Cloud Vision API and TensorFlow

+Kazunori Sato
@kazunori_279

Kaz Sato

Staff Developer Advocate,
Tech Lead for Data & Analytics
Cloud Platform, Google Inc.

Google Cloud Platform

= The Datacenter as a Computer


B2


Enterprise

Jupiter network

40 G ports

10 G x 100 K = 1 Pbps total

CLOS topology

Software Defined Network

Borg

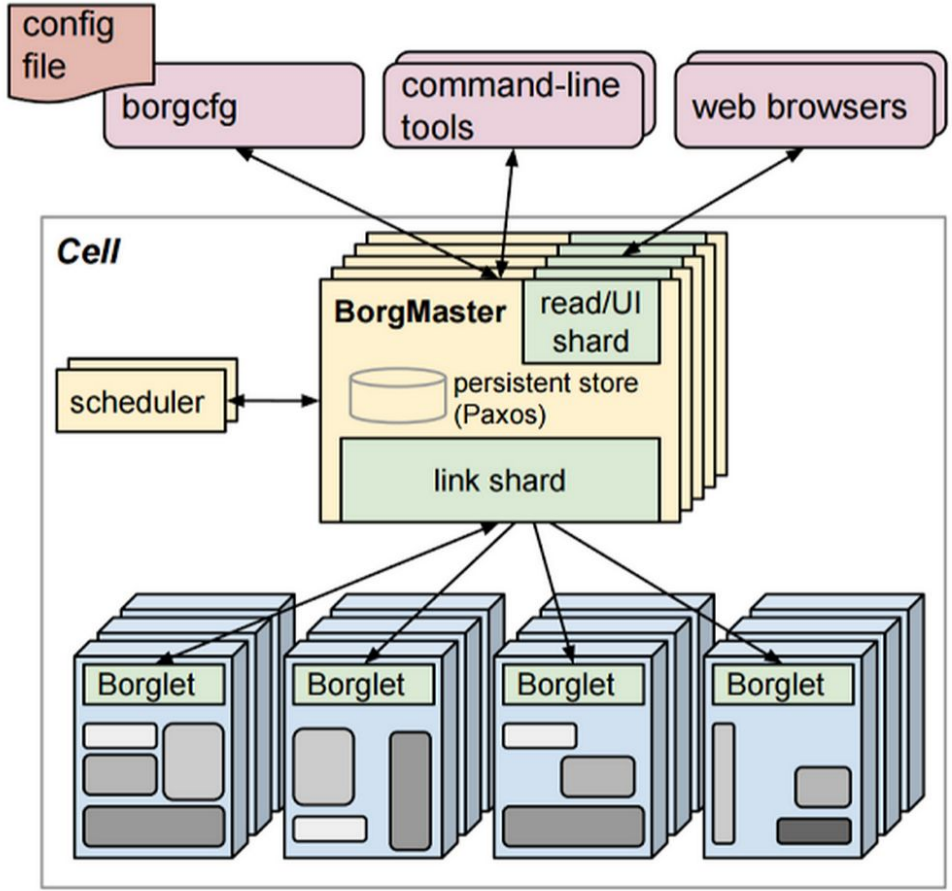
No VMs, pure containers

Manages 10K machines / Cell

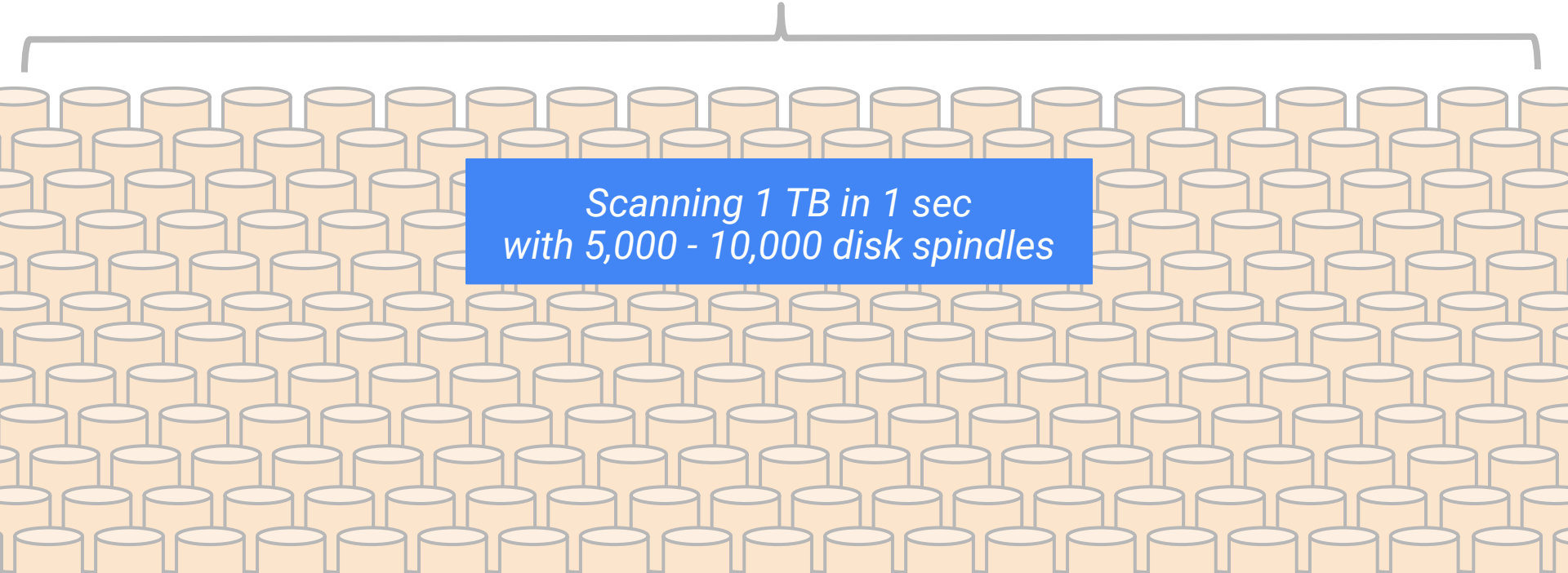
DC-scale *proactive* job sched

(CPU, mem, disk IO, TCP ports)

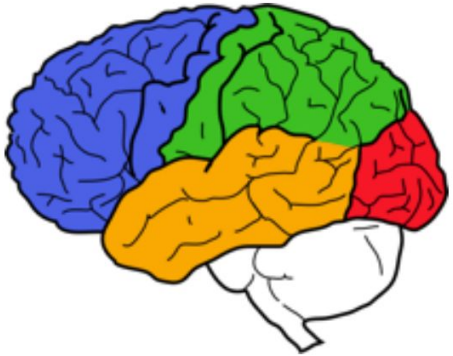
Paxos-based metadata store



```
SELECT your_data FROM billions_of_rows
 WHERE full_disk_scan_required = true;
```



