Tools from Cloud Computing Every Economist Should Know

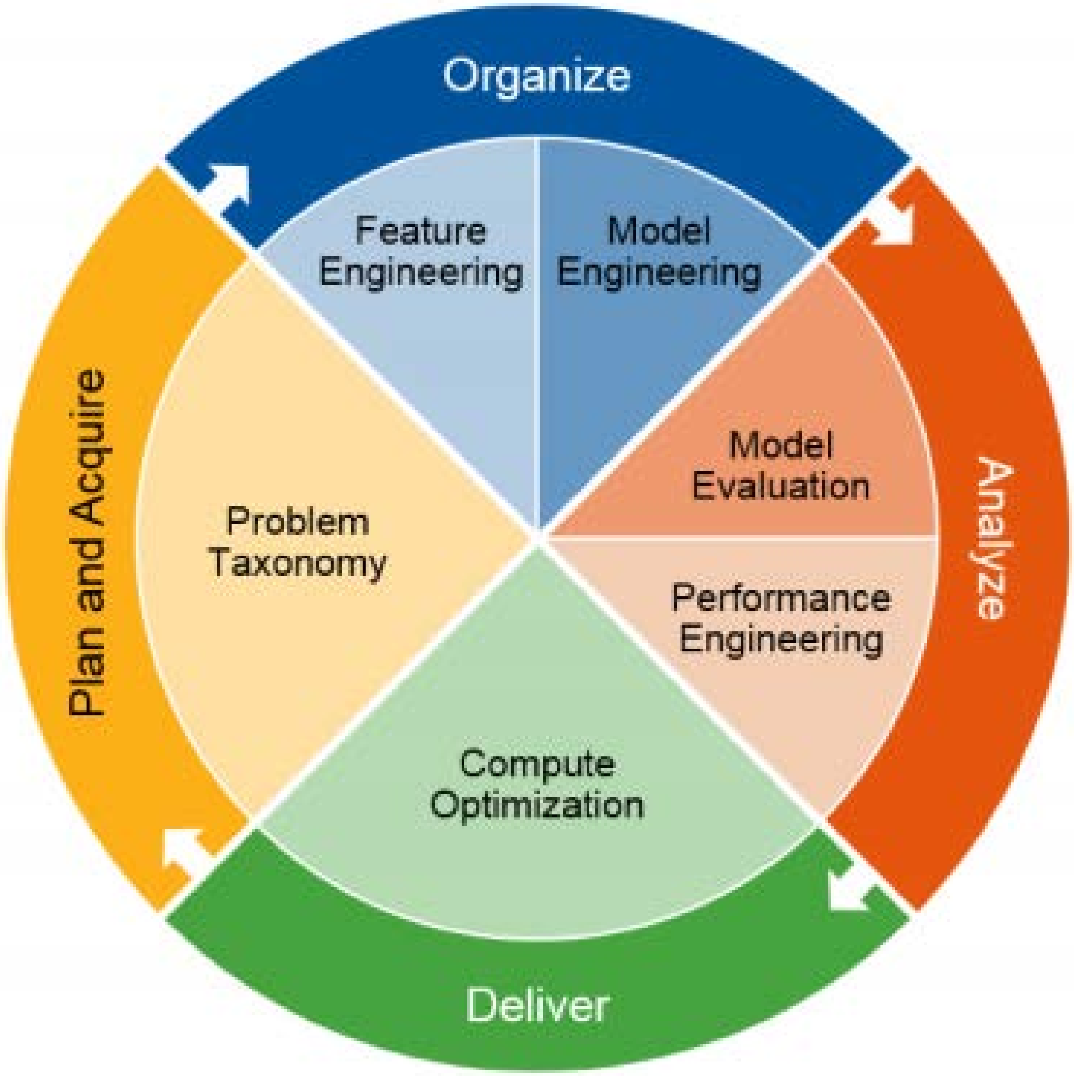
Ashish Mishra
Core AI, Amazon

Nov 4, 2019

Core AI, Amazon Consumer

- Core tenet: “Peak Jumping”
- Intersection of science, engineering and business
- Generating tangible value, and optimizing for Amazon’s customers
- Partner with internal and external teams
  - Supply Chain Optimization
  - Pricing
  - Amazon Fulfilment
  - Finance teams
  - among others
- Growing the scientist community in consumer business

Model Application



Why does Cloud Computing matter?

Big Data

“There is little doubt, at least in our own minds, that over the next decades “big data” will change the landscape of economic policy and economic research. As we emphasized throughout, we don’t think that big data will substitute for common sense, economic theory, or the need for careful research designs. Rather, it will complement them. How exactly remains to be seen.”

Liran Einav and Jonathan Levin, *The data revolution and economic analysis*
Technical report, NBER Innovation Policy and the Economy

https://www.nber.org/papers/w19035

» Scalability

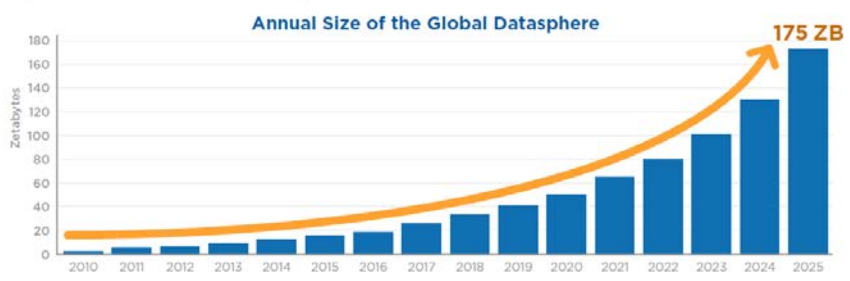


Figure 1 - Annual Size of the Global Datasphere

- Applications of Data Science growing exponentially
- Dedicated hardware is inelastic, and has a short shelf-life

» Technical Innovation

- Qualitatively different models may be more effective in this space

» Collaboration

- Dispersed groups of people can interact virtually
- Easily share code and data in real-time
- Always-on availability and mobility
- Cloud-based workflow and sharing apps also help interaction with business

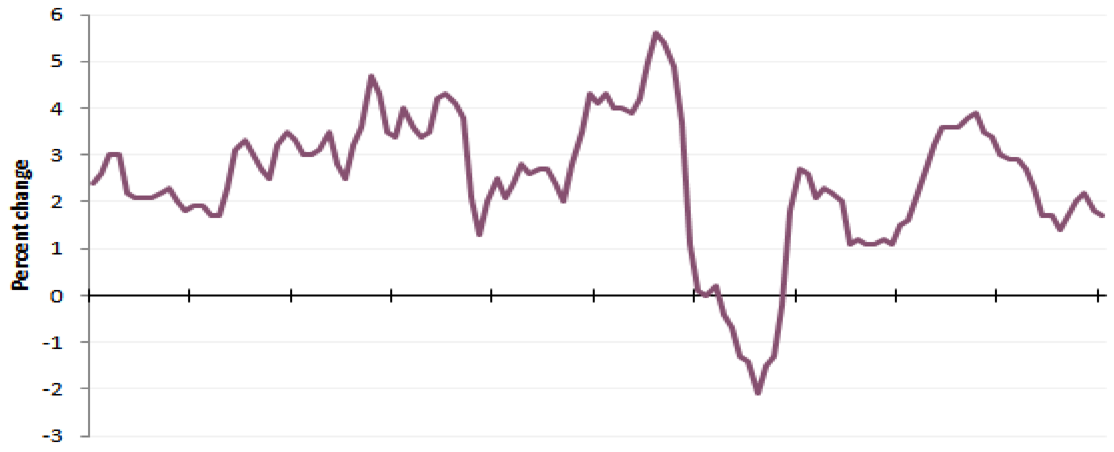
Case Study

- Hedonic Price Regression to construct price indices

$$ln(p_{i,t}) = f(\alpha, x_{i,t}'\beta_t, \gamma_i' f_t, \varepsilon_{i,t})$$

  - β: time varying implicit prices
    γ: latent product attributes
    f: time specific loadings

