



Hops

Scaling the AI Hierarchy of Needs with Hopsworks

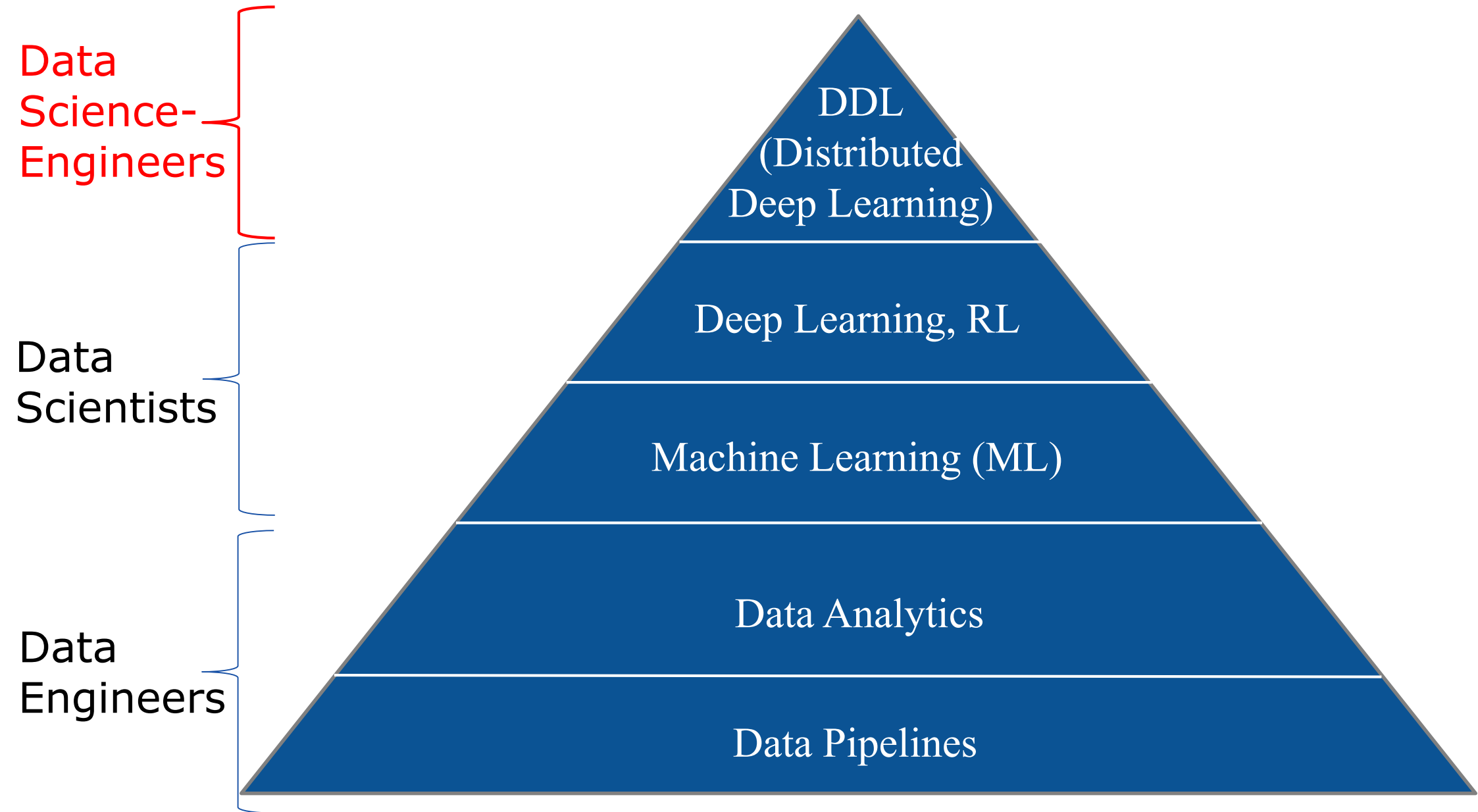
Jim Dowling

Assoc Prof @ KTH

CEO @ Logical Clocks AB

CERN Colloquium, April 16th 2018

AI Hierarchy of Needs



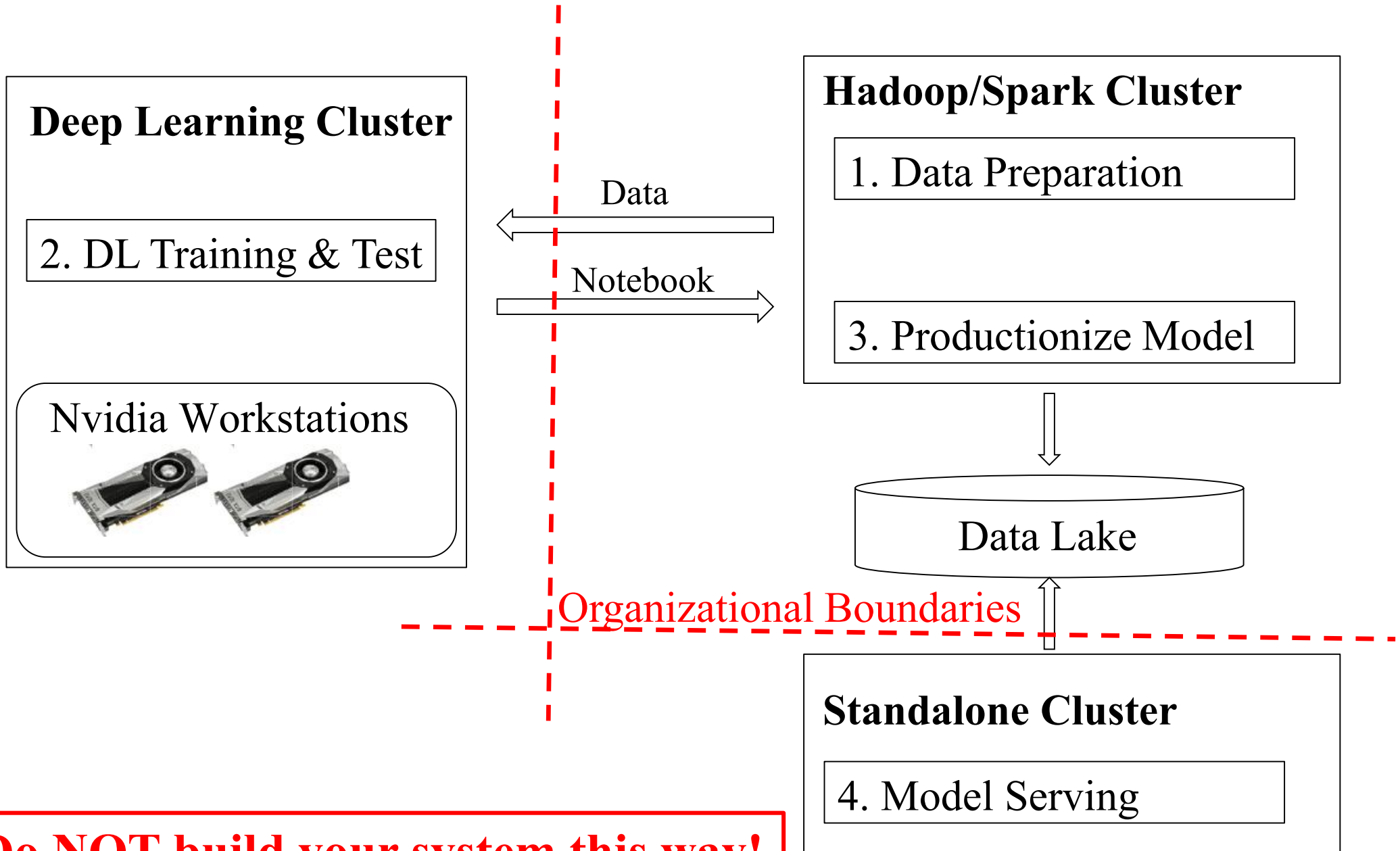
Hopsworks: Collaborative Data Science*

“Open data science platforms [should] do the following:

- Provide a shared platform for all data science contributors
- Open data science tools (such as Python and R) at scale
- Provide self-service access to data, storage, and compute
- Support a complete pipeline from data to deployment
- Include collaborative development tools`
- Ensure asset management and reproducibility”

State-of-the-Art for On-Premise ML Pipelines

Separate Data Lake and Deep Learning Clusters*



***Do NOT build your system this way!**

Why not Kubeflow?

- Operational Reasons

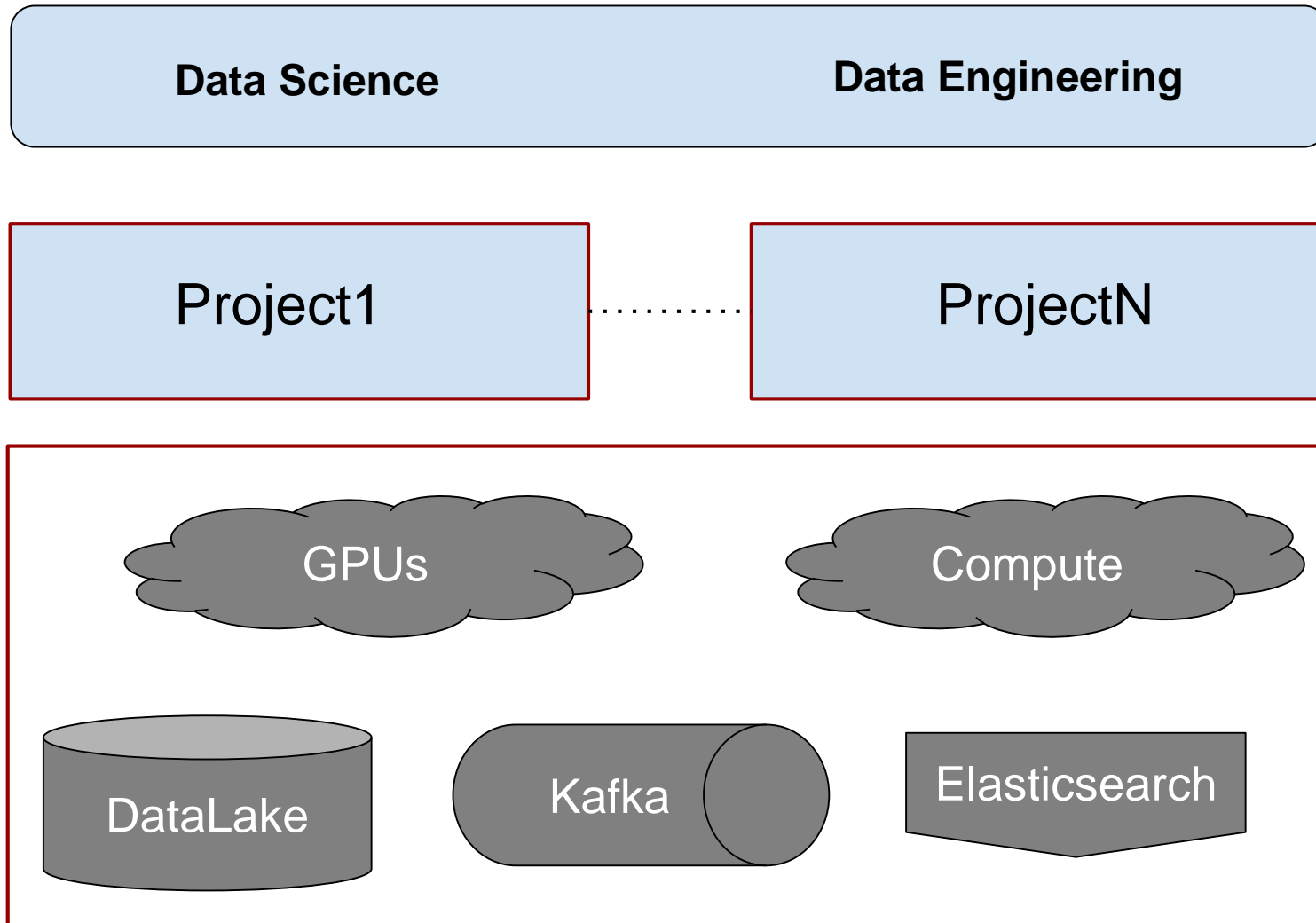
- Limited Enterprise Security
 - Data Lake Integration? Encryption-at-Rest?
- Stateful services still needed – external to Kubernetes

- Usability Reasons

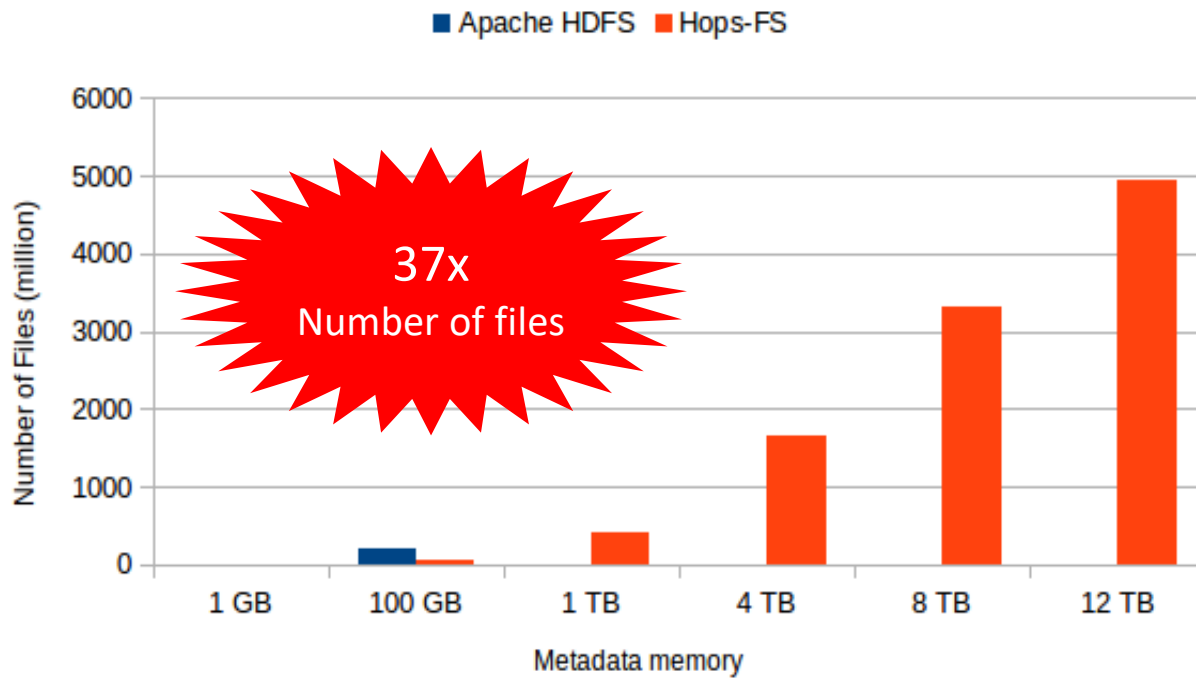
- Install a new Python library?
 - Need to update/write a Dockerfile
- Interactive Analytics?
 - Data Scientists have to write YML files and restart their cluster
 - Can be slow to spawn a new cluster for development
- Secure Data Ingestion?

