

Distributed Machine Learning with H2O

Joint Statistical Meeting 2018
Vancouver, British Columbia, Canada



Navdeep Gill
@Navdeep_Gill_

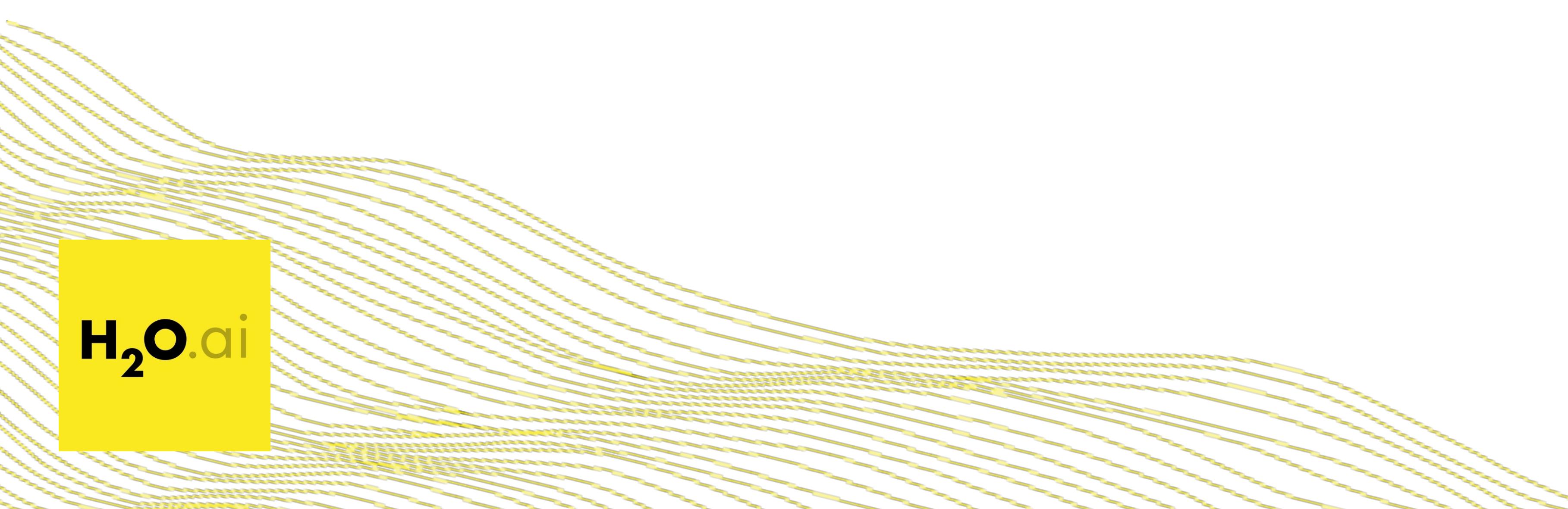
About Me

- Data Scientist & Software Engineer at H2O.ai in Mountain View, California, USA
- B.S. in Statistics, B.A. in Psychology (minor in Mathematics), & M.S. in Computational Statistics from California State University, East Bay
- Work I have done at H2O.ai:
 - H2O-3 (<https://github.com/h2oai/h2o-3>)
 - #12 contributor (based on Github contributions)
 - Co-author of AutoML/Stacked Ensembles
 - Worked heavily on the R/Python APIs
 - Development of H2O-3 Java backend (data munging tasks, bug fixes, etc.)
 - Day-to-day software stuff...
 - H2O4GPU (<https://github.com/h2oai/h2o4gpu>)
 - #2 contributor (based on Github contributions)
 - Implemented Truncated SVD and PCA in CUDA
 - Heavy development of Python/R API's
 - Day-to-day software stuff...
 - Rsparkling (<https://github.com/h2oai/rsparkling>)
 - #1 contributor ((based on Github contributions)
 - Running H2O's Sparkling Water in R
 - Day-to-day software stuff...
- Currently focused on machine learning interpretability (MLI)
 - Co-developed MLI capabilities in H2O Driverless.ai
 - <https://www.h2o.ai/driverless-ai/>
 - <http://docs.h2o.ai/driverless-ai/latest-stable/docs/booklets/DriverlessAIBooklet.pdf>
 - <http://docs.h2o.ai/driverless-ai/latest-stable/docs/booklets/MLIBooklet.pdf>
 - <https://www.safaribooksonline.com/library/view/an-introduction-to/9781492033158/>
 - <https://github.com/h2oai/mli-resources>
 - Day-to-day software stuff...

Agenda

- H2O Introduction
- H2O Core Overview
- H2O API
- Demo

H2O Introduction



H₂O.ai

Company Overview

Founded

2011 Venture-backed, Debuted in 2012

- **H₂O Open Source In-Memory AI Prediction Engine**

Products

- Sparkling Water (H₂O + Spark)
- H2O4GPU (H2O on GPUs)
- Enterprise Steam
- Driverless AI

Mission

Operationalize Data Science & Provide a Platform to Build Beautiful Data Products

Team

75+ employees

- Distributed Systems Engineers doing Machine Learning
- World-class Visualization Designers

Headquarters

Mountain View, CA



Scientific Advisory Council



Dr. Trevor Hastie

- PhD in Statistics, Stanford University
- John A. Overdeck Professor of Mathematics, Stanford University
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Co-author, *Generalized Additive Models*
- 108,404 citations (via Google Scholar)