- After estimating the hedonic models, build Laspeyres, Paasche, and chain weighted indices



Case Study: Hedonic Price Indices

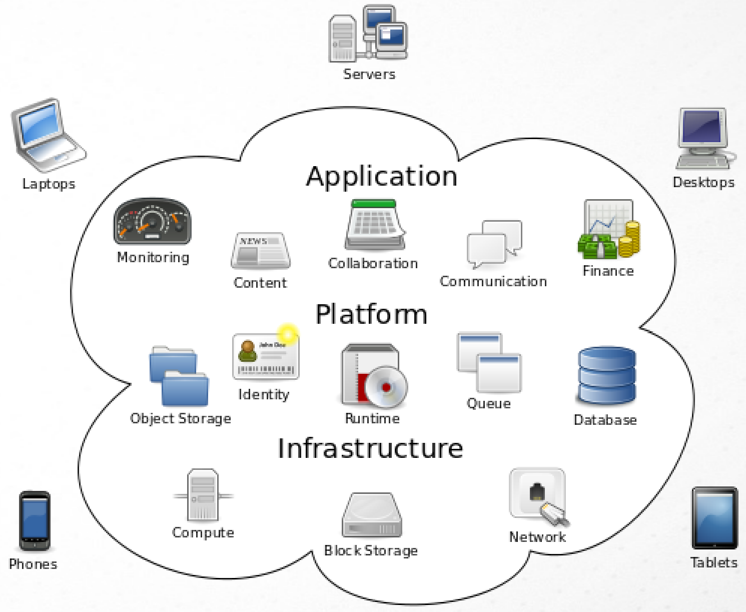
- » Single Business Unit
  - 4 years of history
  - Over 1Bn products
- » Feature engineering
  - Structured text parsing
  - NLP of product attributes
  - Image processing (thumbnails)
  - Behavioral: Clicks, adds, purchases
  - Sales across Amazon.com, FBA, 3rd-party merchants
  - Product search metadata
- » Over 200Tb of feature data

Case Study: Hedonic Price Indices

- Not feasible to manually encode regressors
- Complexity of models
  - Need to go through hundreds of iterations of input data engineering, parameter tweaking, and experimentation with the algorithms themselves
- Integration with downstream consumer teams, operational cost
- This is a typical problem-space

Functionality

- Cloud Storage
- Cloud Compute
- Development
- Collaboration



Cloud computing

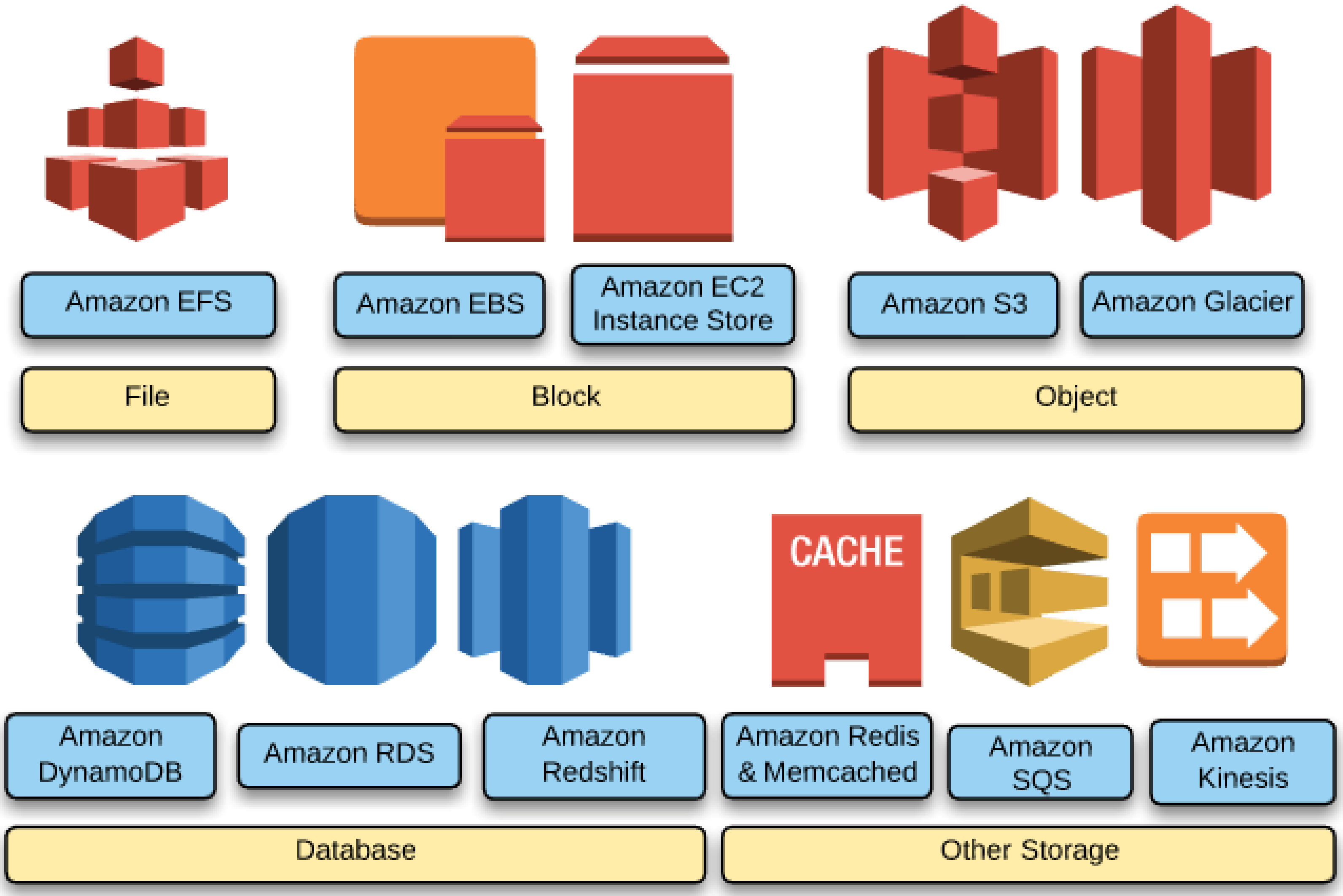
Storage (Data)

Aspects of Cloud Storage

- » Durability
  - Replication
  - Backup and Disaster Recovery
- » Multi-tenancy
- » Access Control
- » Security
  - Can include lifecycle policy, etc.