The Hops Way

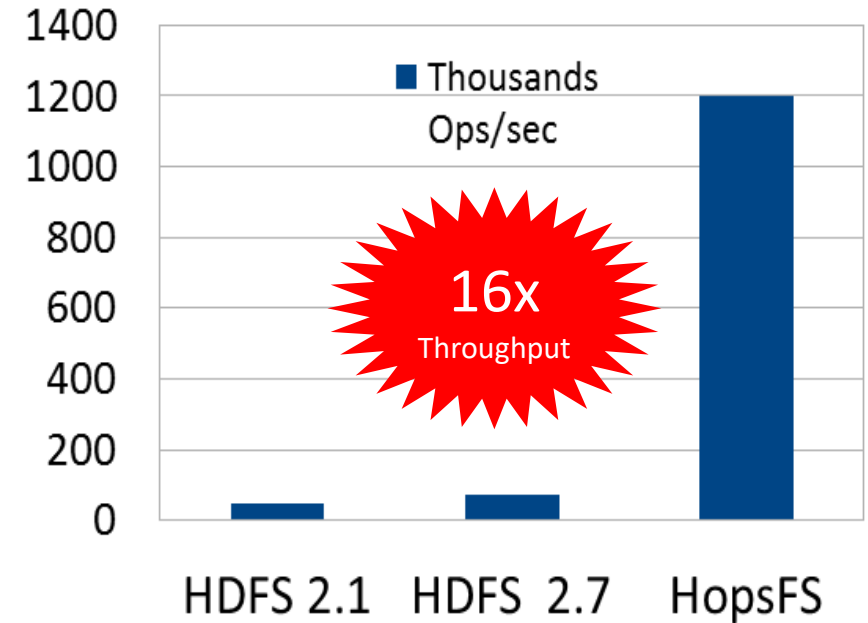
Hopsworks: Single ML and Big Data Cluster



HopsFS: Next Generation HDFS*



Bigger



Faster

 **IEEE** Scale Challenge Winner (2017)

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

GPU Resource Requests in Hops YARN



4 GPUs on any host
10 GPUs on 1 host



100 GPUs on 10 hosts with 'Infiniband'

HopsYARN



Hops supports a Hetrogenous Mix of GPUs

Lots of good GPUs > A few great GPUs

Hops



100 x Nvidia 1080Ti (DeepLearning11)

VS



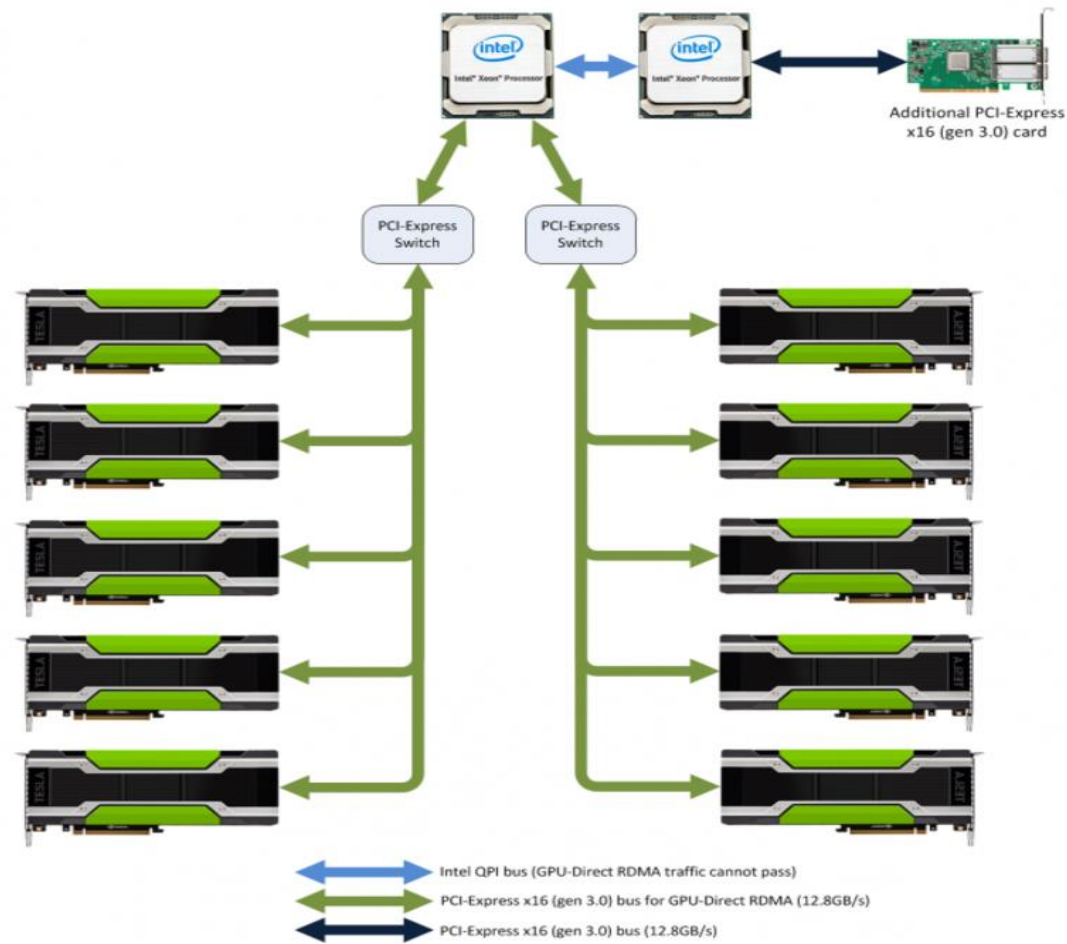
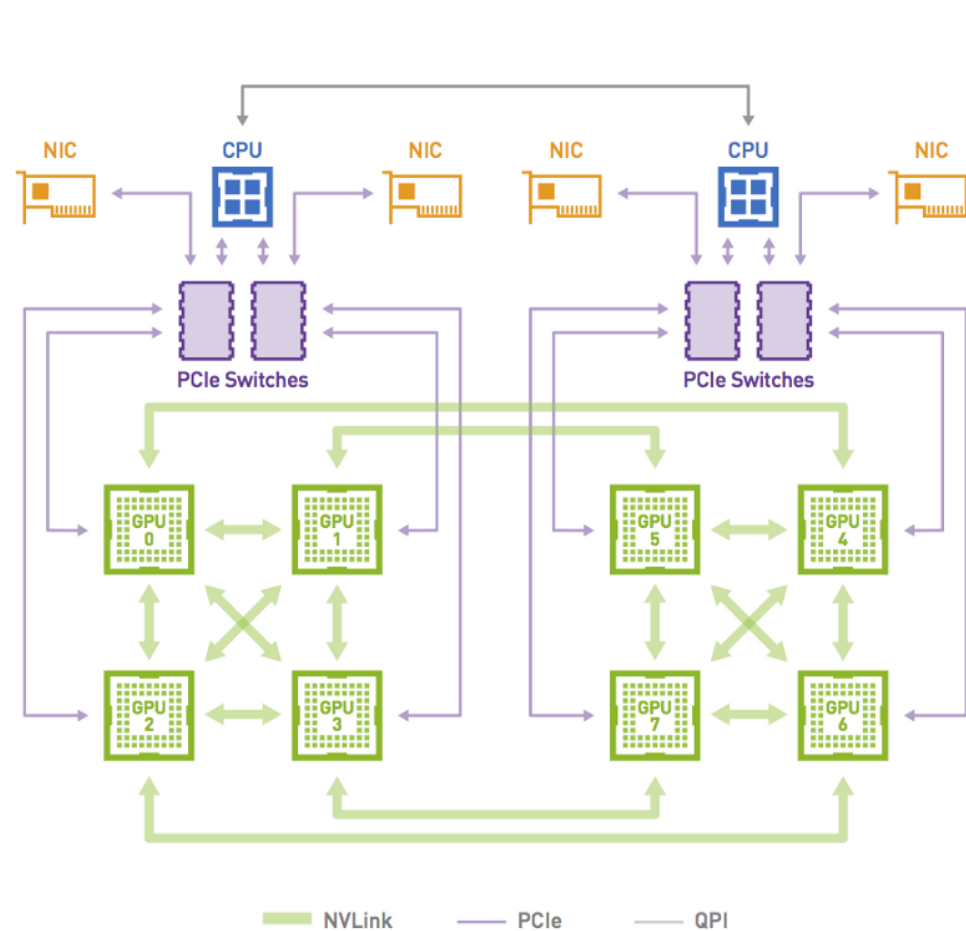
8 x Nvidia P/V100 (DGX-1)

Both top (100 GPUs) and bottom (8 GPUs) cost the same: 150K Euro.

