




LOGICAL CLOCKS

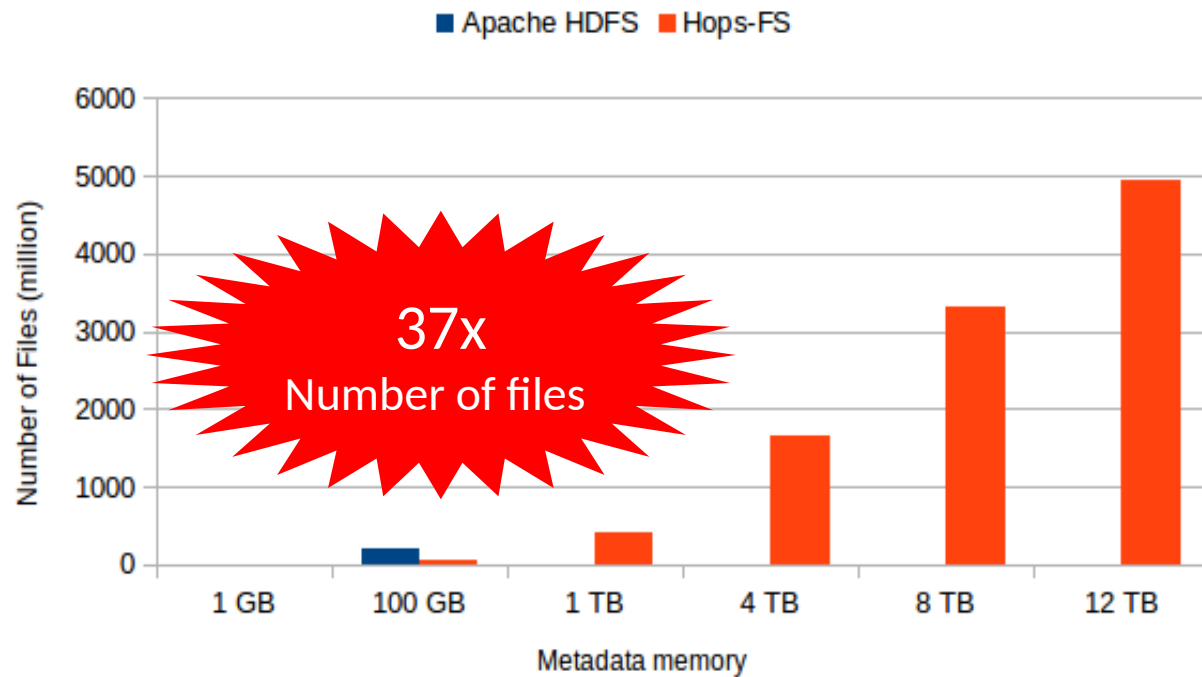
TensorFlow-on-Hops: Hotdog or Not with Toppings

 jim_dowling

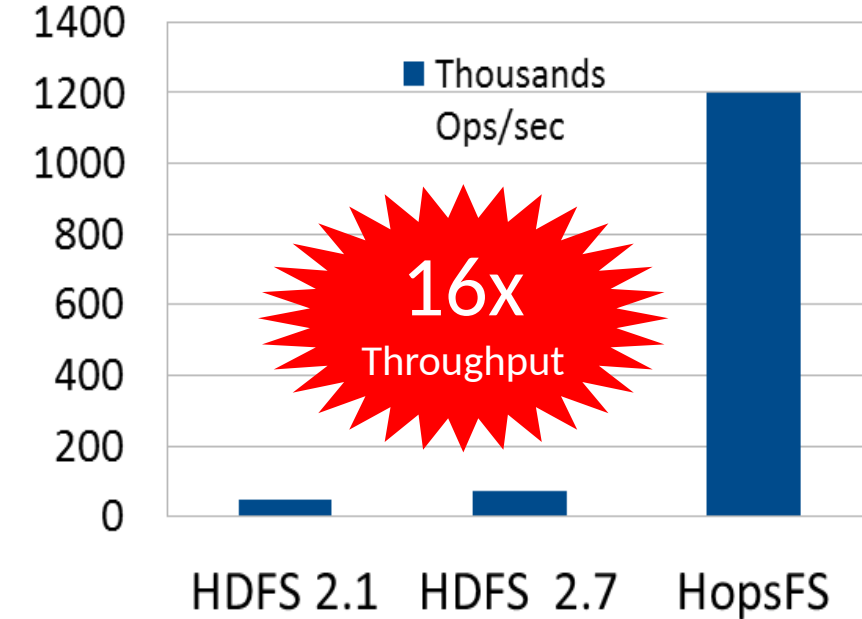
Register an account at:

<http://hops.io/tf>

World's Fastest/Biggest Hadoop*



Bigger



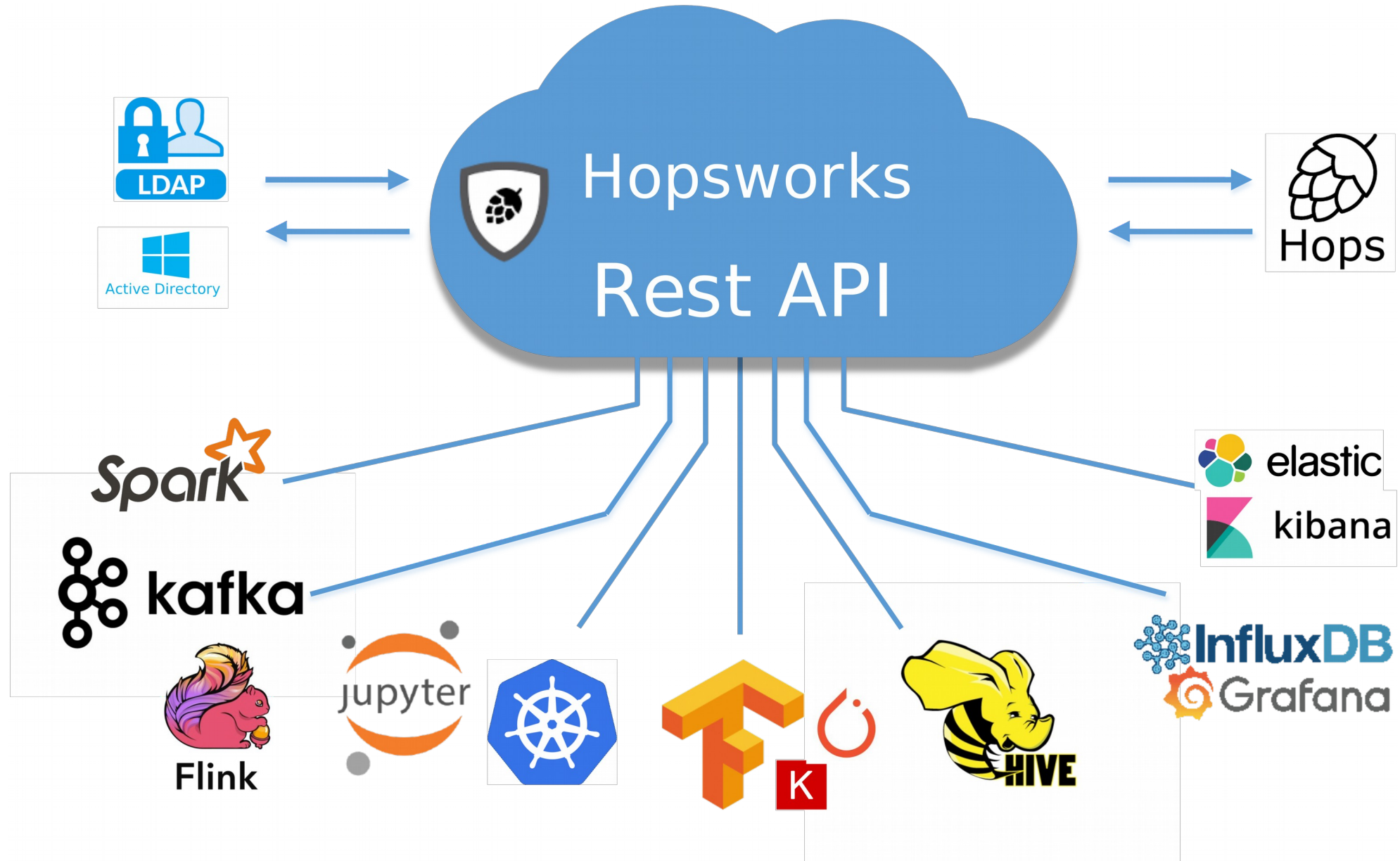
Faster – 1.2M ops/s

 **IEEE** Scale Challenge Winner (2017)

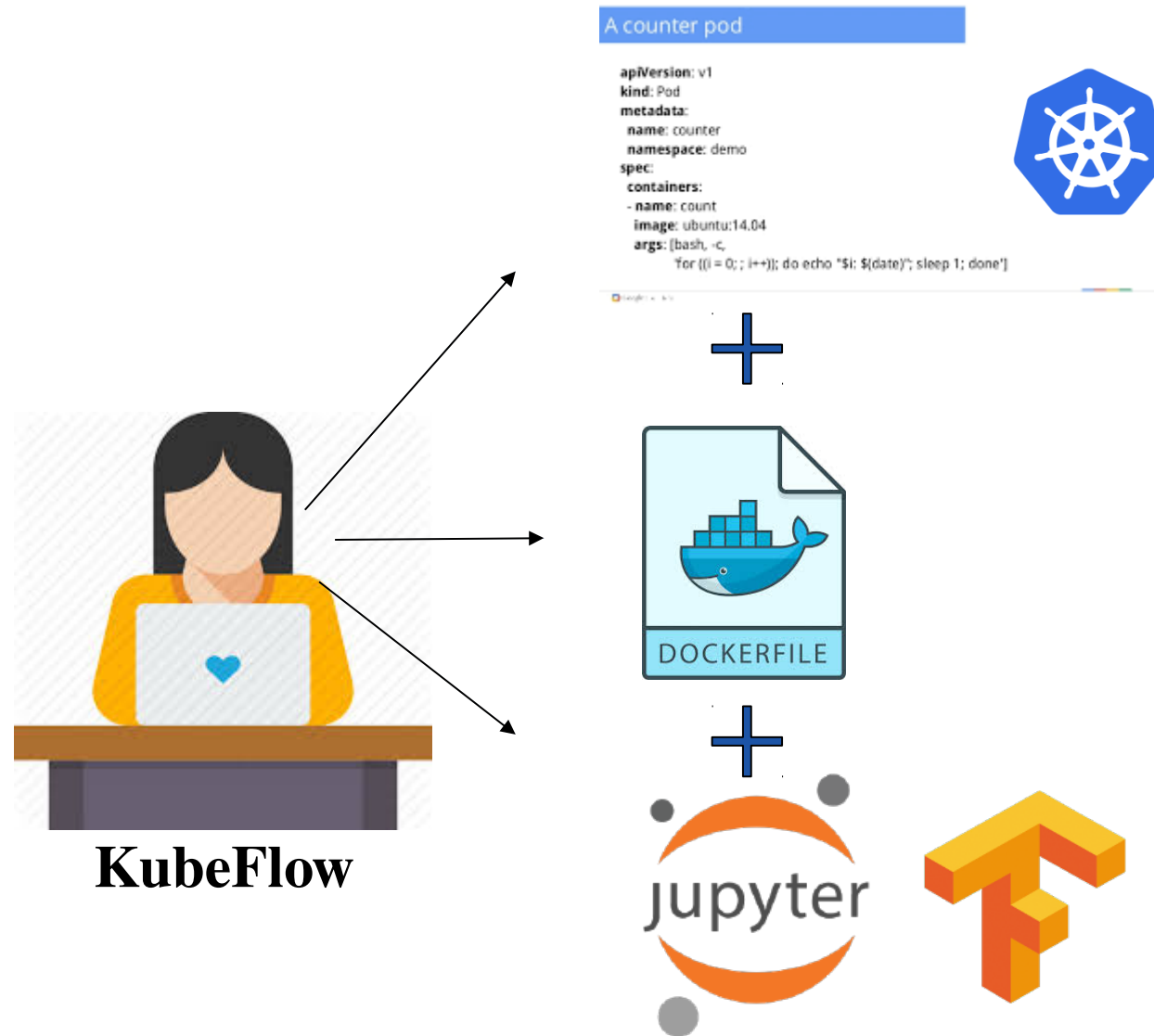


*<https://www.usenix.org/conference/fast17/technical-sessions/presentation/niazi>

Hops Data Platform



Why not KubeFlow?



Hops – Projects and GPUs

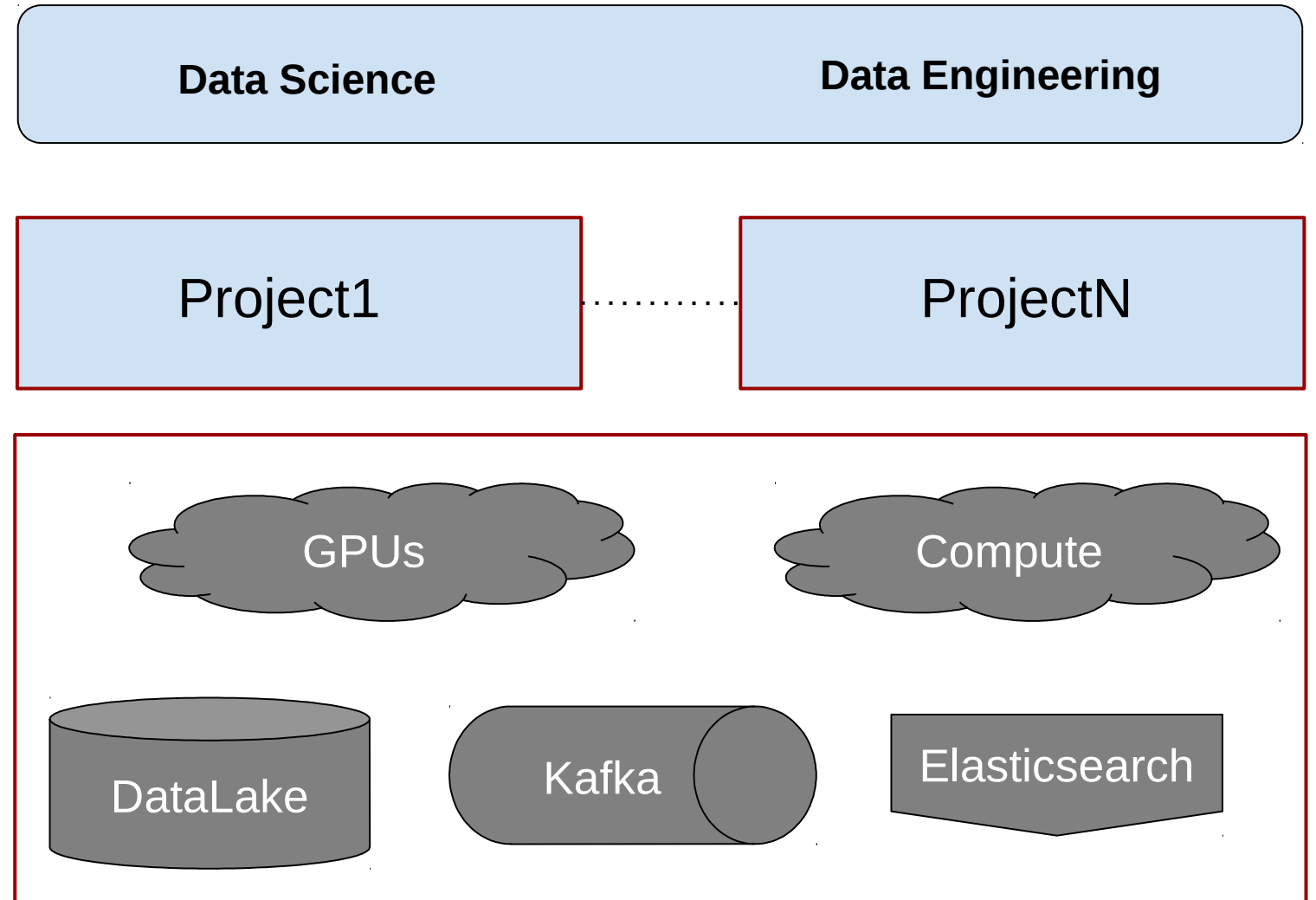
1. Security by Design:

Projects as a Sandbox for self-service and teams

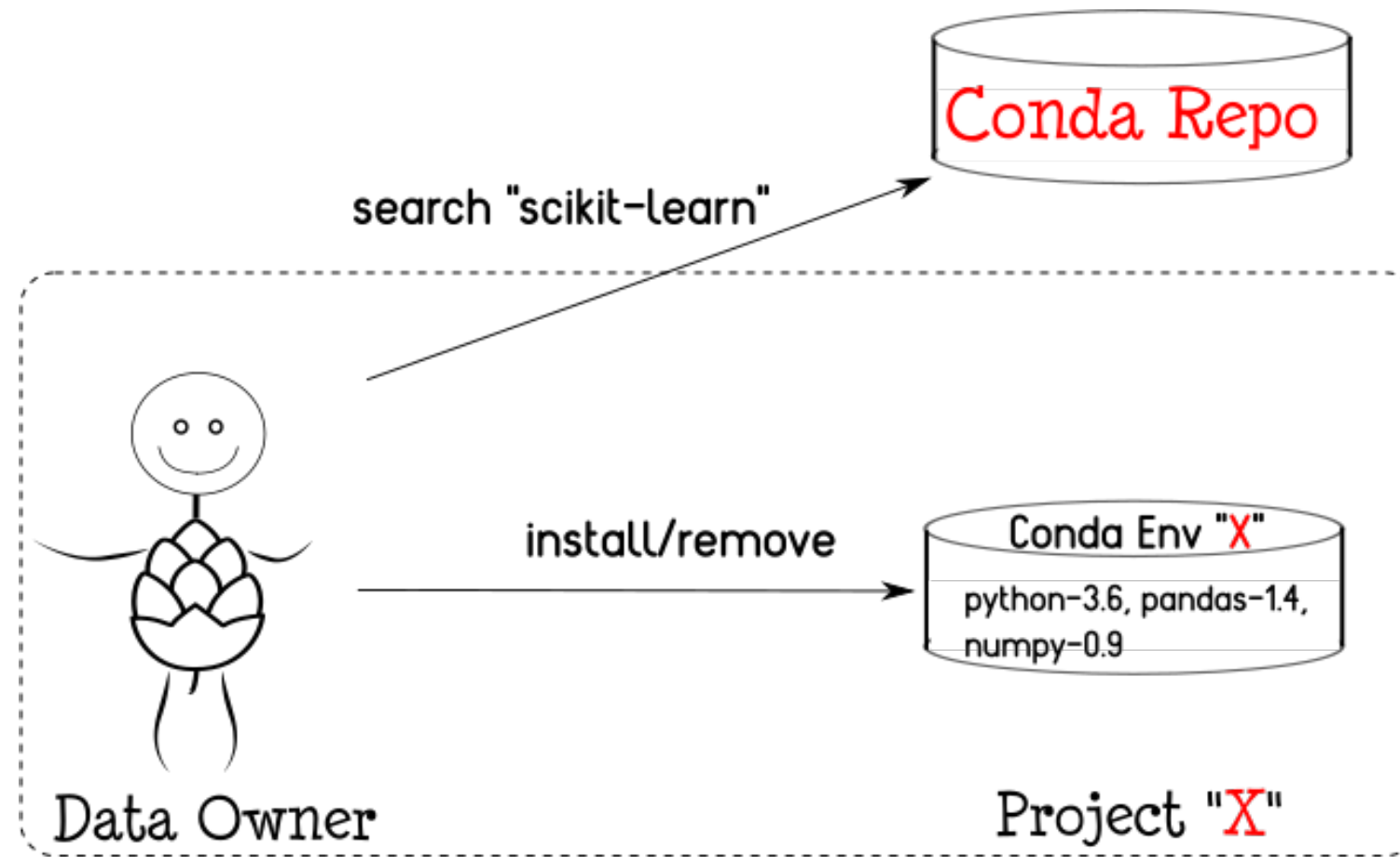
2. Ease of use: Data Scientists need only code Python

3. Scale-out Deep Learning:

Parallel experiments, Distributed Training



Python in the Cluster: Per-Project Conda Envs



Python libraries are usable by Spark/Tensorflow



Hops: TLS Certs for Security (not Kerberos)

- User Certificates:
 - Per-project users
- Service Certificates:
 - NameNode, ResourceManager, Kafka, HiveServer2, Livy, etc
- Application Certificates
- Supports Certification Revocation, Renewal, Reloading.