- » Structured vs unstructured data
- » File-system based vs tabular
- » Relational (SQL) vs NoSQL
- » Serial vs Random Access

Common Offerings



Amazon EFS

File

Amazon EBS

Amazon EC2 Instance Store

Block

Amazon S3

Amazon Glacier

Object

Amazon DynamoDB

Amazon RDS

Amazon Redshift

Database

Amazon Redis & Memcached

Amazon SQS

Amazon Kinesis

Other Storage

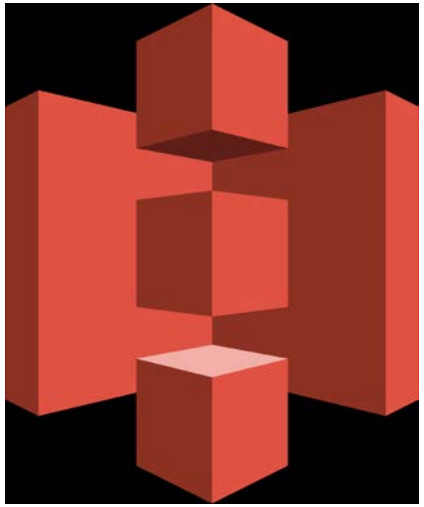
Cloud Storage for Econometric Datasets

» Characteristics
- Generally quantitative, semi-structured data
- Batch access

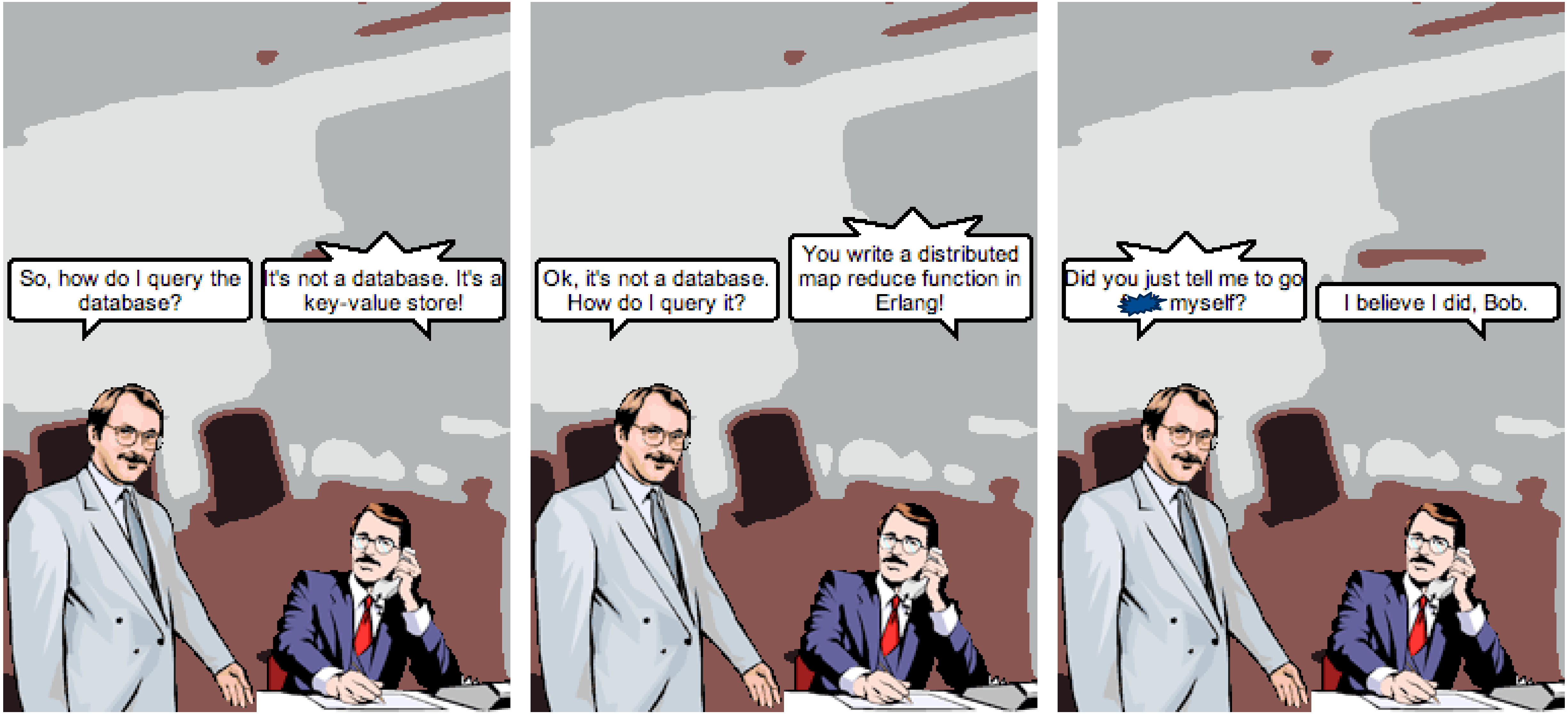
» Object-store with HDFS layer works well