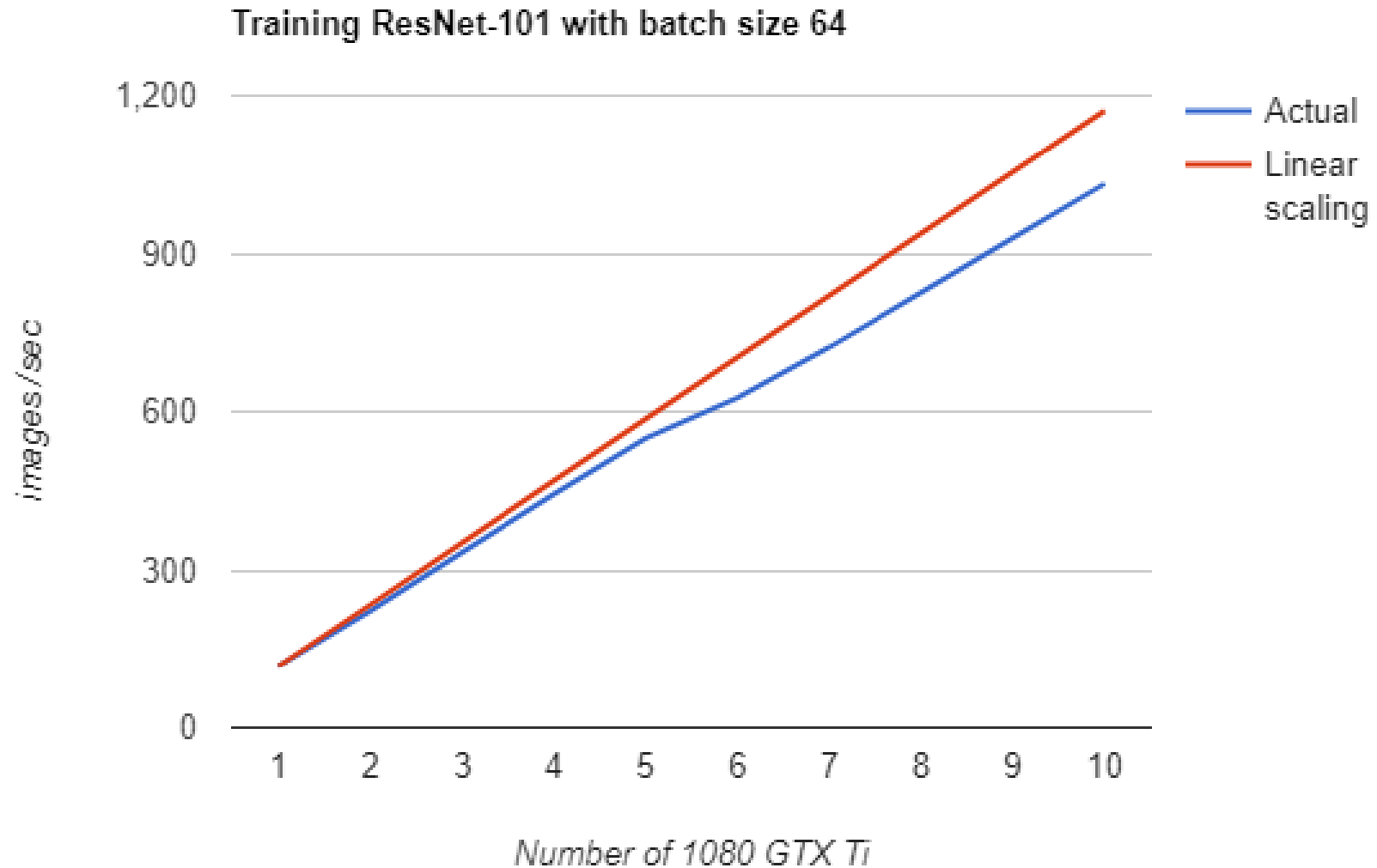
NVLink vs PCI-E Single Root Complex

NVLink – 80 GB/s

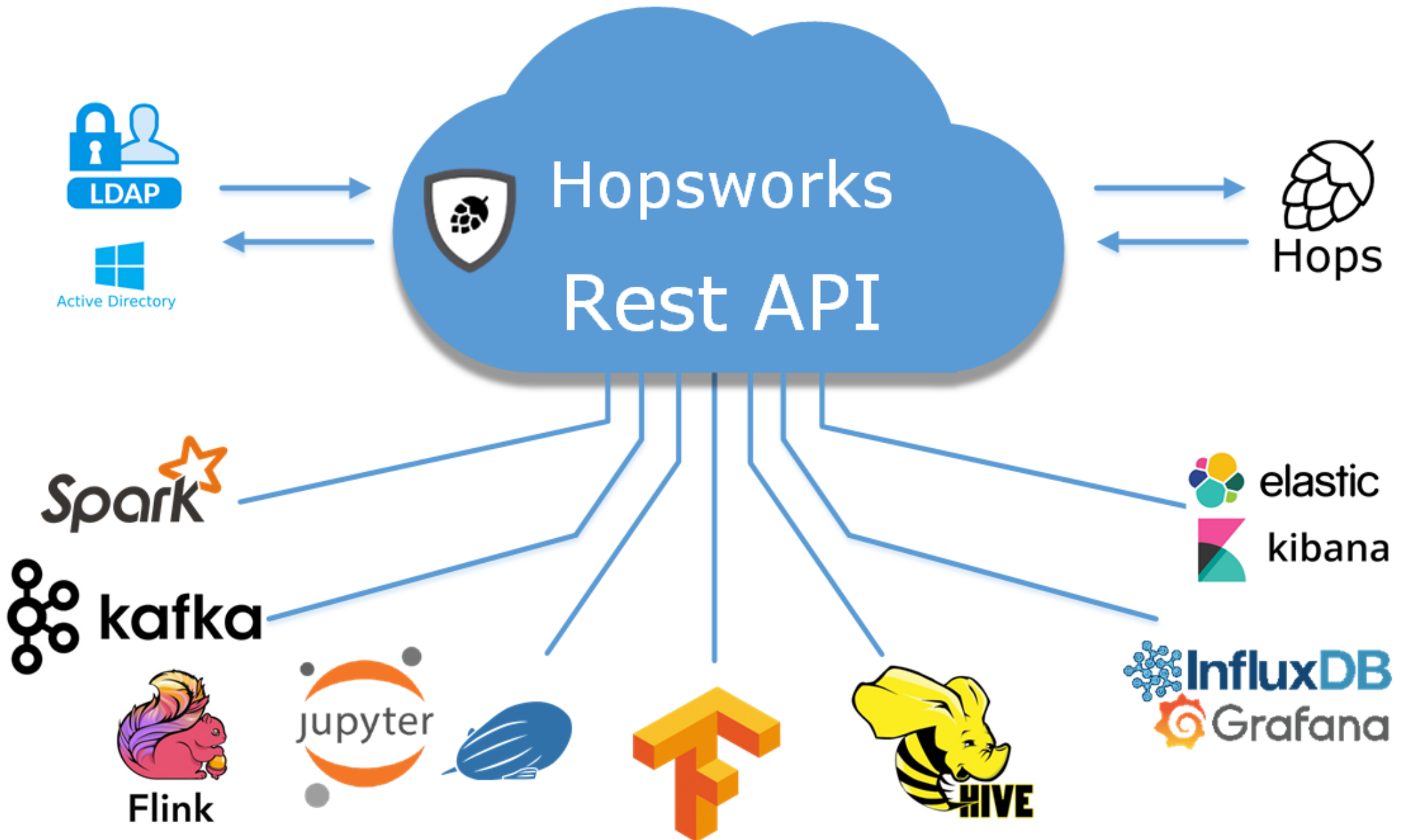
PCI-E – 16 GB/s



TensorFlow/Hops on 10 1080Ti GPUs



Hopsworks*



Hopsworks: Projects, Users, Datasets

Privacy by Design in Hopsworks

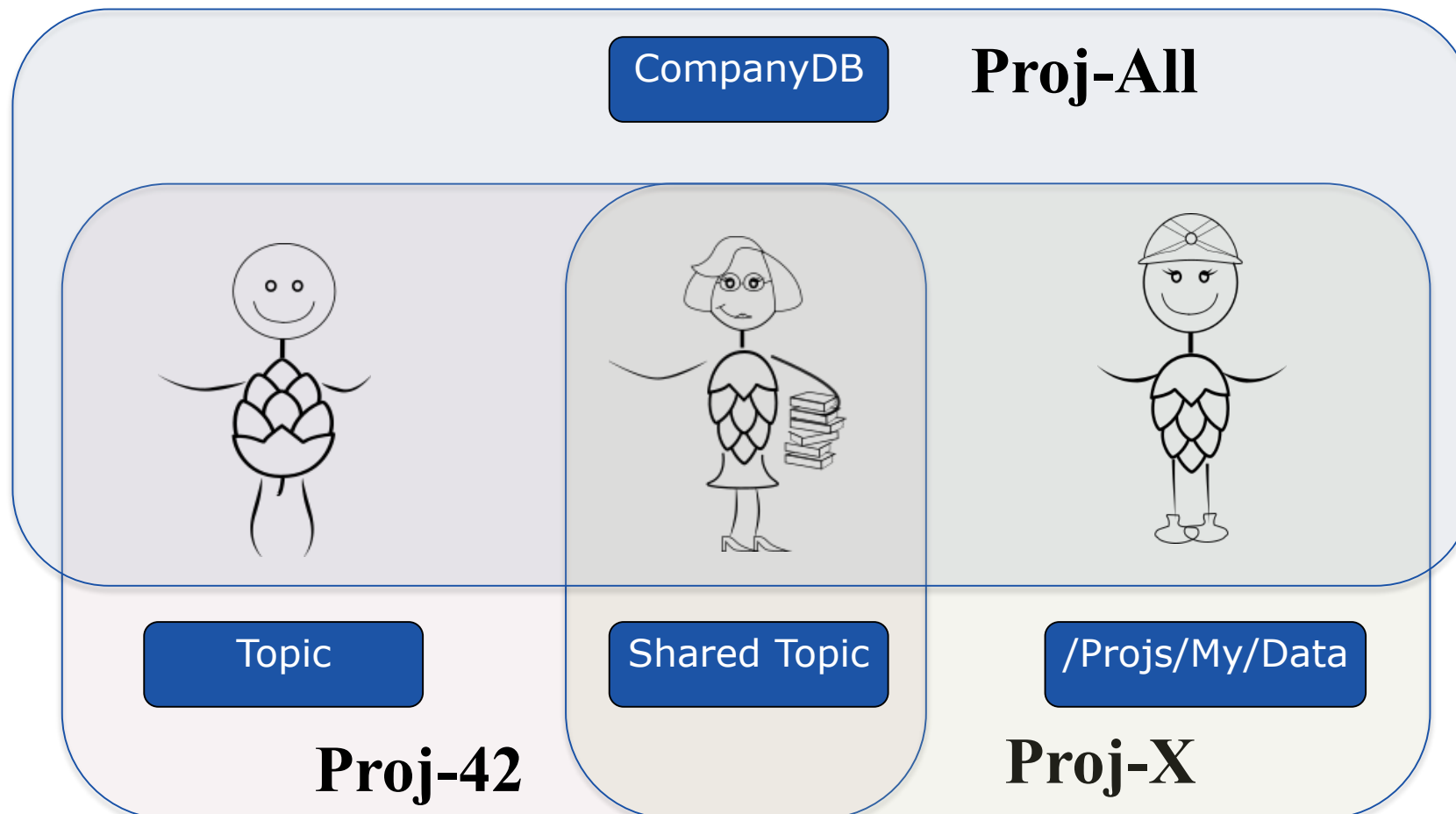
Ostrich Day: 2018-05-25



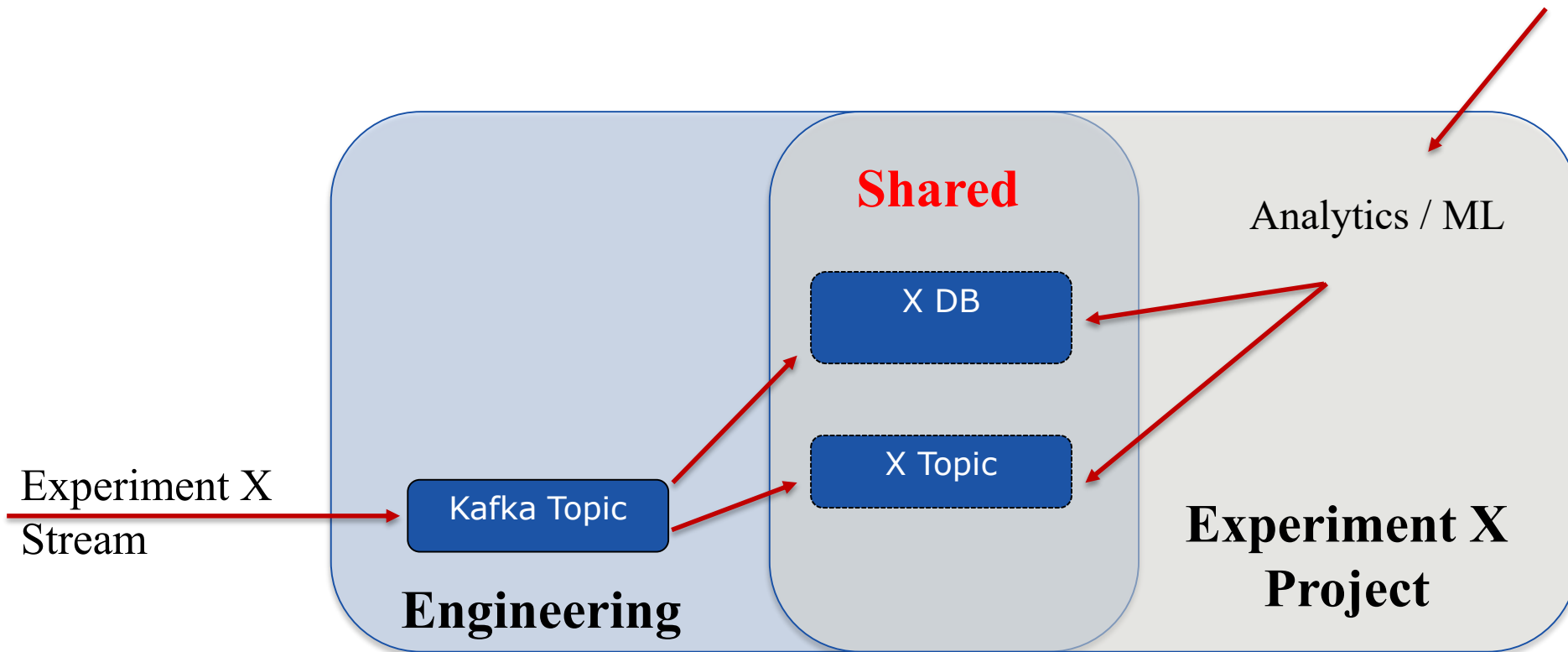
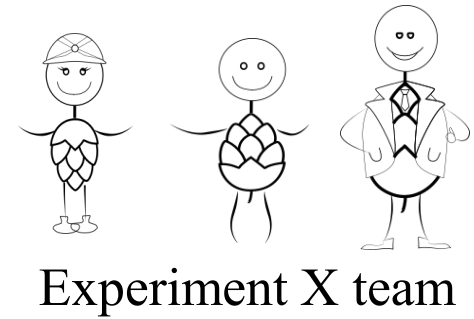
<http://www.computerweekly.com/news/560295538/D-Day-for-GDPR-is-25-May-2018>

Projects for Software-as-a-Service

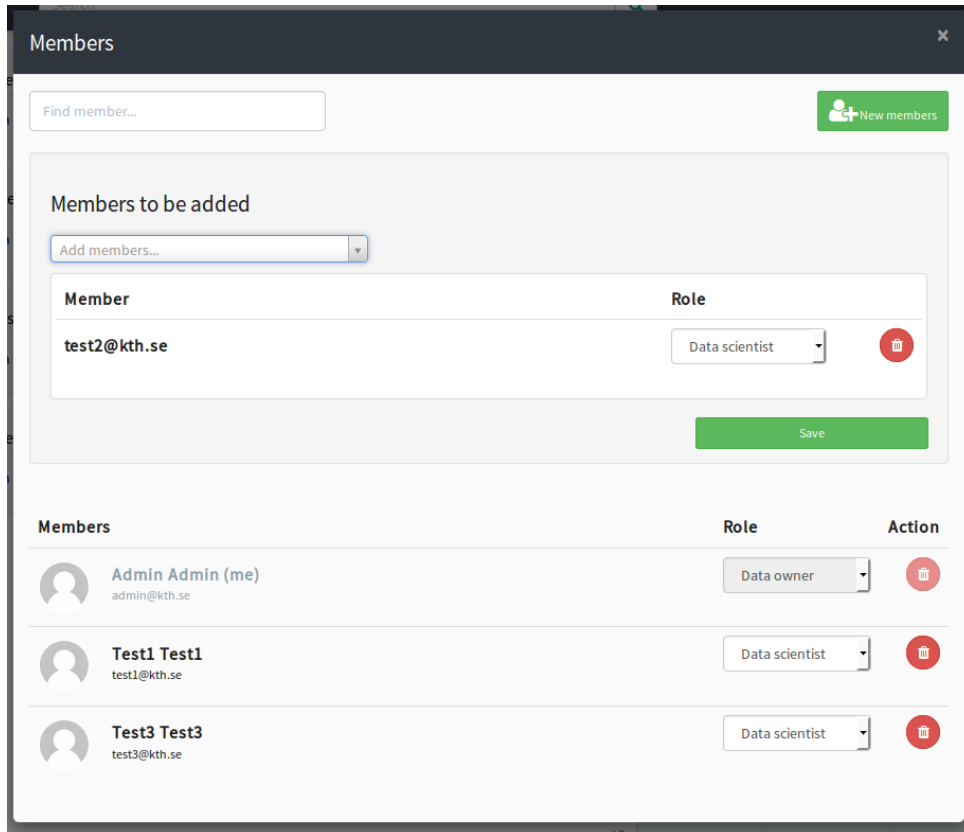
A **Project** is a Grouping of **Users** and **Data**