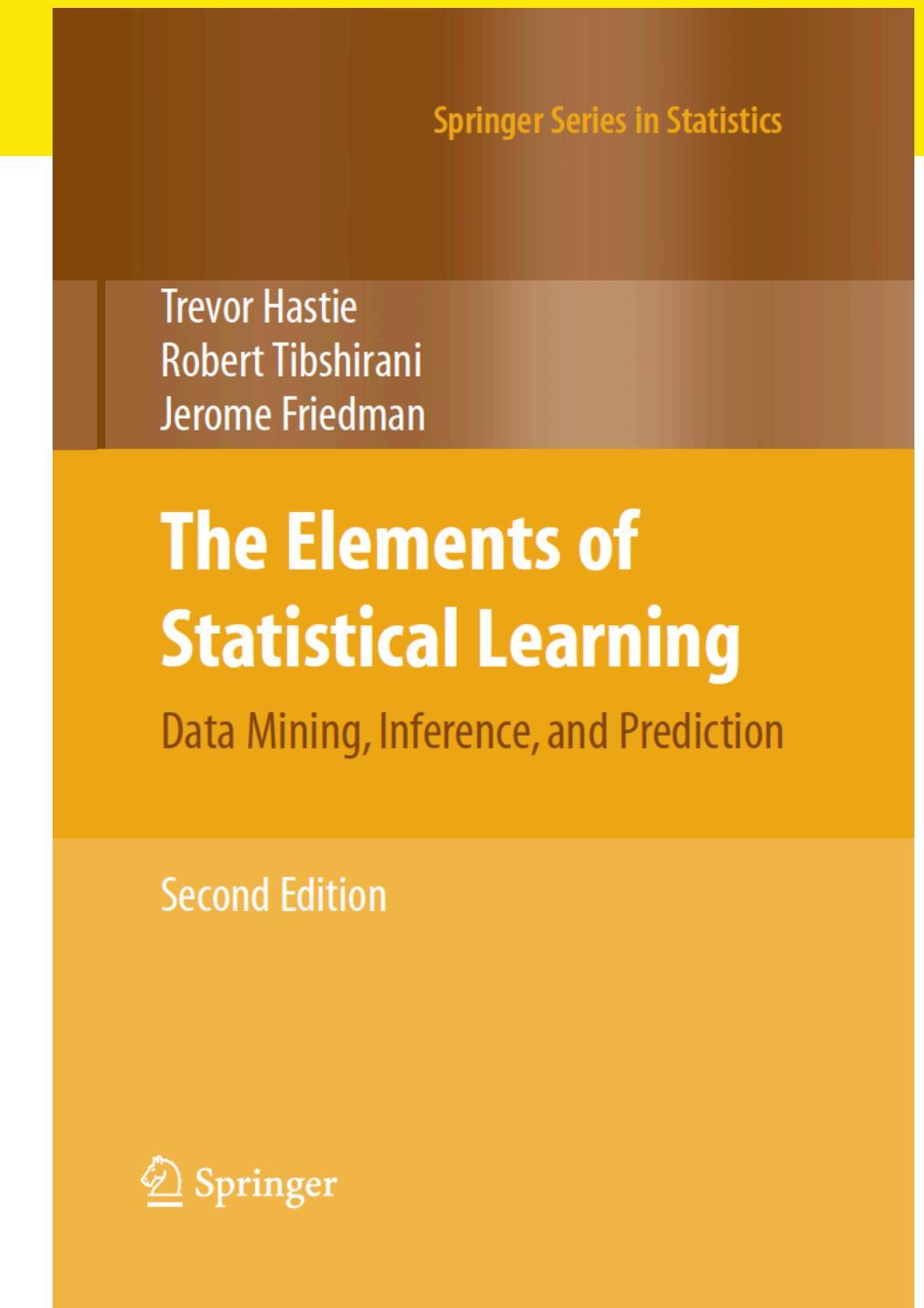
Dr. Robert Tibshirani

- PhD in Statistics, Stanford University
- Professor of Statistics and Health Research and Policy, Stanford University
- COPPS Presidents' Award recipient
- Co-author, *The Elements of Statistical Learning: Prediction, Inference and Data Mining*
- Author, *Regression Shrinkage and Selection via the Lasso*
- Co-author, *An Introduction to the Bootstrap*



Dr. Steven Boyd

- PhD in Electrical Engineering and Computer Science, UC Berkeley
- Professor of Electrical Engineering and Computer Science, Stanford University
- Co-author, *Convex Optimization*
- Co-author, *Linear Matrix Inequalities in System and Control Theory*
- Co-author, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*



What is H2O?

Java-Based Software for In-Memory Data Modeling

Open Source



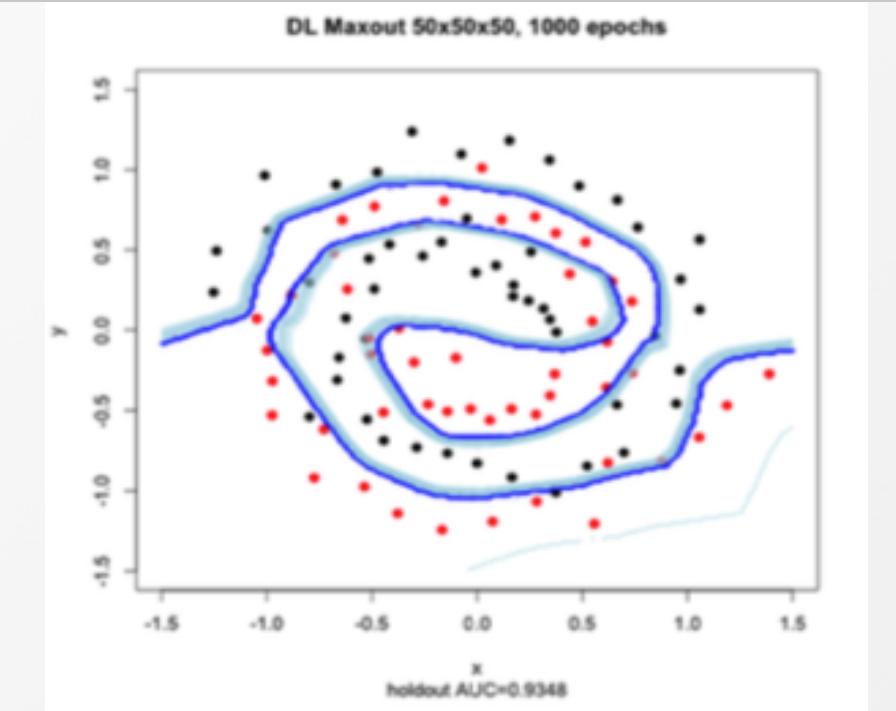
Big Data Ecosystem



Flexible Interface

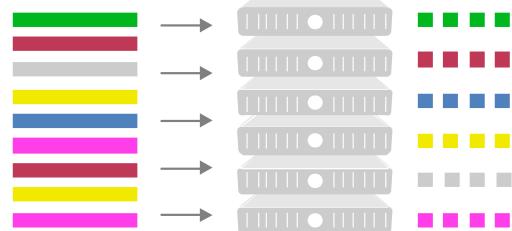
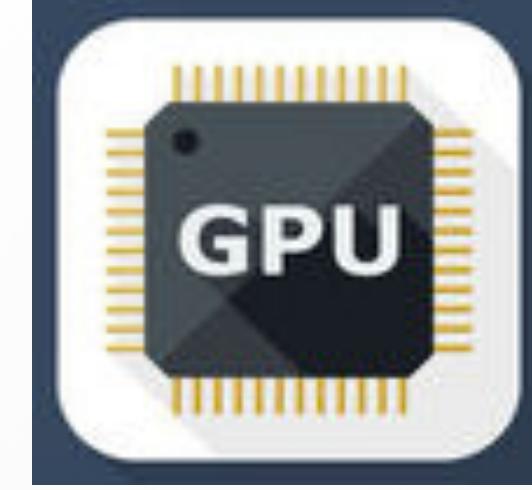