» Storage Format
- Columnar, Binary formats: Parquet or ORC

» Organization: Data Catalog



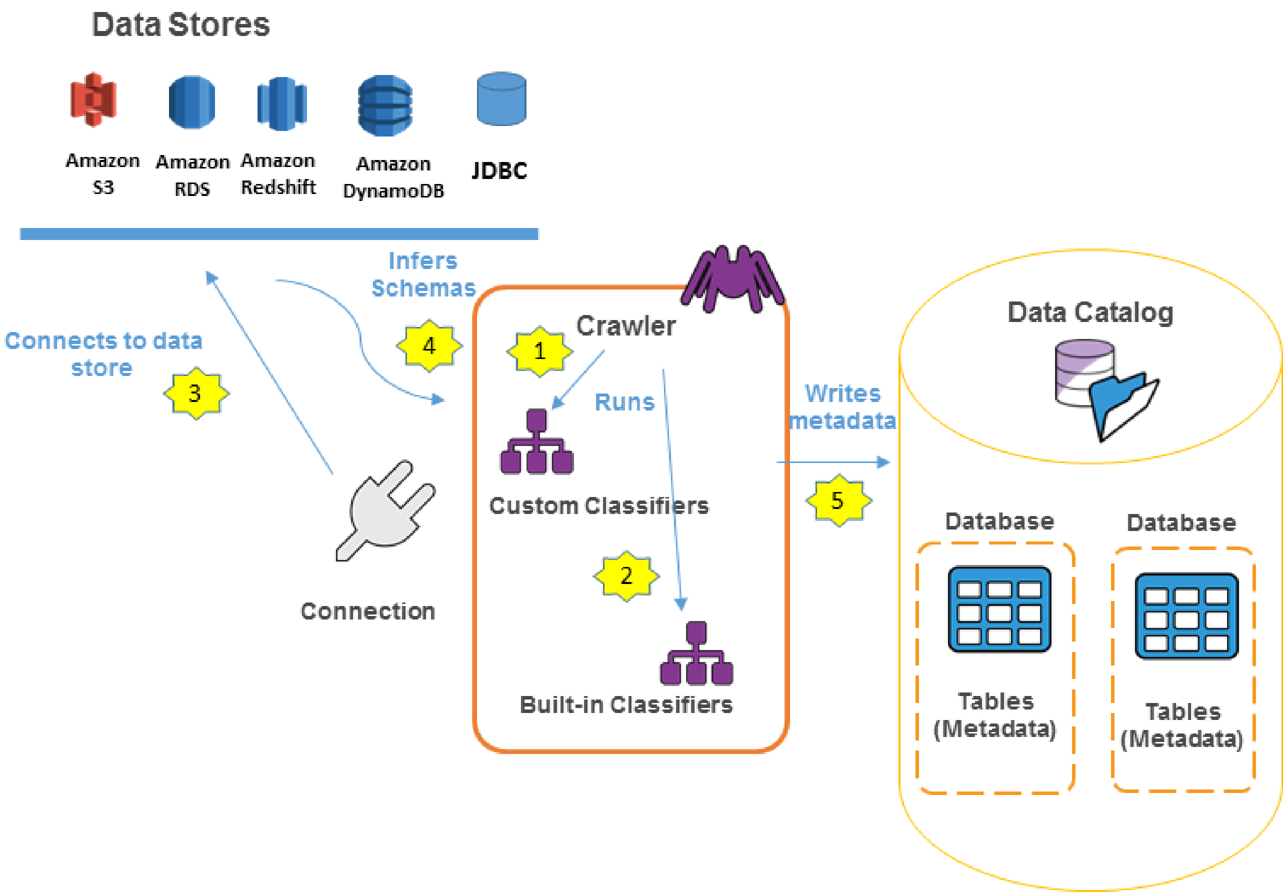
Data Discovery


So, how do I query the database?
It's not a database. It's a key-value store!
Ok, it's not a database. How do I query it?
You write a distributed map reduce function in Erlang!
Did you just tell me to go [redacted] myself?
I believe I did, Bob.

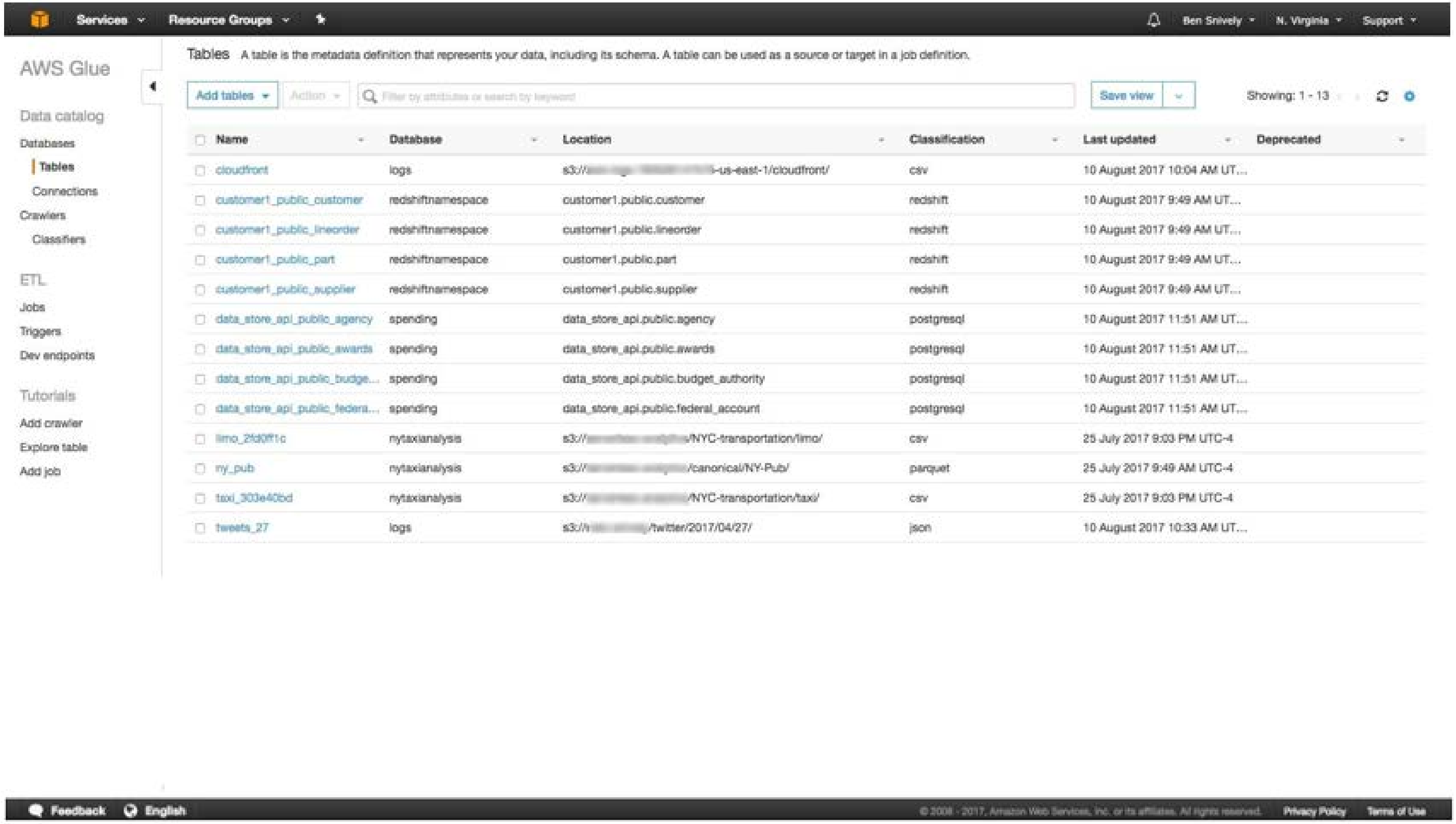
Cloud Data Catalog

» Discover, annotate, and share metadata about your datasets

» Automatic crawling and classification



Cloud Data Catalog



Tables A table is the metadata definition that represents your data, including its schema. A table can be used as a source or target in a job definition.

	Name	Database	Location	Classification	Last updated	Deprecated
□	cloudfront	logs	s3://[redacted]-us-east-1/cloudfront/	csv	10 August 2017 10:04 AM UT...	
□	customer1_public_customer	redshiftnamespace	customer1.public.customer	redshift	10 August 2017 9:49 AM UT...	
□	customer1_public_lineorder	redshiftnamespace	customer1.public.lineorder	redshift	10 August 2017 9:49 AM UT...	
□	customer1_public_part	redshiftnamespace	customer1.public.part	redshift	10 August 2017 9:49 AM UT...	
□	customer1_public_supplier	redshiftnamespace	customer1.public.supplier	redshift	10 August 2017 9:49 AM UT...	
□	data_store_api_public_agency	spending	data_store_api.public.agency	postgresql	10 August 2017 11:51 AM UT...	
□	data_store_api_public_awards	spending	data_store_api.public.awards	postgresql	10 August 2017 11:51 AM UT...	
□	data_store_api_public_budge...	spending	data_store_api.public.budget_authority	postgresql	10 August 2017 11:51 AM UT...	
□	data_store_api_public_federa...	spending	data_store_api.public.federal_account	postgresql	10 August 2017 11:51 AM UT...	
□	limo_2fd0ff1c	nytaxianalysis	s3://[redacted]/NYC-transportation/limo/	csv	25 July 2017 9:03 PM UTC-4	
□	ny_pub	nytaxianalysis	s3://[redacted]/canonical/NY-Pub/	parquet	25 July 2017 9:49 AM UTC-4	
□	taxi_303e40bd	nytaxianalysis	s3://[redacted]/NYC-transportation/taxi/	csv	25 July 2017 9:03 PM UTC-4	
□	tweets_27	logs	s3://[redacted]/twitter/2017/04/27/	json	10 August 2017 10:33 AM UT...	