Hopsworks Projects in Practice



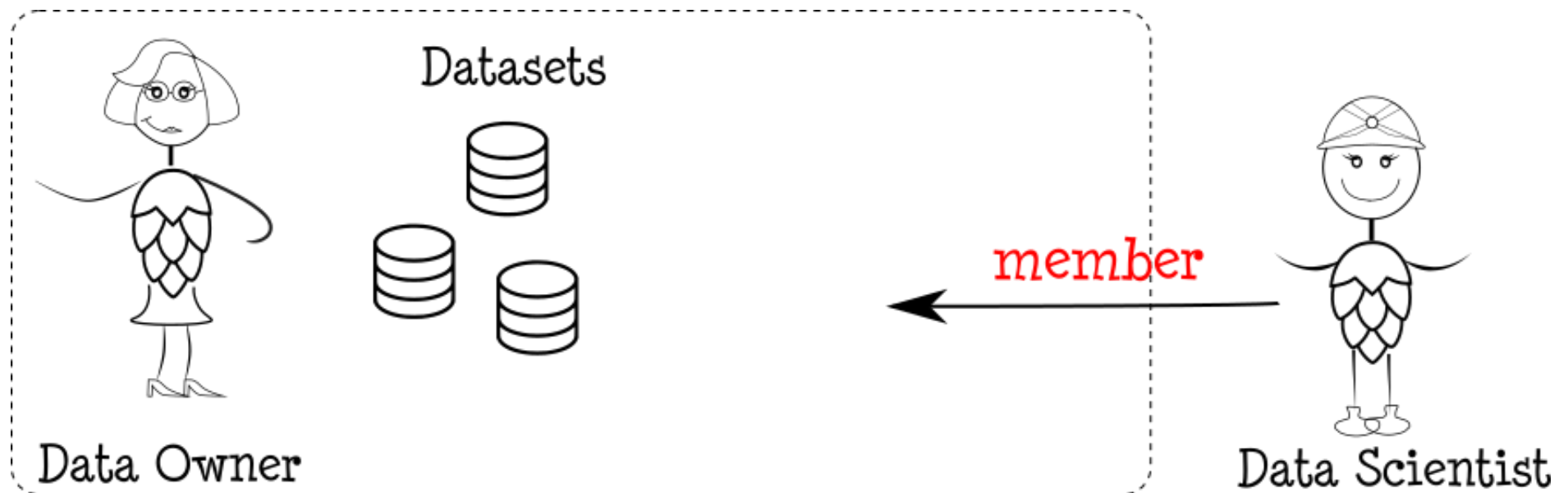
Project Roles



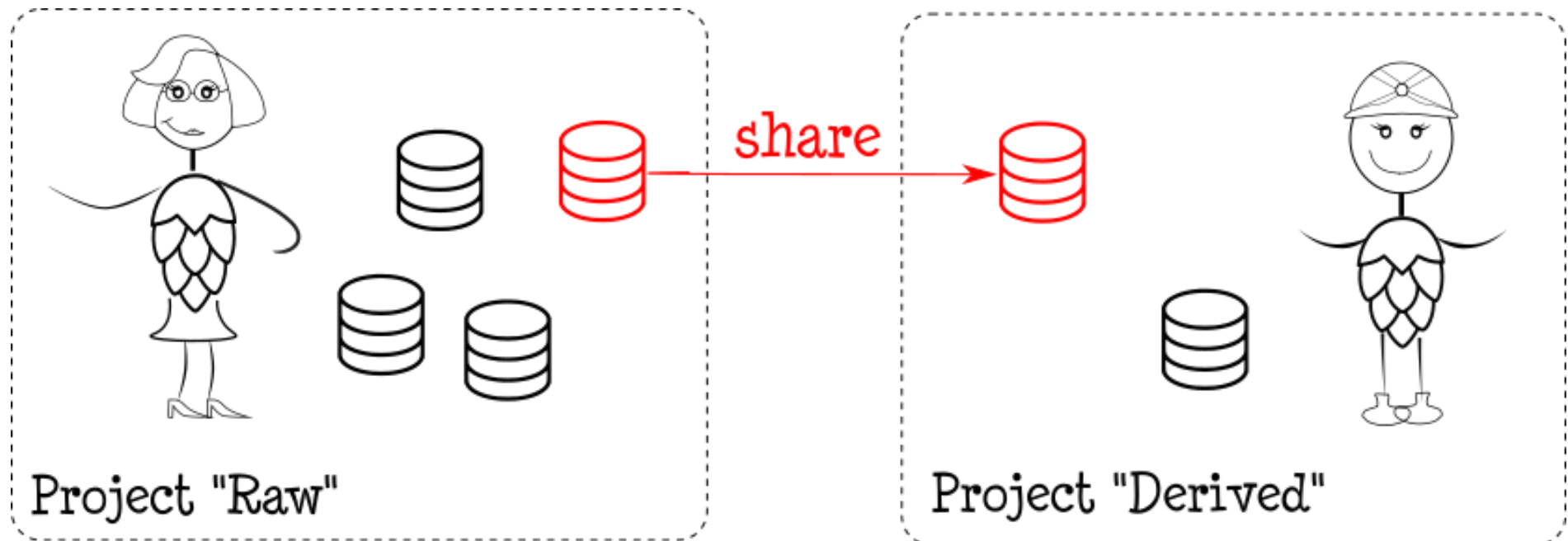
- Data Owner Privileges
 - Import/Export data
 - Manage Membership
 - Share DataSets, Topics
- Data Scientist Privileges
 - Write and Run code

We delegate administration of privileges to users

Manage Projects like GitHub

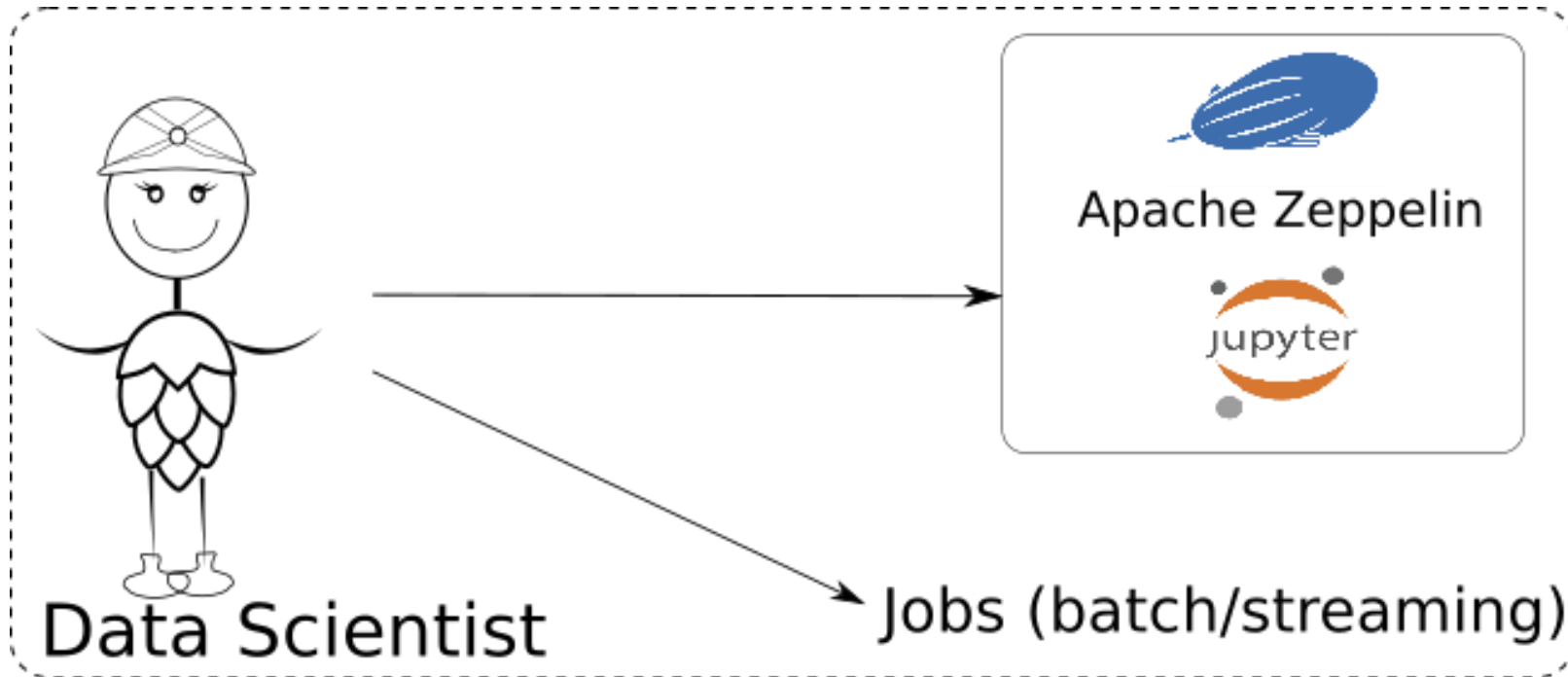


Share like in Dropbox

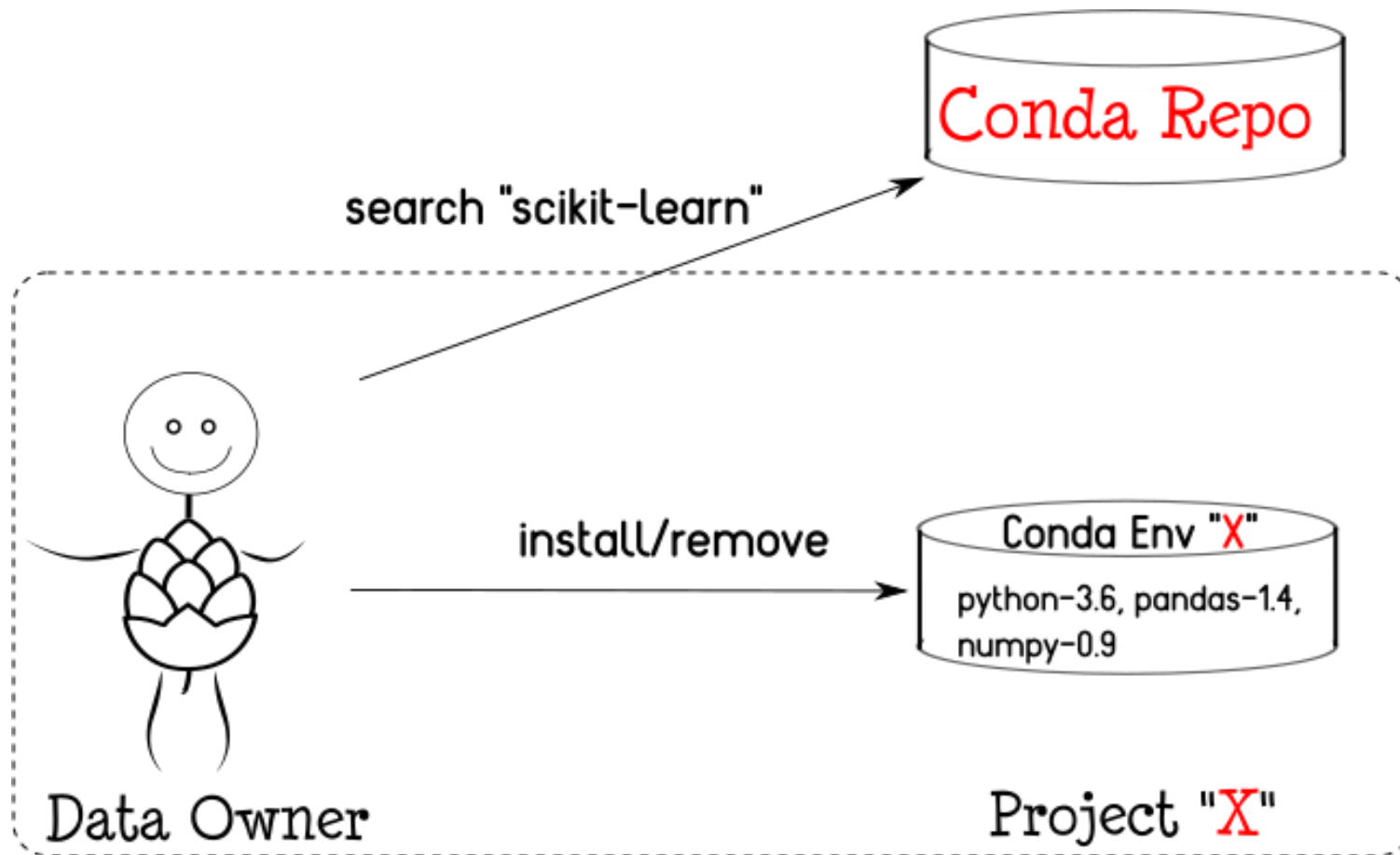


Share any Data Source/Sink: HDFS Datasets, Kafka Topics, etc

Workflow/Jobs and Notebook Support



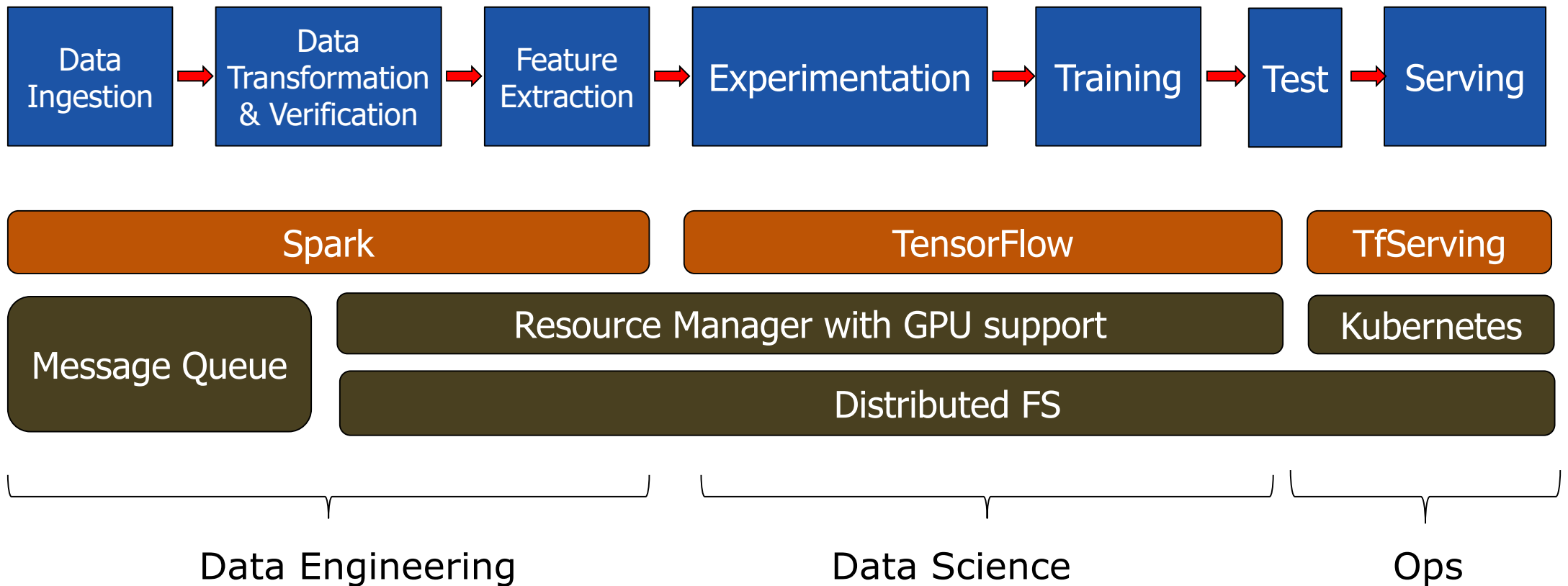
Custom Python Environments with Conda



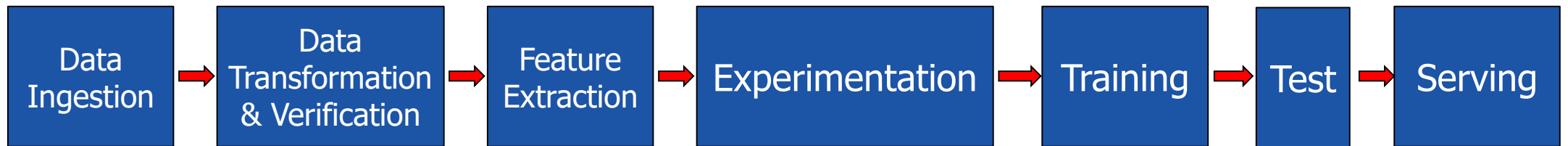
Python libraries are usable by Spark/Tensorflow

