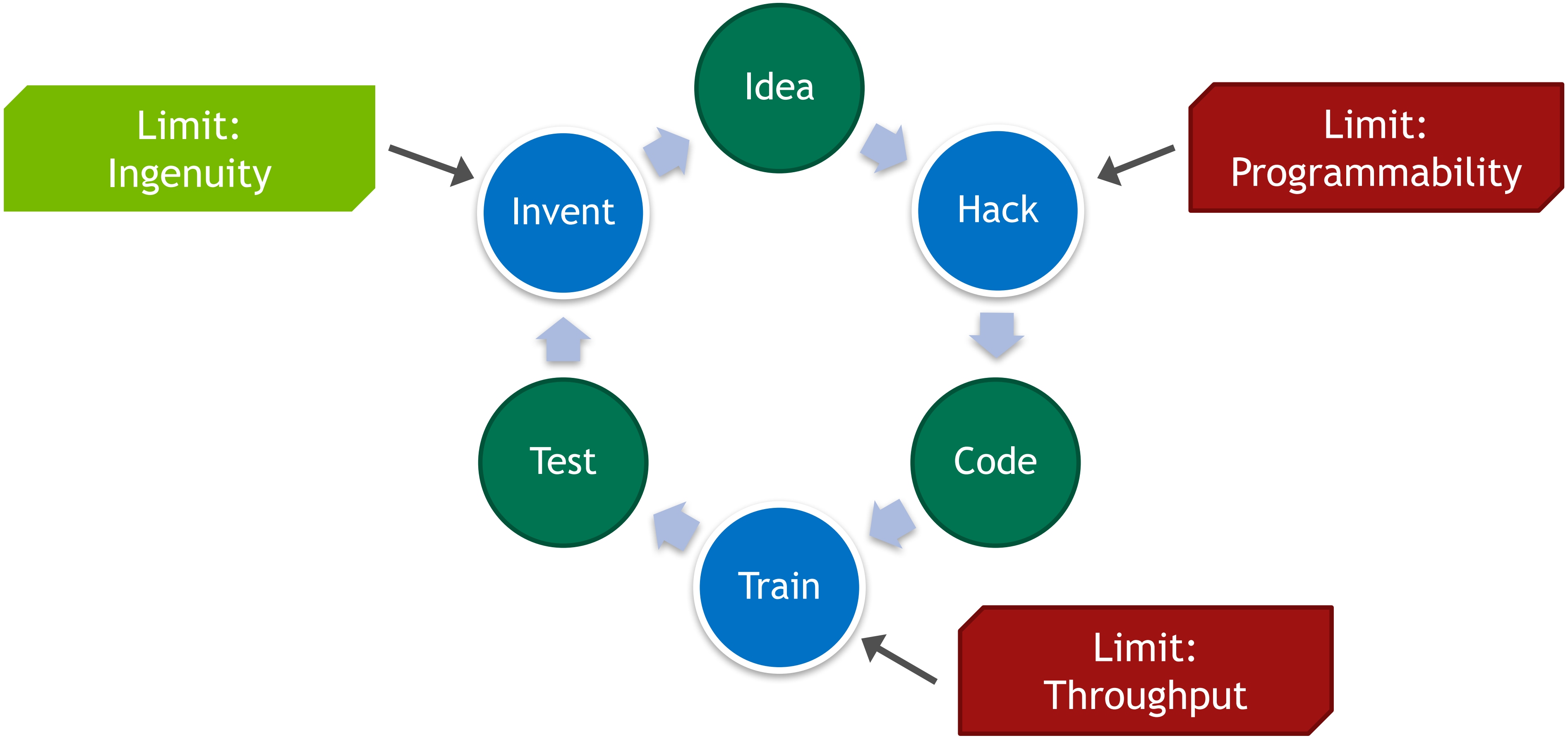
Computing hardware

Focus of this talk



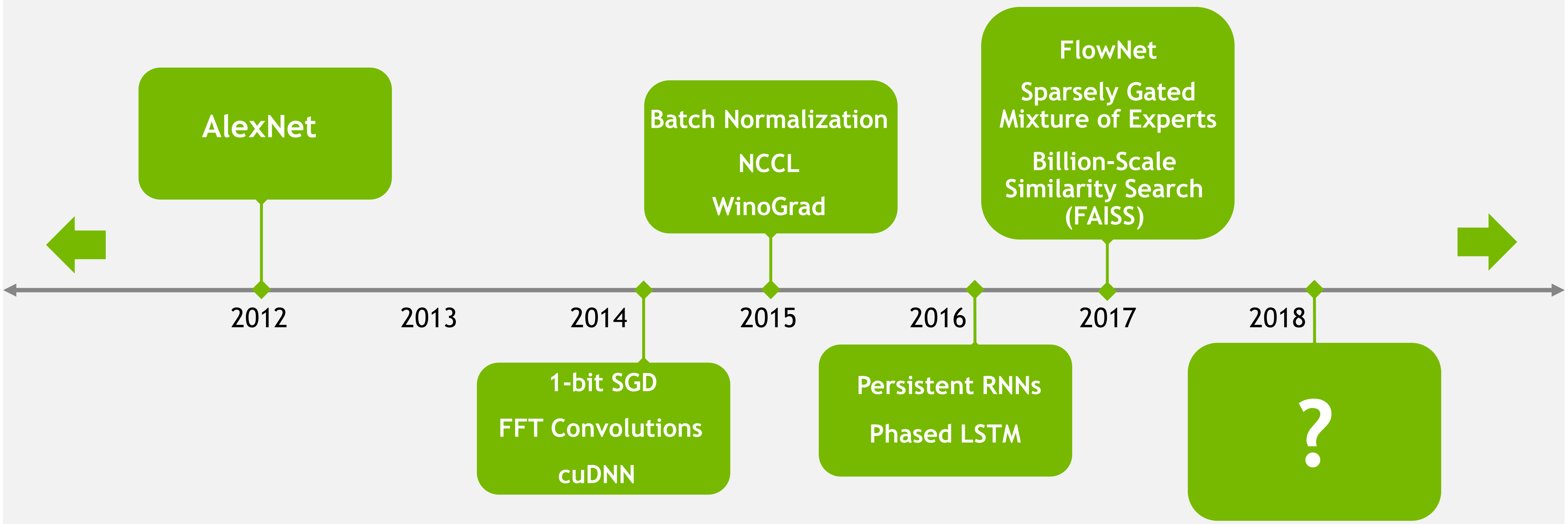
RESEARCH AS A SEQUENTIAL PROCESS

Goal: reduce latency of idea generation



COMPUTATIONAL EVOLUTION

Deep learning changes every day



New solvers, new layers, new scaling techniques, new applications for old techniques, and much more…

CUDA

Programming system for accelerated computing