Smart and Fast Algorithms

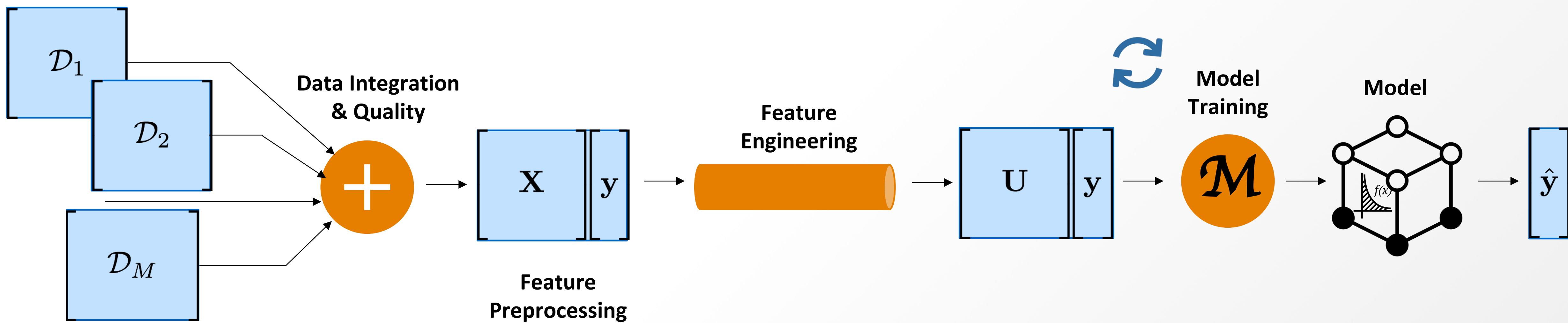


What is H2O?

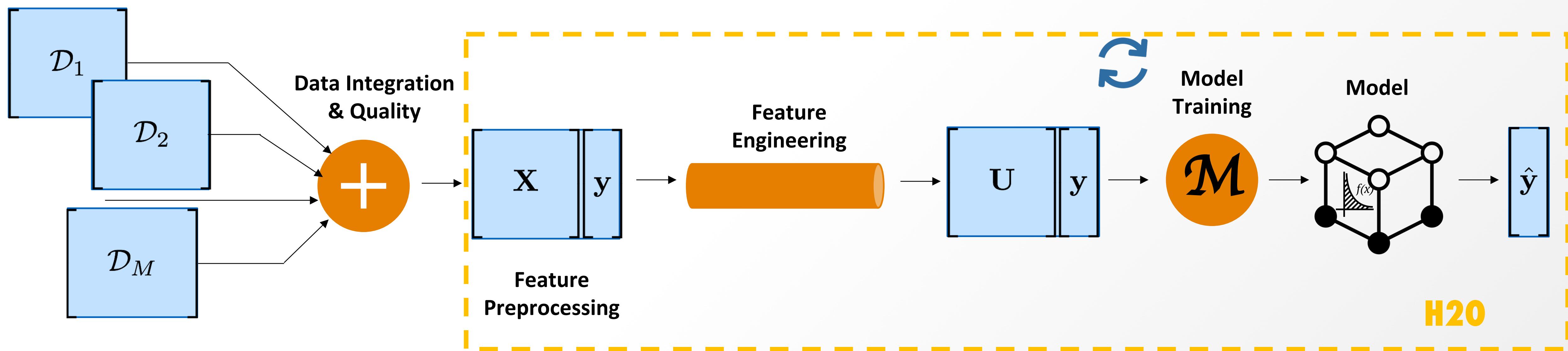
Java-Based Software for In-Memory Data Modeling

Scalability and Performance	Rapid Model Deployment	GPU Enablement*	Cloud Integration
 <ul style="list-style-type: none">Distributed In-Memory Computing PlatformDistributed AlgorithmsFine-Grain MapReduce	<ul style="list-style-type: none">Highly portable models deployed in Java (POJO)Automated and streamlined scoring service deployment with Rest API*		  

The Machine Learning Pipeline



Where H2O Fits



Current Algorithm Overview

Statistical Analysis

- Linear Models (GLM)
- Naïve Bayes

Ensembles

- Random Forest
- Gradient Boosting Machine
- Stacking / Super Learner

Deep Neural Networks

- MLP
- Autoencoder
 - Anomaly Detection
 - Deep Features

H2O AutoML

- Automatic Machine Learning in H2O

Clustering

- K-Means (Auto-K)

Dimension Reduction

- Principal Component Analysis
- Generalized Low Rank Models

Word Embedding

- Word2Vec

Time Series

- iSAX

Machine Learning Tuning

- Hyperparameter Search
- Early Stopping

H2O Core Overview

Behind the Scenes



How H2O Core Works



How H2O Core Works



(just a java application)