Cloud Compute

Cloud Compute

- » Typically separate from storage
- » Cluster compute: massive acceleration of most analytic jobs
- » Parallelism has a different set of bottlenecks
  - Network
  - Disk
  - Memory
  - Threading

Cloud Computing frameworks

- MapReduce (original)
- Apache Flink, Beam
- Spark
  - High level of community input
  - Supports SQL, streaming, and analytics
  - Interactive shells for code in Python, R, Scala
  - Supported by most cloud providers
- Other libraries with cluster support
  - e.g. TF-Cluster, R.parallel, etc.

Breadth of Models with Spark/MLLib

» Classification

- Logistic regression
  - Binomial logistic regression
  - Multinomial logistic regression
- Decision tree classifier
- Random forest classifier
- Gradient-boosted tree classifier
- Multilayer perceptron classifier
- One-vs-Rest classifier (a.k.a. One-vs-All)
- Naive Bayes

» Regression

- Linear regression
- Generalized linear regression
- Decision tree regression
- Random forest regression
- Gradient-boosted tree regression
- Survival regression
- Isotonic regression

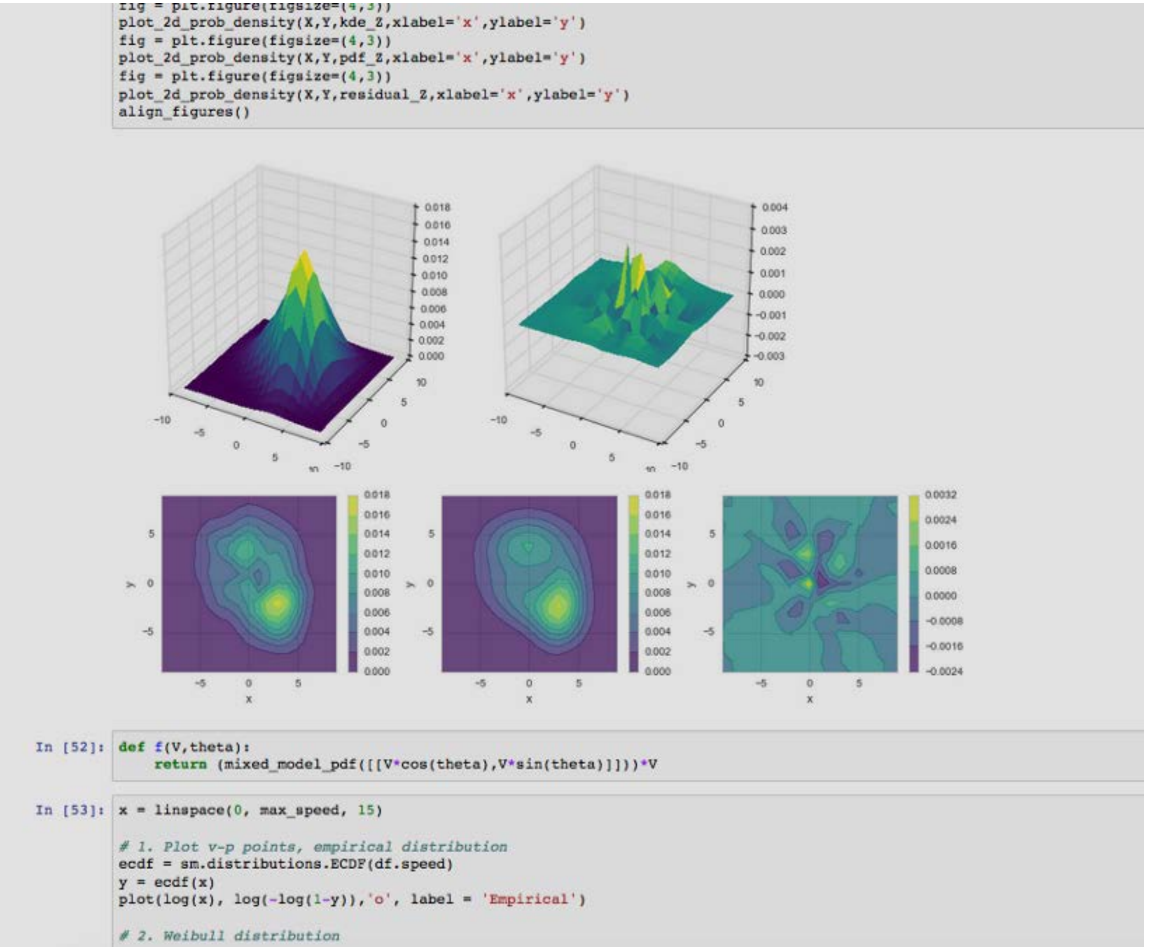
Development

Cloud Notebooks

- Fully managed online compute instances
  - Kernel runs on cloud servers
  - Minimal setup effort
- Portable, highly available
- Blend together live code, equations, visualizations, and explanatory text
- New generation
  - Full-featured IDE support
  - Launch into automated pipelines

Cloud Notebooks

- » Effortless to spin up more powerful instances or clusters
- » Sharing the compute environment along with code
- » Supported languages: Python, R, Scala…

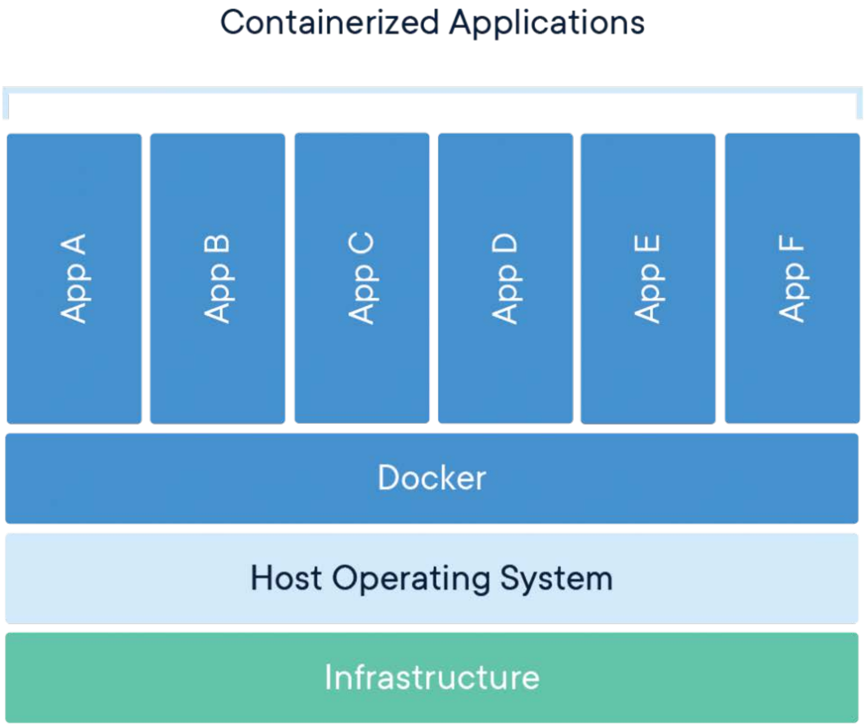


```
fig = plt.figure(figsize=(4,3))
plot_2d_prob_density(X,Y,kde_Z,xlabel='x',ylabel='y')
fig = plt.figure(figsize=(4,3))
plot_2d_prob_density(X,Y,pdf_Z,xlabel='x',ylabel='y')
fig = plt.figure(figsize=(4,3))
plot_2d_prob_density(X,Y,residual_Z,xlabel='x',ylabel='y')
align_figures()
```