GPU Resource Requests in Hops YARN

10 GPUs on 1 host

4 GPUs on any host

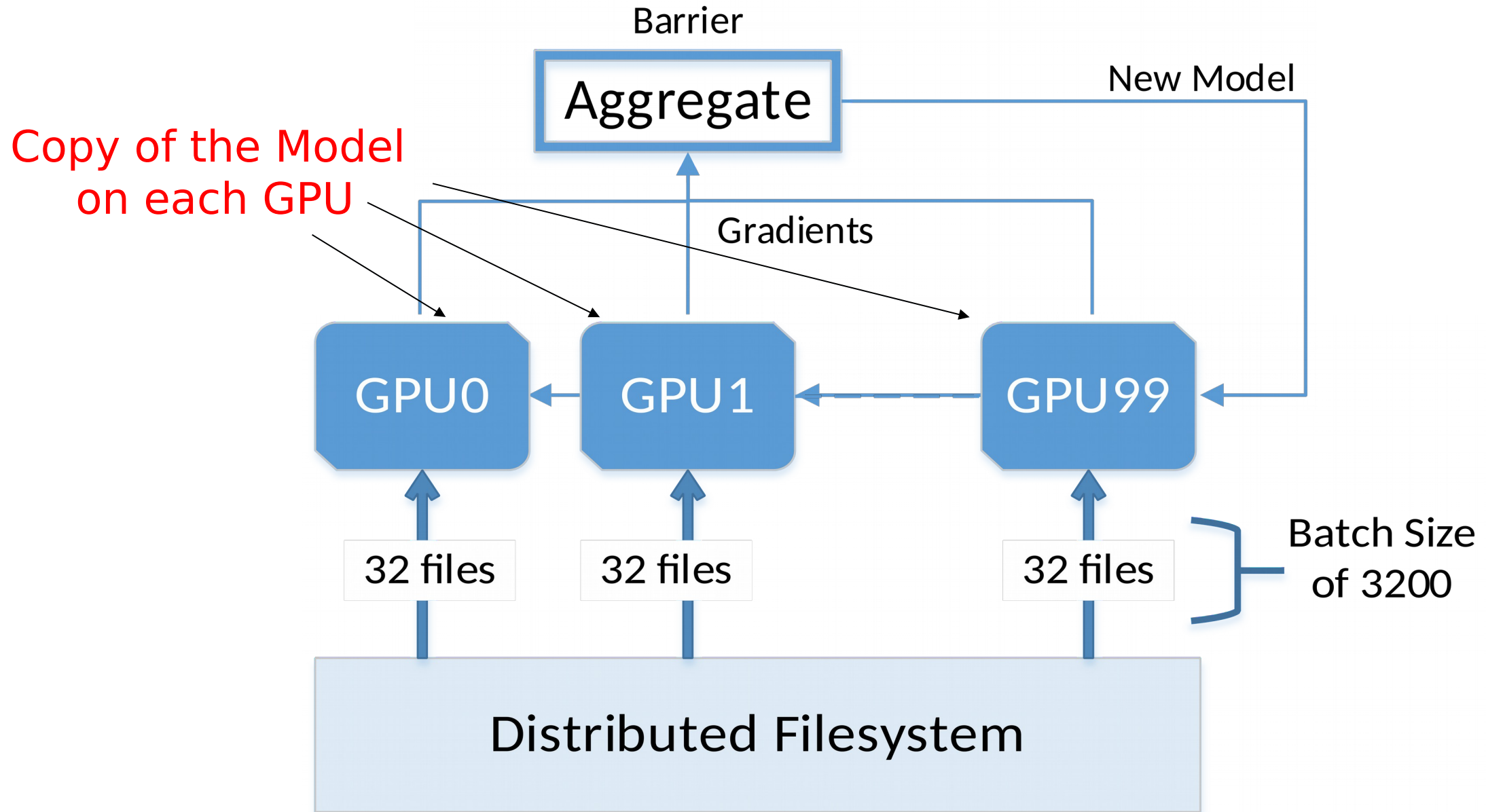
100 GPUs on 10 hosts with 'Infiniband'



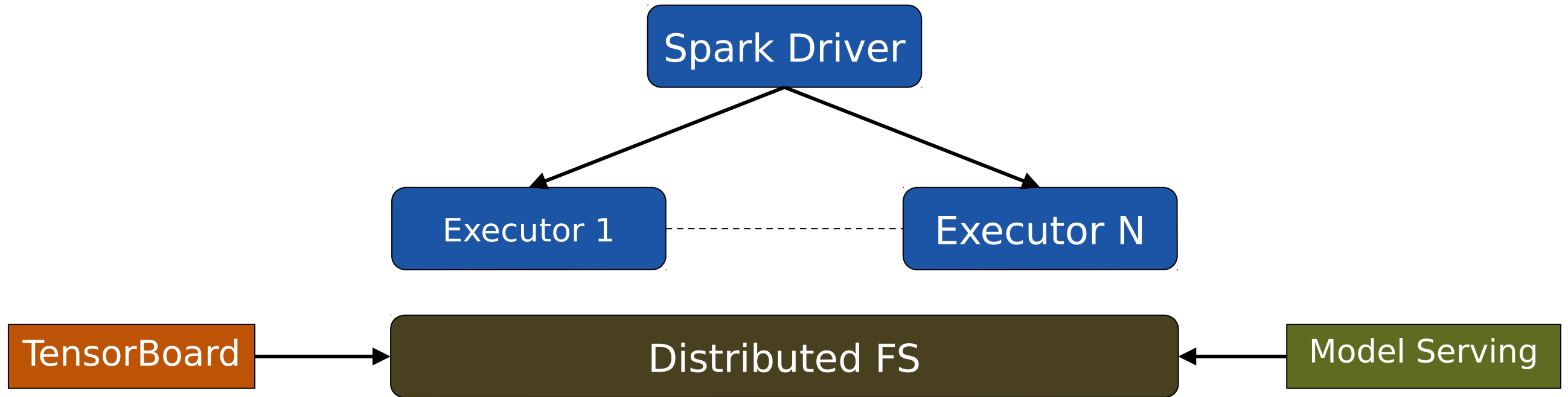
Hops supports a Hetrogenous Mix of GPUs



Data Parallelism on Hops/TensorFlow



Parallel Experiments on Hops

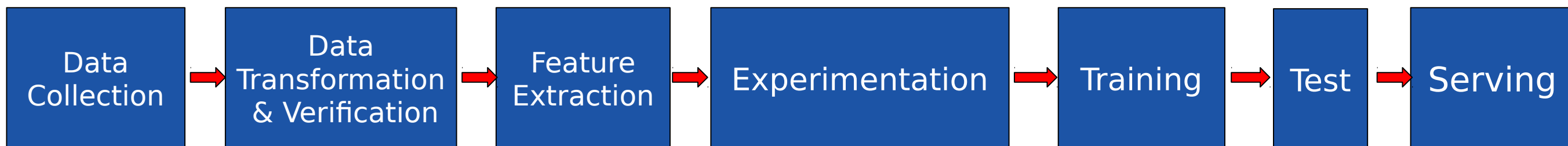


- Big Training/test datasets
- Model checkpointing
- Evaluation
- Distribute Training, Parallel Experiments
- Model Serving

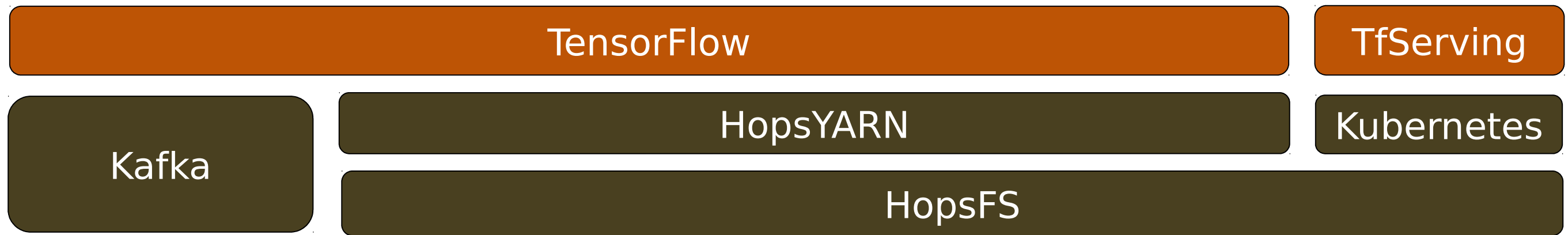


ML in Production: Machine Learning Pipelines

A Machine Learning Pipeline



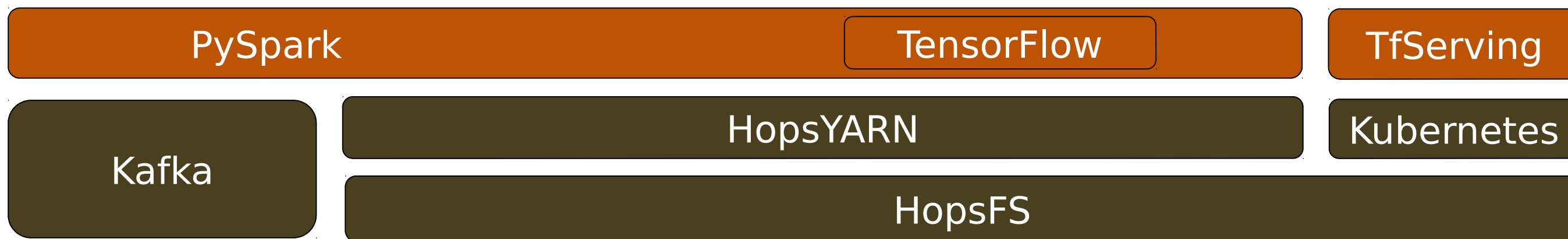
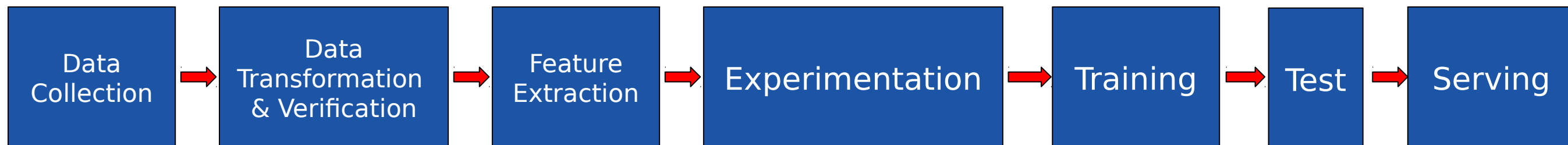
Hops Small Data ML Pipeline



Project Teams (Data Engineers/Scientists)