Google Brain

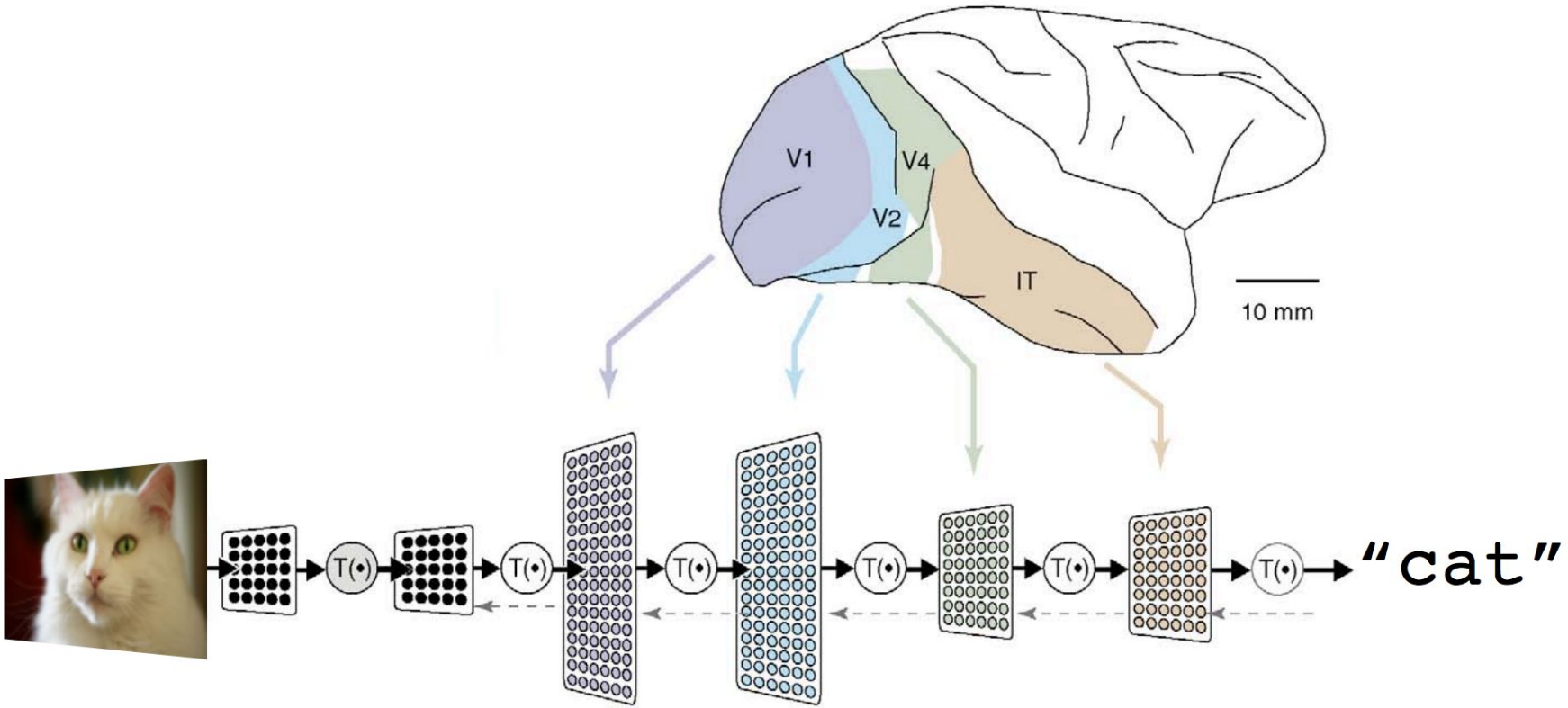


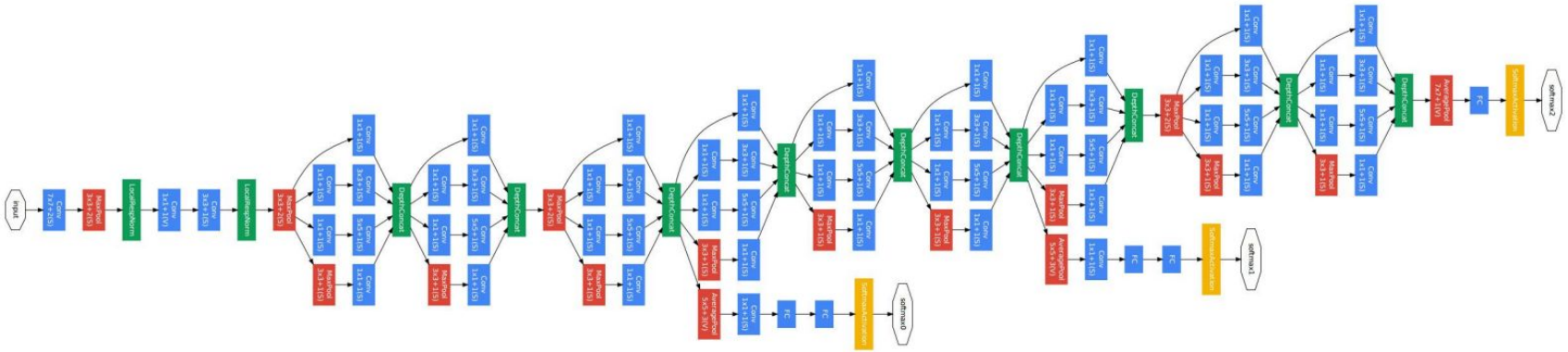
Large Scale Deep Learning

Jeff Dean

Google

Joint work with many colleagues at Google

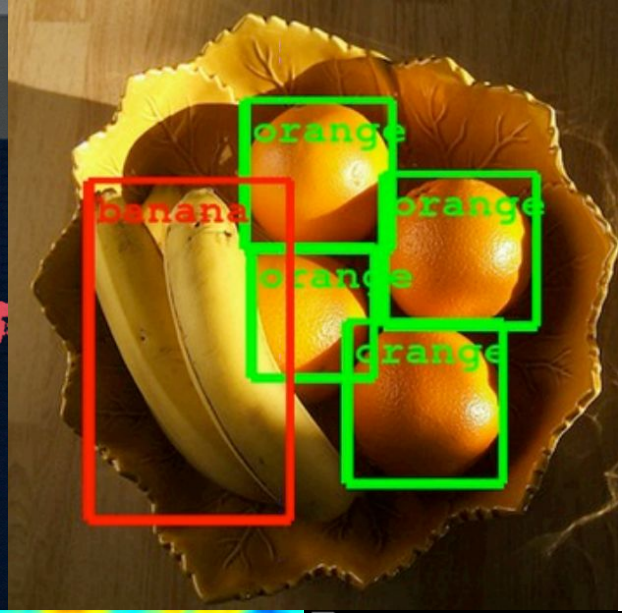

V1
V4
V2
IT
10 mm
T(•)
T(•)
T(•)
T(•)
T(•)
T(•)
"cat"

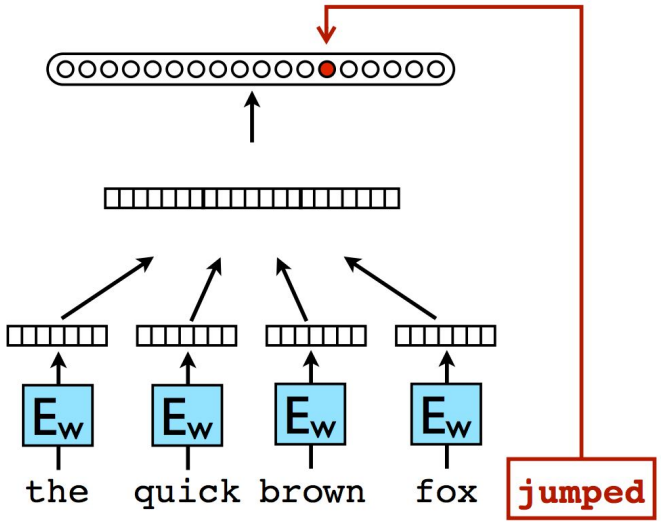


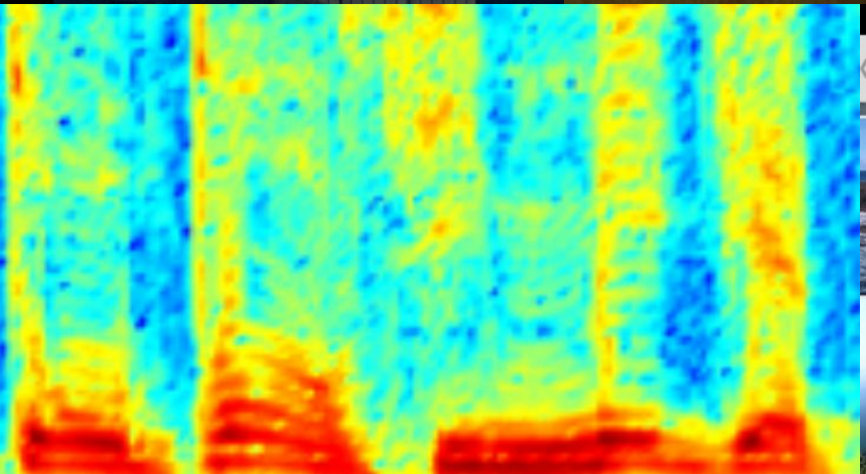
The Inception Architecture (GoogLeNet, 2015)

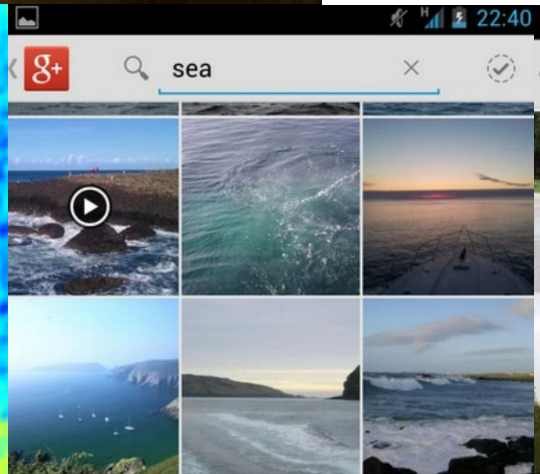

Neuron 1
Neuron 2
Neuron 3
Neuron 4
Neuron 5

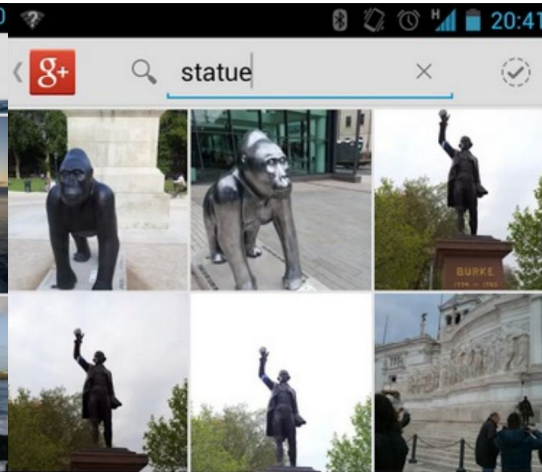

CIANO MOTOR ENGINEERS
MECHANICAL REPAIRS TO ALL MAKES AND MODELS
Specialising in BMW, MINI & TOYOTA
8 REGATTA ROAD FIVE DOCK 9745 3173