C++ for accelerated processors

On-chip memory management

Asynchronous, parallel API

Programmability makes it possible to innovate

10 years of investment

New layer?

No problem.

CUDA LIBRARIES

Optimized Kernels

CUBLAS: Linear algebra

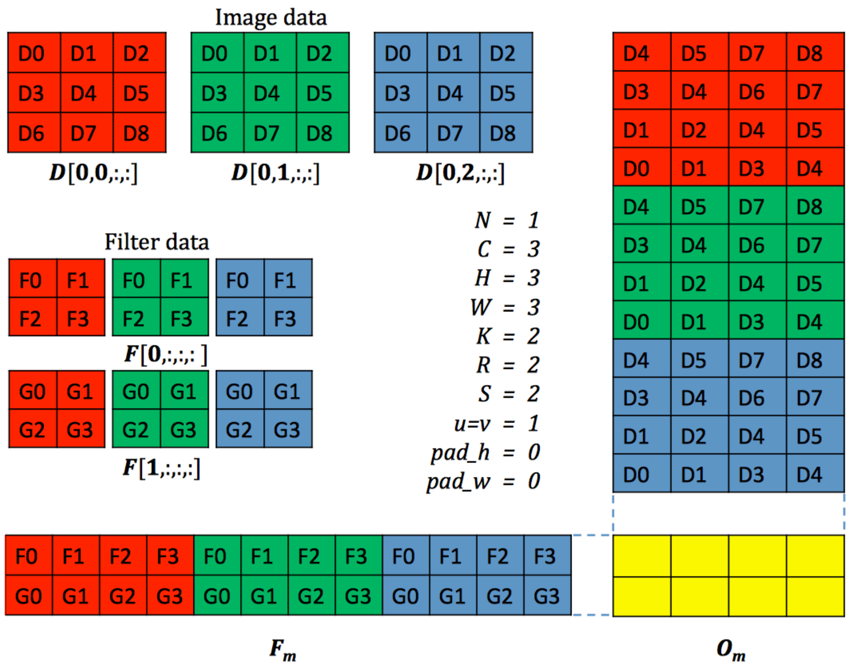
So many flavors of GEMM

CUDNN: Neural network kernels

Convolutions (direct, Winograd, FFT)

Can achieve > Speed of Light!