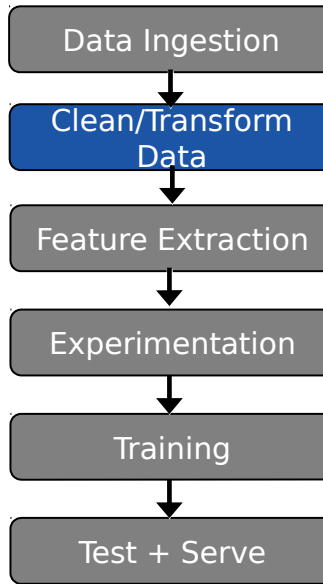
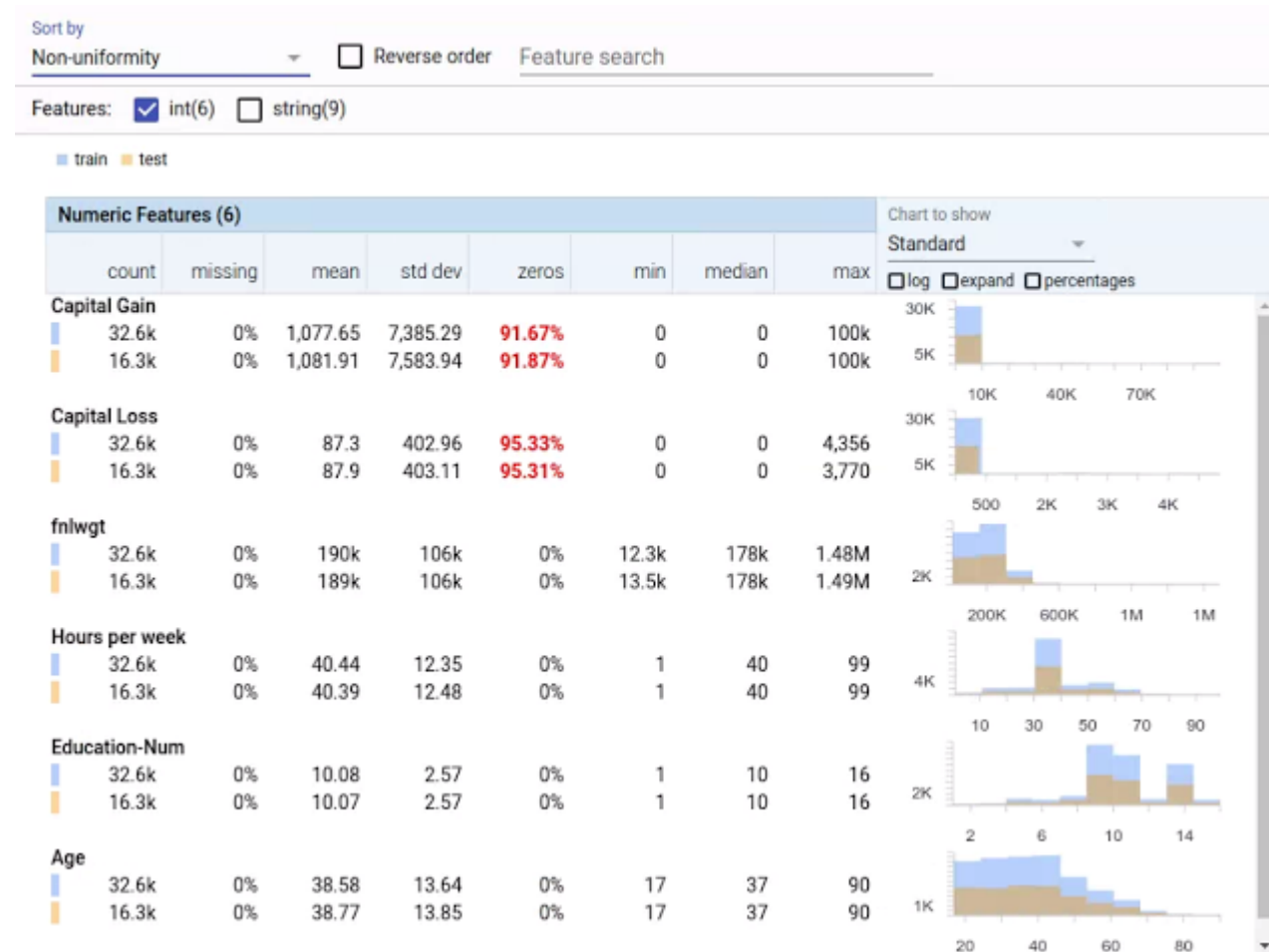


Hops Big Data ML Pipeline



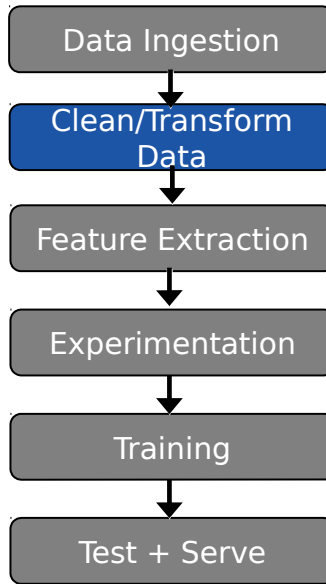
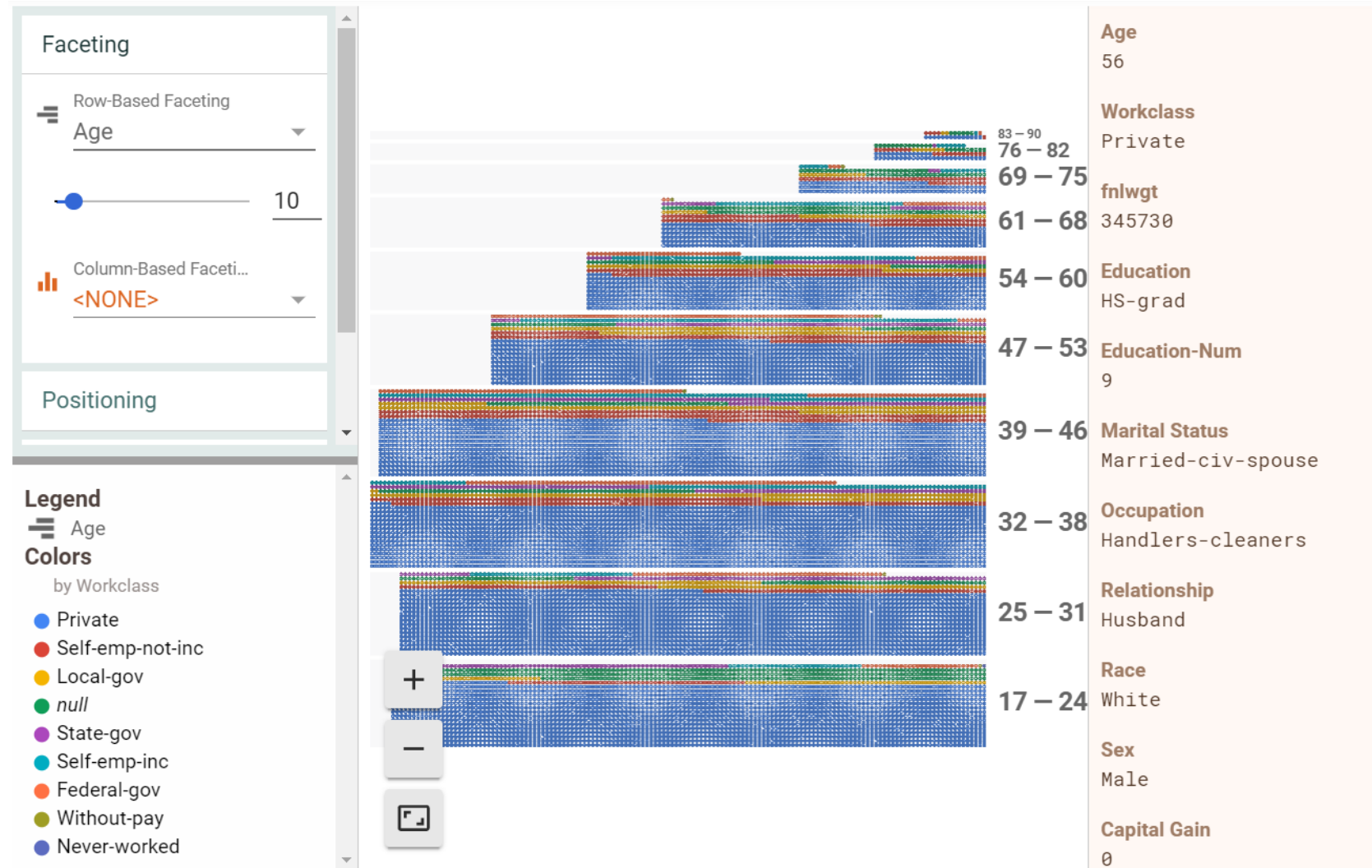
Google Facets Overview

- Visualize data distributions
- Min/max/mean/median values for features
- Missing values in columns
- Facets Overview expects test/train datasets as input



Google Facets Dive

- Visualize the relationship between the data points across the different features of a dataset.