How H2O Core Works



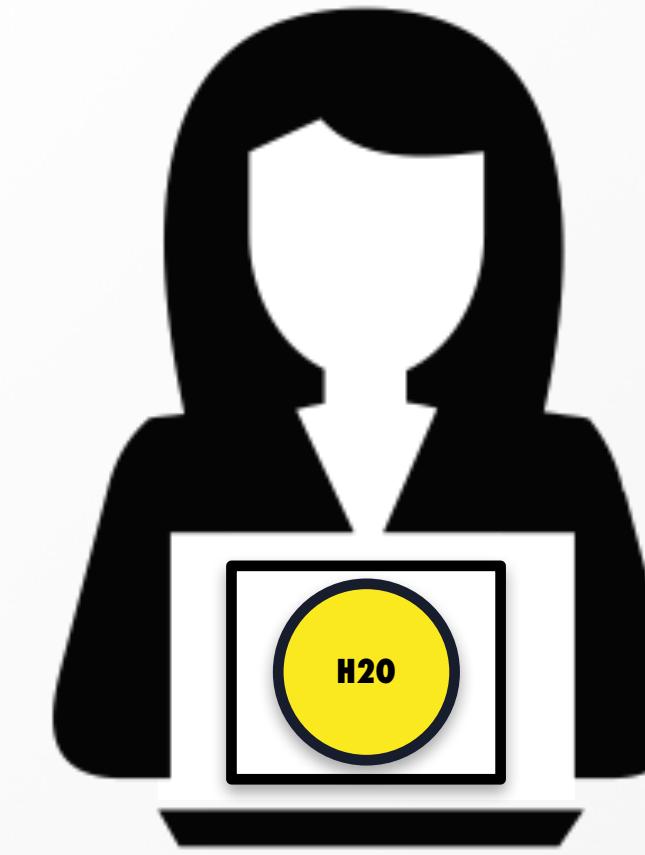
How H2O Core Works



on a laptop

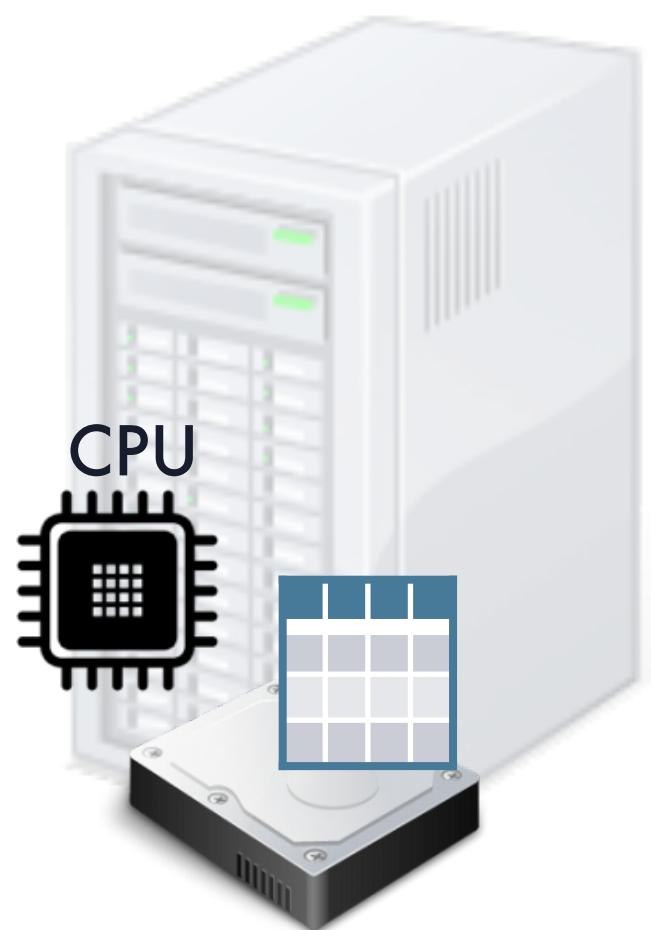


on a virtual machine

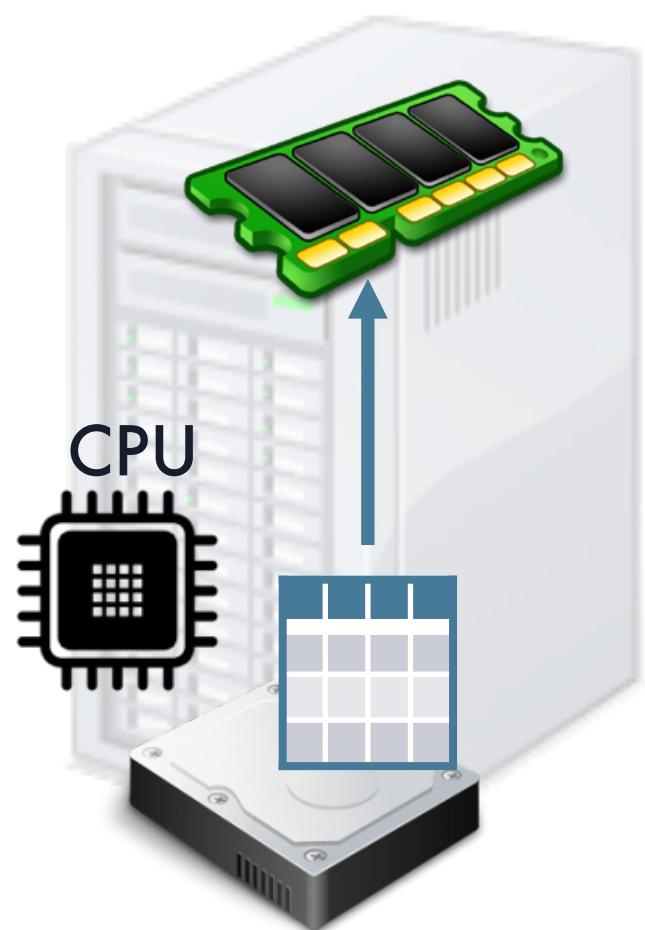


in a container