Recurrent Neural Networks



COMMUNICATION LIBRARIES

NCCL, MPI

NCCL: Optimized intra-node & inter-node communication

Library with sophisticated topology aware collective algorithms

MPI: Library for inter-node communication

CUDA-aware MPI means you can run MPI programs using GPUs

Scalable, distributed code in a familiar environment for HPC

All-reduce: king of data parallel training

FRAMEWORKS

Cambrian explosion of AI

Need programmability

Lots of AI frameworks

Let researchers prototype rapidly

All are GPU accelerated

SIMULATION

Many important AI tasks involve agents interacting with the real world

For this, you need simulators

Physics

Appearance

Simulation has a big role to play in AI progress

NVIDIA Project Isaac: simulator for RL

DEEP NEURAL NETWORKS

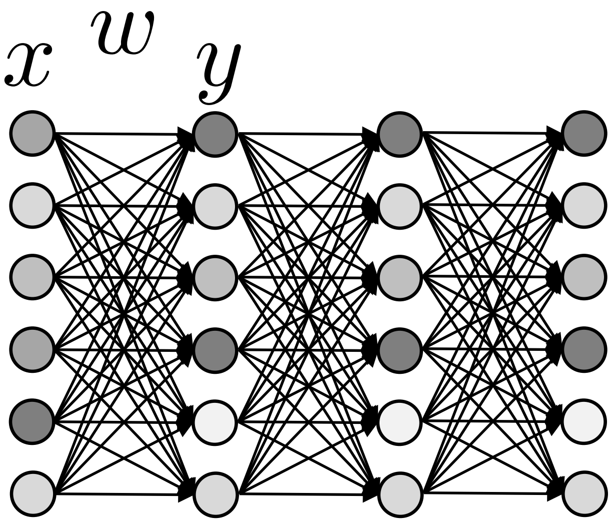
Simple, powerful function approximators

$$y_j = f\left(\sum_i w_{ij} x_i\right)$$

One layer

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}$$

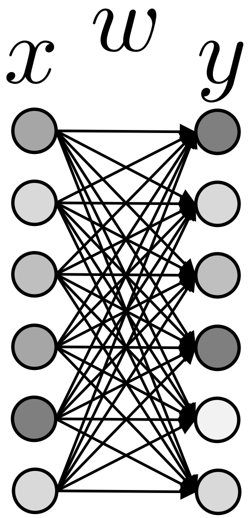
nonlinearity



Deep Neural Network

TRAINING NEURAL NETWORKS

$$y_j = f\left(\sum_i w_{ij} x_i\right)$$



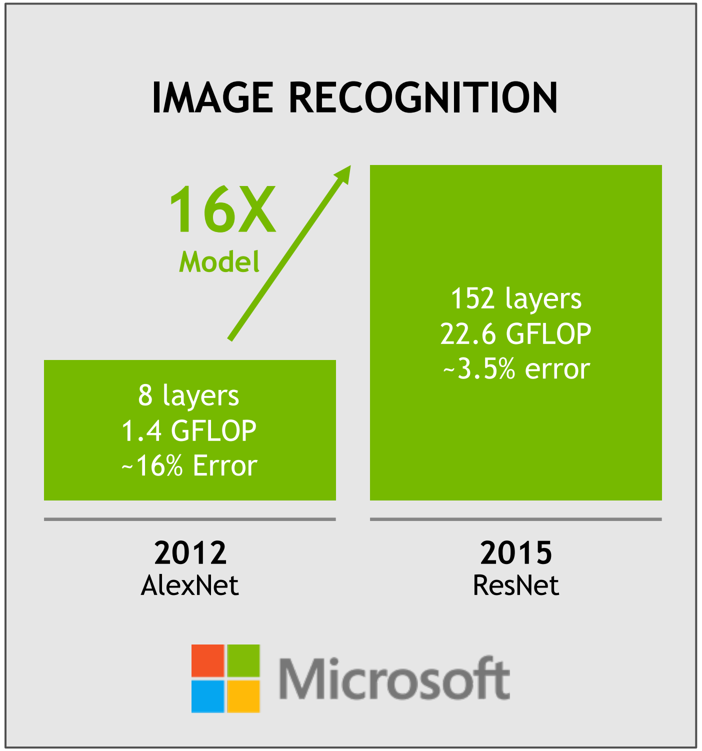
Computation dominated by dot products

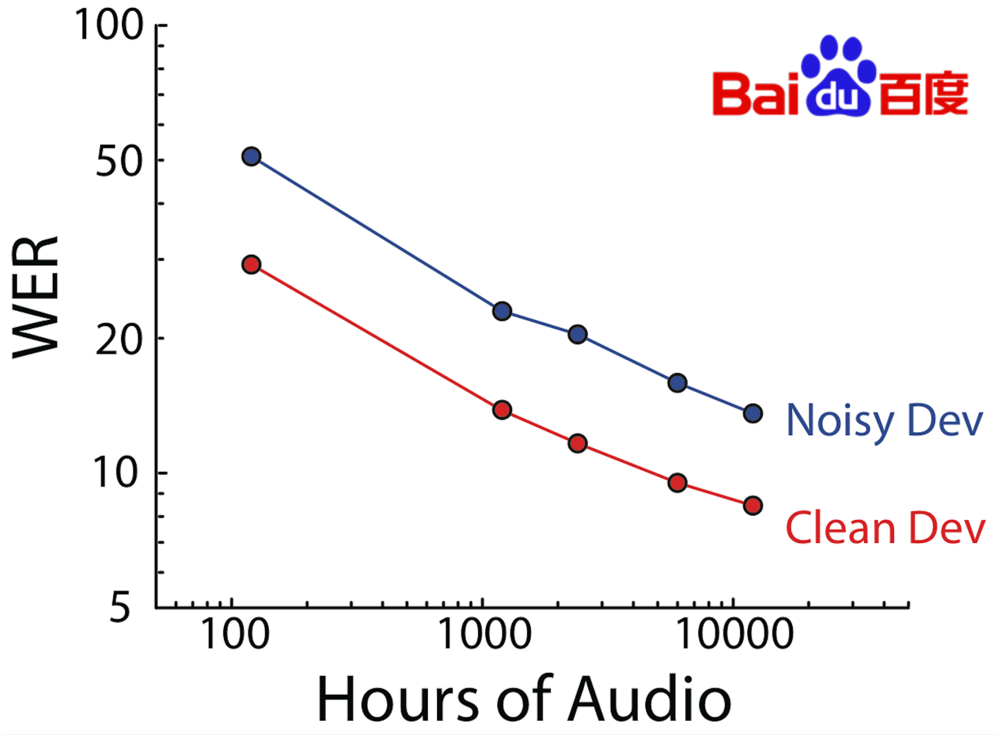
Multiple inputs, multiple outputs, batch means it is compute bound