orange
banana
orange
orange
orange

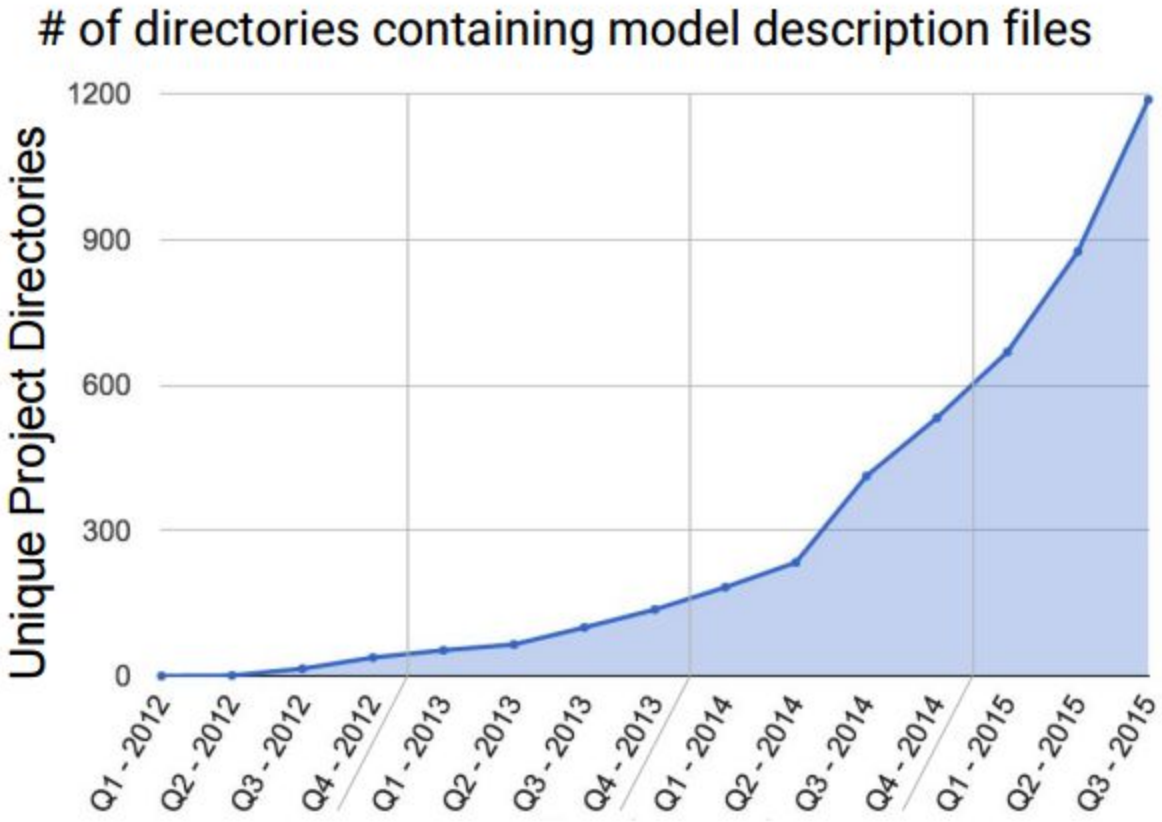

Ew
Ew
Ew
Ew
the
quick
brown
fox
jumped




22:40
sea


20:4
statue

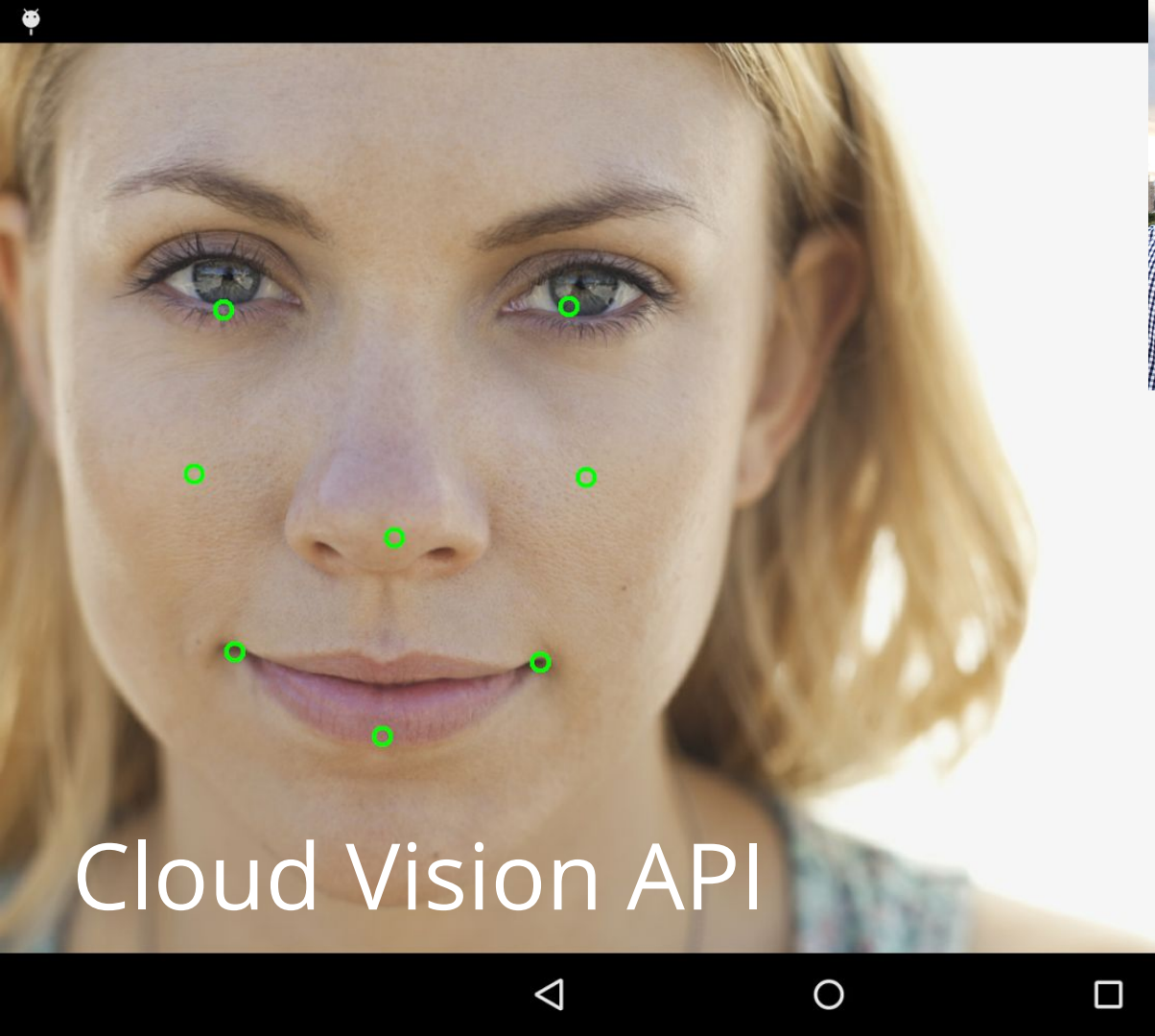
Growing Use of Deep Learning at Google

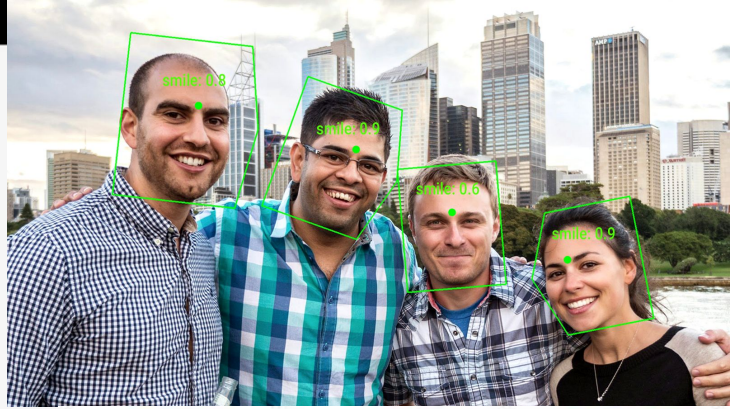


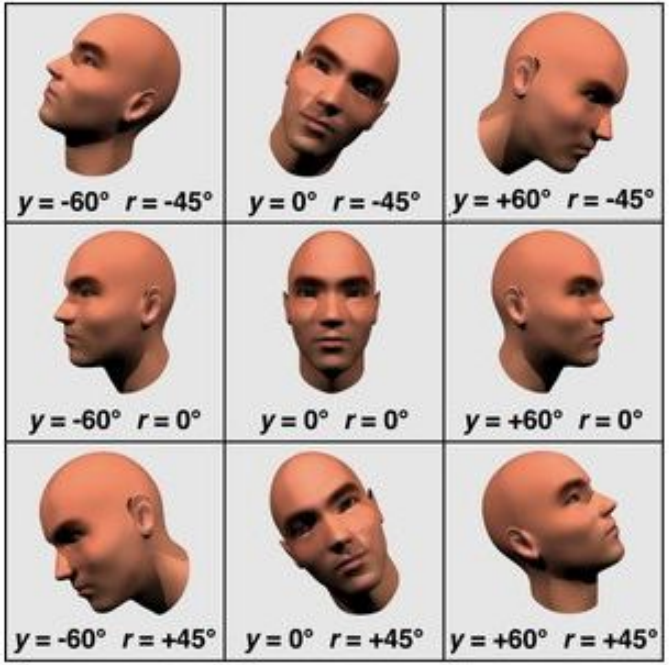
Across many products/areas:

Android
Apps
drug discovery
Gmail
Image understanding
Maps
Natural language understanding
Photos
Robotics research
Speech
Translation
YouTube
... many others ...