H2O Core

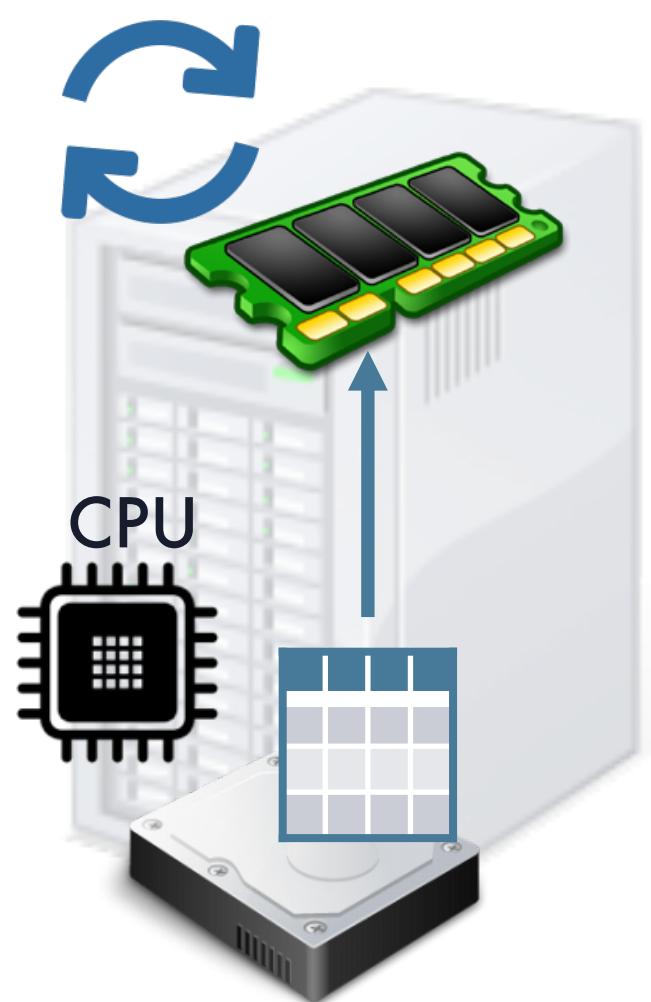


H2O Core



H2O Core

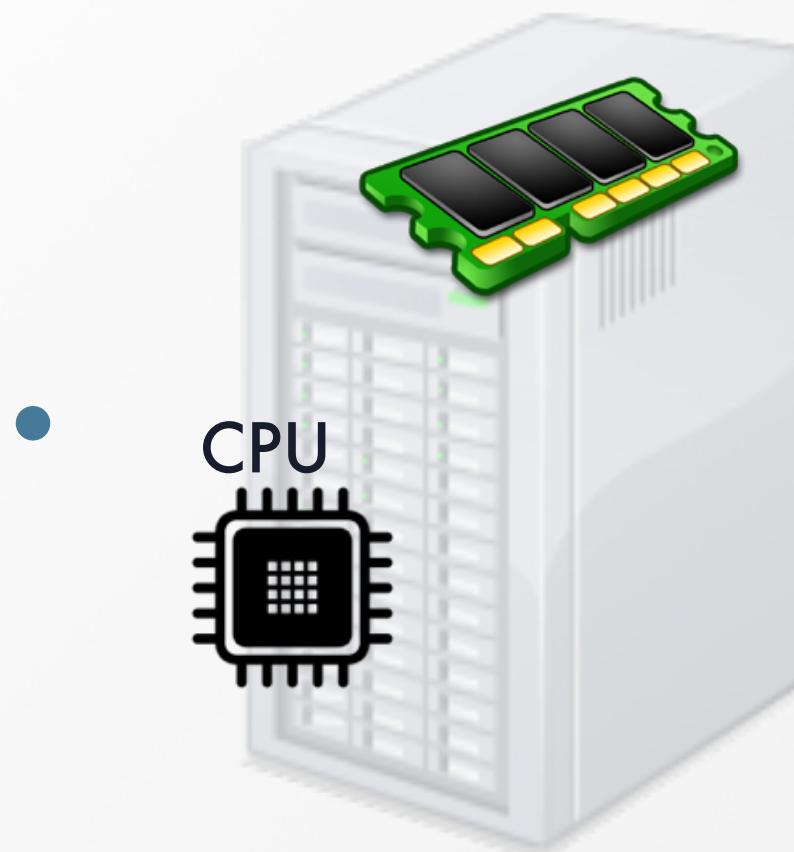
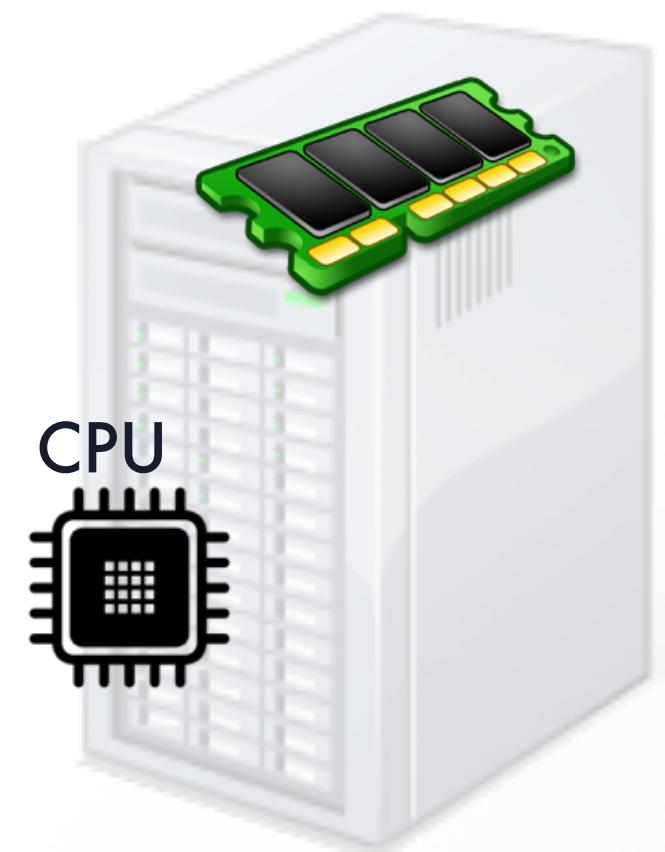
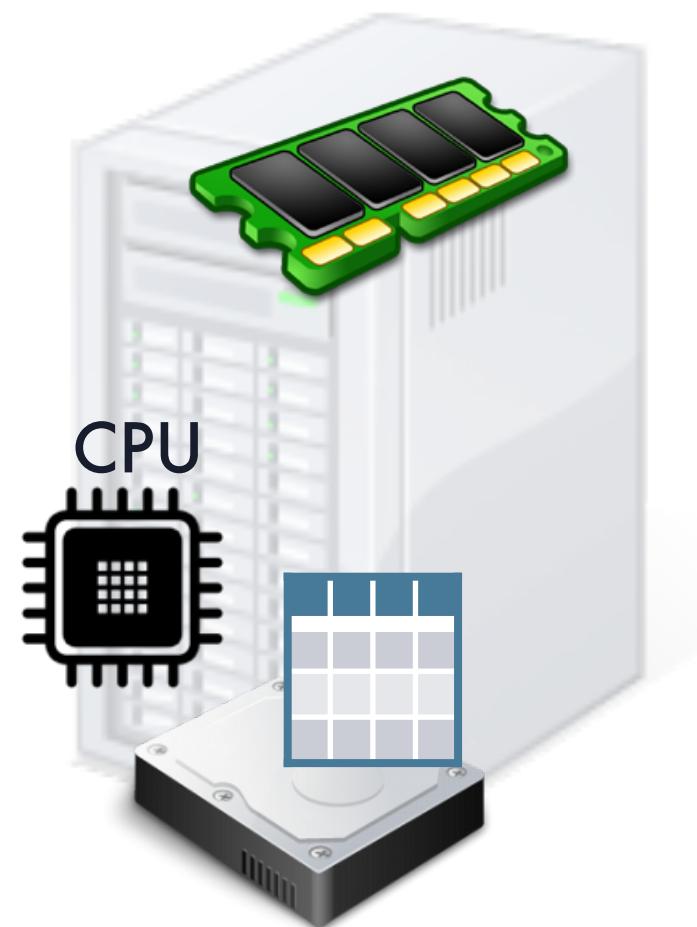
Model Building