Machine Learning Pipelines in Hops

A Scale-Out Machine Learning Pipeline



Hops Small Data ML Pipeline



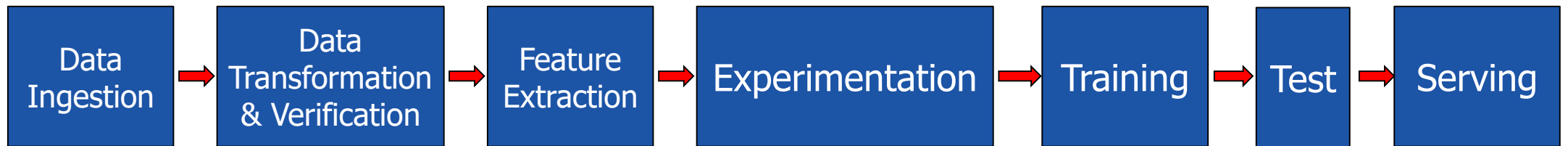
TensorFlow

TfServing

Hops (Kafka/HopsFS/Spark/TensorFlow/Kubernetes)

Project Teams (Data Engineers/Scientists)

Hops Big Data ML Pipeline



PySpark

TensorFlow

TfServing

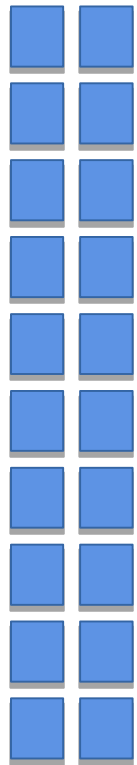
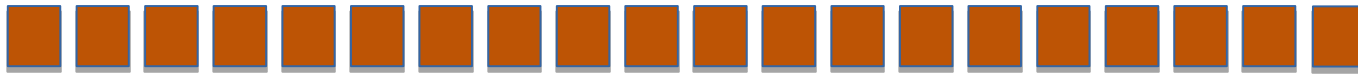
Hops (Kafka/HopsFS/Spark/TensorFlow/Kubernetes)

Project Teams (Data Engineers/Scientists)

Parallel Experiments



Hops



The Outer Loop (hyperparameters):

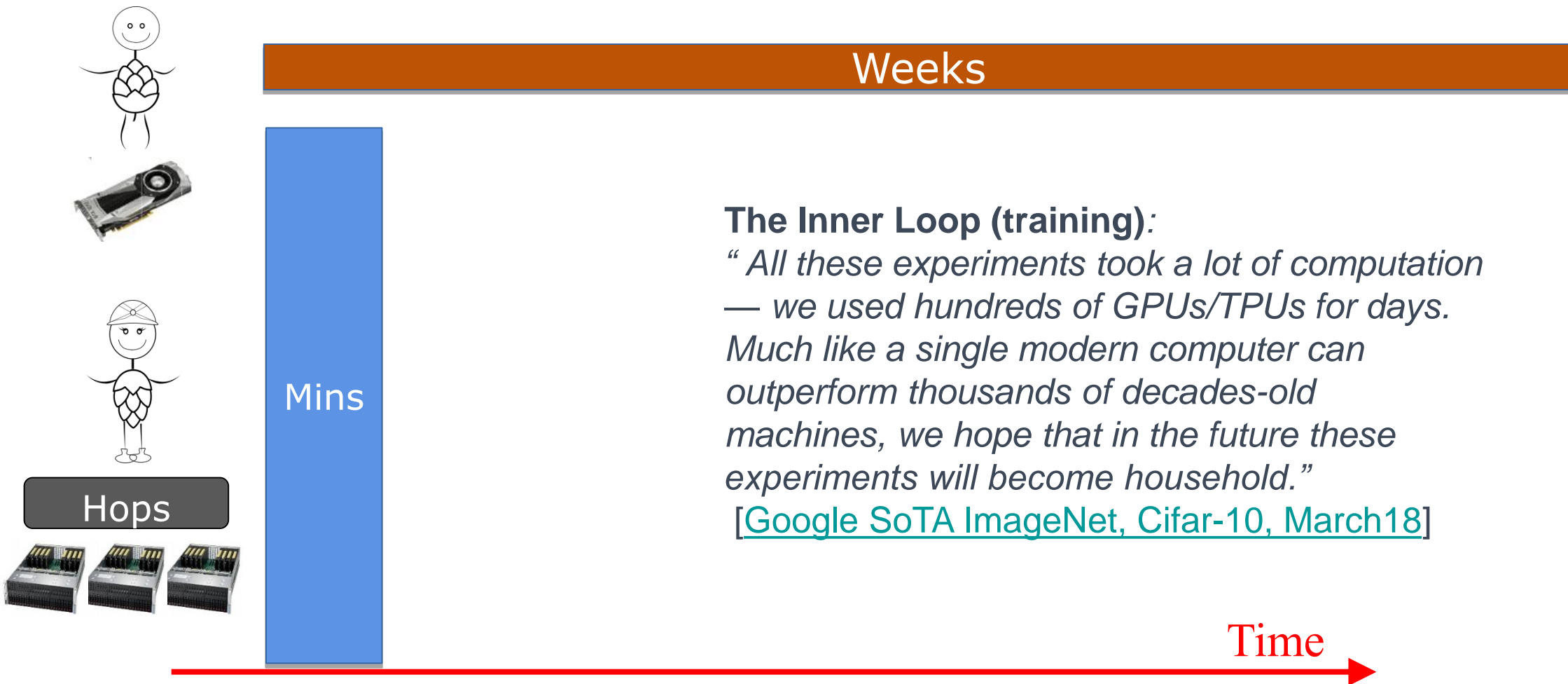
“I have to run a hundred experiments to find the best model,” he complained, as he showed me his Jupyter notebooks. “That takes time. Every experiment takes a lot of programming, because there are so many different parameters.

[\[Rants of a Data Scientist\]](#)

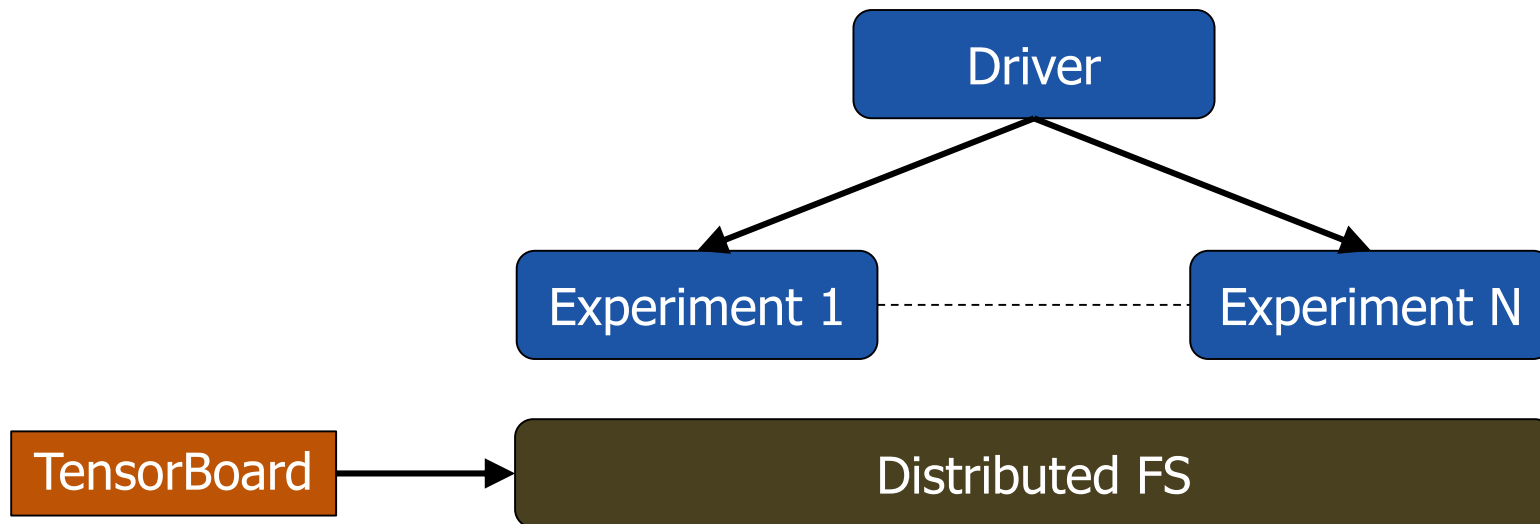
Time



Distributed Training



Need for a Distributed Filesystem



Training/test datasets,
experiment results,
experiment configurations,
model checkpoints,
hyperparameter optimization.

Coding Machine Learning Pipelines in Hops

Jobs or Notebooks

- Hopsworks Jobs
 - REST API
 - Schedule-able
 - PySpark/Scale/Java
 - Workflow API for chaining Jobs

- Jupyter Kernels
 - sparkmagic (livy)
 - python



- Notebooks in HDFS
 - Easily shared in a project
 - hdfscontents plugin

New job

Search:

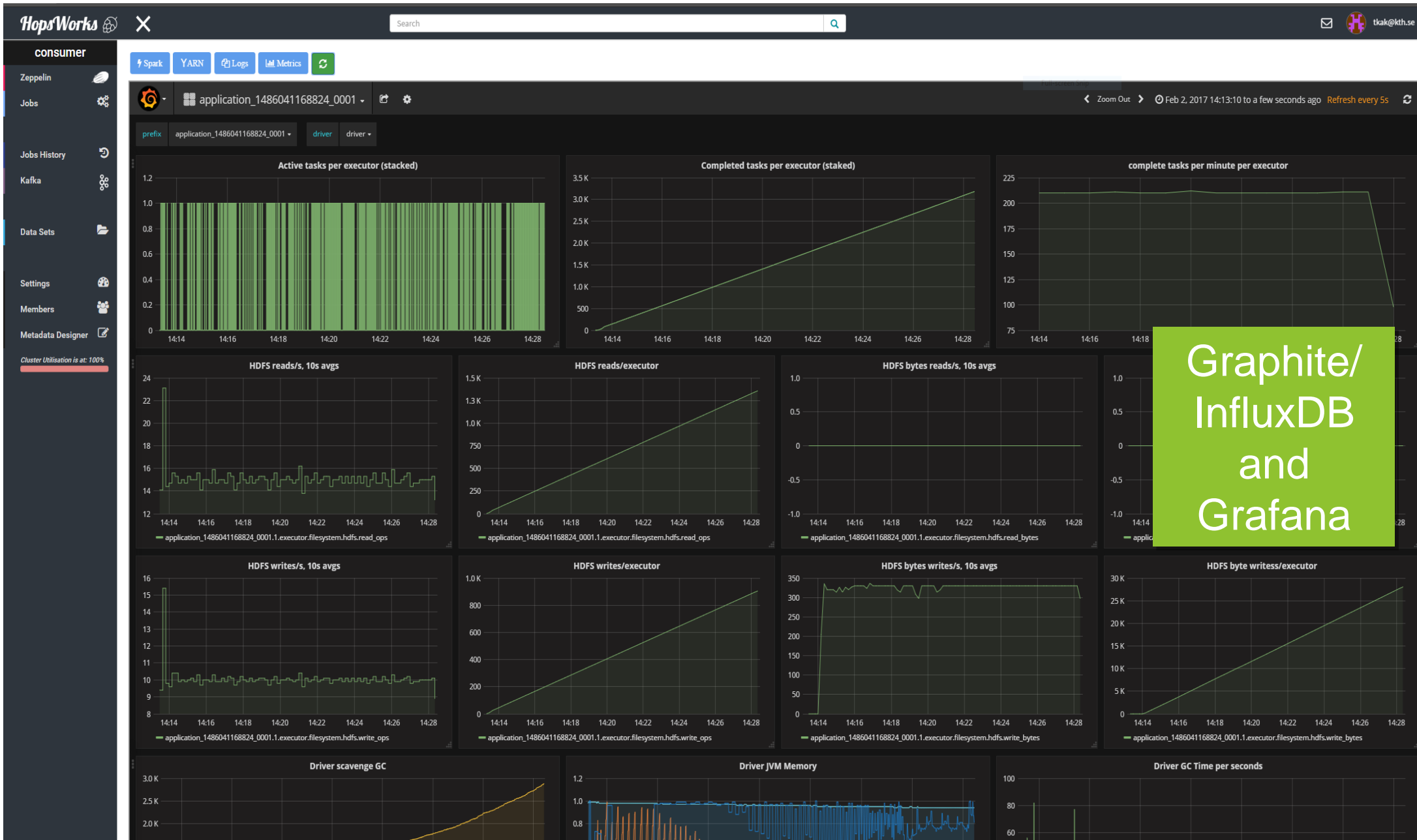
jobs per page:

	Name	Created on	Type	Owner	Submitted at	State	Status	Progress	Duration	Actions
<input type="button" value="Run"/>	Job-1521630234.799	Mar 21, 2018 12:04:02 PM	SPARK	James Dowling	Wed Mar 21 12:04:05 CET 2018	Finished	Succeeded	<div style="width: 100%;"><div style="width: 100%;"></div></div> 100%	00:21	<input type="button" value="Delete"/> <input type="button" value="View"/> <input type="button" value="More"/>