Train one model: 20+ Exaflops

SCALE MATTERS

More data, more compute: More AI





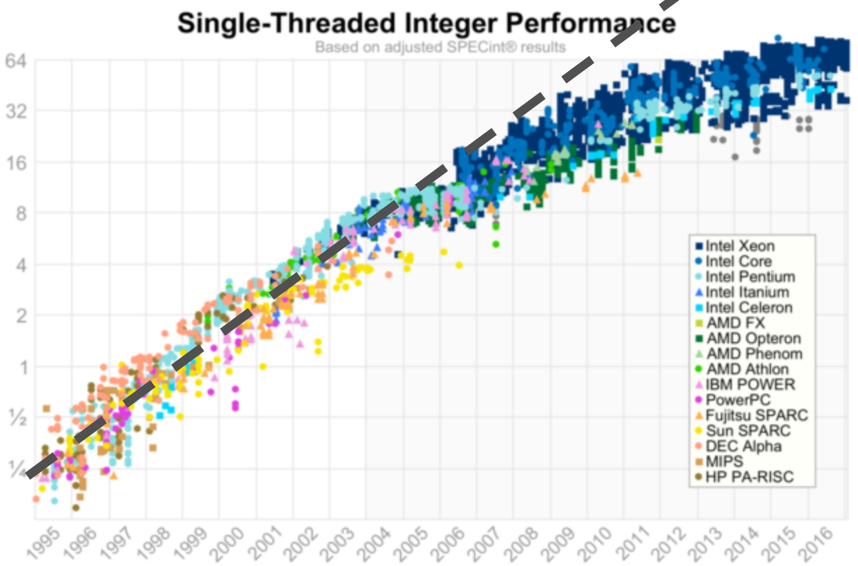
LAWS OF PHYSICS

Successful AI uses Accelerated Computing

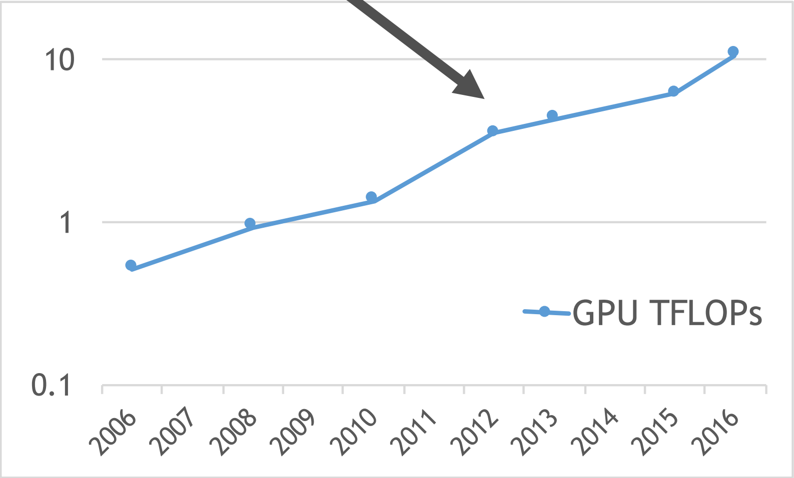
Volta

20X gap
and growing…

20X in 10 years



General Purpose Performance



Accelerated Performance

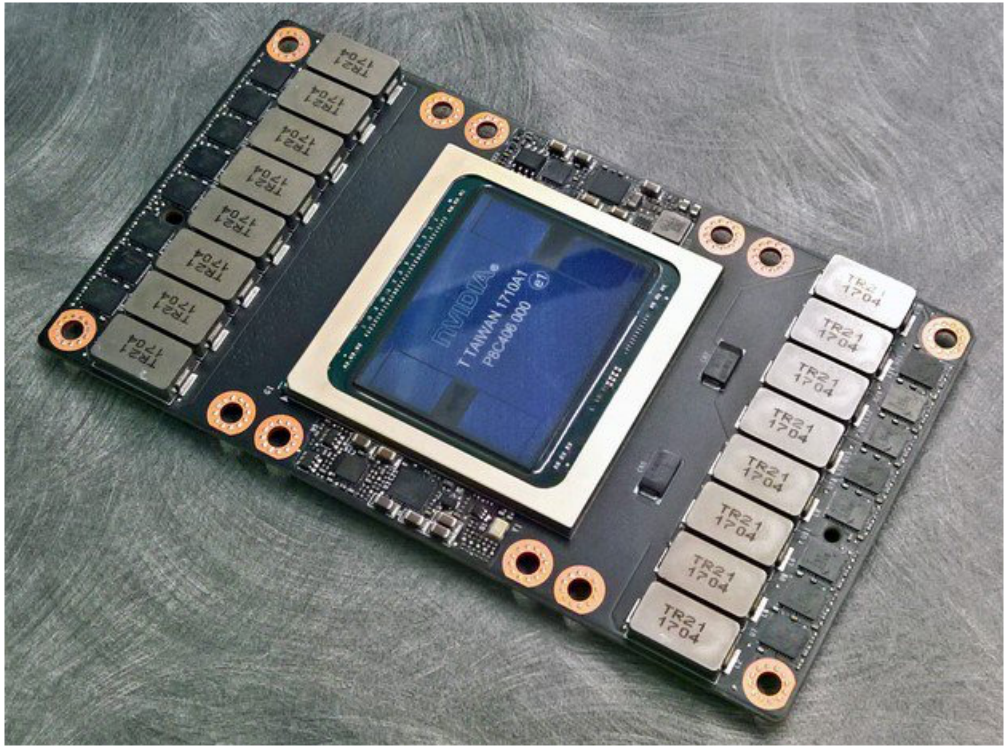
ACCELERATED COMPUTING

Find economically important problem
 that needs compute

Make hardware for it to take it to speed of light

GPUs are accelerators

AI is huge focus for our GPU



V100 GPU

TESLA V100

21B transistors
815 mm²

80 SM
5120 CUDA Cores
640 Tensor Cores

16 GB HBM2
900 GB/s HBM2
300 GB/s NVLink

@ctnzr

*full GV100 chip contains 84 SMs

GPU PERFORMANCE COMPARISON

	P100	V100	Ratio
Training acceleration	10 TOPS	120 TOPS	12x
Inference acceleration	21 TFLOPS	120 TOPS	6x
FP64/FP32	5/10 TFLOPS	7.5/15 TFLOPS	1.5x
HBM2 Bandwidth	720 GB/s	900 GB/s	1.2x
NVLink Bandwidth	160 GB/s	300 GB/s	1.9x
L2 Cache	4 MB	6 MB	1.5x
L1 Caches	1.3 MB	10 MB	7.7x