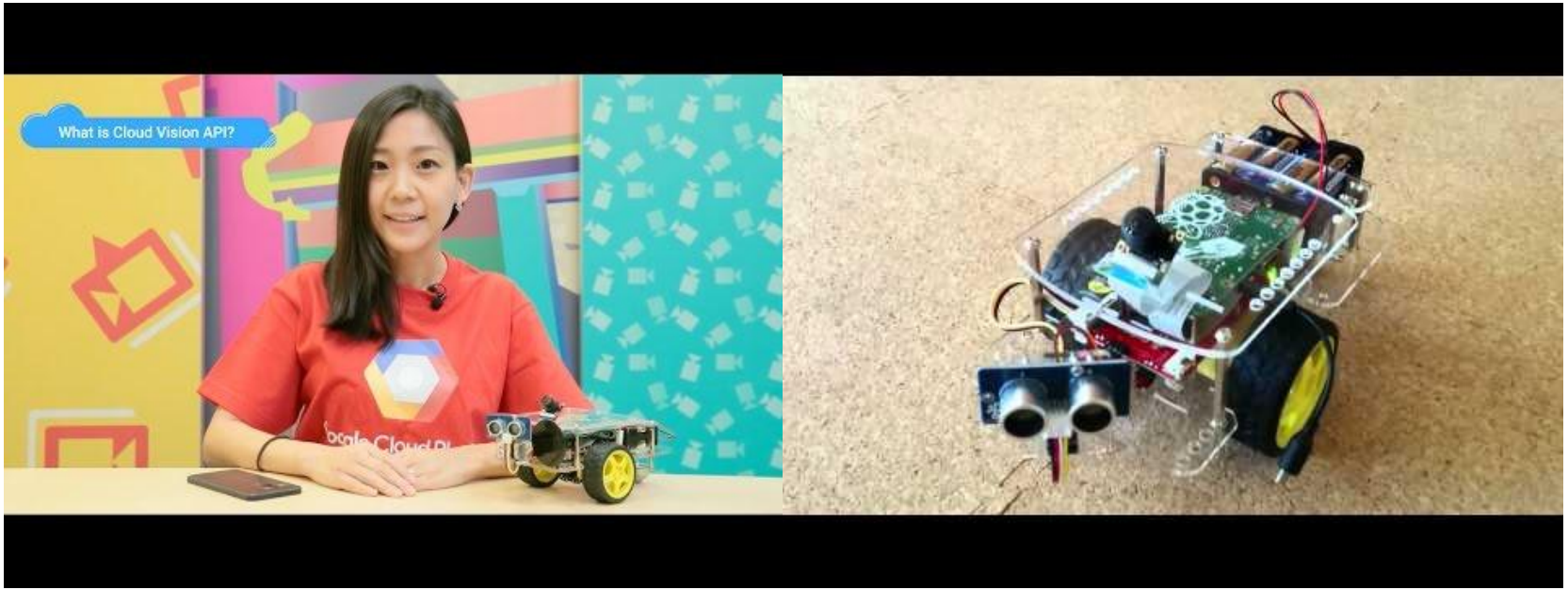
Cloud Vision API


Cloud Vision API


smile: 0.8
smile: 0.9
smile: 0.6
smile: 0.9

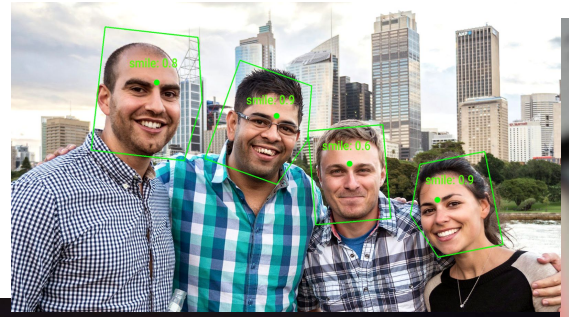

y = -60° r = -45°
y = 0° r = -45°
y = +60° r = -45°
y = -60° r = 0°
y = 0° r = 0°
y = +60° r = 0°
y = -60° r = +45°
y = 0° r = +45°
y = +60° r = +45°

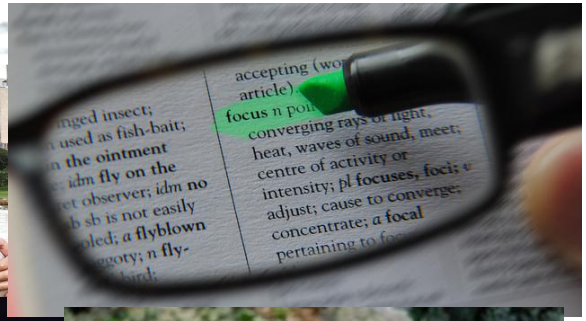
Demo Video

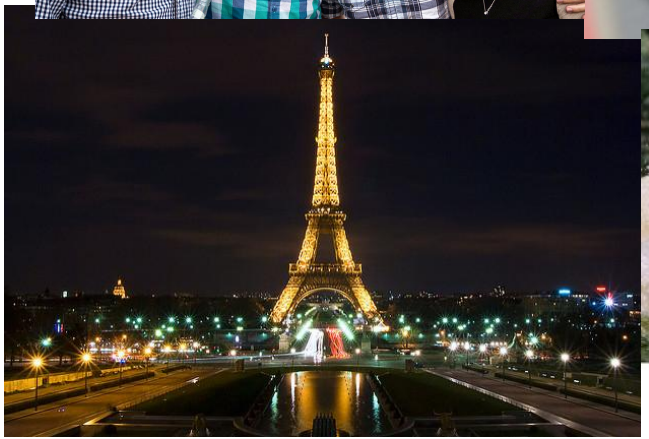


Types of Detection

- Label
- Landmark
- Logo
- Face
- Text
- Safe search







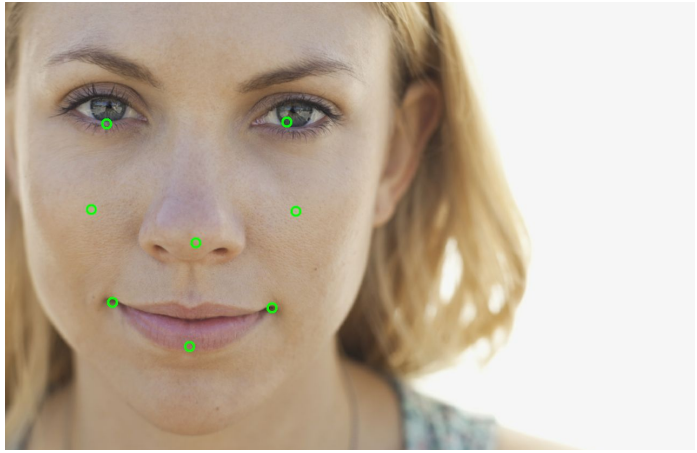
"rapid","score":0.88886356,
"canoe slalom","score":0.88697785
"kayak","score":0.86466473

@SRobTweets

Types of Detection

Face Detection

- Find multiple faces
- Location of eyes, nose, mouth
- Detect emotions: joy, anger, surprise, sorrow



Entity Detection

- Find common objects and landmarks, and their location in the image
- Detect explicit content

TensorFlow

What is TensorFlow?

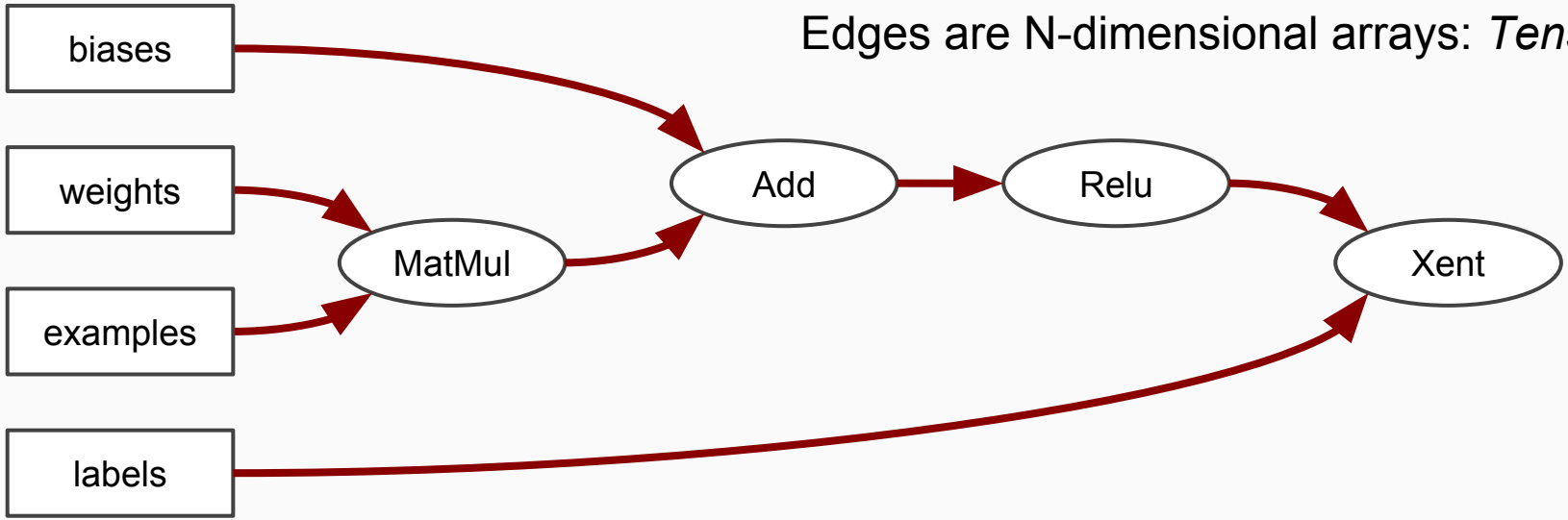
Google's **open source** library for machine intelligence