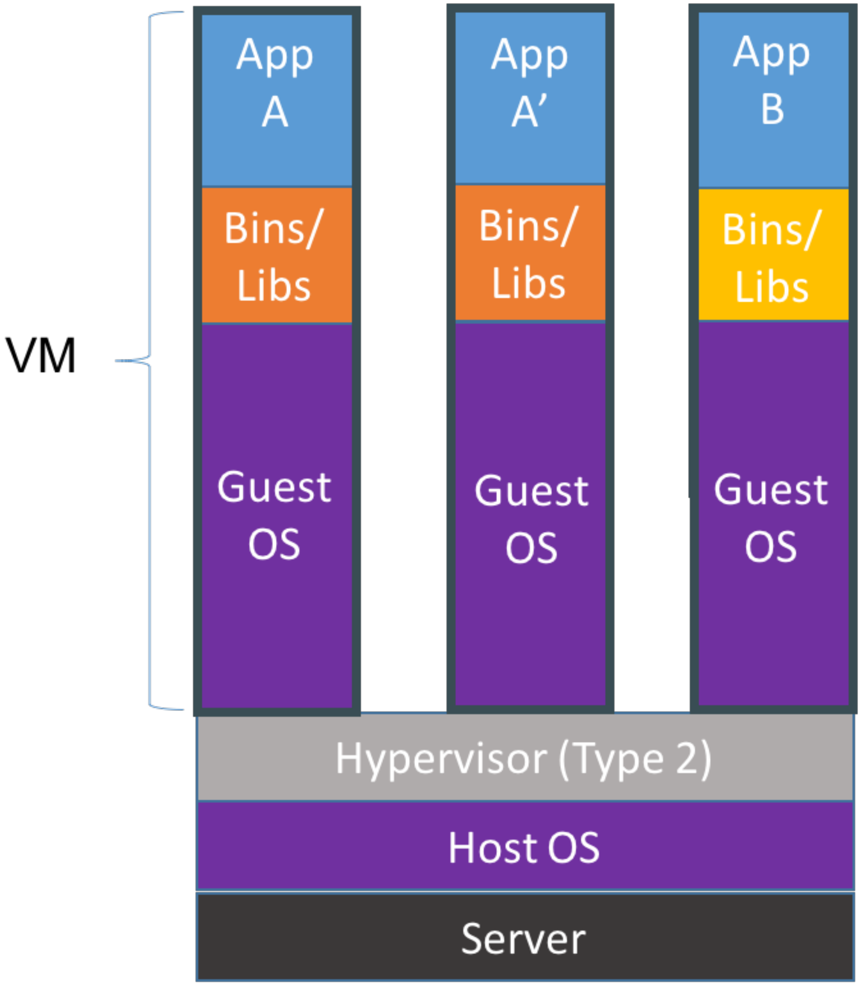
```
In [52]: def f(V,theta):
             return (mixed_model_pdf([[V*cos(theta),V*sin(theta)]]))*V

In [53]: x = linspace(0, max_speed, 15)

         # 1. Plot v-p points, empirical distribution
         ecdf = sm.distributions.ECDF(df.speed)
         y = ecdf(x)
         plot(log(x), log(-log(1-y)),'o', label = 'Empirical')

         # 2. Weibull distribution
```

Bundling other code packages

- Most common for cloud deployments: Docker containers
- Industry standard, lightweight, application isolation

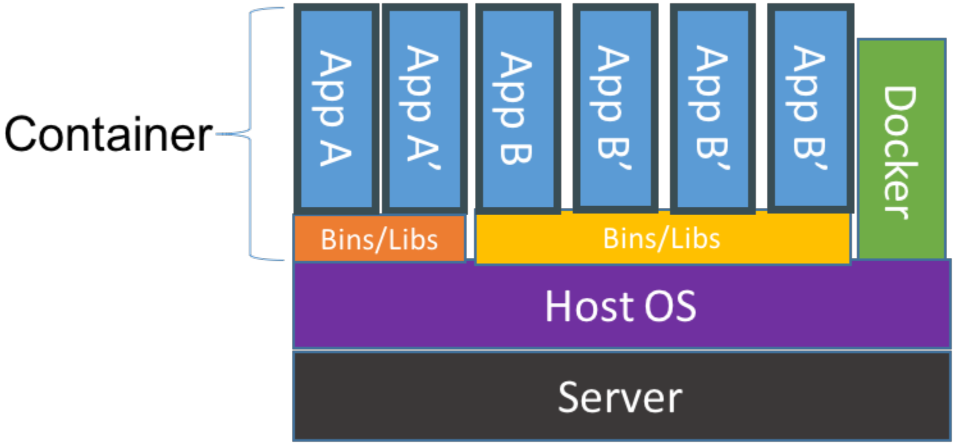


Containers



Containers are isolated, but share OS and, where appropriate, bins/libraries

…result is significantly faster deployment, much less overhead, easier migration, faster restart



Why Dockerize?

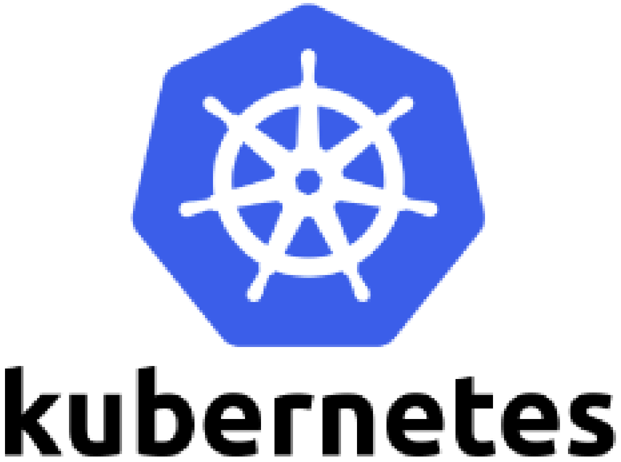
» Infrastructure

- Agility and scaling
- Standardized environments
- Portability
- Resource efficiency

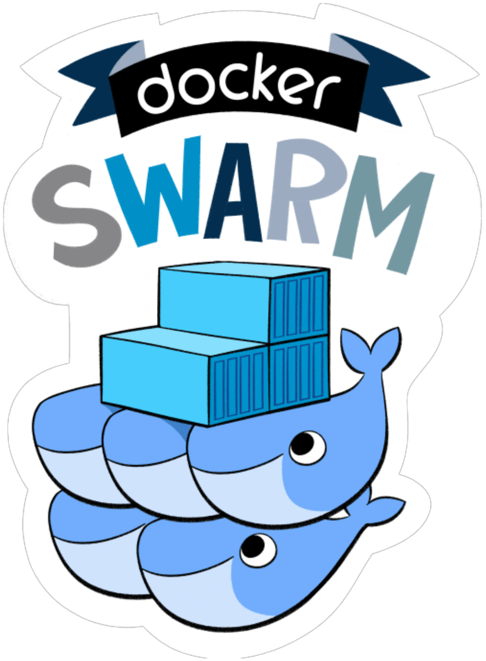
» Application

- “Batteries included”
- Repeatable builds and orchestration
- Faster development cycles
- Lightweight