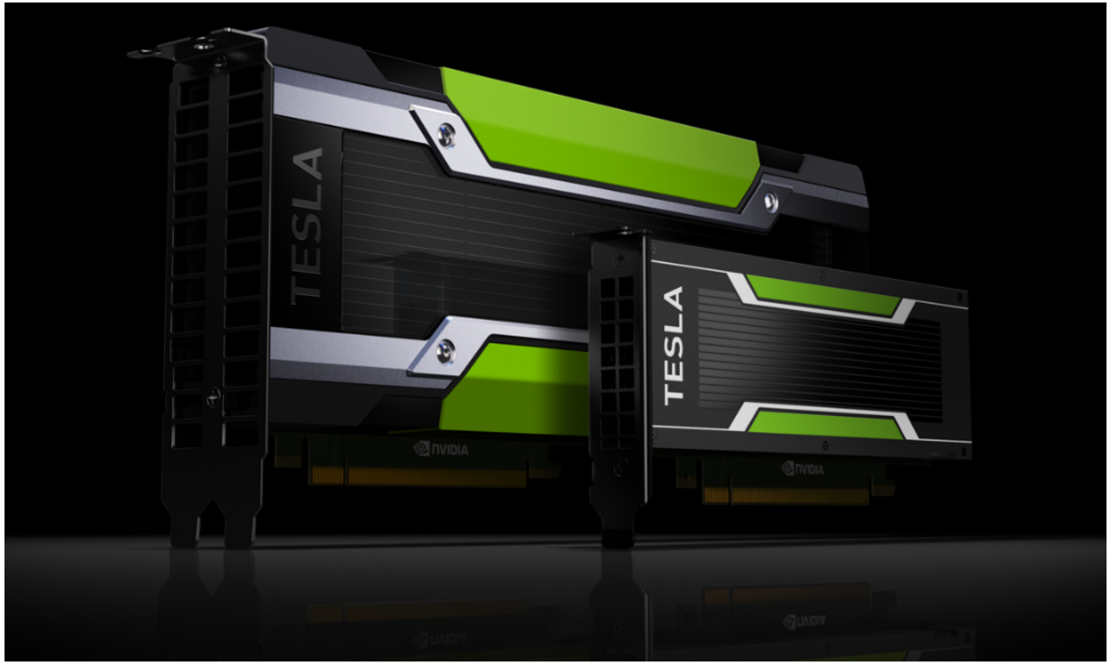
ARITHMETIC

Mixed precision for training

FP32 + FP16

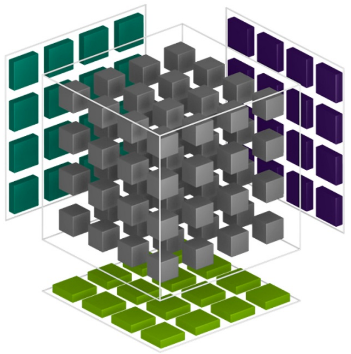
Lower precision integer for inference

Int8



TENSOR CORE

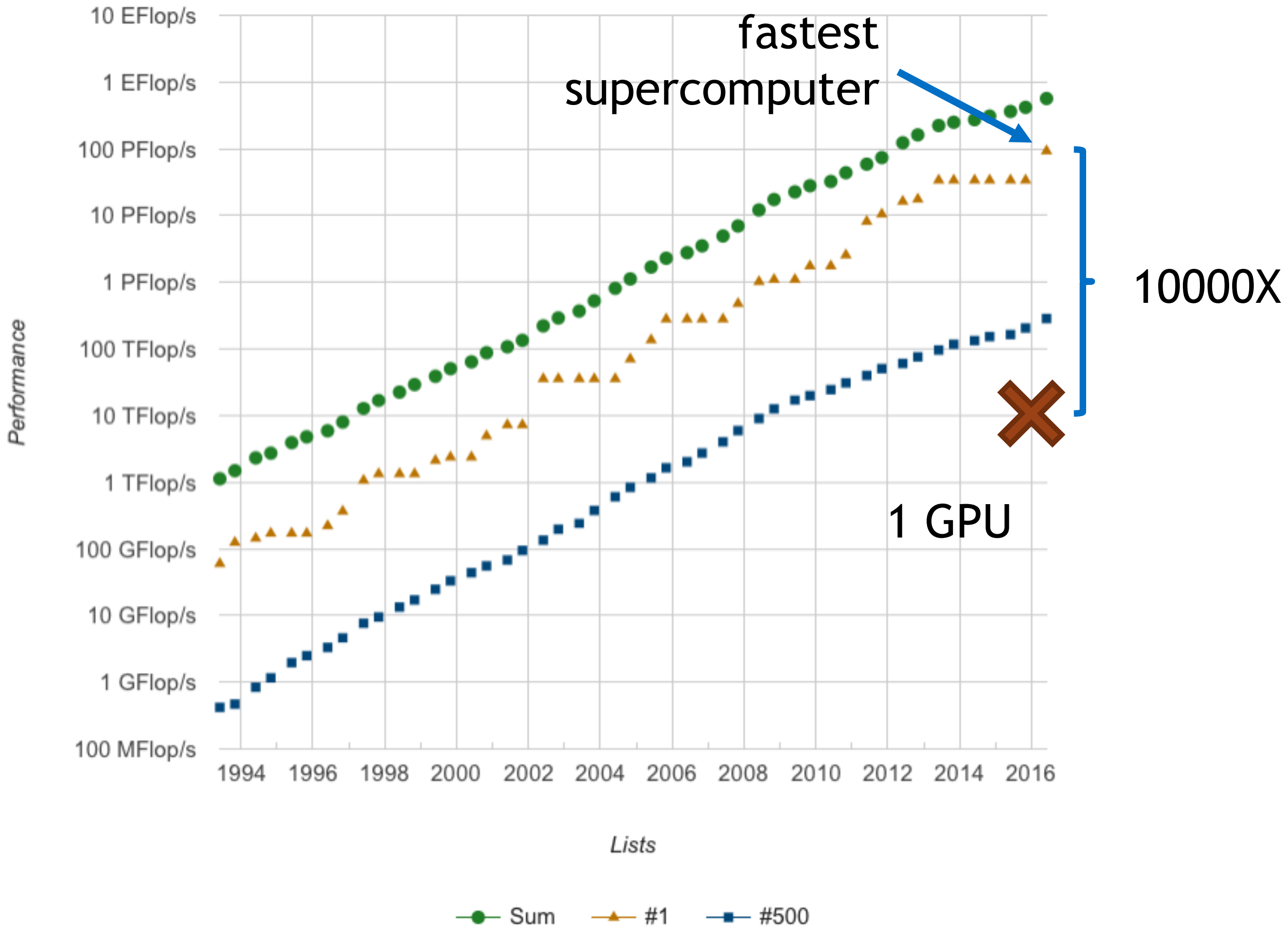
Mixed Precision Matrix Math
4x4 matrices



$$\underbrace{D}_{\text{FP16 or FP32}} = \underbrace{\begin{pmatrix} A_{0,0} & A_{0,1} & A_{0,2} & A_{0,3} \\ A_{1,0} & A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,0} & A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,0} & A_{3,1} & A_{3,2} & A_{3,3} \end{pmatrix}}_{\text{FP16}} \underbrace{\begin{pmatrix} B_{0,0} & B_{0,1} & B_{0,2} & B_{0,3} \\ B_{1,0} & B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,0} & B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,0} & B_{3,1} & B_{3,2} & B_{3,3} \end{pmatrix}}_{\text{FP16}} + \underbrace{\begin{pmatrix} C_{0,0} & C_{0,1} & C_{0,2} & C_{0,3} \\ C_{1,0} & C_{1,1} & C_{1,2} & C_{1,3} \\ C_{2,0} & C_{2,1} & C_{2,2} & C_{2,3} \\ C_{3,0} & C_{3,1} & C_{3,2} & C_{3,3} \end{pmatrix}}_{\text{FP16 or FP32}}$$

$$D = AB + C$$

SCALABILITY



Thesis: AI is most important problem

How can we use our best computers for it?