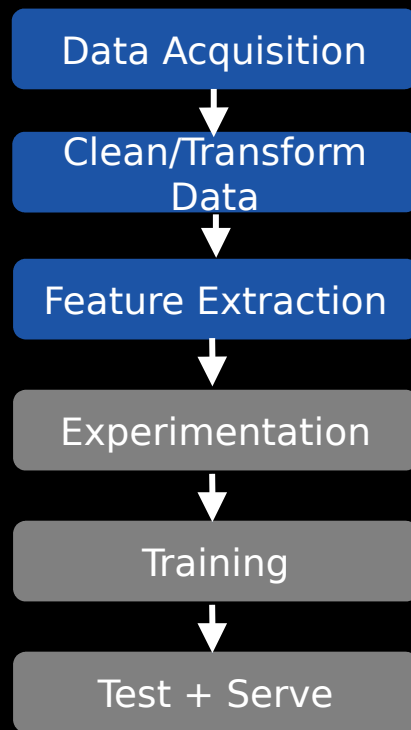


Data Ingestion and Google Facets

```
features = ["Age", "Occupation", "Sex", ..., "Country"]
```

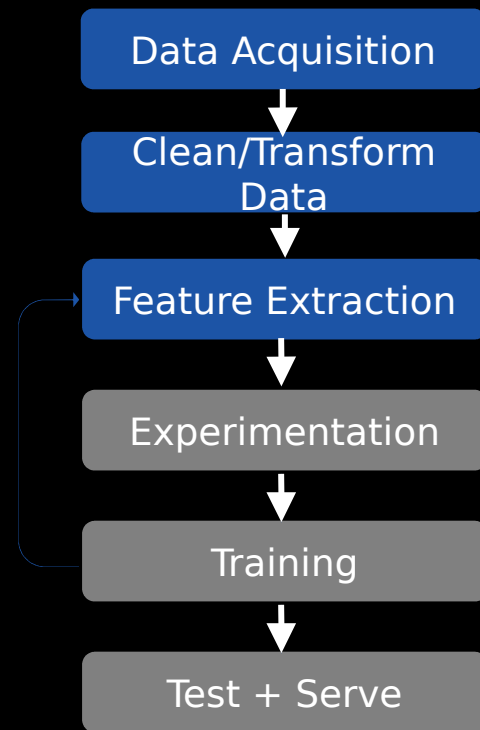
```
h = hdfs.get_fs()
with h.open_file(hdfs.project_path() +
    "/TestJob/data/census/adult.data", "r") as trainFile:
    train_data = pd.read_csv(trainFile, names=features,
        sep=r'\s*,\s*', engine='python', na_values="?")
    test_data = ...
```

```
facets.overview(train_data, test_data)
facets.dive(test_data.to_json(orient='records'))
```



Small Data Preparation with tf.data API

```
def input_fn(batch_size):  
    files = tf.data.Dataset.list_files(IMAGES_DIR)  
  
    def tfrecord_dataset(filename):  
        return tf.data.TFRecordDataset(filename,  
            num_parallel_reads=32, buffer_size=8*1024*1024)  
  
    dataset = files.apply(tf.data.parallel_interleave  
        (tfrecord_dataset, cycle_length=32, sloppy=True))  
    dataset = dataset.apply(tf.data.map_and_batch(parser_fn, batch_size,  
        num_parallel_batches=4))  
    dataset = dataset.prefetch(4)  
    return dataset
```

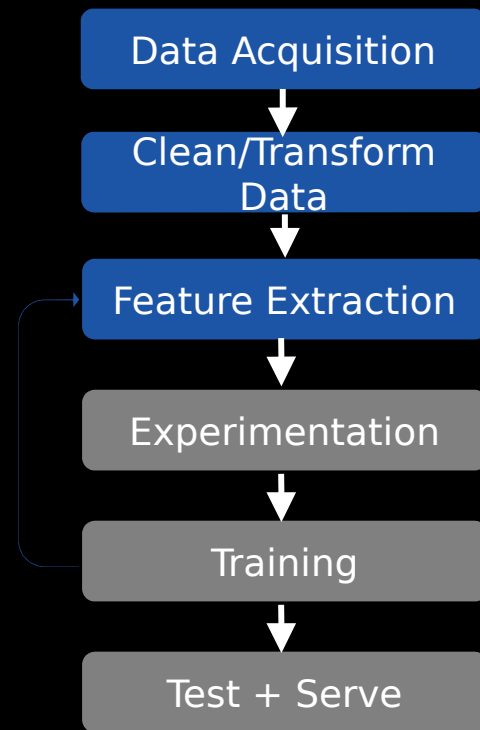


Big Data Preparation with PySpark

```
images = spark.readImages(IMAGE_PATH, recursive = True,
                          numPartitions=10, sampleRatio = 0.1).cache()

tr = (ImageTransformer().setOutputCol("transformed")
     .resize(height = 200, width = 200)
     .crop(0, 0, height = 180, width = 180) )
smallImages = tr.transform(images).select("transformed")

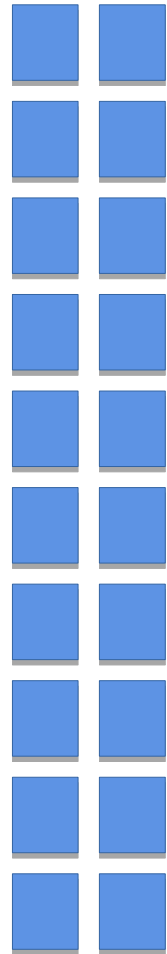
# Output .tfrecords using TensorFlowOnSpark utility
dfutil.saveAsTFRecords(smallImages, OUTPUT_DIR)
```



Parallel Experiments



Hops



The Outer Loop (hyperparameters):
"I have to run a hundred experiments to find the best model," he complained, as he showed me his Jupyter notebooks.
"That takes time. Every experiment takes a lot of programming, because there are so many different parameters.
[\[Rants of a Data Scientist\]](#)

Time



Hyperparam Opt. with Tf/Spark on Hops

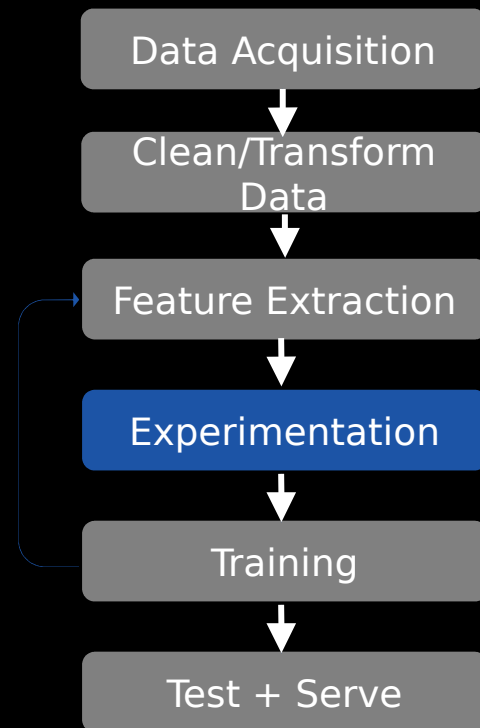
```
def train(learning_rate, dropout):
```

```
[TensorFlow Code here]
```

```
args_dict = {'learning_rate': [0.001, 0.005, 0.01],  
            'dropout': [0.5, 0.6]}
```

```
experiment.launch(spark, train, args_dict)
```

Launch 6 Spark Executors



HyperParam Opt. Visualization on TensorBoard

The screenshot displays the TensorBoard interface for hyperparameter optimization. The left sidebar shows the HopsWorks logo and navigation options like Jupyter, Zeppelin, Jobs, Data Sets, Settings, Members, and Metadata Designer. The main area features a search bar and navigation tabs for SCALARS, IMAGES, GRAPHS, PROJECTOR, and INACTIVE. The SCALARS tab is active, showing three line charts: accuracy, global_step/sec, and loss. The accuracy chart shows a steady increase from 0.00 to approximately 0.80 over 200 steps. The global_step/sec chart shows fluctuating values between 0.00 and 5.00. The loss chart shows a decrease from 2.20 to approximately 0.60 over 200 steps. A red box highlights the 'Runs' section, which lists three hyperparameter configurations: learning_rate=0.001.dropout=0.45, learning_rate=0.0001.dropout=0.7, and learning_rate=0.0005.dropout=0.7. The 'Runs' section also includes a 'Write a regex to filter runs' input field and a 'TOGGLE ALL RUNS' button.

Hyperparam Opt Results Visualization



Model Architecture Search on TensorFlow/Hops

```
def train_cifar10(learning_rate, dropout):
```

```
    [TensorFlow Code here]
```

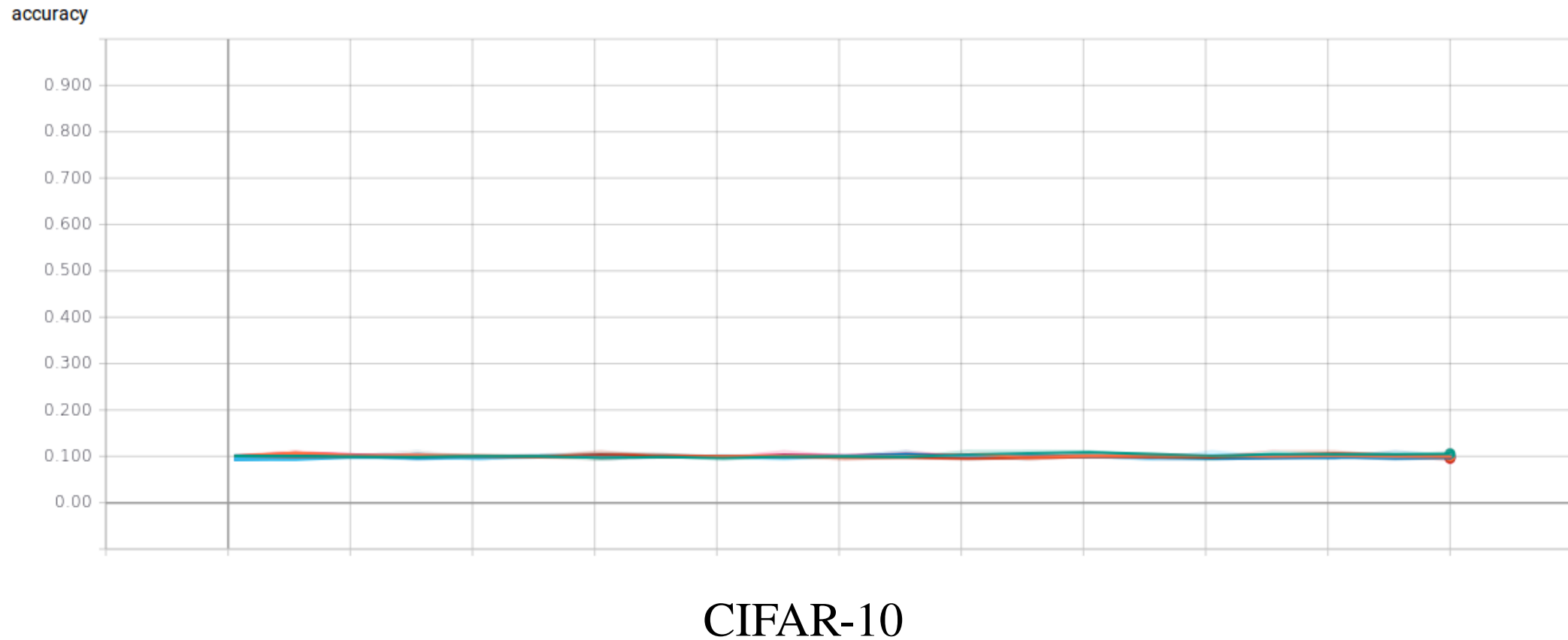
```
dict =
```

```
{'learning_rate': [0.005, 0.00005], 'dropout': [0.01, 0.99],  
'num_layers': [1,3]}
```