- **tensorflow.org** launched in Nov 2015
- The second generation (after DistBelief)
- Used by many production ML projects at Google

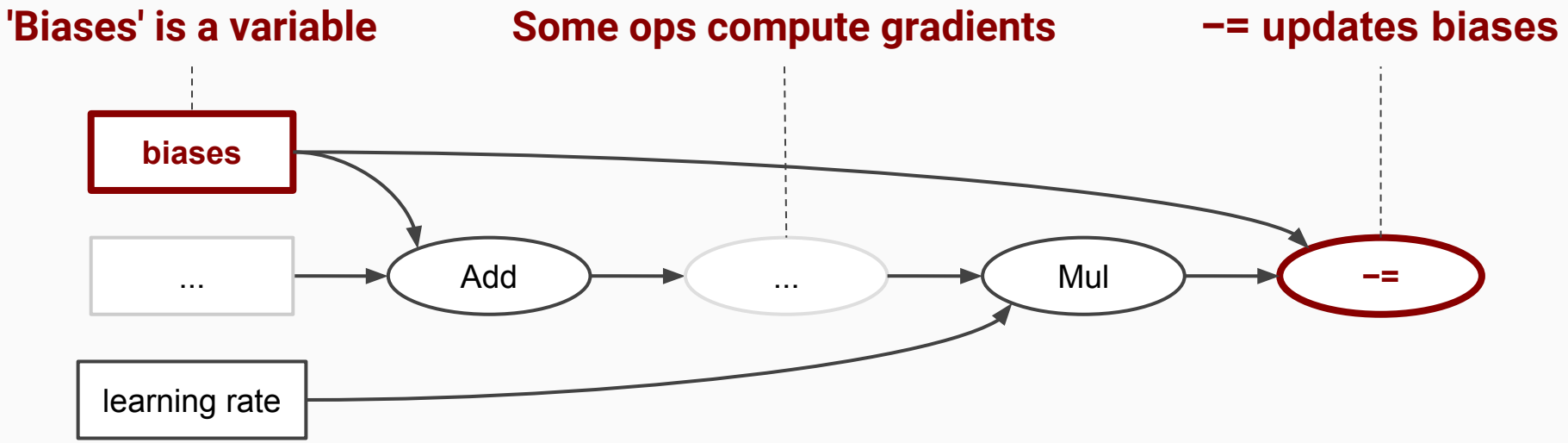
What is TensorFlow?

- **Tensor**: N-dimensional array
  - Vector: 1 dimension
  - Matrix: 2 dimensions
- **Flow**: data flow computation framework (like MapReduce)
- **TensorFlow**: a data flow based numerical computation framework
  - Best suited for Machine Learning and Deep Learning
  - Or any other **HPC** (High Performance Computing) applications

Yet another dataflow system with tensors

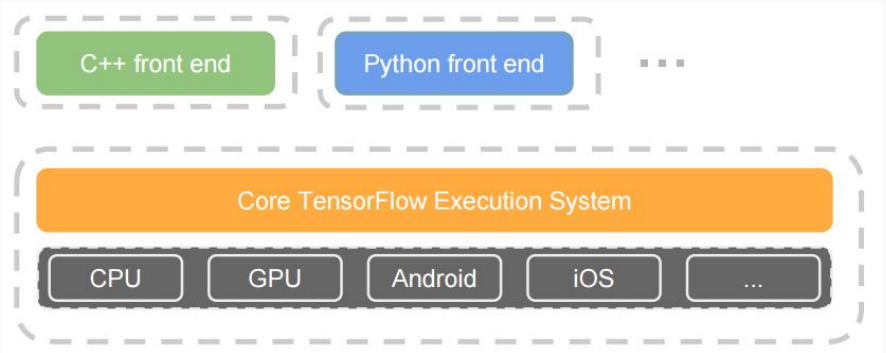


Yet another dataflow system with state

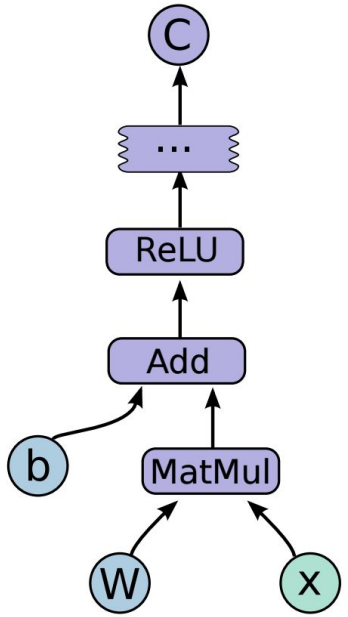


Portable

- Training on:
  - Data Center
  - CPUs, GPUs and etc
- Running on:
  - Mobile phones
  - IoT devices

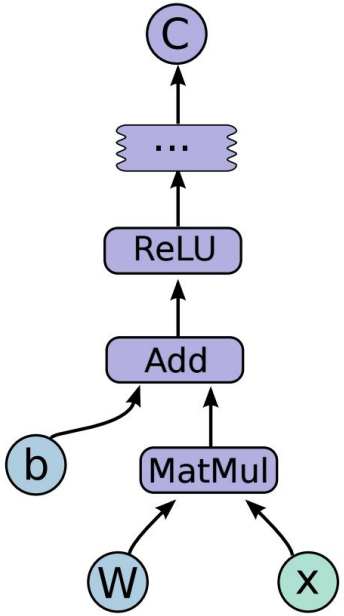


Simple Example




```
# define the network
import tensorflow as tf
x = tf.placeholder(tf.float32, [None, 784])
W = tf.Variable(tf.zeros([784, 10]))
b = tf.Variable(tf.zeros([10]))
y = tf.nn.softmax(tf.matmul(x, W) + b)

# define a training step
y_ = tf.placeholder(tf.float32, [None, 10])
xent = -tf.reduce_sum(y_*tf.log(y))
step = tf.train.GradientDescentOptimizer(0.01).minimize(xent)
```

Simple Example



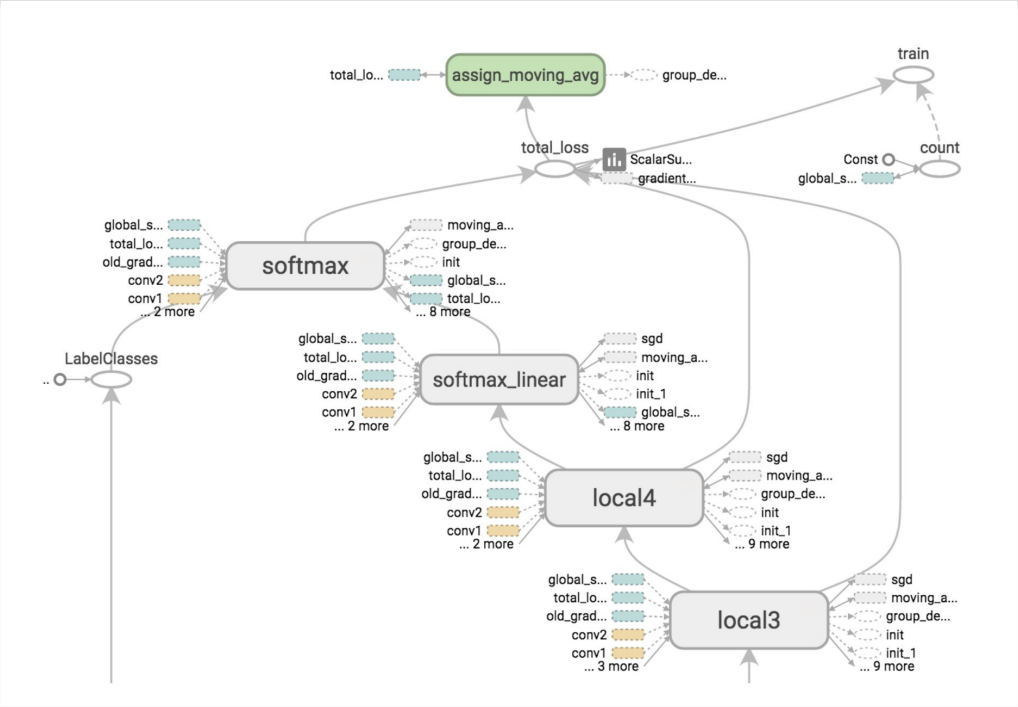

```
# initialize session
init = tf.initialize_all_variables()
sess = tf.Session()
sess.run(init)

# training
for i in range(1000):
  batch_xs, batch_ys = mnist.train.next_batch(100)
  sess.run(step, feed_dict={x: batch_xs, y_: batch_ys})
```