Current best practices use ~128 GPUs

Often people use 1-8

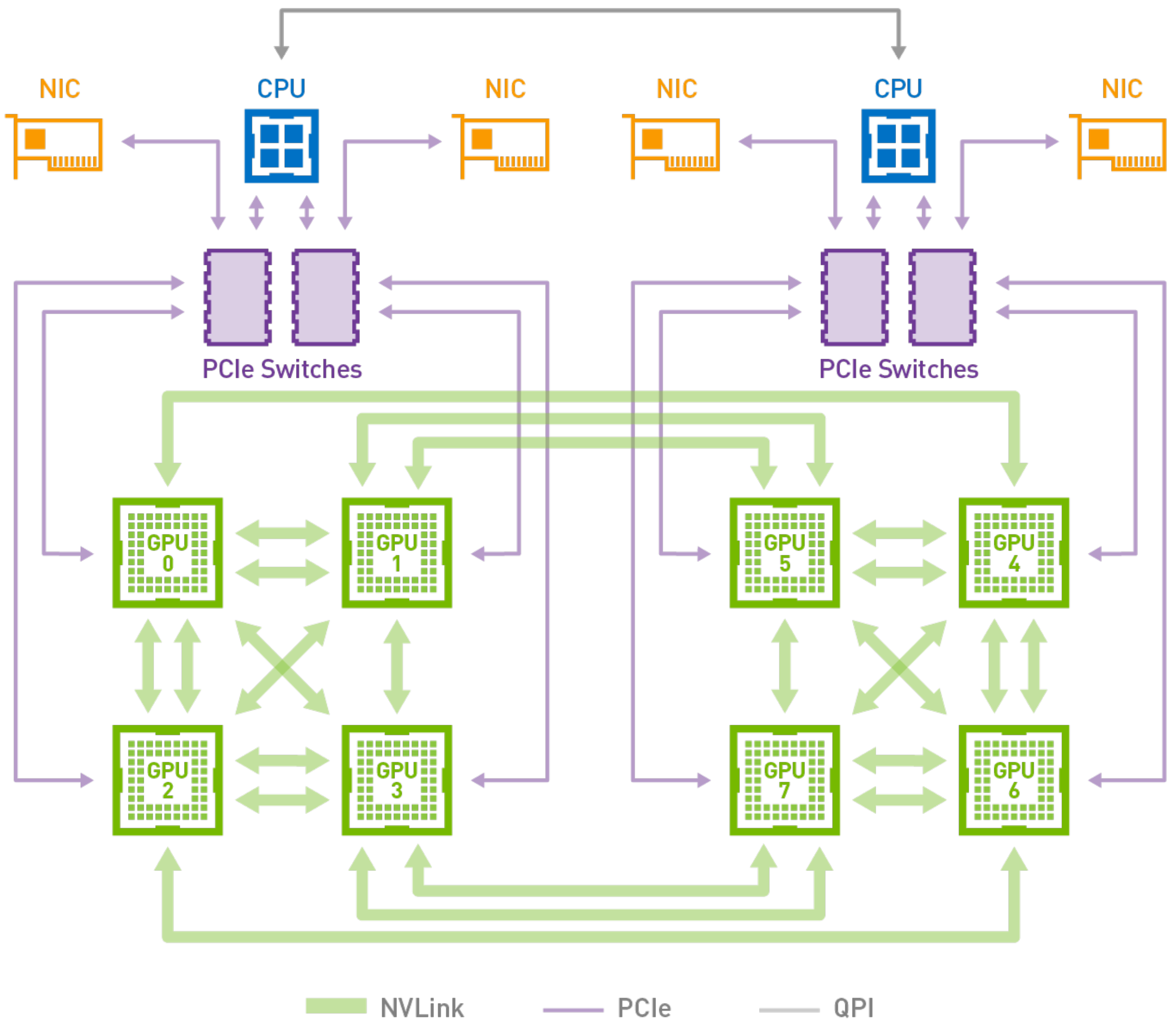
Research problem: how can we use 10000?