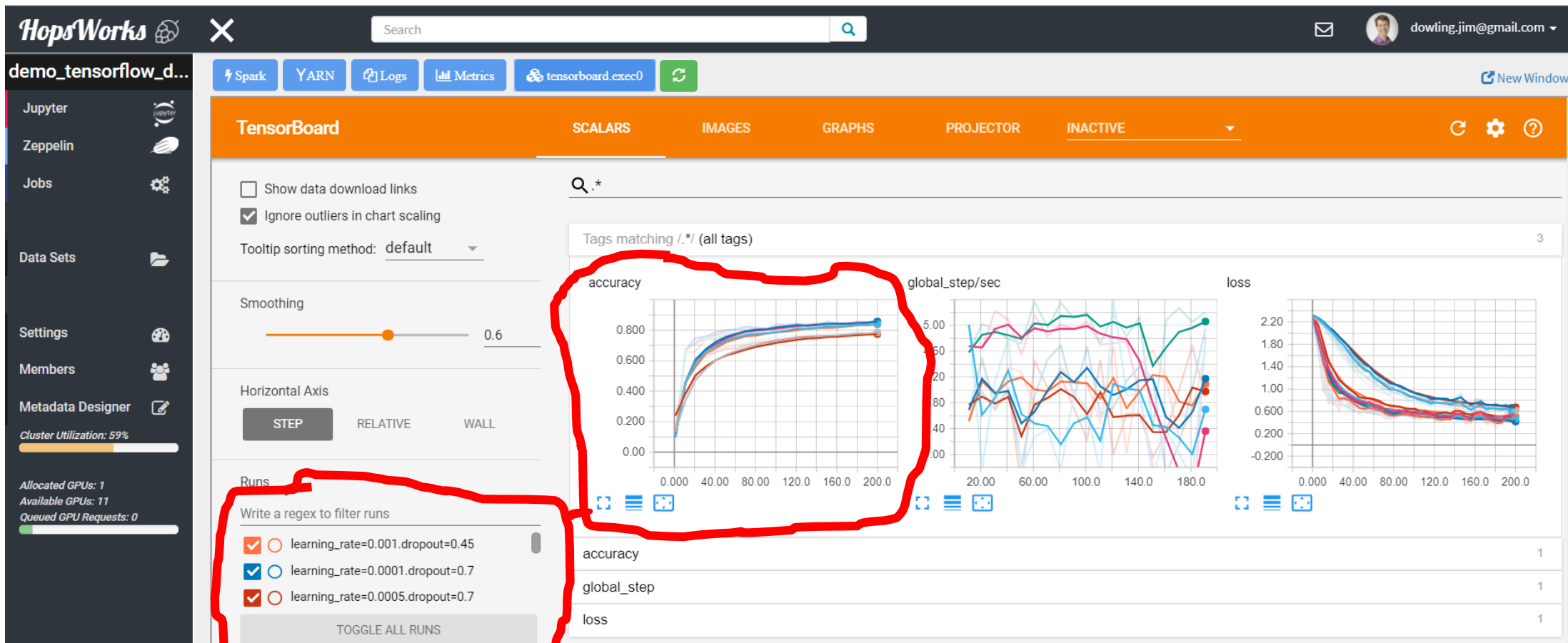
Other Development Tools

- Real-time Logging with Kibana
- Performance Monitoring with Grafana (InfluxDB)
- TensorFlow Debugging with TensorBoard
- SparkMeasure (from CERN)
- Dr Elephant (should be back soon with Spark 2.3)

Resource Monitoring/Alerting



TensorBoard



Hyperparam Opt Results Visualization

HopsWorks Beta

SIGN IN TO CONTINUE.

jdowling@kth.se

.....

[Need support?](#)

[Login help?](#)

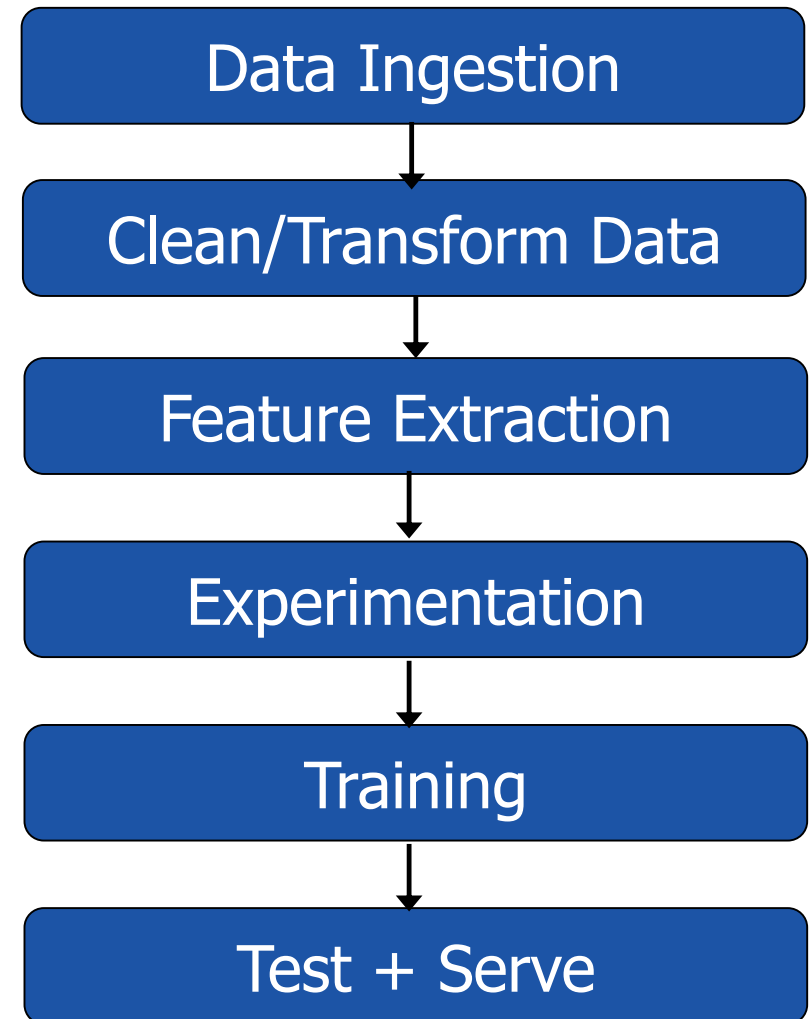
Login

Register



Hops API

- Python (also Java/Scala)
 - Manage tensorboard, Load/save models in HDFS
 - Horovod, TensorFlowOnSpark
 - Parallel experiments
 - Hyperparameter search0
 - Model Architecture Search with Genetic Algorithms
 - Secure Streaming Analytics with Kafka/Spark/Flink
 - SSL/TLS certs, Avro Schema, Endpoints for Kafka/Zookeeper



Kafka Self-Service UI

Manage & Share

- Topics
- ACLs
- Avro Schemas

Data Ingestion

Clean/Transform Data

Feature Extraction

Experimentation

Training

Test + Serve

The screenshot displays the HopsWorks Kafka Self-Service UI. The interface includes a top navigation bar with the HopsWorks logo, a search bar, and a user profile for 'admin@kth.se'. A left sidebar contains navigation options: producer, Zeppelin, Jobs, Jobs History, Kafka, Data Sets, Settings, Members, and Metadata Designer. The main content area shows the 'Topics' tab for a topic named 'hellotopic'. It indicates '1 of 10 topics in use' and provides a 'New Topic +' button. Below this, there are columns for 'Topic Name', 'Schema (v)', 'ACL', 'Share', 'Advanced', and 'Remove'. The 'hellotopic' entry shows a schema of 'kafkaschema (1)', an ACL with a '+' icon, a share icon, an advanced icon, and a remove icon. A detailed table below lists project permissions for 'producer' and 'consumer' users, including their UserEmail, Permission, Operation, Host, Role, and Remove/Edit actions. At the bottom, a table shows partition details for 'hellotopic', including Partition id, Partition leader, Partition replicas, and Insync replicas.

Project	UserEmail	Permission	Operation	Host	Role	Remove	Edit
producer	admin@kth.se	allow	*	*	*		
consumer	tkak@kth.se	allow	*	*	*		

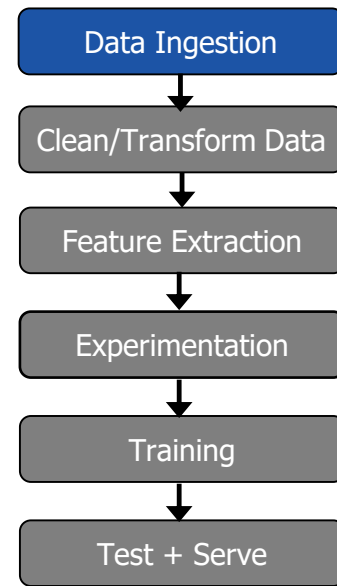
Partition id	Partition leader	Partition replicas	Insync replicas
1	10.0.2.15	["10.0.2.15"]	["10.0.2.15"]
0	10.0.2.15	["10.0.2.15"]	["10.0.2.15"]

Data Ingestion (Kafka)

The Hops API simplifies consuming events from and producing events to Kafka.

```
Properties props = new Properties();
props.put(ProducerConfig.BOOTSTRAP_SERVERS_CONFIG, brokerList);
props.put(SCHEMA_REGISTRY_URL, restApp.restConnect);
props.put(ProducerConfig.KEY_SERIALIZER_CLASS_CONFIG,
org.apache.kafka.common.serialization.StringSerializer.class);
props.put(ProducerConfig.VALUE_SERIALIZER_CLASS_CONFIG,
io.confluent.kafka.serializers.KafkaAvroSerializer.class);
props.put("producer.type", "sync");
props.put("serializer.class", "kafka.serializer.StringEncoder");
props.put("request.required.acks", "1");
props.put("ssl.keystore.location", "/var/ssl/kafka.client.keystore.jks")
props.put("ssl.keystore.password", "test1234")
props.put("ssl.key.password", "test1234")
ProducerConfig config = new ProducerConfig(props);
String userSchema = "{\"namespace\": \"example.avro\", \"type\":
\"record\", \"name\": \"User\", \" +
    \"fields\": [{\"name\": \"name\", \"type\":
    \"string\"}]}}";
Schema.Parser parser = new Schema.Parser();
Schema schema = parser.parse(userSchema);
GenericRecord avroRecord = new GenericData.Record(schema);
avroRecord.put("name", "testUser");
Producer<String, String> producer = new Producer<String,
String>(config);
ProducerRecord<String, Object> message = new
ProducerRecord<>("topicName", avroRecord );
producer.send(data);
```

```
SparkProducer producer =
HopsUtil.getSparkProducer();
```



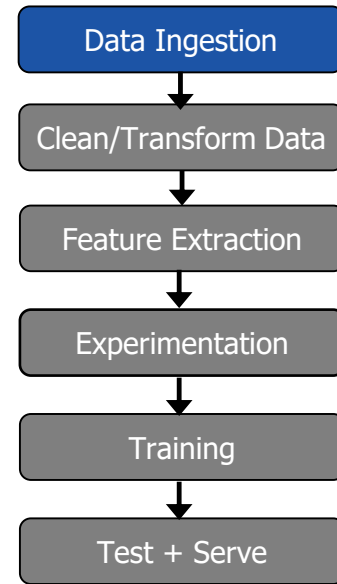
Hive LLAP vs SparkSQL

- Hive LLAP

- ORC format
- Fast startup (LLAP Daemons)
- Zeppelin support
- Integration with BI tools (Tableau, Qlik, etc)

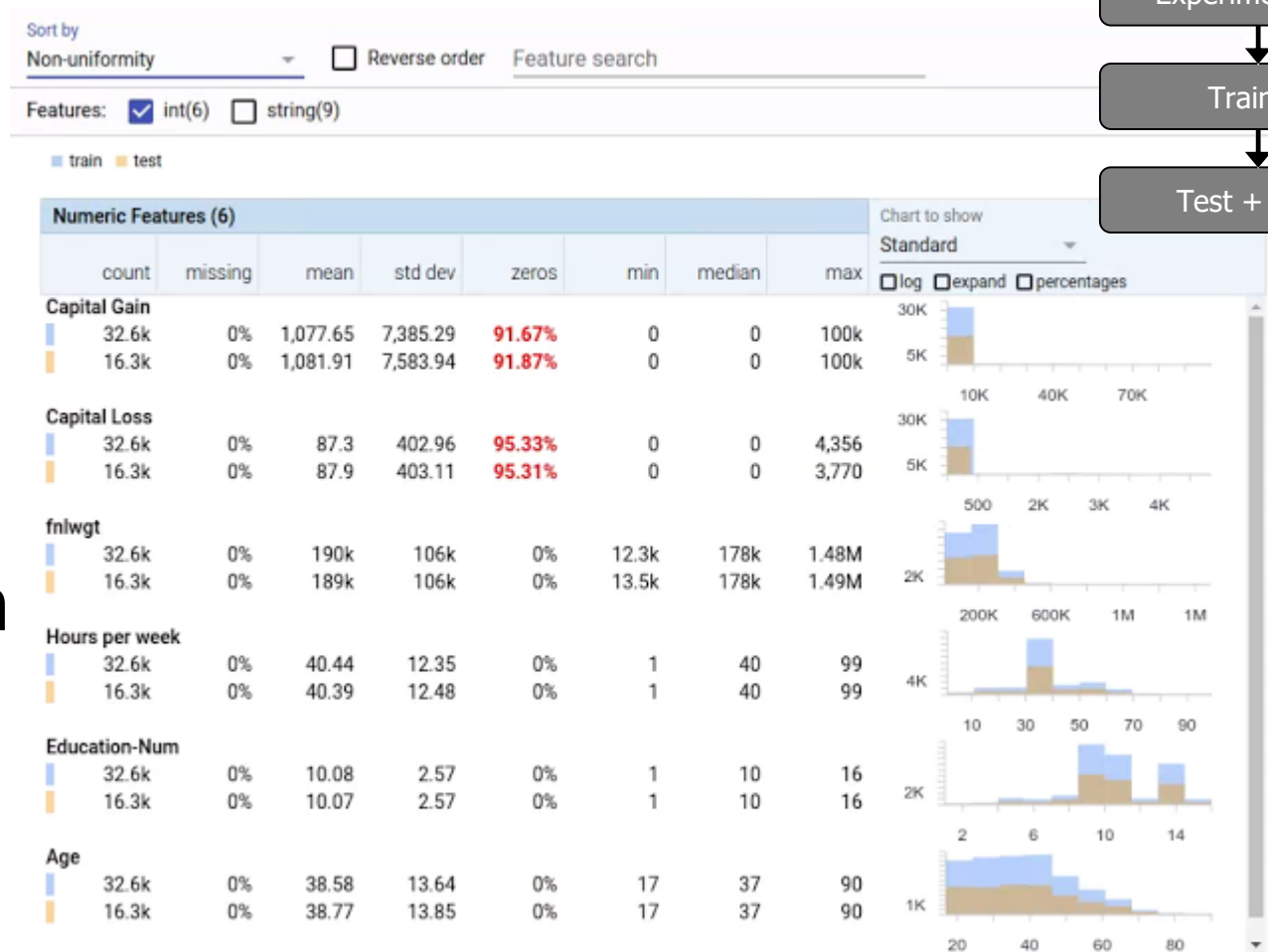
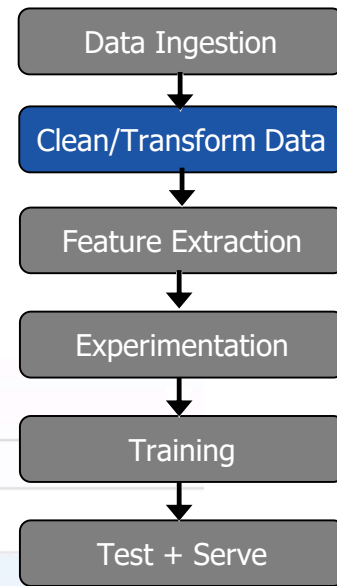
- Spark SQL