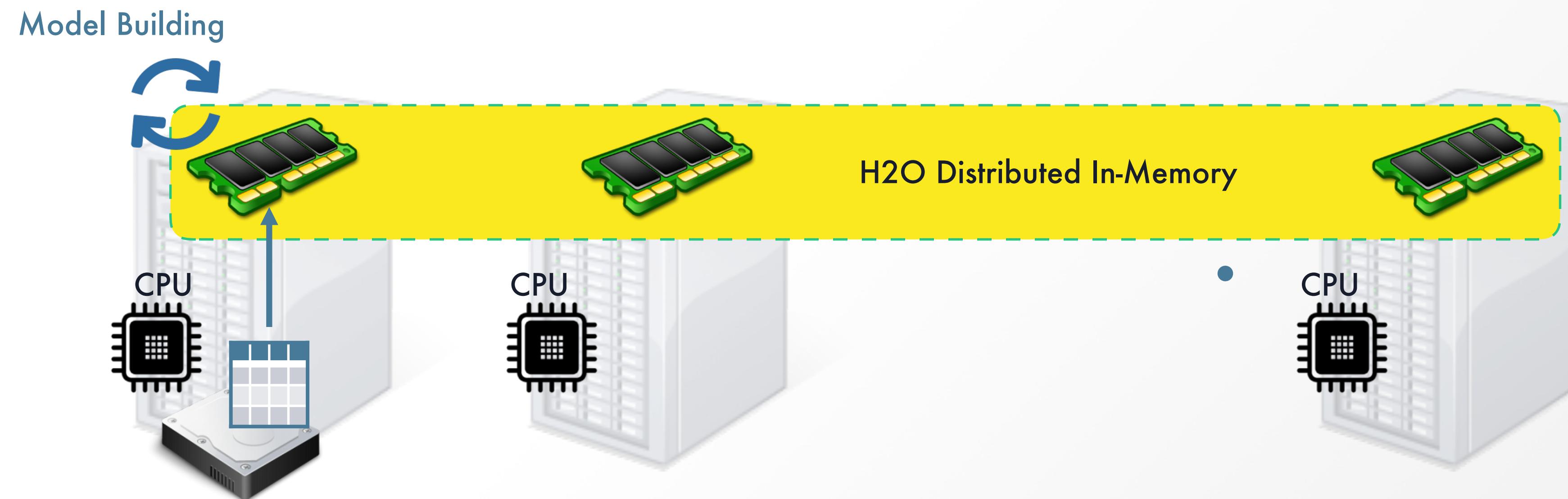
H2O Core



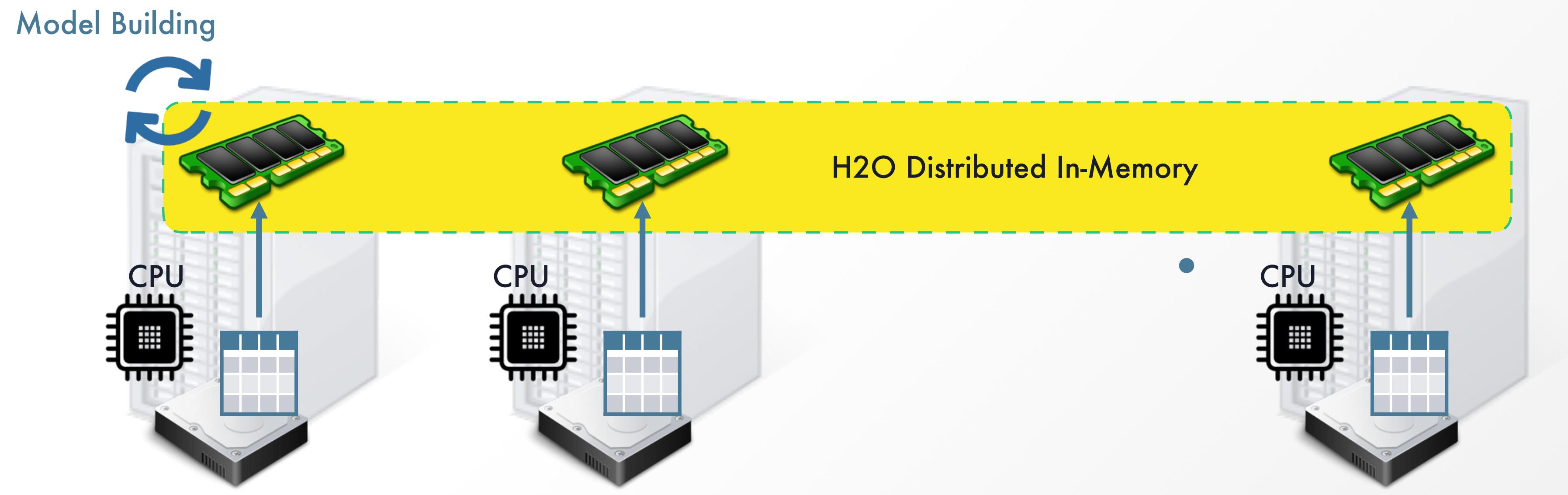
H2O Core



H2O Core



H2O Core with Hadoop

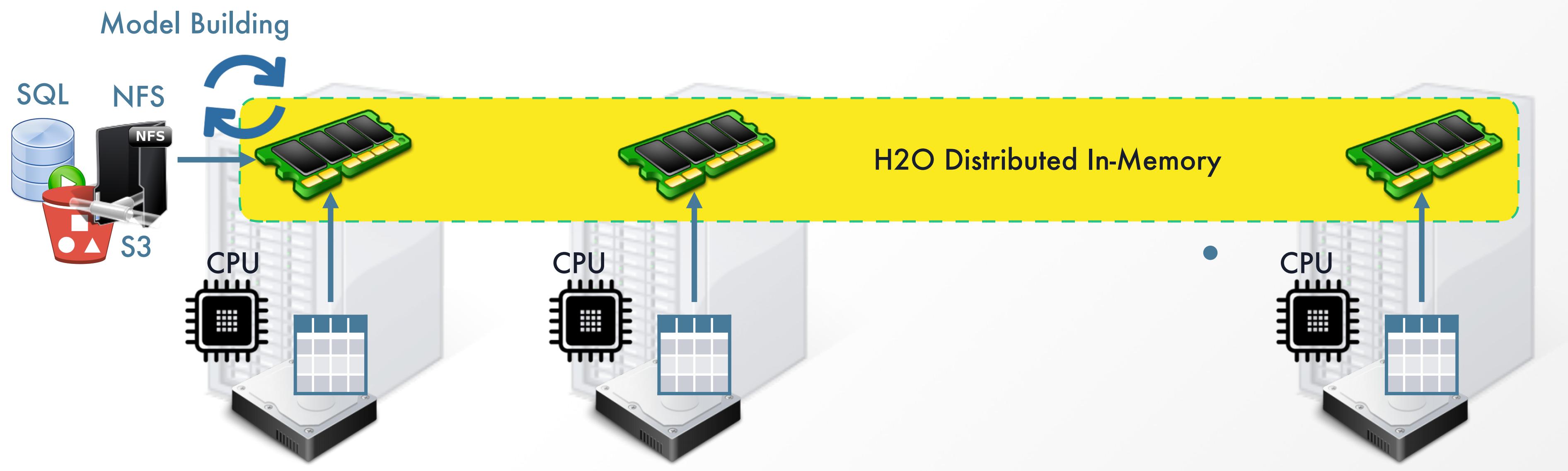


YARN

cloudera Hortonworks

MAPR

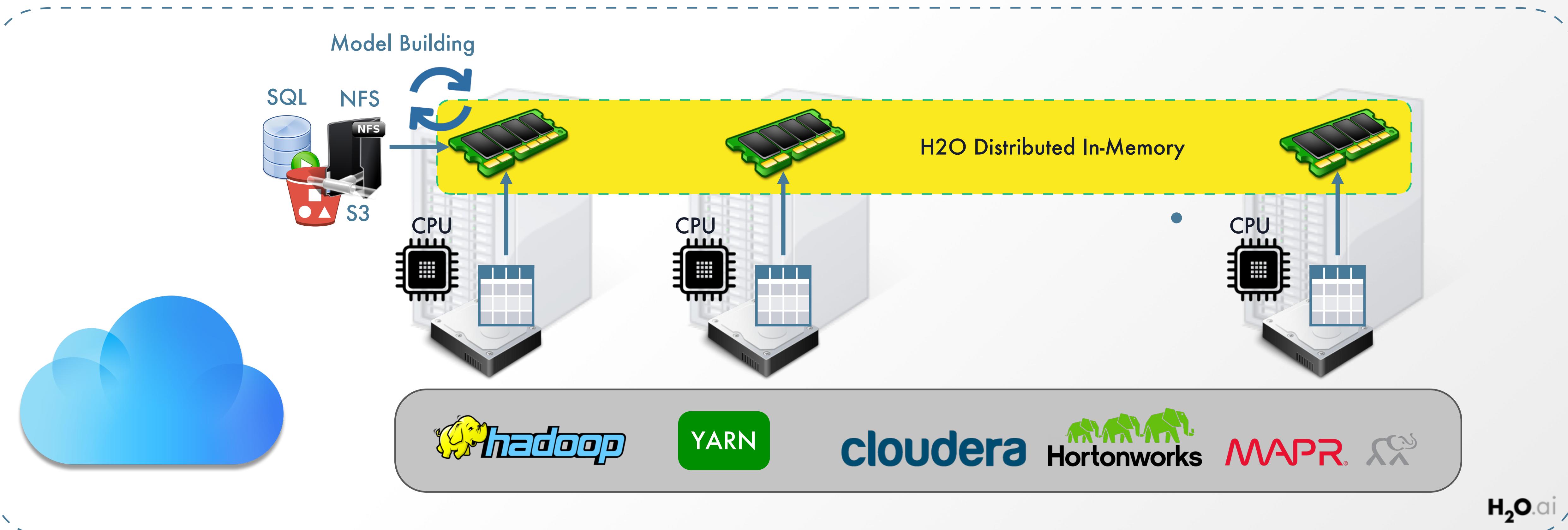