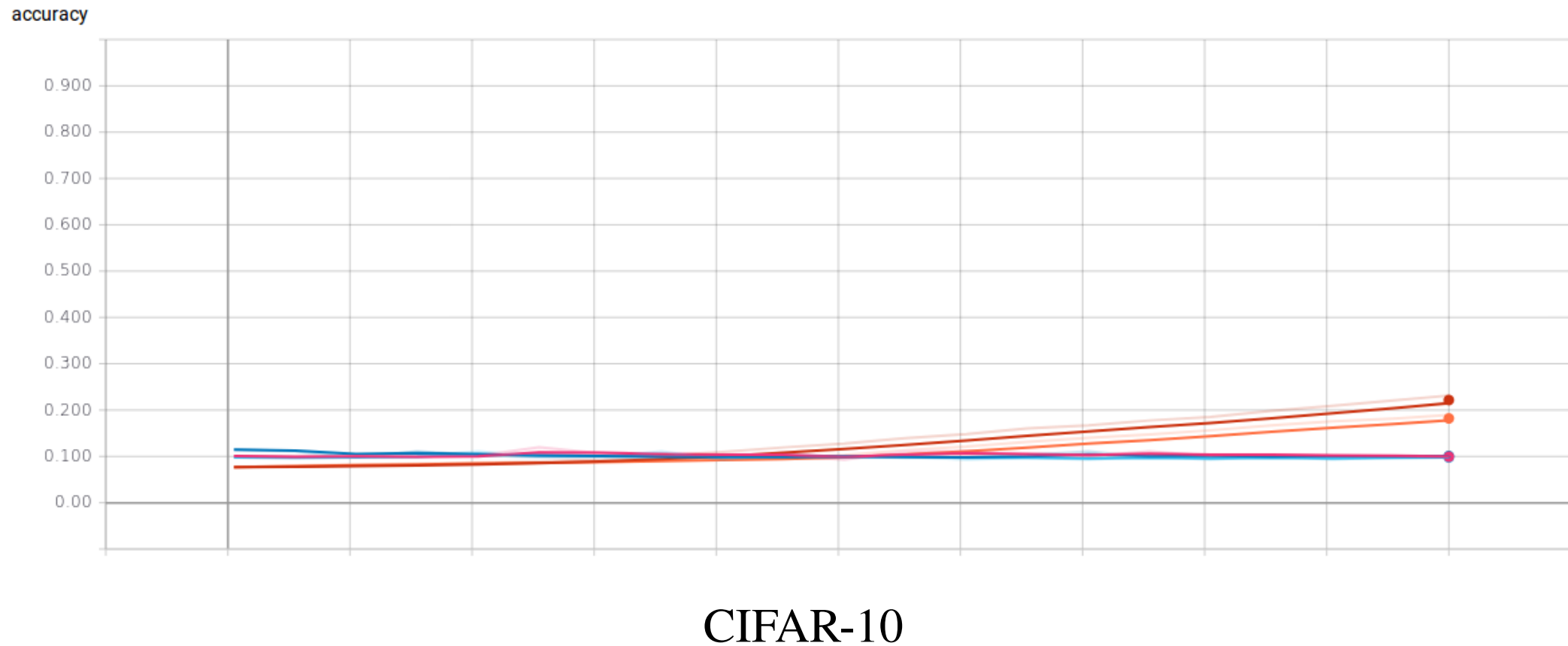
```
experiment.evolutionary_search(spark, train_cifar10, dict,  
direction='max',  
popsize=10, generations=3, crossover=0.7, mutation=0.5)
```



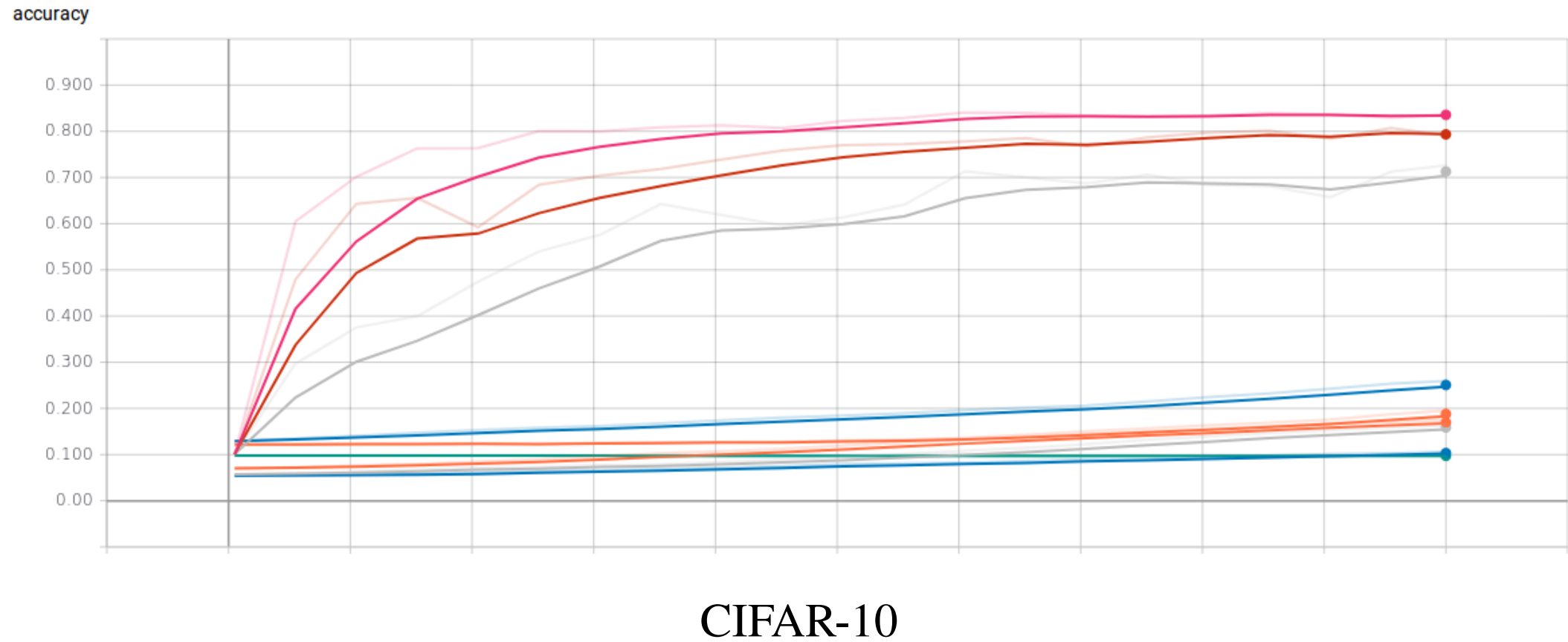
Differential Evolution in Tensorboard (1/4)



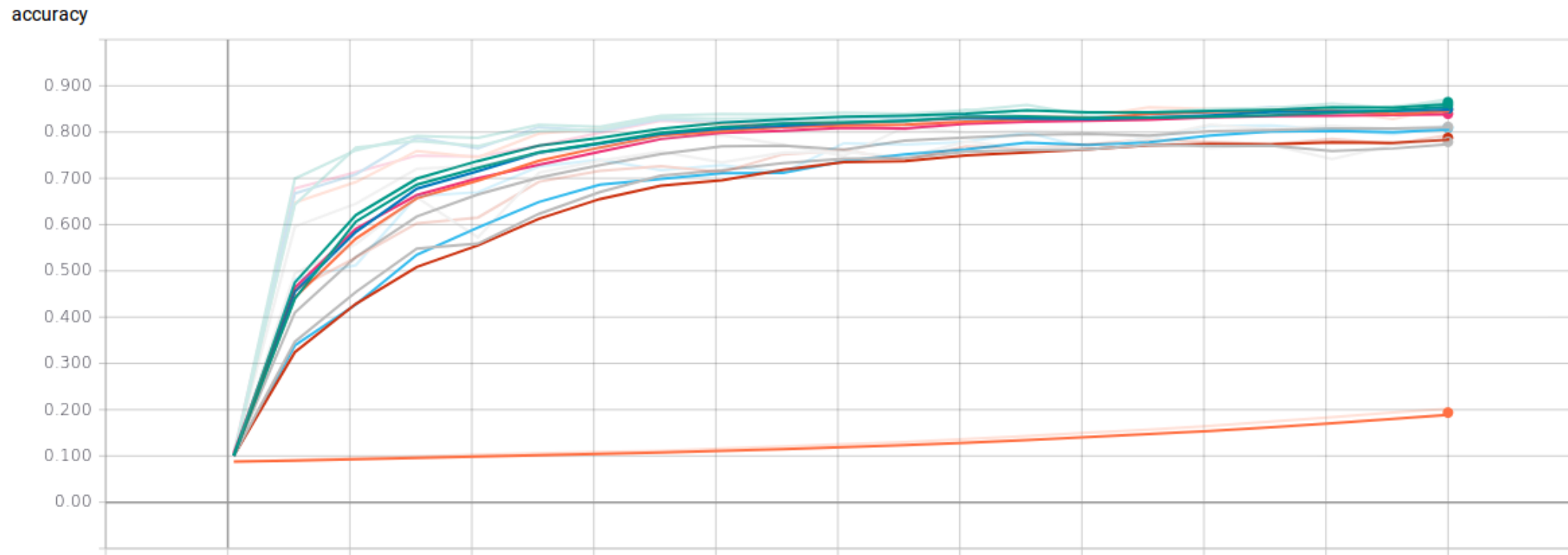
Differential Evolution in Tensorboard (2/4)



Differential Evolution in Tensorboard (3/4)