- Parquet Format
- Slow startup w/ YARN
- Integrated with SparkML / GraphX
- DataFrames can be written as TfRecords



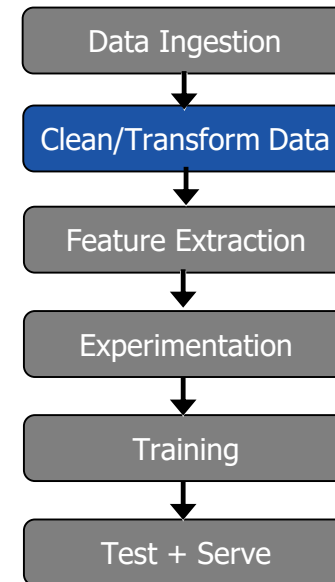
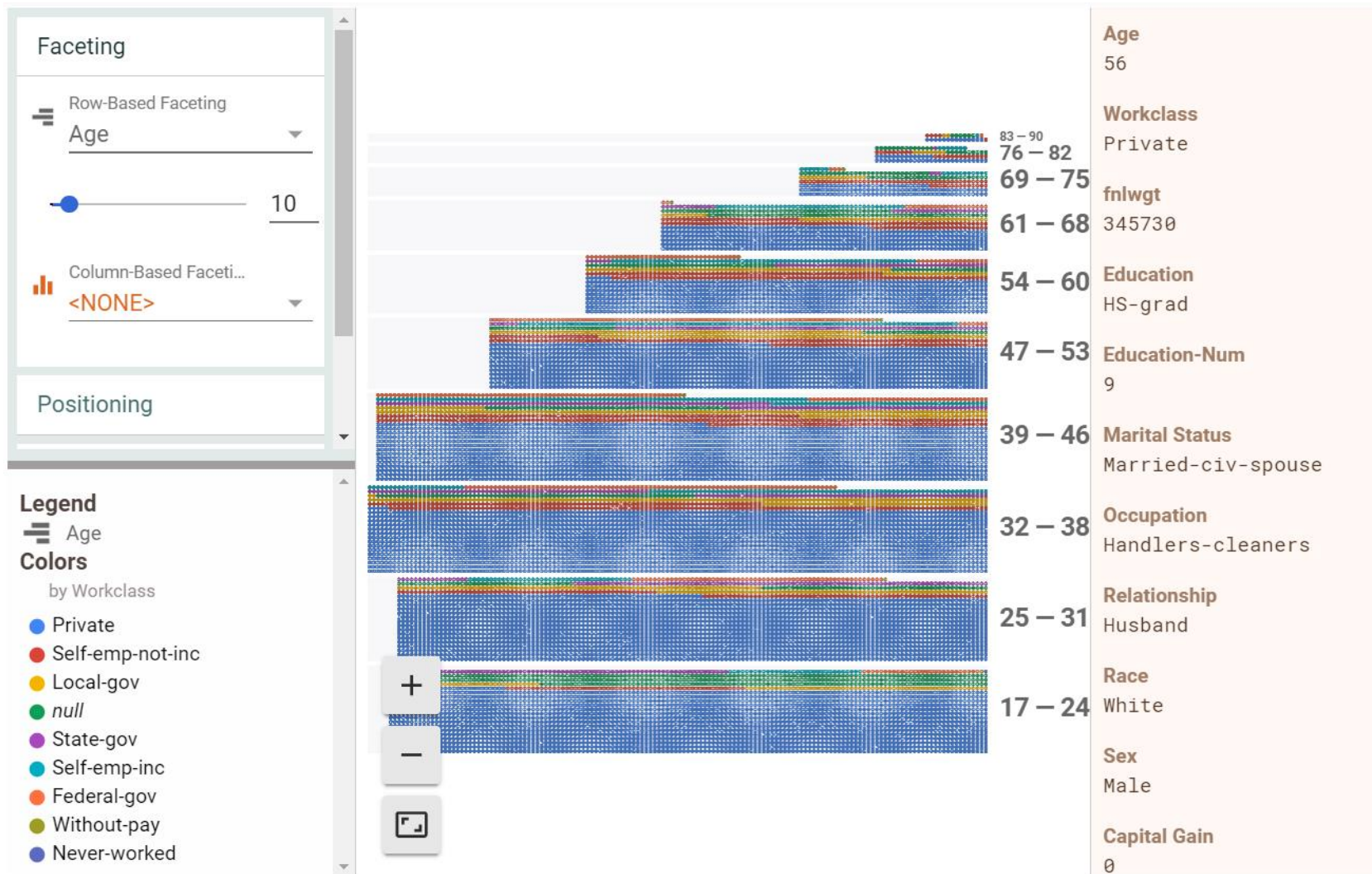
Google Facets Overview

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

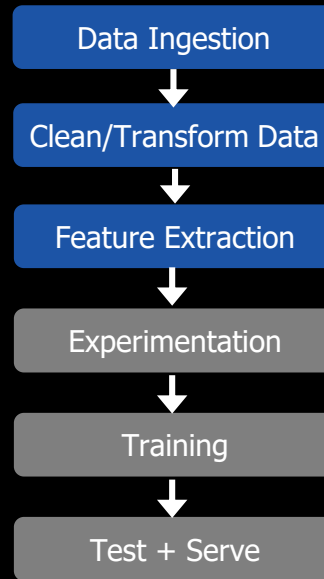


Google Facets Dive

- Visualize the relationship between the data points across the different features of a dataset
- Facets Dive expects input dataset as json



Data Ingestion (HopsFS) and Google Facets



```
import hops.hdfs as hdfs

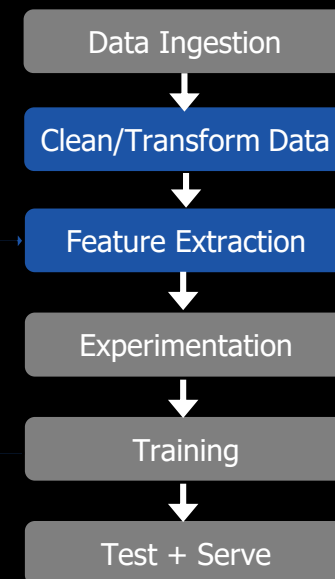
features = ["Age", "Occupation", "Sex", ..., "Country"]
h = hdfs.get_fs()
with h.open_file(hdfs.project_path() +
  "/TestJob/data/census/adult.data", "r") as trainFile:
  train_data =pd.read_csv(trainFile, names=features,
    sep=r'\s*,\s*', engine='python', na_values="?")
with h.open_file(hdfs.project_path() +
  "/TestJob/data/census/adult.test", "r") as testFile:
  test_data =pd.read_csv(testFile, names=features, sep=r'\s*,\s*',
    engine='python', skiprows=[0], na_values="?")

from hops import facets
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_sz):  
    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_sz,  
        num_parallel_batches=4))  
    dataset = dataset.prefetch(4)  
    return dataset
```

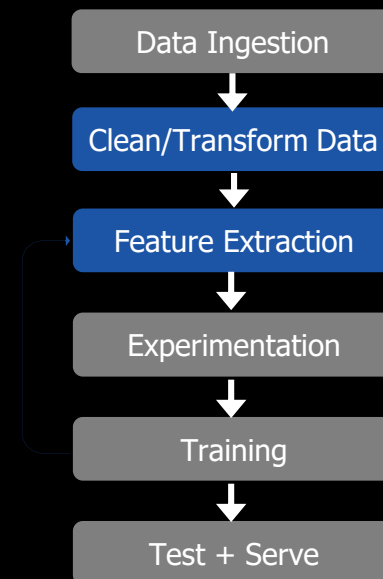


Big Data Preparation with PySpark

```
from mmlspark import ImageTransformer

images = spark.readImages(IMAGE_PATH, recursive = True,
                           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")
```



Hyperparam Opt. with Tf/Spark on Hops

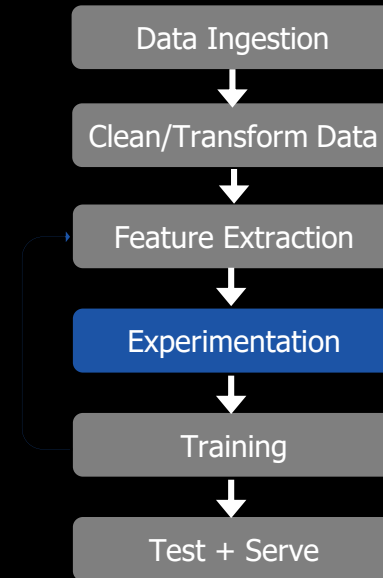
```
from hops import experiment

def model_fn(learning_rate, dropout):
    import tensorflow as tf
    from hops import tensorboard, hdfs, devices
```

[TensorFlow Code here]

```
args_dict = {'learning_rate': [0.001, 0.005, 0.01],
             'dropout': [0.5, 0.6]}
experiment.launch(spark, model_fn, args_dict)
```

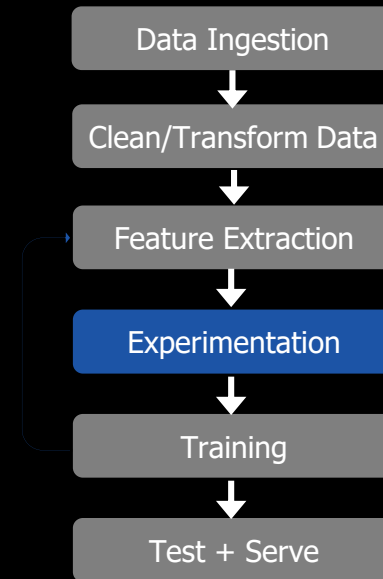
Launch TF jobs in Spark Executors



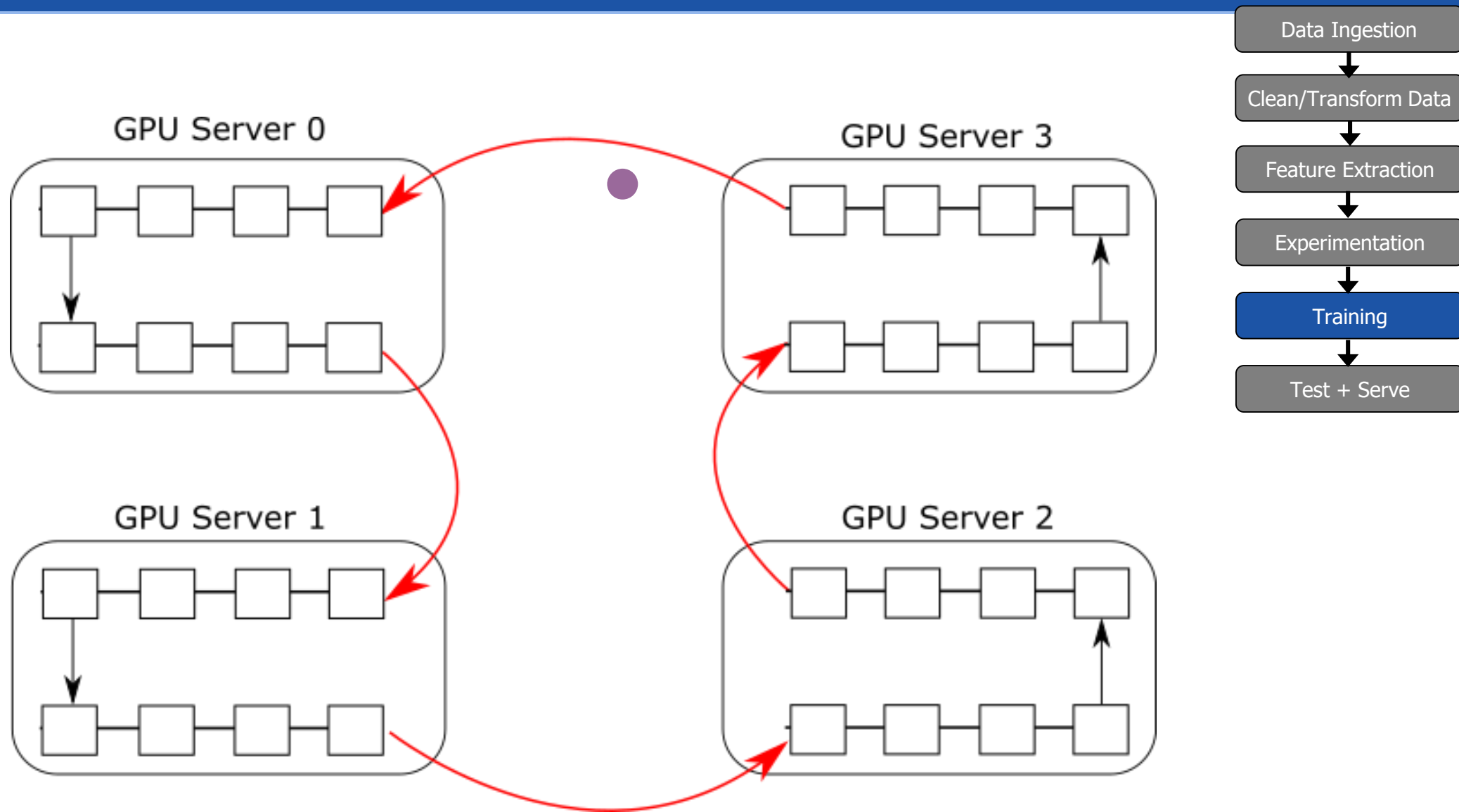
Model Architecture Search Tf/Spark on Hops

```
def model_fn(learning_rate, dropout):  
    import tensorflow as tf  
    from hops import tensorboard, hdfs, devices  
  
    [TensorFlow Code here with Estimator/Experiment]
```

```
from hops import experiment  
boundary_dict = {'learning_rate': [0.005, 0.00005],  
                'dropout': [0.01, 0.99], 'num_layers': [1,3]}  
  
# Differential Evolution searches for good models  
tensorboard_hdfs_logdir =  
experiment.evolutionary_search(spark, wrapper,  
boundary_dict, direction='max', popsize=10, generations=3,  
crossover=0.7, mutation=0.5)
```



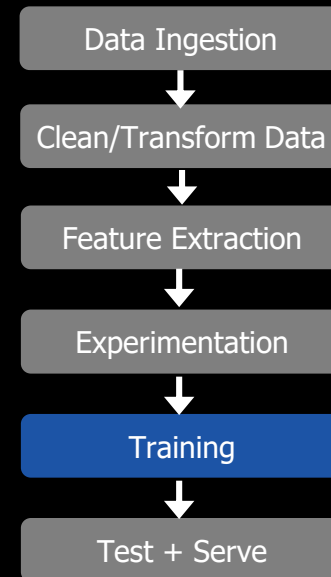
Distributed Training with Ring-AllReduce



Only one slow worker or bus or n/w link is needed to bottleneck training time.