H2O Core with Other Data Sources



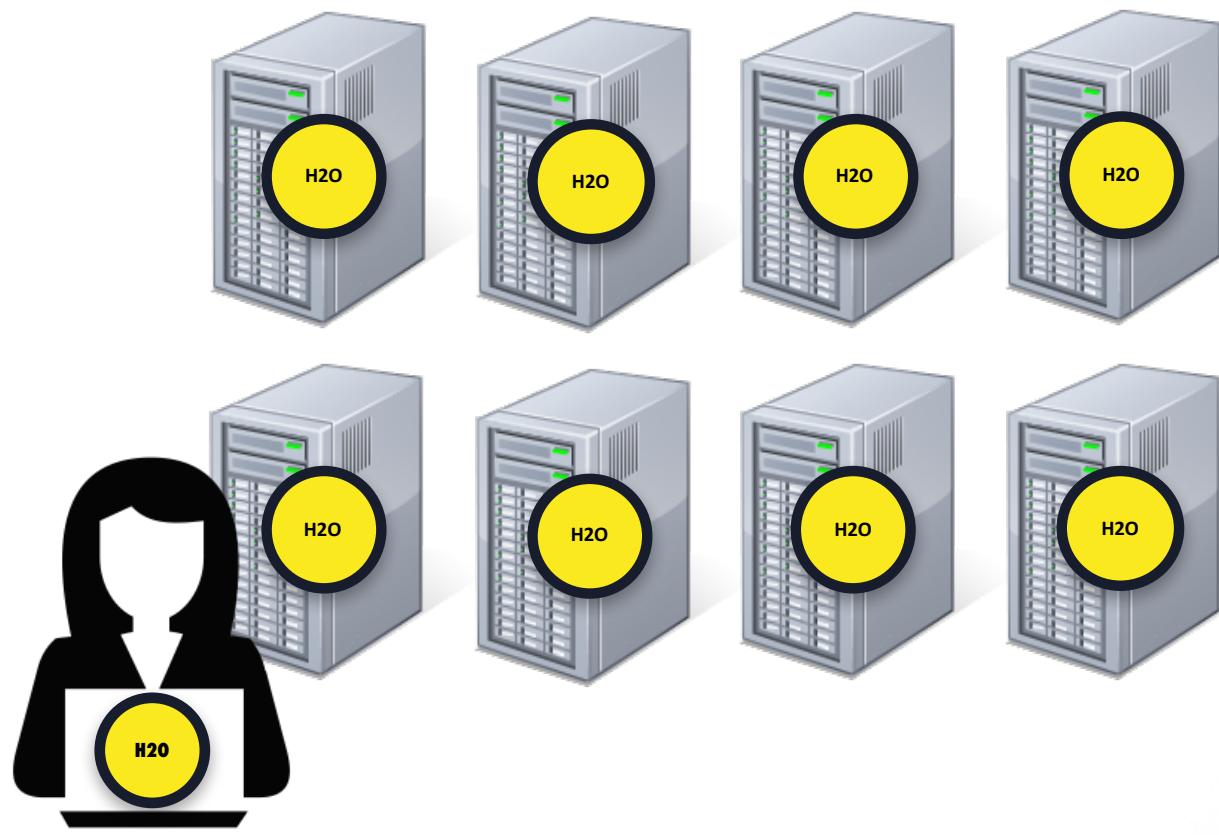
YARN

The Cloudera and Hortonworks logos, both featuring green elephant icons.The MAPR logo, featuring a red elephant icon and the word "MAPR" in red uppercase letters.