Many Docker image managers



AWS Batch



AWS ECS

Collaboration

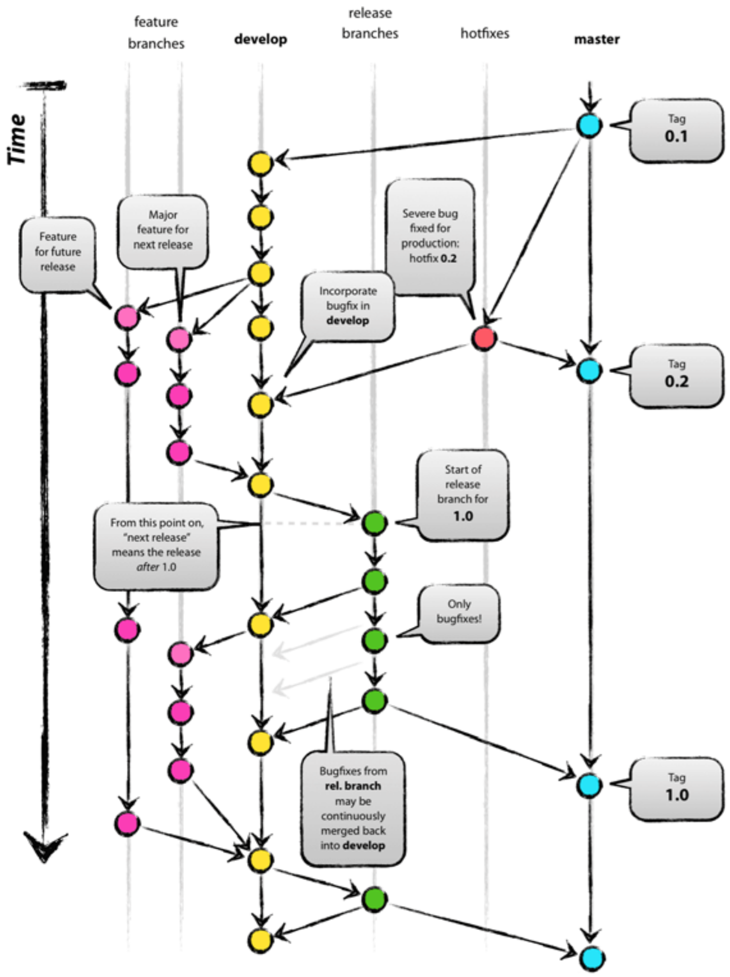
Source Control System

» Benefits

- Long-term version history
  - Auditing, reversion, compare multiple versions
- Code Review
- Merging changes across contributors

» Cloud hosted git repo

- e.g. GitHub, CodeCommit, Cloud Source Repositories
- Scalable, secure, automatic backup
- Easy to share across a distributed team
- Code search
- Cloud notebook integration

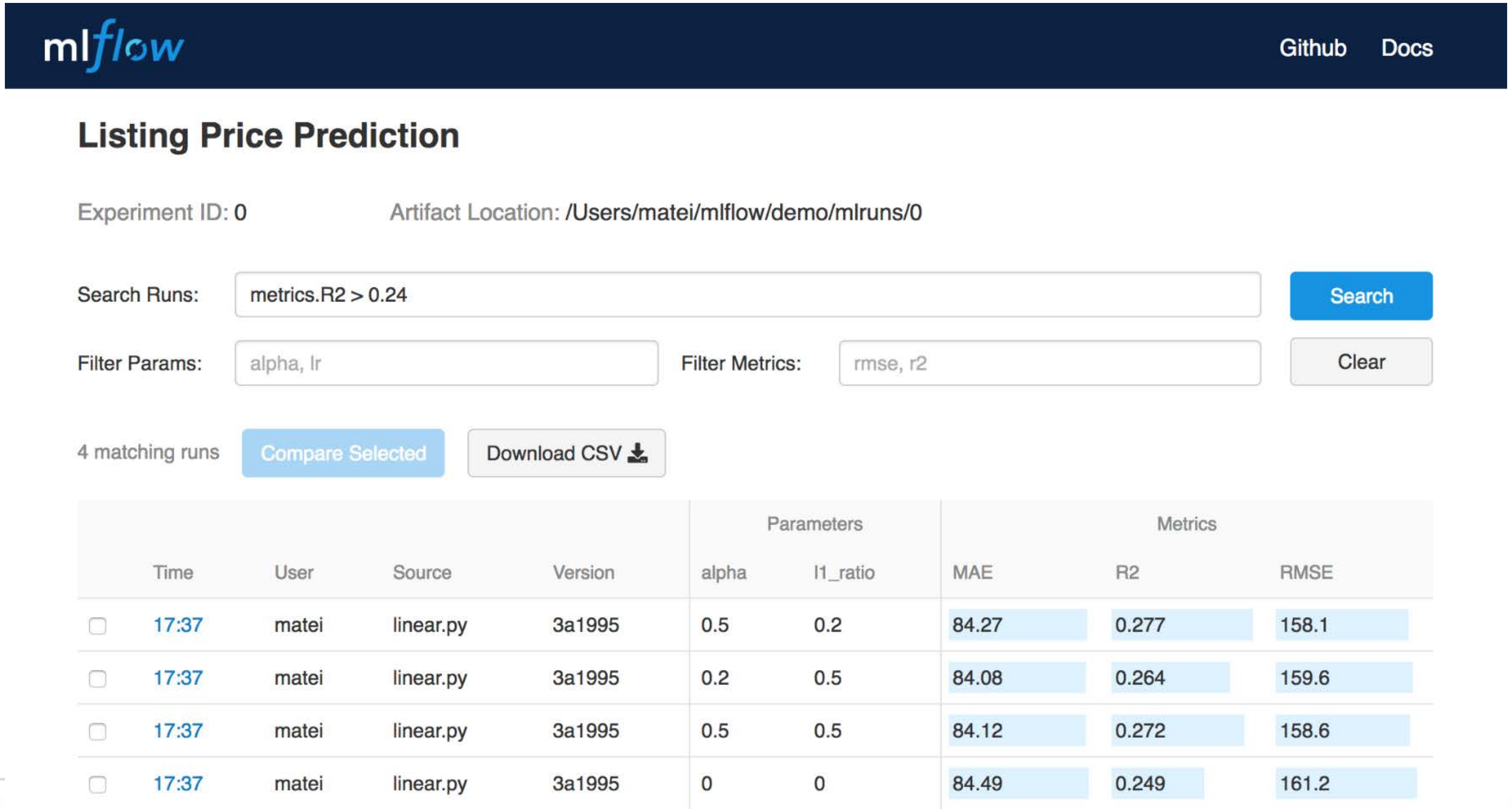

feature branches
develop
release branches
hotfixes
master
Time
Tag 0.1
Feature for future release
Major feature for next release
Severe bug fixed for production: hotfix 0.2
Incorporate bugfix in develop
Tag 0.2
Start of release branch for 1.0
From this point on, "next release" means the release after 1.0
Only bugfixes!
Bugfixes from rel. branch may be continuously merged back into develop
Tag 1.0