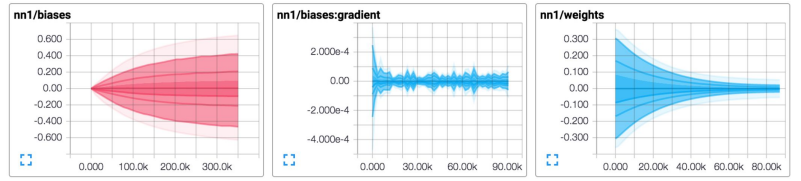
Operations, plenty of them

Category	Examples
Element-wise mathematical operations	Add, Sub, Mul, Div, Exp, Log, Greater, Less, Equal, ...
Array operations	Concat, Slice, Split, Constant, Rank, Shape, Shuffle, ...
Matrix operations	MatMul, MatrixInverse, MatrixDeterminant, ...
Stateful operations	Variable, Assign, AssignAdd, ...
Neural-net building blocks	SoftMax, Sigmoid, ReLU, Convolution2D, MaxPool, ...
Checkpointing operations	Save, Restore
Queue and synchronization operations	Enqueue, Dequeue, MutexAcquire, MutexRelease, ...
Control flow operations	Merge, Switch, Enter, Leave, NextIteration

Table 1: Example TensorFlow operation types

TensorBoard: visualization tool

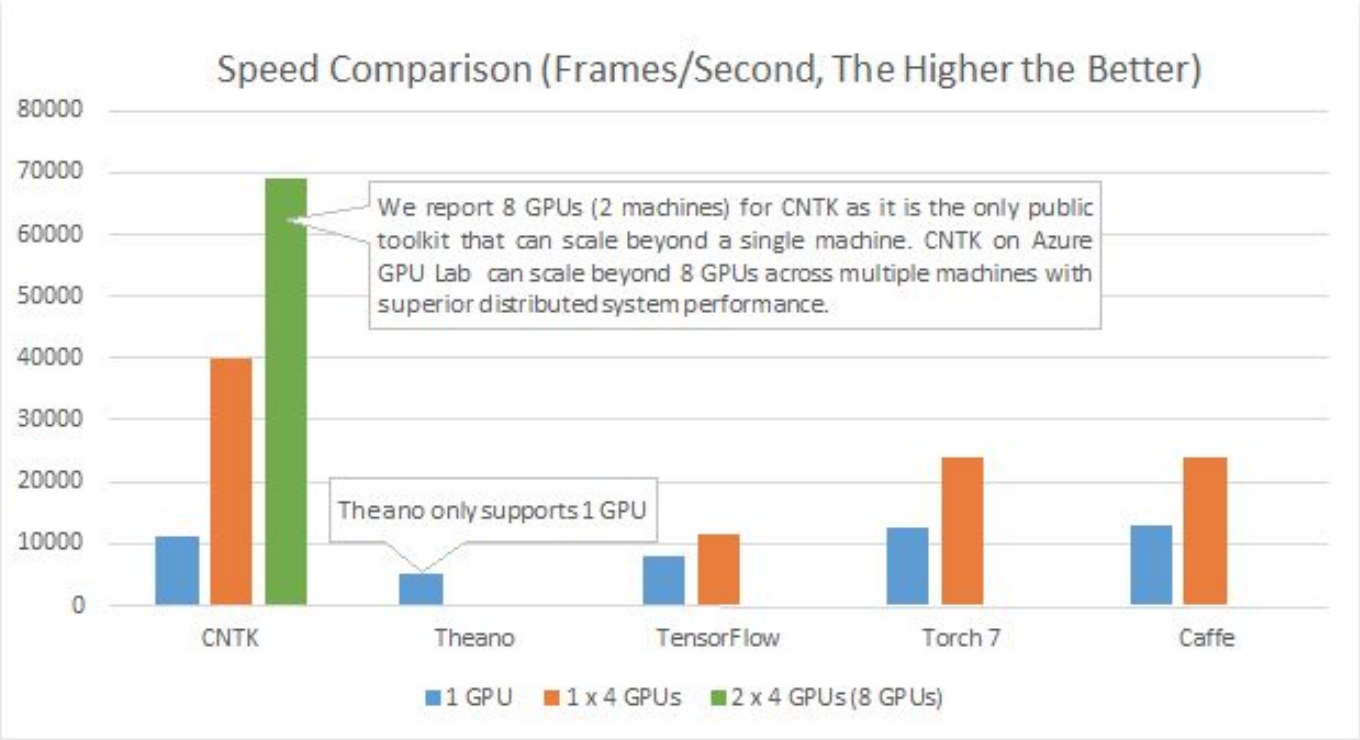




Distributed Training with TensorFlow

Single GPU server
for production service?

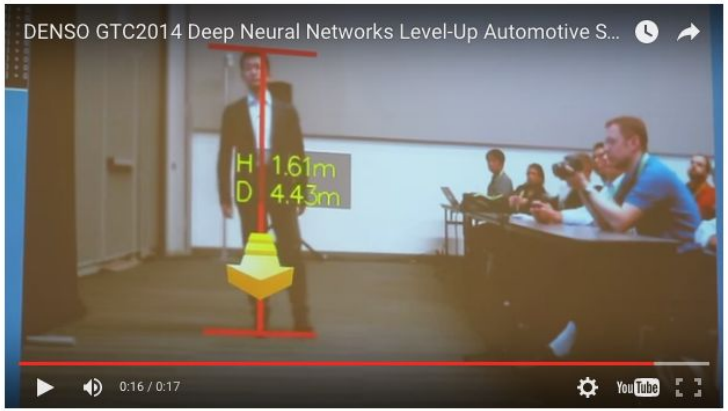
Microsoft: CNTK benchmark with 8 GPUs



From: [Microsoft Research Blog]

Denso IT Lab:

- TIT TSUBAME2 supercomputer with **96 GPUs**
- Perf gain: **dozens** of times



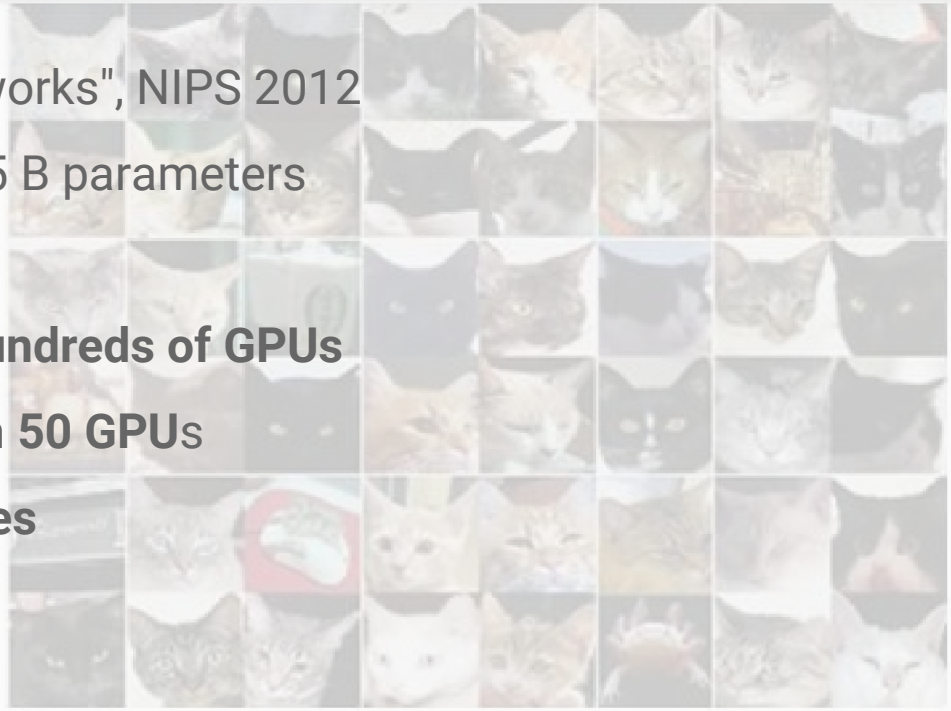
From: DENSO GTC2014 Deep Neural Networks Level-Up Automotive Safety

Preferred Networks + Sakura:

- Distributed GPU cluster with InfiniBand for **Chainer**
- In summer, 2016

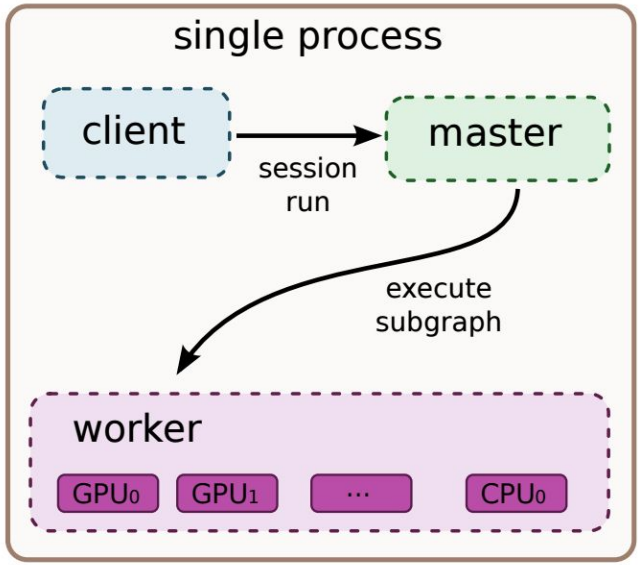
From: http://www.titech.ac.jp/news/2013/022156.html