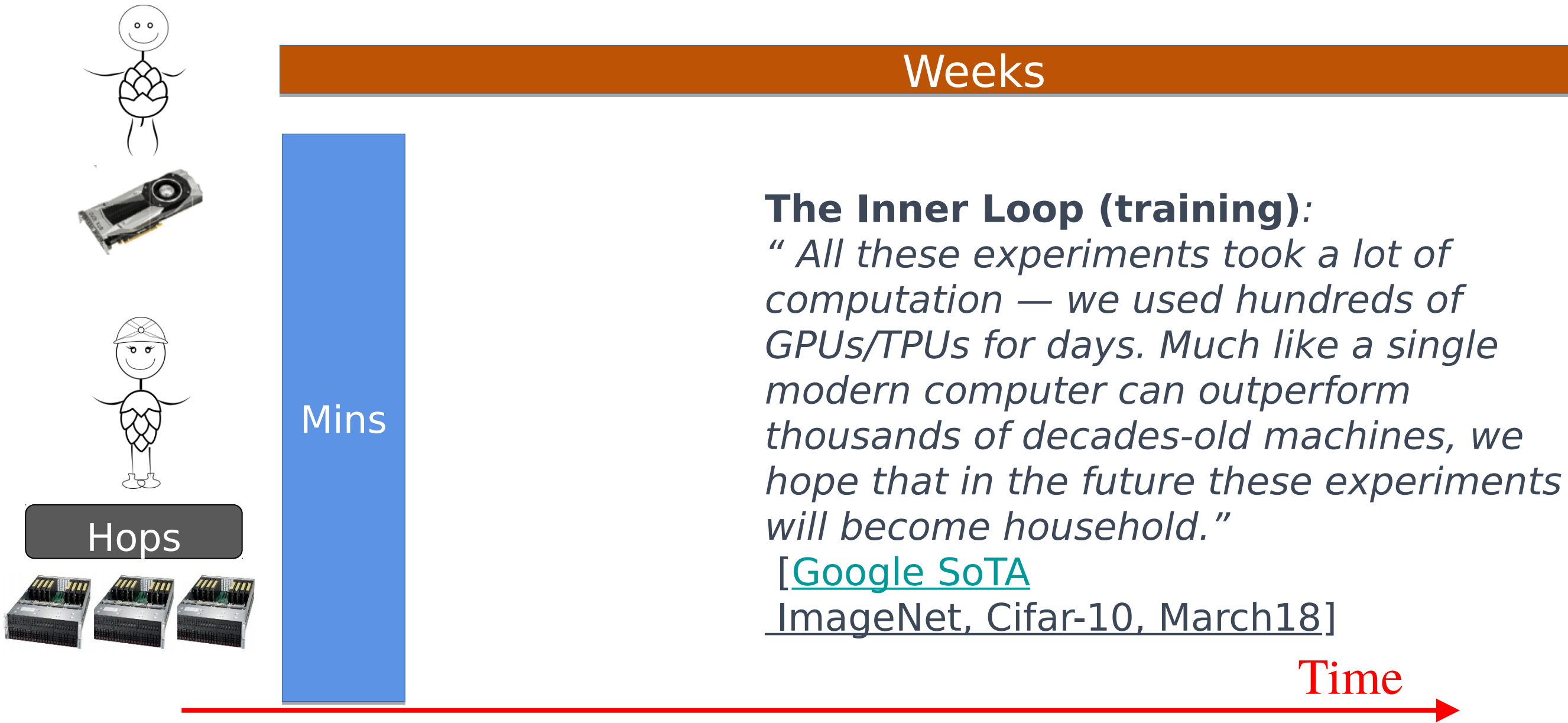


Differential Evolution in Tensorboard (4/4)



CIFAR-10

Distributed Training



Distributed Training: Theory and Practice



Image from @hardmaru on Twitter.



Ring-AllReduce vs Parameter Server



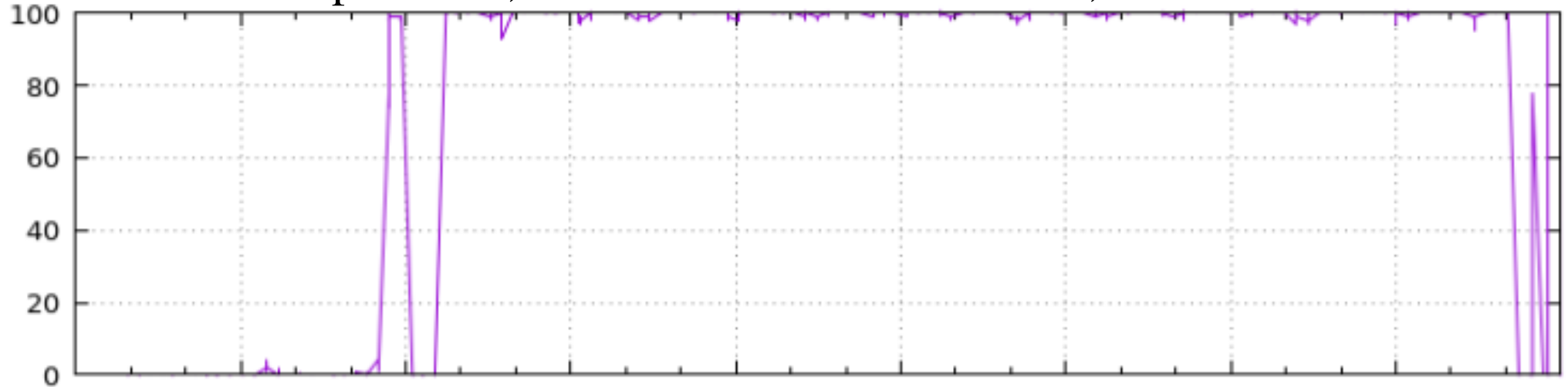
Network Bandwidth is the Bottleneck for Distributed Training



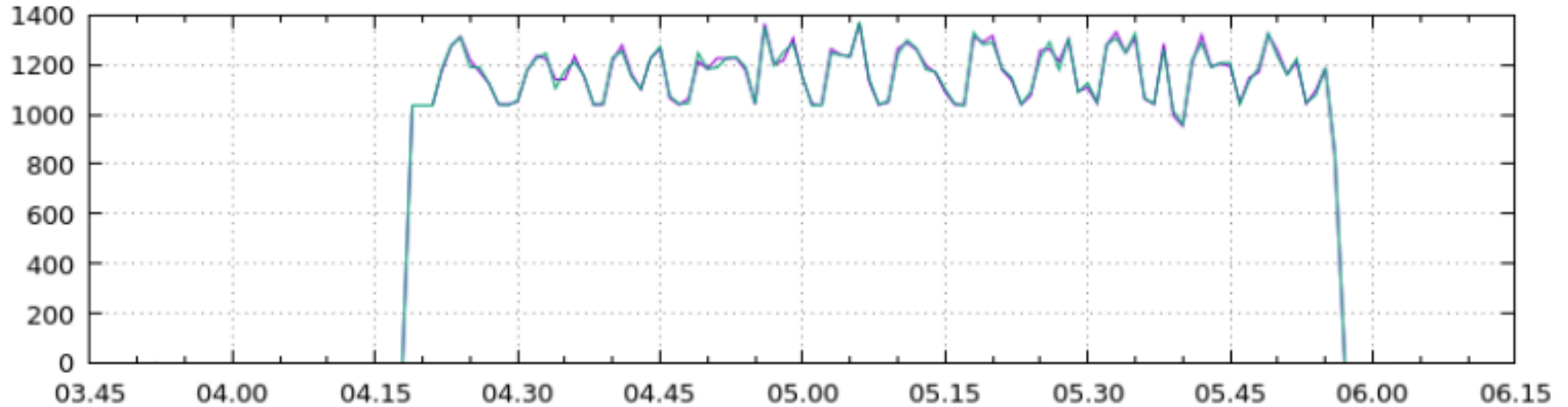
Horovod - AllReduce Inception V4 Performance

Inception V4, 20 Nodes on 2 GPU Servers, 40 Gb/s

% GPU
Utilization



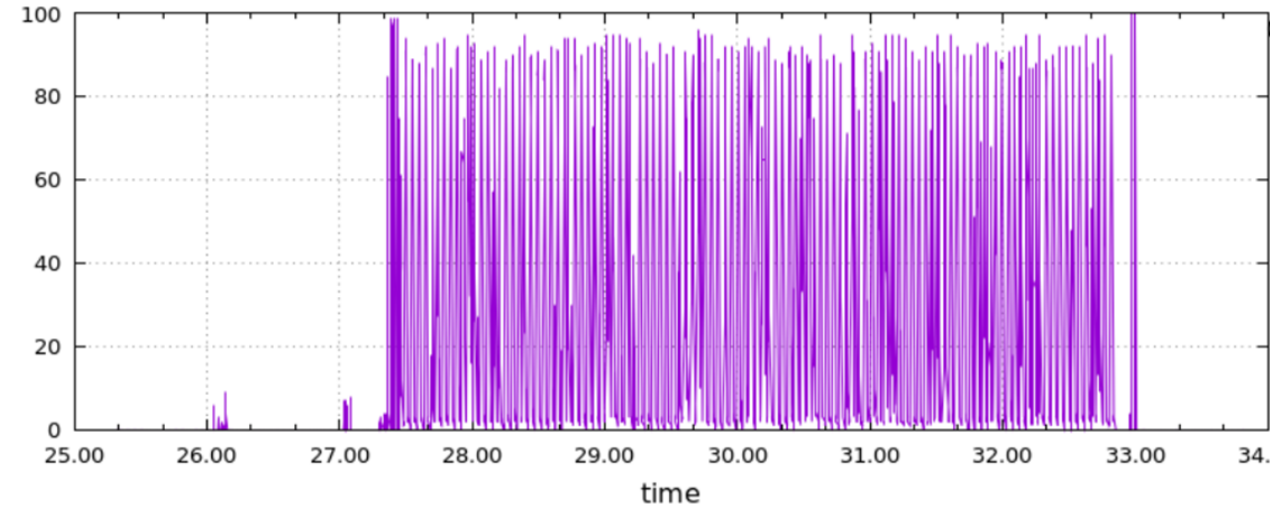
% Network
Bandwidth



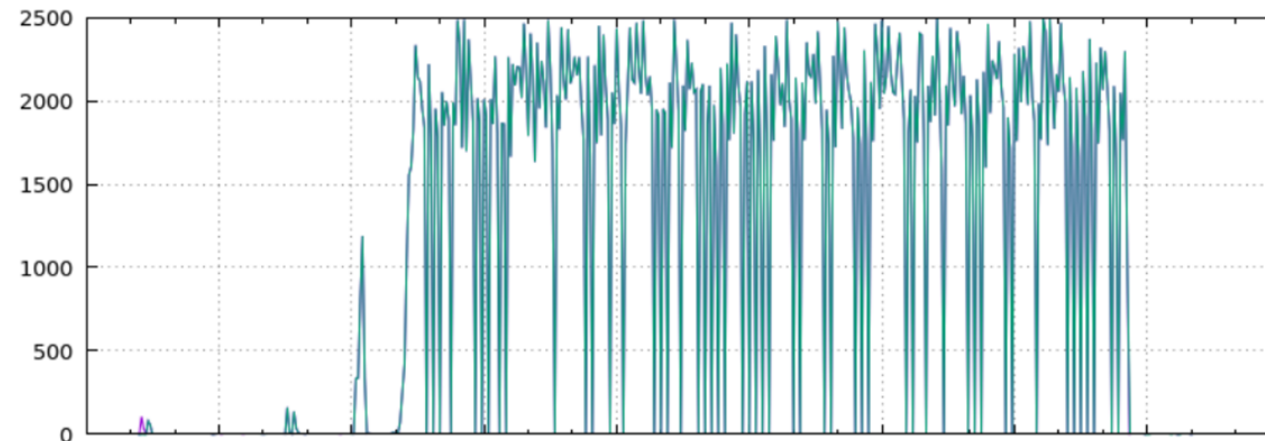
Parameter Server - Inception V4 Performance