Model & experiment tracking

- Models often have dozens of configurable parameters
- Code changes over time
- Need to explicitly track parameters, code, and input data that went into a given run
- Hard to reproduce
- Exponentially more challenging in a larger organization or team

Hosted Model Registry

- » e.g. MLflow Tracking
- » Similar idea to VCS, but parameters and input data snapshots are first-class entities



Listing Price Prediction

Experiment ID: 0 Artifact Location: /Users/matei/mlflow/demo/mlruns/0

Search Runs: metrics.R2 > 0.24

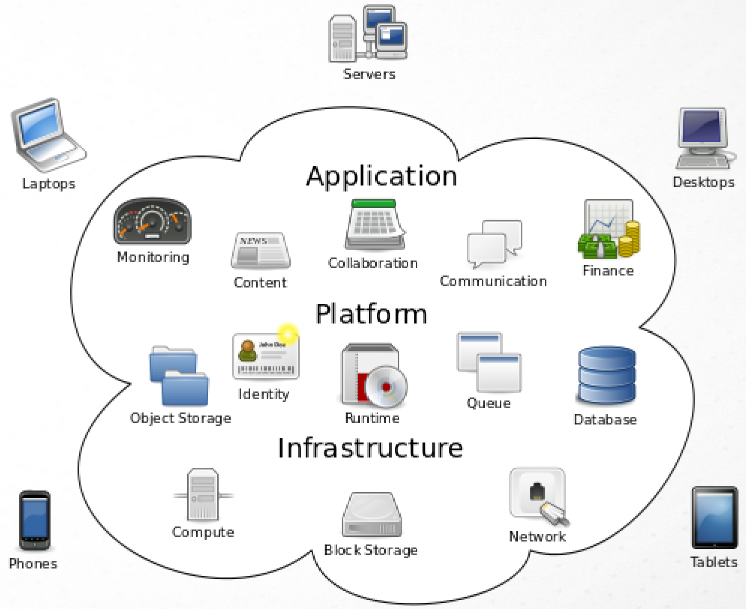
Filter Params: alpha, lr Filter Metrics: rmse, r2

4 matching runs

					Parameters		Metrics		
	Time	User	Source	Version	alpha	l1_ratio	MAE	R2	RMSE
☐	17:37	matei	linear.py	3a1995	0.5	0.2	84.27	0.277	158.1
☐	17:37	matei	linear.py	3a1995	0.2	0.5	84.08	0.264	159.6
☐	17:37	matei	linear.py	3a1995	0.5	0.5	84.12	0.272	158.6
☐	17:37	matei	linear.py	3a1995	0	0	84.49	0.249	161.2

Recap

- Cloud Storage
- Cloud Compute
- Development
- Collaboration



Cloud computing

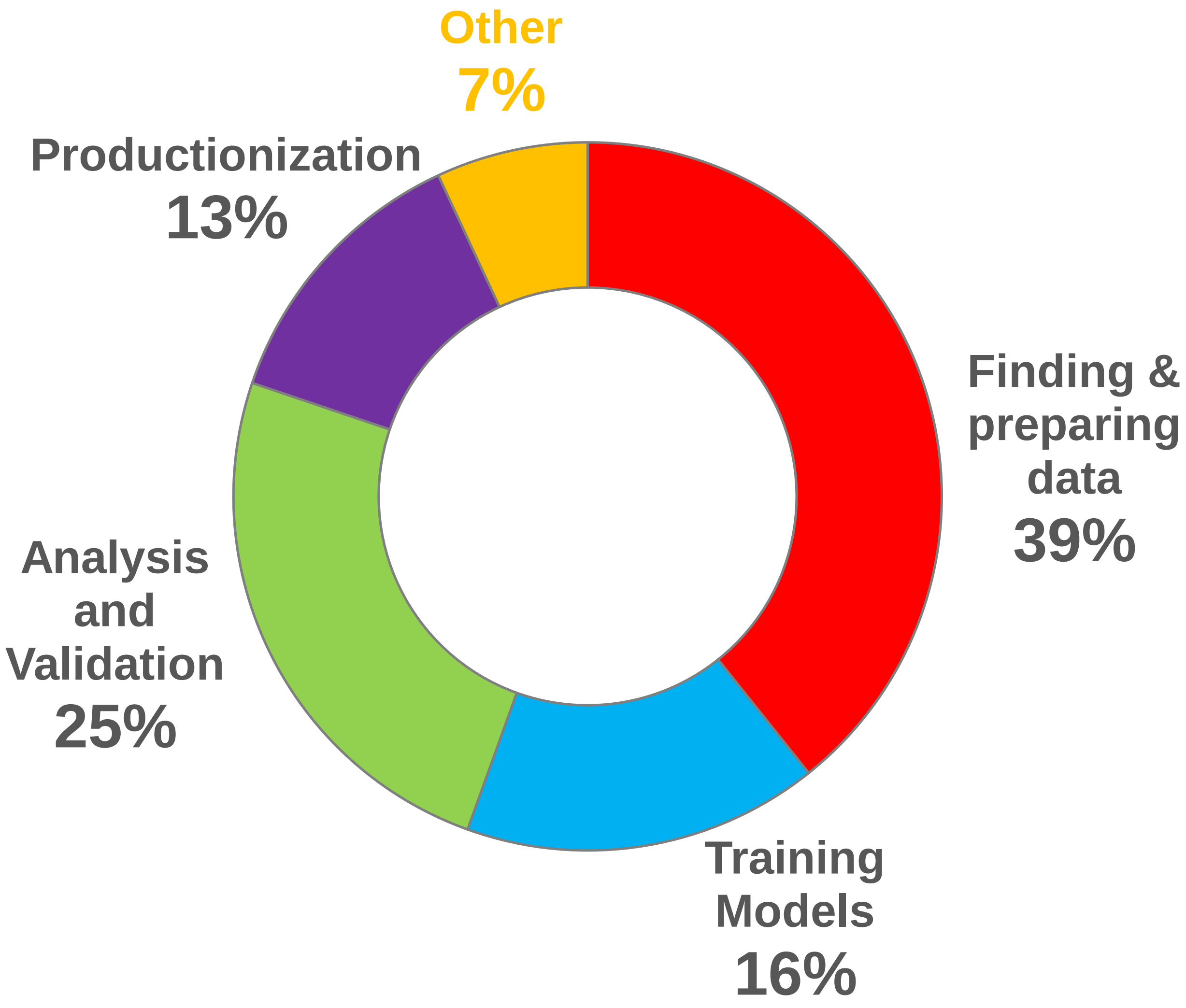
What’s next?

On the horizon

- » Beyond Spark
  - Ray
  - Avoids block-synchronous compute paradigm
  - Supported libraries including Modin, RLLib, TUNE
- » ML-Economics integration
  - Cloud support for DL-type workloads
- » Deeper model dependency/lineage

https://arxiv.org/abs/1712.05889

Environment Improvements



How Economists Spend their time? – Economist Bi-annual Survey 2019

Questions?

coreai-org@amazon.com

amazon