VOLTA NVLINK

300GB/sec

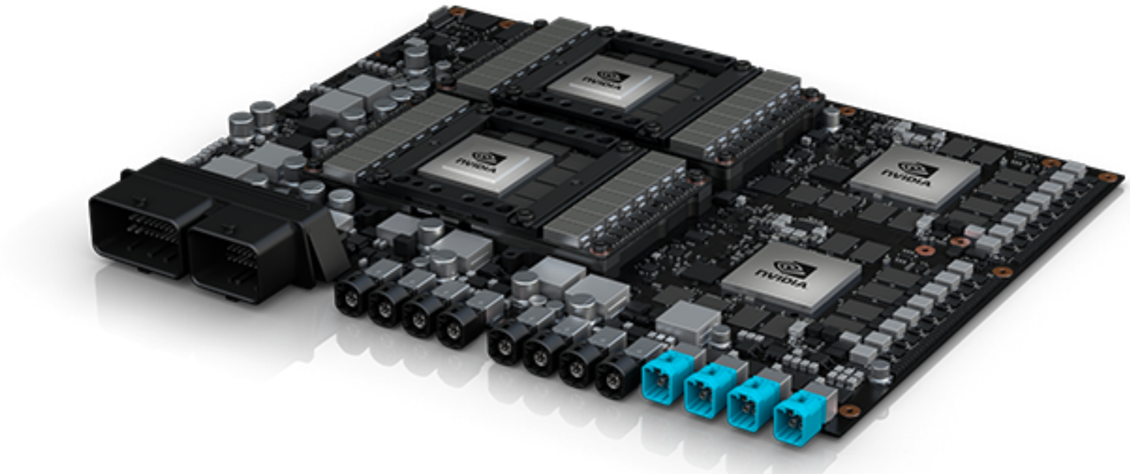
50% more links

28% faster signaling



HARDWARE PLATFORMS

Systems, not just GPUs



Drive PX Pegasus:

320 TOPS

For Self-Driving Cars

DGX:

960 TOPS, 8 TB SSD, 3.2 kW

128 GB HBM2, 7.2 TB/s Mem BW

512 GB DRAM, 4x EDR IB

TENSOR RT

Optimized Inference

Horizontal and vertical fusion

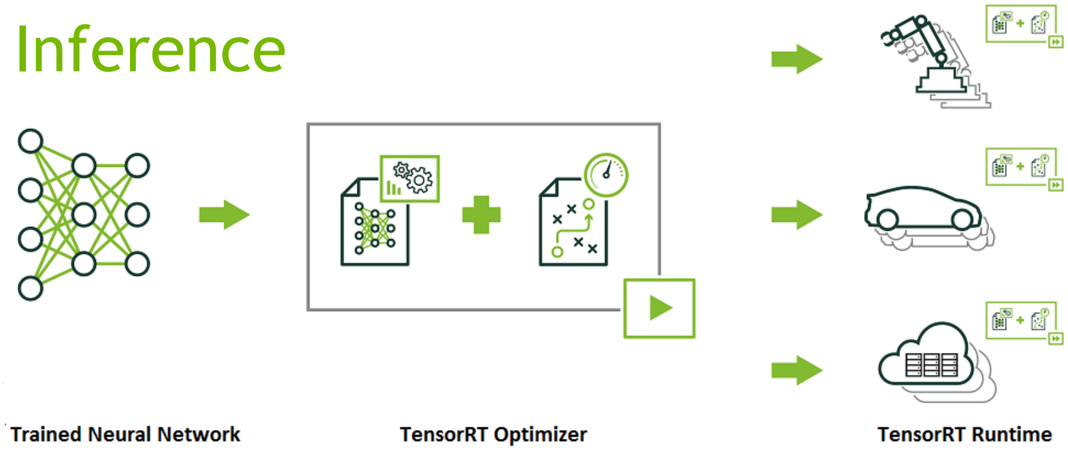
Saves memory bandwidth

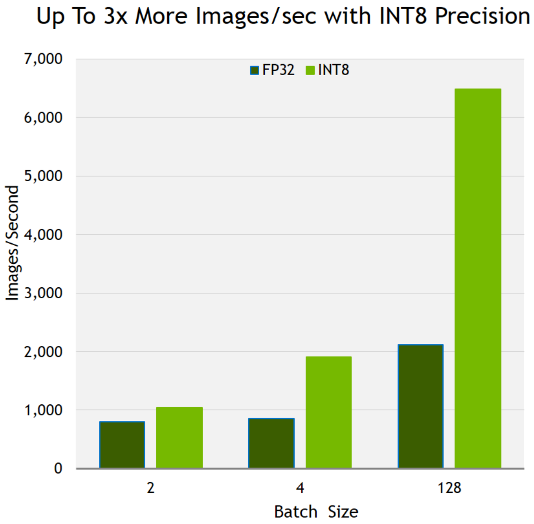
Low batch-size optimizations

Inference batch sizes are small

Int8 support

Helps choose scaling factors





GoogLenet, FP32 vs INT8 precision + TensorRT on
Tesla P40 GPU, 2 Socket Haswell E5-2698 v3@2.3GHz with HT off

ACCELERATED COMPUTING FOR AI

Tremendous excitement in systems for AI

Programmability & flexibility fundamental

High computational intensity also required

Make human ingenuity the limiting factor for AI research & deployment

Bryan Catanzaro
@ctnzr

NVIDIA®