Distributed Training with Horovod on Hops

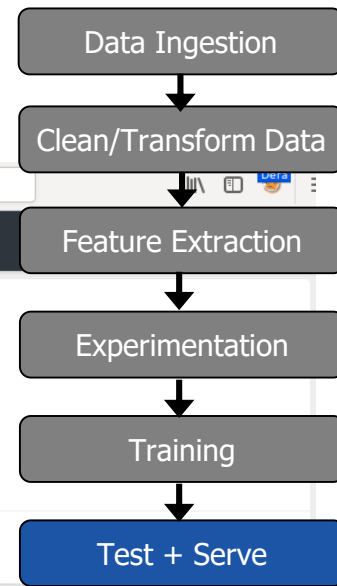
```
def conv_model(feature, target, mode)
  ...
  hvd.init()
  opt = hvd.DistributedOptimizer(opt)
  if hvd.local_rank()==0:
    hooks = [hvd.BroadcastGlobalVariablesHook(0), ..]
    ...
  else:
    hooks = [hvd.BroadcastGlobalVariablesHook(0), ..]
    ...
  from hops import allreduce
  allreduce.launch(spark, 'hdfs:///Projects/.../all_reduce.ipynb')
```



TensorFlow code



TensorFlow Model Serving



Model Enable batching

	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 ...
```

Deep Learning Hierarchy of Scale

Training Time for ImageNet

Minutes

DDL
AllReduce
on GPU Servers

Hours

DDL with GPU Servers
and Parameter Servers

Days/Hours

Parallel Experiments on GPU Servers

Days

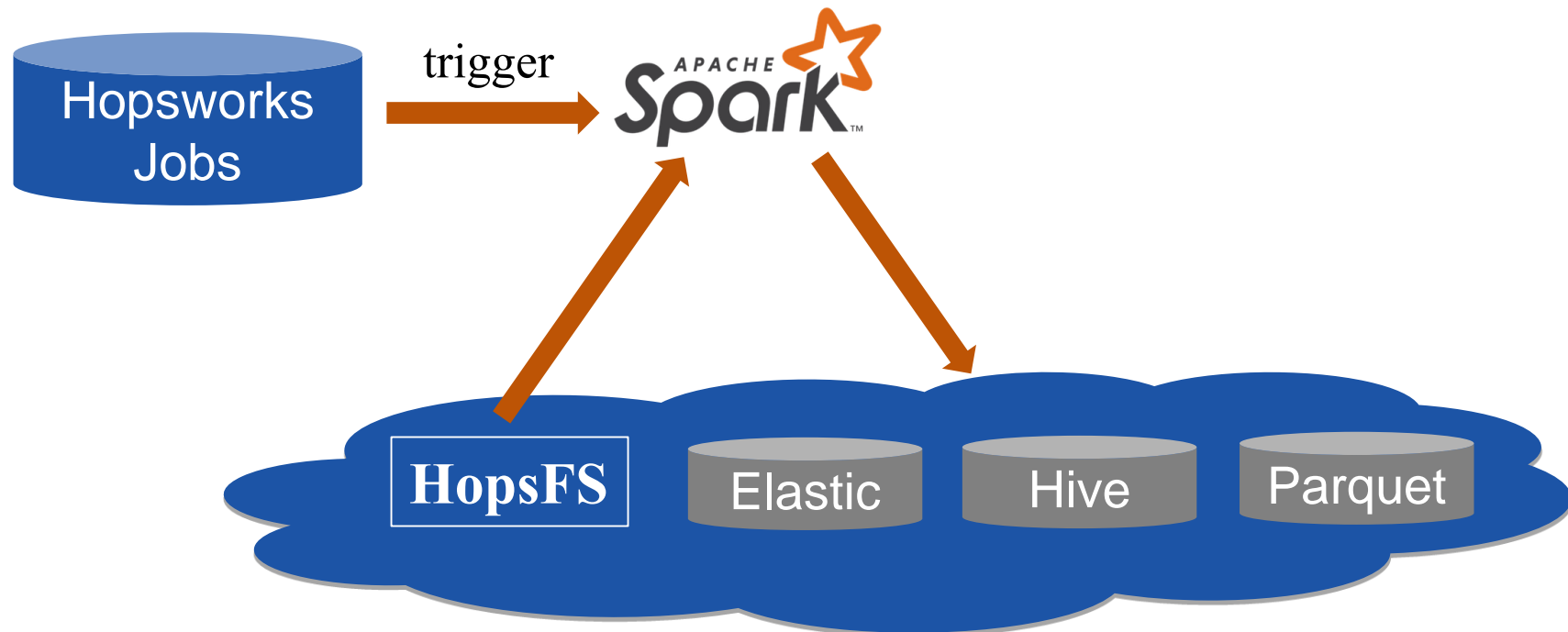
Many GPUs on a Single GPU Server

Weeks

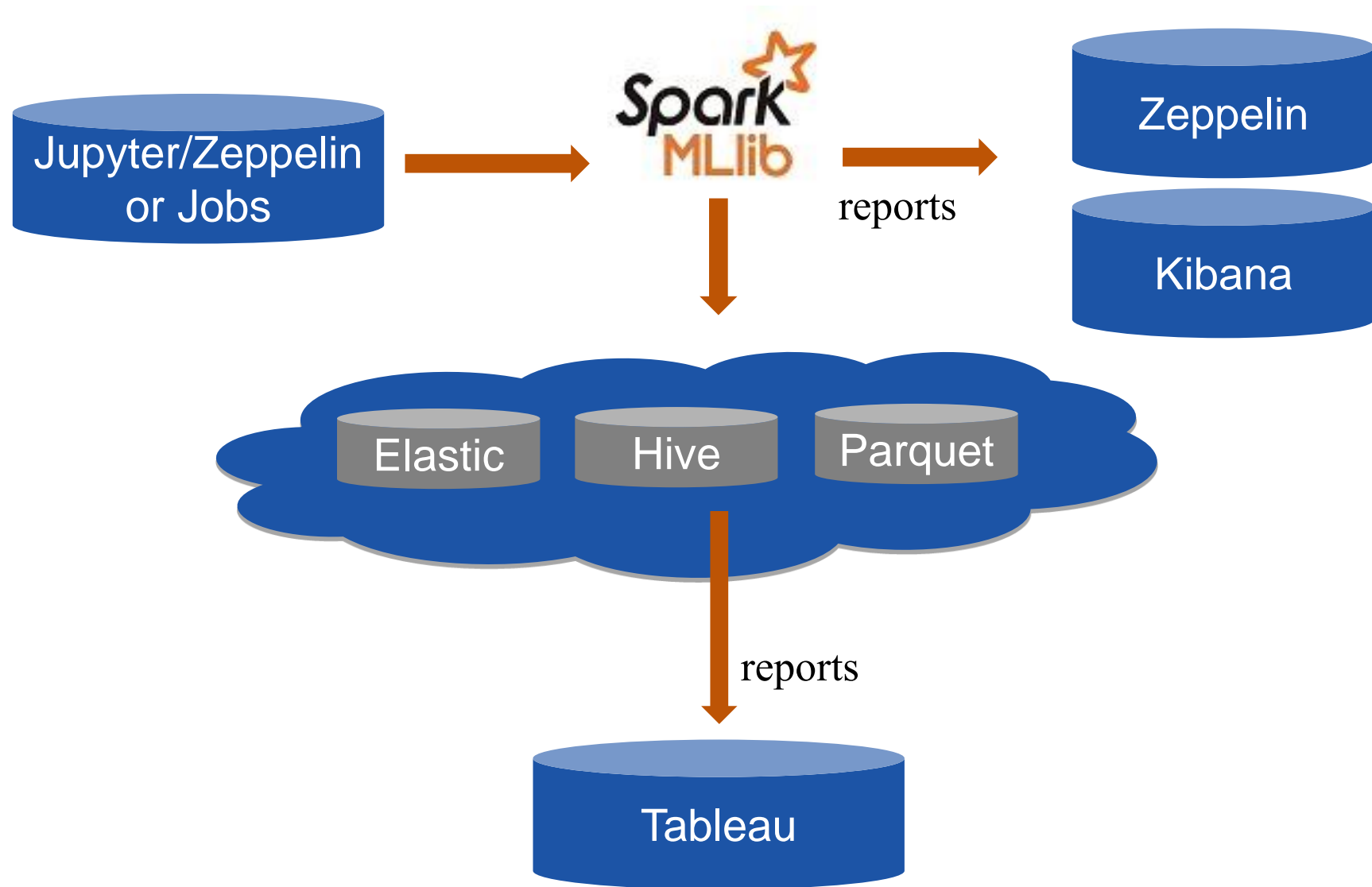
Single GPU

How else is Hopsworks Used?

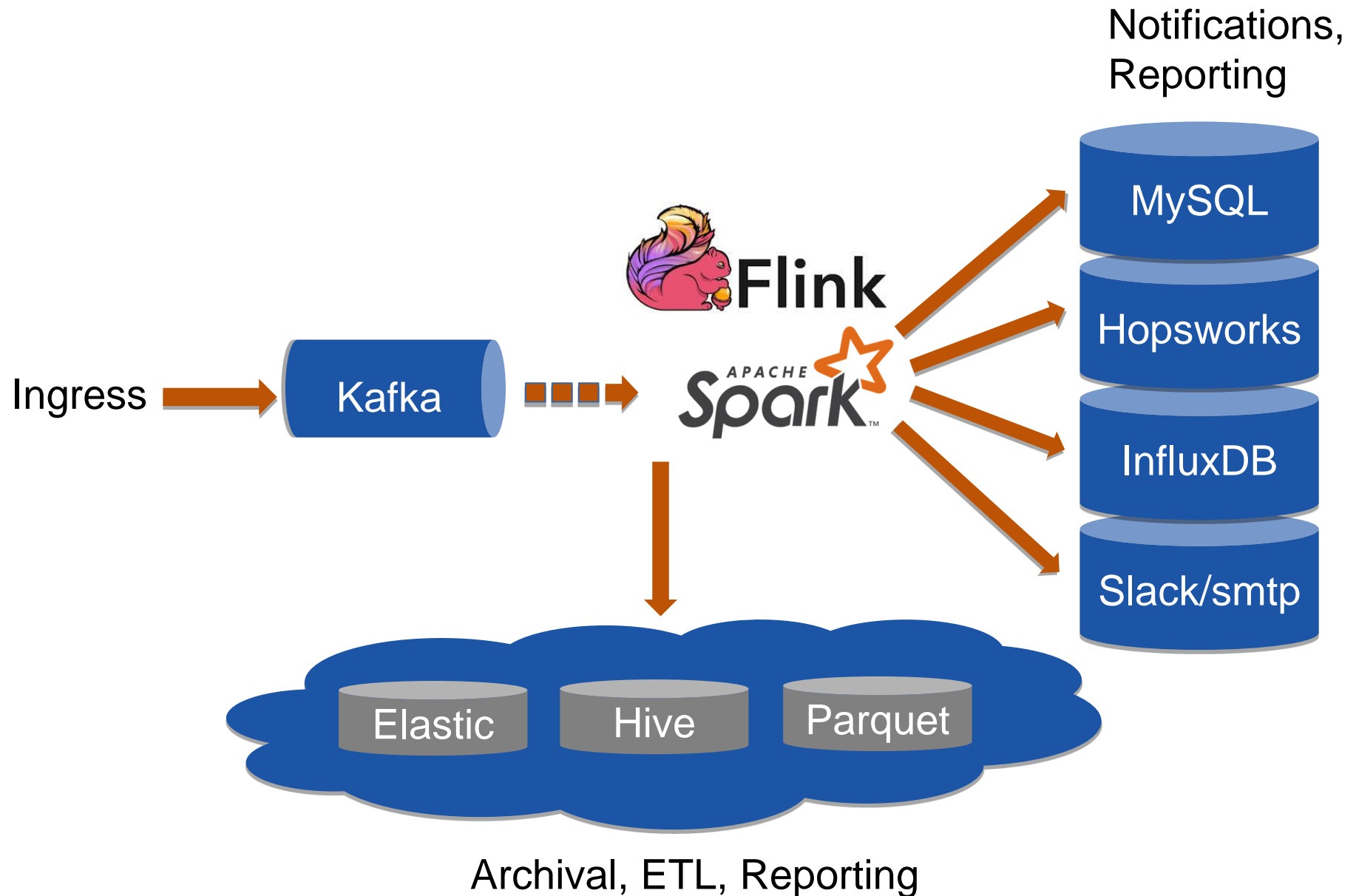
ETL Workloads



Business Intelligence Workloads

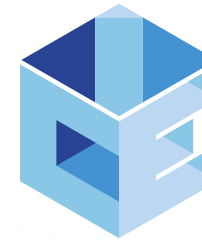


Streaming Analytics in Hopsworks



Hadoop-as-a-Service in Sweden

- www.hops.site
 - 450+ Users
 - ~1000 cores
 - ~12 TB RAM
 - ~1 PB storage
 - ~50 GPUs (Nvidia 1080 Ti)
- **RISE SICS ICE**
 - 250 kW Datacenter, ~1000 servers



Hops Near-Term Roadmap

- Kubernetes/Docker for Model Serving + Jupyter
- PixieDust (Jupyter Plugin)
- Experiment Management
 - Reproducible experiment runs
- Feature Store
- Geographical Replication for HopsFS

Summary

- Europe's Only Hadoop Distribution – Hops Hadoop
 - Fully Open-Source
 - Supports larger/faster Hadoop Clusters with GPUs
- Hopsworks is a new Data Platform built on HopsFS with first-class support for Data Science
 - Spark
 - TensorFlow
 - Support services for ML

The Team

Active:

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



www.hops.io

 @hopshadoop

Alumni:

Vasileios Giannokostas, Johan Svedlund Nordström, Rizvi Hasan, Paul Mälzer, Bram Leenders, Juan Roca, Misganu Dessalegn, K "Sri" Srijevantham, 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.



Karolinska
Institutet



yanzi
networks



Hops

Thank You.

Follow us: @hopshadoop

Star us: <http://github.com/hopshadoop/hopsworks>

Join us: <http://www.hops.io>

Another Talk at 15.00 on Hops Hadoop with Q&A

Room: 513-1-024 (CERN)

<https://indico.cern.ch/event/716789/>