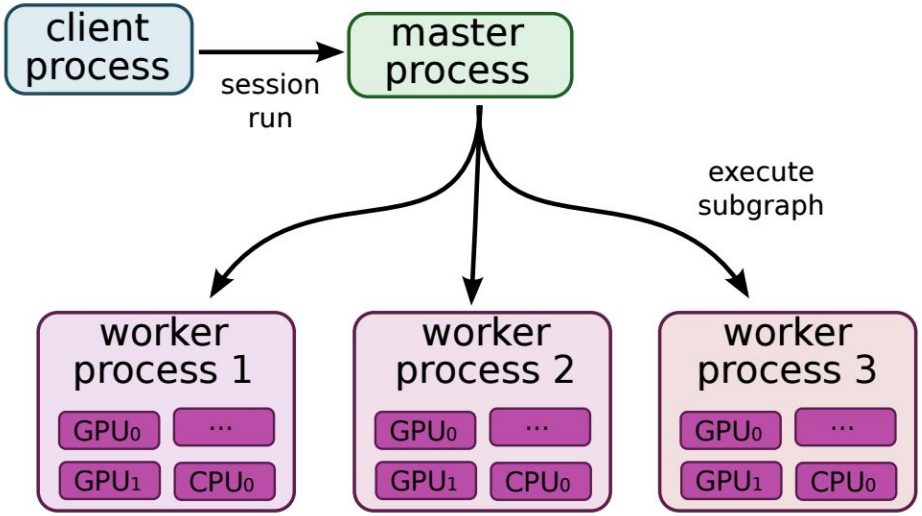
Google Brain:
Embarrassingly parallel for many years



- "Large Scale Distributed Deep Networks", NIPS 2012
  - 10 M images on YouTube, 1.15 B parameters
  - **16 K CPU cores** for 1 week
- **Distributed TensorFlow**: runs on **hundreds of GPUs**
  - Inception / ImageNet: **40x** with **50 GPU**s
  - RankBrain: **300x** with **500 nodes**

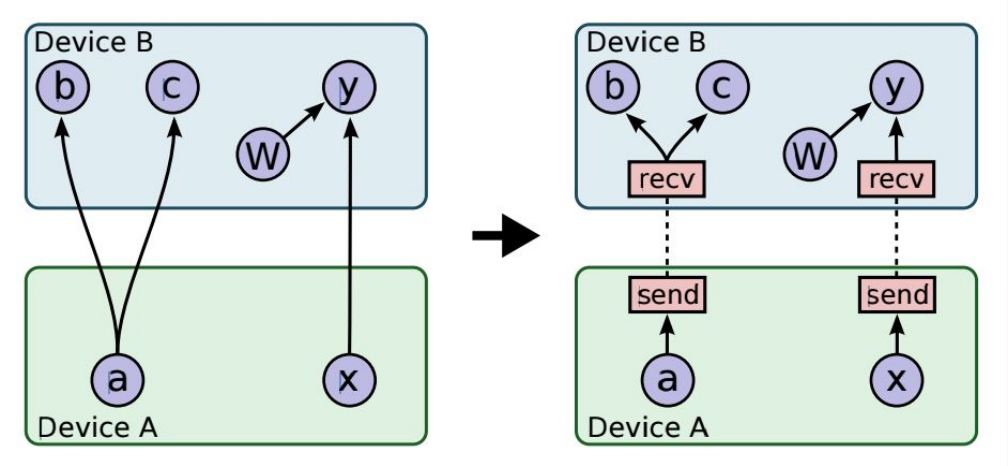
Distributed TensorFlow





Distributed TensorFlow

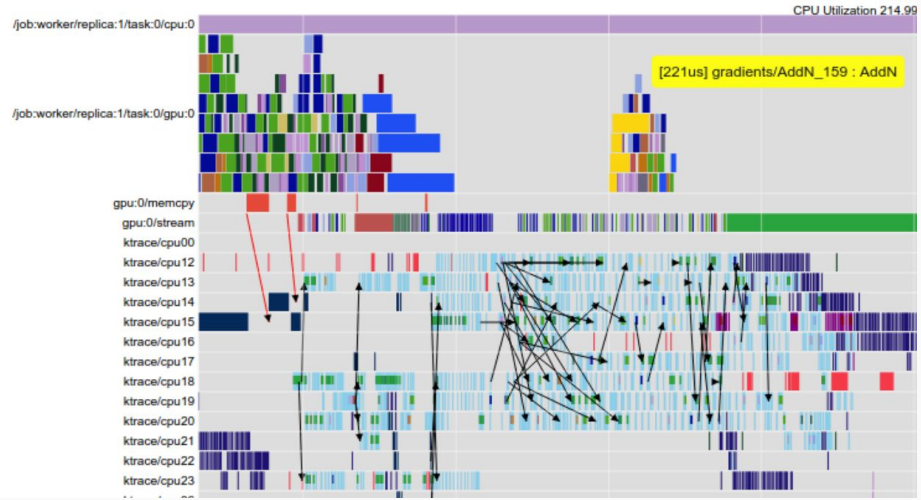
- CPU/GPU scheduling
- Communications
  - Local, RPC, RDMA
  - 32/16/8 bit quantization
- Cost-based optimization
- Fault tolerance



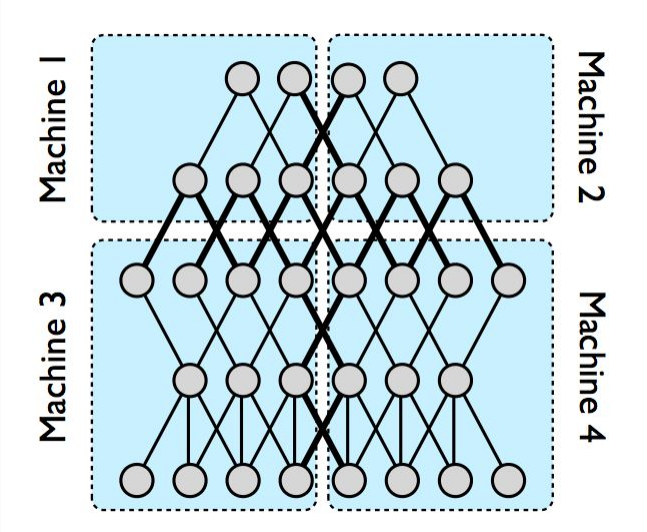
Distributed TensorFlow

- Fully managed
  - **No major changes** required
  - Automatic optimization
- with Device Constraints
  - hints for better optimization

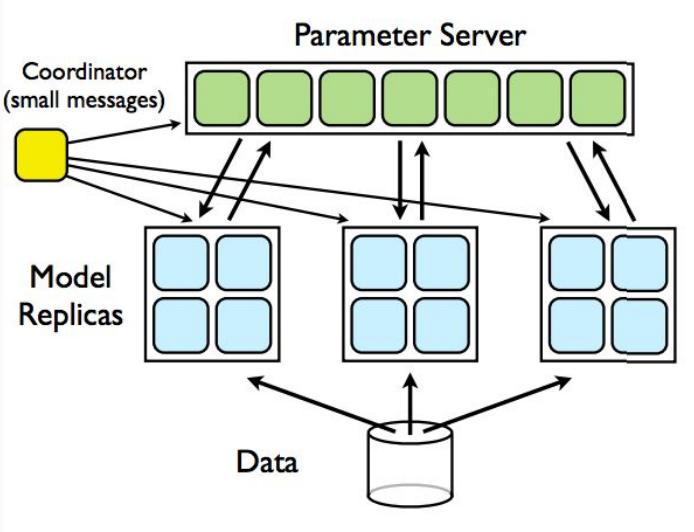
```
/job:localhost/device:cpu:0
/job:worker/task:17/device:gpu:3
/job:parameters/task:4/device:cpu:0
```



Model Parallelism vs Data Parallelism



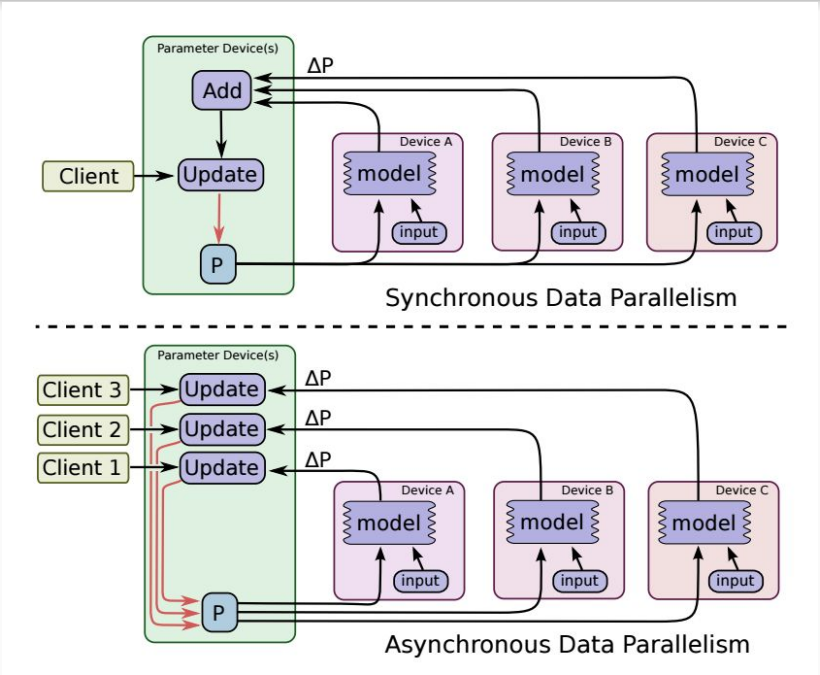
Model Parallelism
(split parameters, share training data)