Horovod, Inception V4, 10 workers, 2 PSeS, 40 Gb/s, Nvidia 1080Ti

% GPU
Utilization

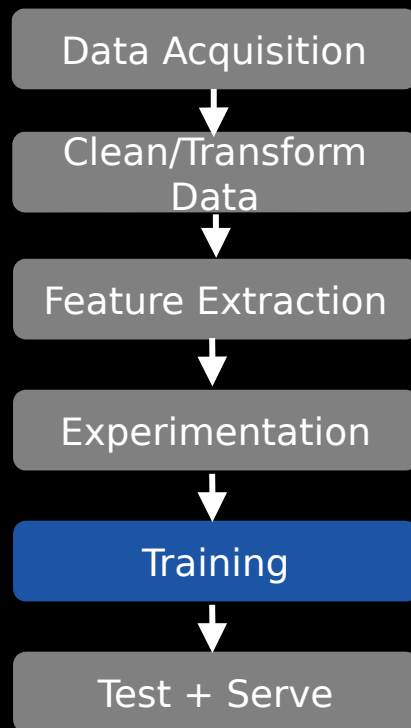


% Network
Bandwidth



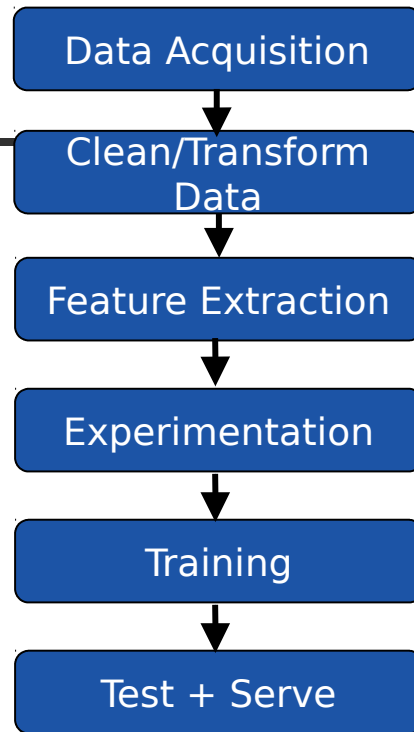
Distributed Training with Horovod on Hops

```
hvd.init()
opt = hvd.DistributedOptimizer(opt)
if hvd.local_rank() == 0:
    [TensorFlow Code here]
.....
else:
    [TensorFlow Code here]
.....
```

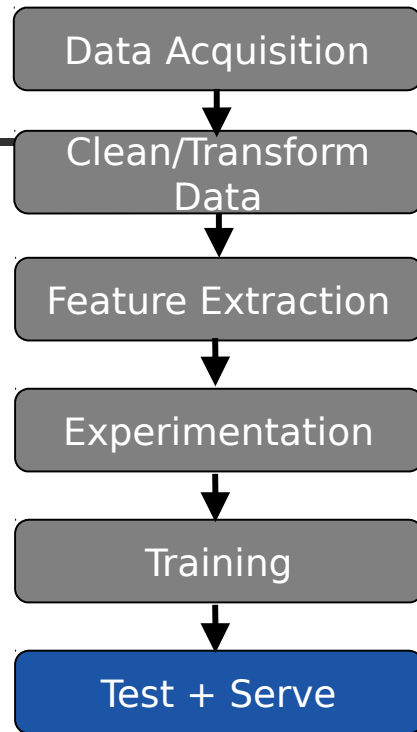


Hops API

- Python (also Java/Scala)
 - Manage tensorboard, Load/save models in HDFS
 - Horovod, TensorFlowOnSpark
 - Parallel experiments
 - Gridsearch
 - Model Architecture Search with Genetic Algorithms
 - Secure Streaming Analytics with Kafka/Spark/Flink
 - SSL/TLS certs, Avro Schema, Endpoints for Kafka/Zookeeper/etc



TensorFlow Model Serving



Model

Enable batching

Create Serving

	Model	Version	Batching	Status	Host	Port	Created	Actions
<input type="button" value="Stop"/>	inception	1	true	Running	10.0.2.15	56778	Jan 16, 2018 5:32:08 PM	<input type="button" value="Logs"/>
<input type="button" value="Run"/>	cifar100	2	true	Created			Jan 16, 2018 5:32:00 PM	<input type="button" value="Delete"/> <input type="button" value="Change version"/>
<input type="button" value="Run"/>	cifar10	1	true	Created			Jan 16, 2018 5:31:53 PM	<input type="button" value="Delete"/> <input type="button" value="Change version"/>

inception

```
2018-01-16 16:32:14.345247: I tensorflow_serving/model_servers/main.cc:147] Building single TensorFlow model file config: model_name: inception model_base_path: /srv/hops/staging/private_dirs /e34a7c0f2aa65470edc34b13f7a4fb8bf66c280338d260917b13a313cdf7f011/tfserving/model/inception
2018-01-16 16:32:14.345604: I tensorflow_serving/model_servers/server_core.cc:441] Adding/updating models.
2018-01-16 16:32:14.345640: I tensorflow_serving/model_servers/server_core.cc:492] (Re-)adding model: inception
2018-01-16 16:32:14.446217: I tensorflow_serving/core/basic_manager.cc:705] Successfully reserved resources to load servable {name: inception version: 1}
2018-01-16 16:32:14.446267: I tensorflow_serving/core/loader_harness.cc:66] Approving load for servable version {name: inception version: 1}
2018-01-16 16:32:14.446298: I tensorflow_serving/core/loader_harness.cc:74] Loading servable version {name: inception version: 1}
2018-01-16 16:32:14.446339: I external/org_tensorflow/tensorflow/contrib/session_bundle/bundle_shim.cc:360] Attempting to load native SavedModelBundle in bundle-shim from: /srv/hops/staging/private_dirs /e34a7c0f2aa65470edc34b13f7a4fb8bf66c280338d260917b13a313cdf7f011/tfserving/model/inception/1
2018-01-16 16:32:14.446372: I external/org_tensorflow/tensorflow/cc/saved_model/loader.cc:236] Loading SavedModel from: /srv/hops/staging/private_dirs/e34a7c0f2aa65470edc34b13f7a4fb8bf66c280338d260917b13a313cdf7f011/tfserving/model/inception/1
2018-01-16 16:32:14.506313: I external/org_tensorflow/tensorflow/cc/saved_model/loader.cc:155] Restoring SavedModel bundle.
2018-01-16 16:32:14.517111: I external/org_tensorflow/tensorflow/cc/saved_model/loader.cc:190] Running LegacyInitOp on SavedModel bundle.
2018-01-16 16:32:14.521759: I external/org_tensorflow/tensorflow/cc/saved_model/loader.cc:284] Loading SavedModel: success. Took 75374 microseconds.
2018-01-16 16:32:14.521835: I tensorflow_serving/servables/tensorflow/saved_model_bundle_factory.cc:93] Wrapping session to perform batch processing
2018-01-16 16:32:14.521869: I tensorflow_serving/servables/tensorflow/bundle_factory_util.cc:153] Wrapping session to perform batch processing
2018-01-16 16:32:14.522216: I tensorflow_serving/core/loader_harness.cc:86] Successfully loaded servable version {name: inception version: 1}
E0116 16:32:14.525443029 19872 ev_epoll1_linux.cc:1051] grpc epoll fd: 3
2018-01-16 16:32:14.527754: I tensorflow_serving/model_servers/main.cc:288] Running ModelServer at 0.0.0.0:56778 ...
```



Training-Serving Skew

- Monitor differences between performance during training and performance during serving.
 - Differences in how you process data in training vs serving.
 - Differences in the training data and live data for serving.
 - A feedback loop between your model and your algorithm.
- When to retrain?
 - If you look at the input data and use **covariant shift** to see when it deviates significantly from the data that was used to train the model on.



0. Register an account

1. Create a 'tensorflow_tour'

serving/train_and_export_model.ipynb

Try out other notebooks - tensorflow/cnn/grid_search,

- 2. Create a new project called 'yourname_hotdog'

- a) enable Python 3.6 for the project

- b) search for the dataset 'hotdog' and import it into 'yourname_hotdog'

- c) download <http://hopshadoop.com:8080/hotdog.ipynb> to your laptop.

- d) upload **hotdog.ipynb** into the Jupyter dataset in 'yourname_hotdog'

- e) install the conda dependencies: matplotlib, pillow, numpy

- f) Start Jupyter and run **hotdog.ipynb**

[Credit Magnus Pedersson : <https://www.youtube.com/watch?v=oxrcZ9uUblI>]



Summary

- Hops is a new Data Platform with first-class support for Python / Deep Learning / ML / Data Governance / GPUs
- You can do fun stuff on Hops, like "Hotdog or not", as well as serious stuff.



The Team

Active:

Jim Dowling, Seif Haridi, Gautier Berthou, Salman Niazi, Mahmoud Ismail, Theofilos Kakantousis, Ermias Gebremeskel, Antonios Kouzoupis, Alex Ormenisan, Fabio Buso, Robin Andersson, August Bonds.

Alumni:

Vasileios Giannokostas, Johan Svedlund Nordström, Rizvi Hasan, Paul Mälzer, Bram Leenders, Juan Roca, Misganu Dessalegn, K "Sri" Srijevanthan, Jude D'Souza, Alberto Lorente, Andre Moré, Ali Gholami, Davis Jaunzems, Stig Viaene, Hooman Peiro, Evangelos Savvidis, Steffen Grohsschmiedt, Qi Qi, Gayana Chandrasekara, Nikolaos Stanogias, Daniel Bali, Ioannis Kerkinos, Peter Buechler, Pushparaj Motamari, Hamid Afzali, Wasif Malik, Lalith Suresh, Mariano Valles, Ying Lieu, Fanti Machmount Al Samisti, Braulio Grana, Adam Alpire, Zahin Azher Rashid, Aruna Kumari Yedurupaka, Tobias Johansson, Roberto Bampi, Roshan Sedar.



www.hops.io

 @hopshadoop