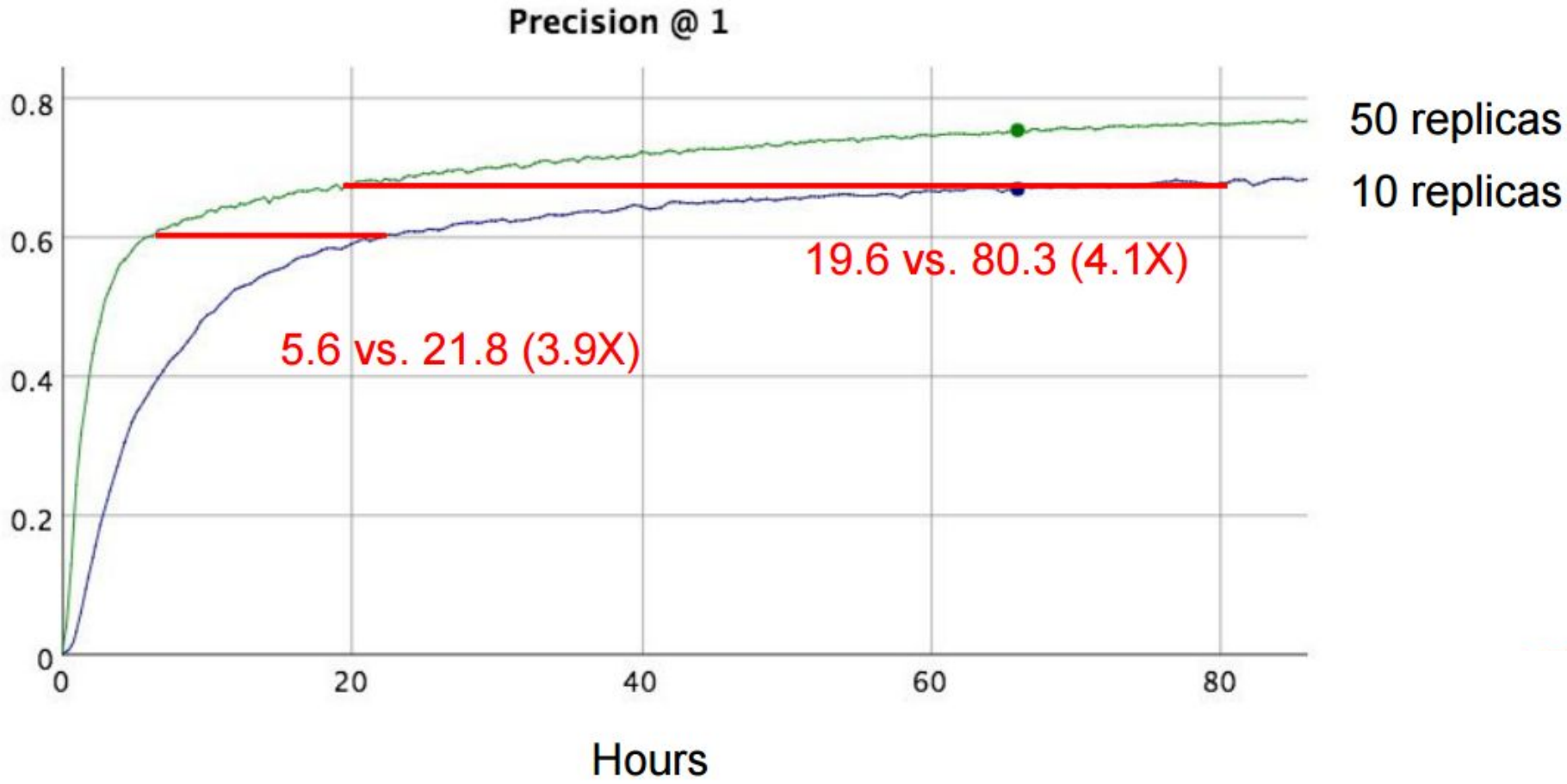
Data Parallelism
(split training data, share parameters)

Data Parallelism

- Google uses Data Parallelism mostly
  - Dense: **10 - 40x** with 50 replicas
  - Sparse: 1 K+ replicas
- Synchronous vs Asynchronous
  - Sync: better **gradient effectiveness**
  - Async: better **fault tolerance**



10 vs 50 Replica Inception Synchronous Training

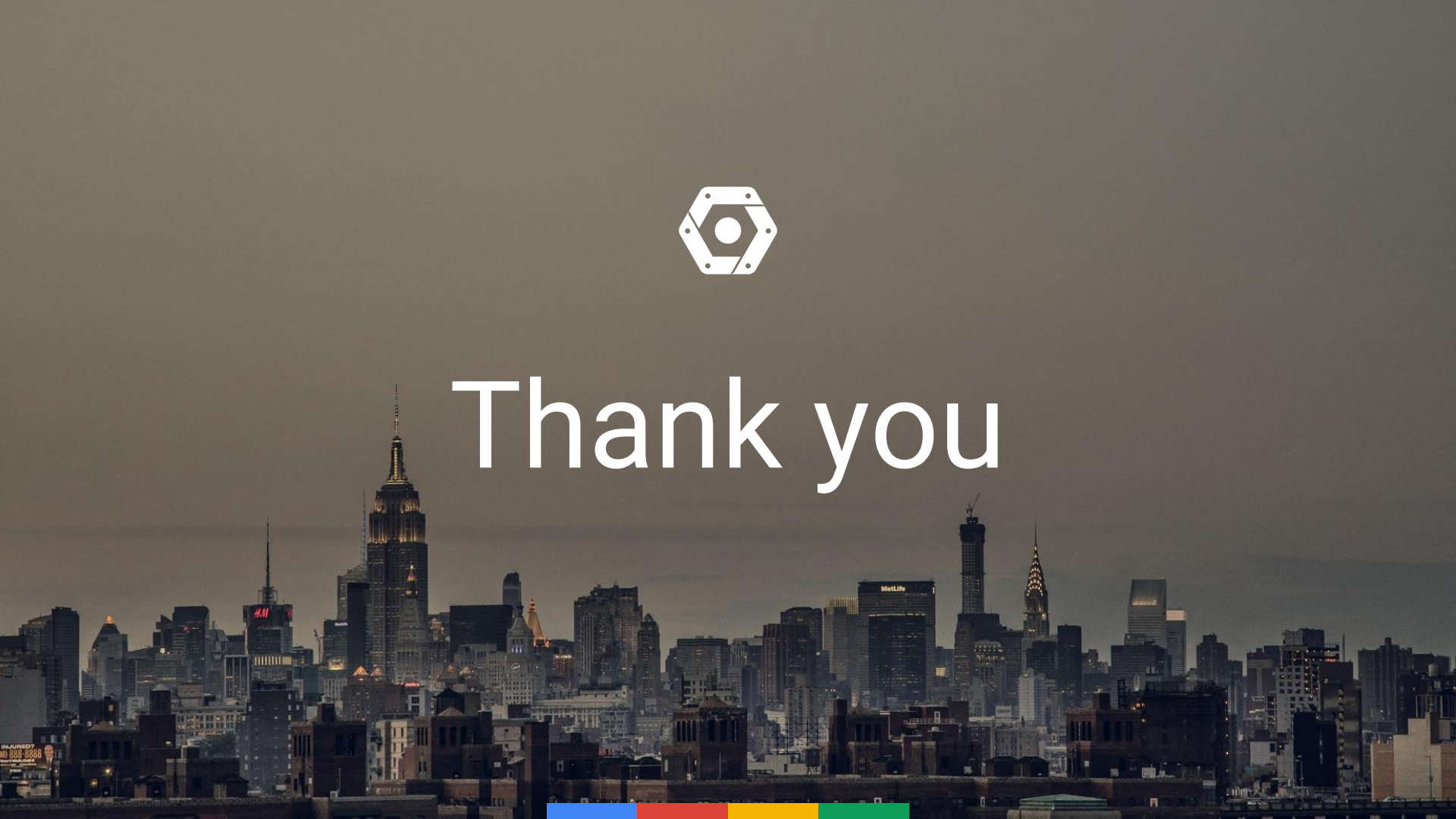


Summary

- Cloud Vision API
  - Easy and powerful API for utilizing Google's latest vision recognition
- TensorFlow
  - Portable: Works from data center machines to phones
  - Distributed and Proven: scales to **hundreds of GPUs** in production
    - will be available soon!

Resources

- tensorflow.org
- TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems, Jeff Dean et al, tensorflow.org, 2015
- Large Scale Distributed Systems for Training Neural Networks, Jeff Dean and Oriol Vinyals, NIPS 2015
- Large Scale Distributed Large Networks, Jeff Dean et al, NIPS 2012


Thank you