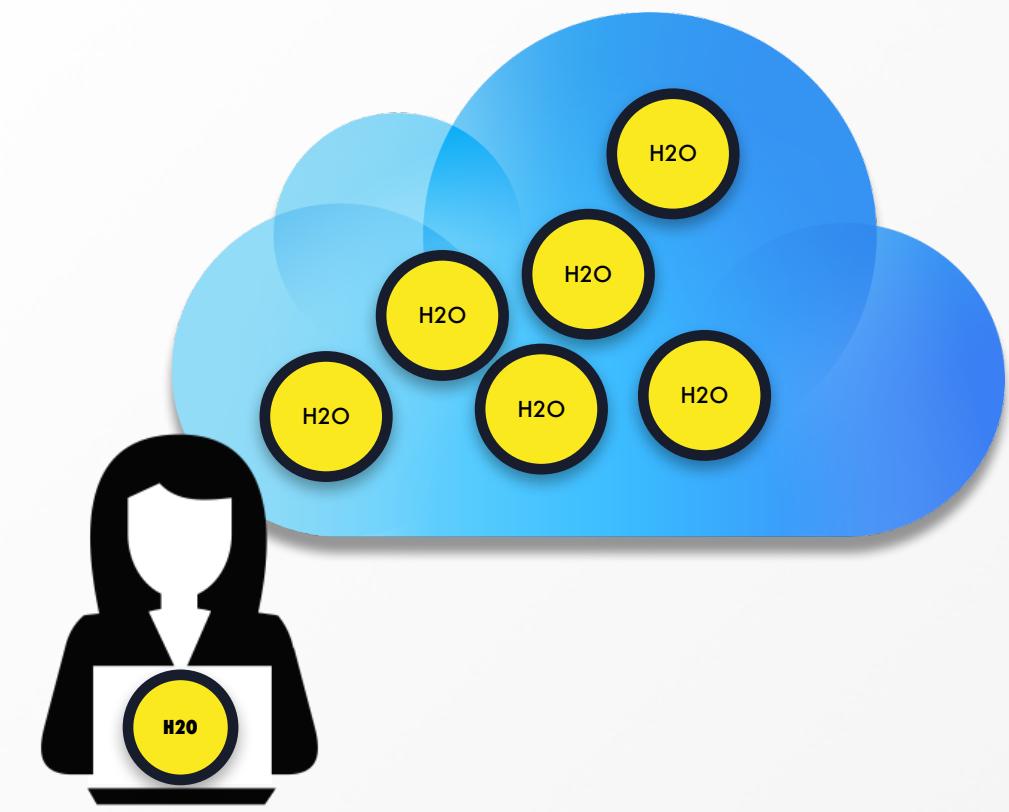
H2O Core on the Cloud



H2O Distributed Environments



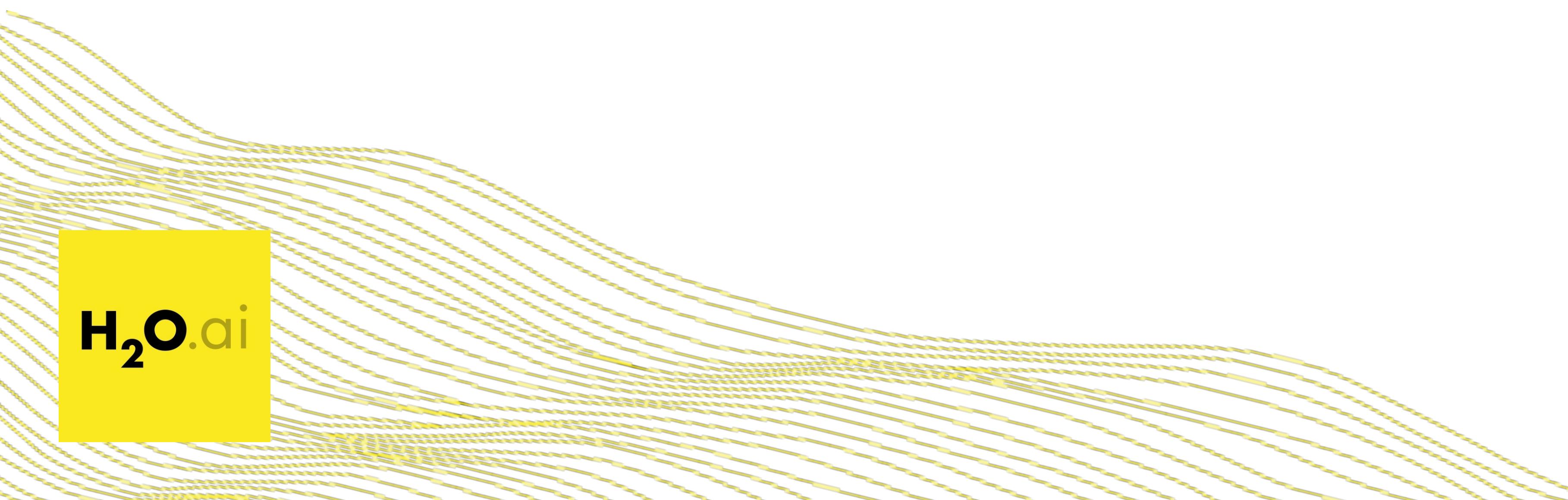
distributed across multiple machines



distributed on the cloud

H2O API

How the client & cluster communicate



H2O API



(ALL GREAT THINGS)

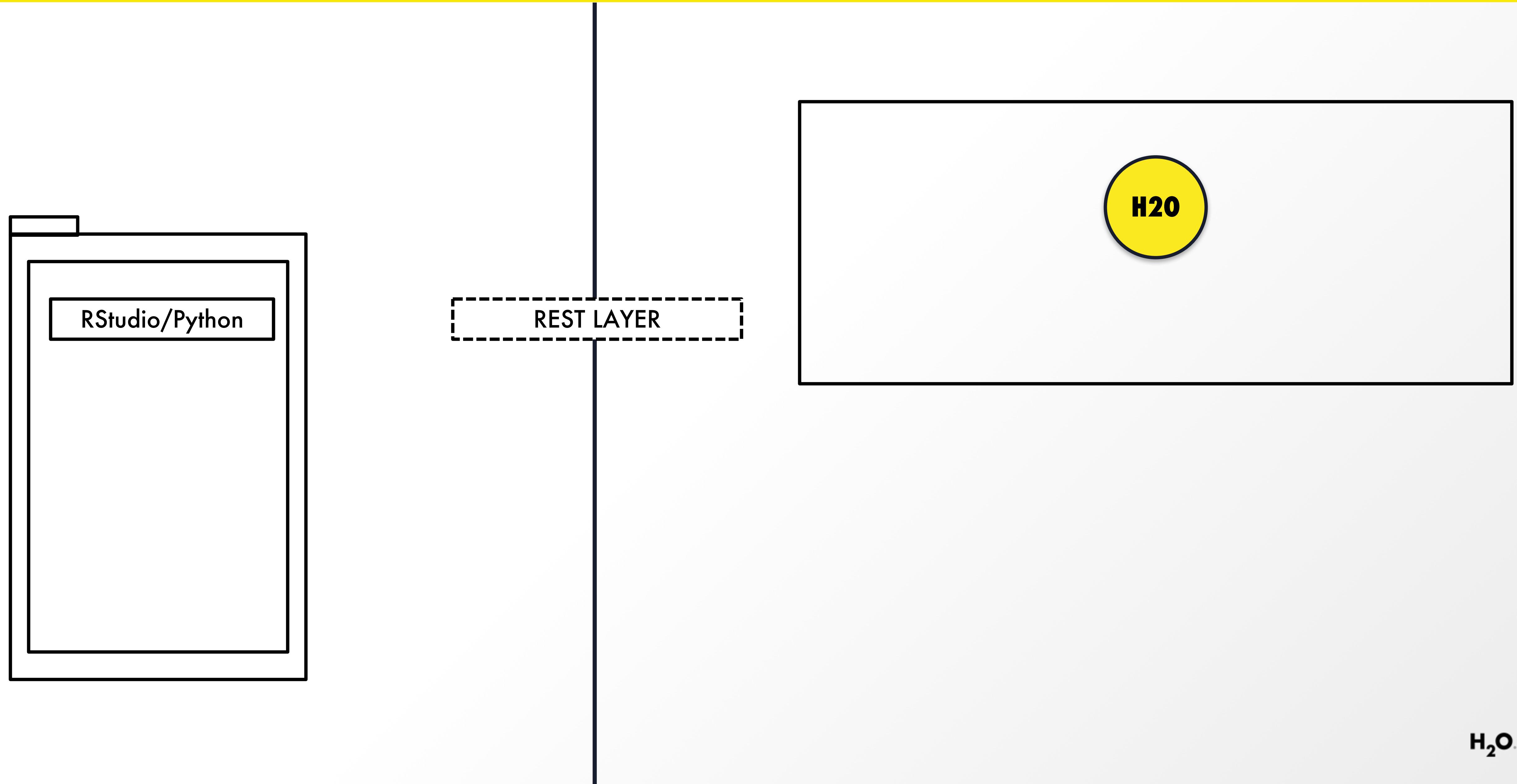
H2O API



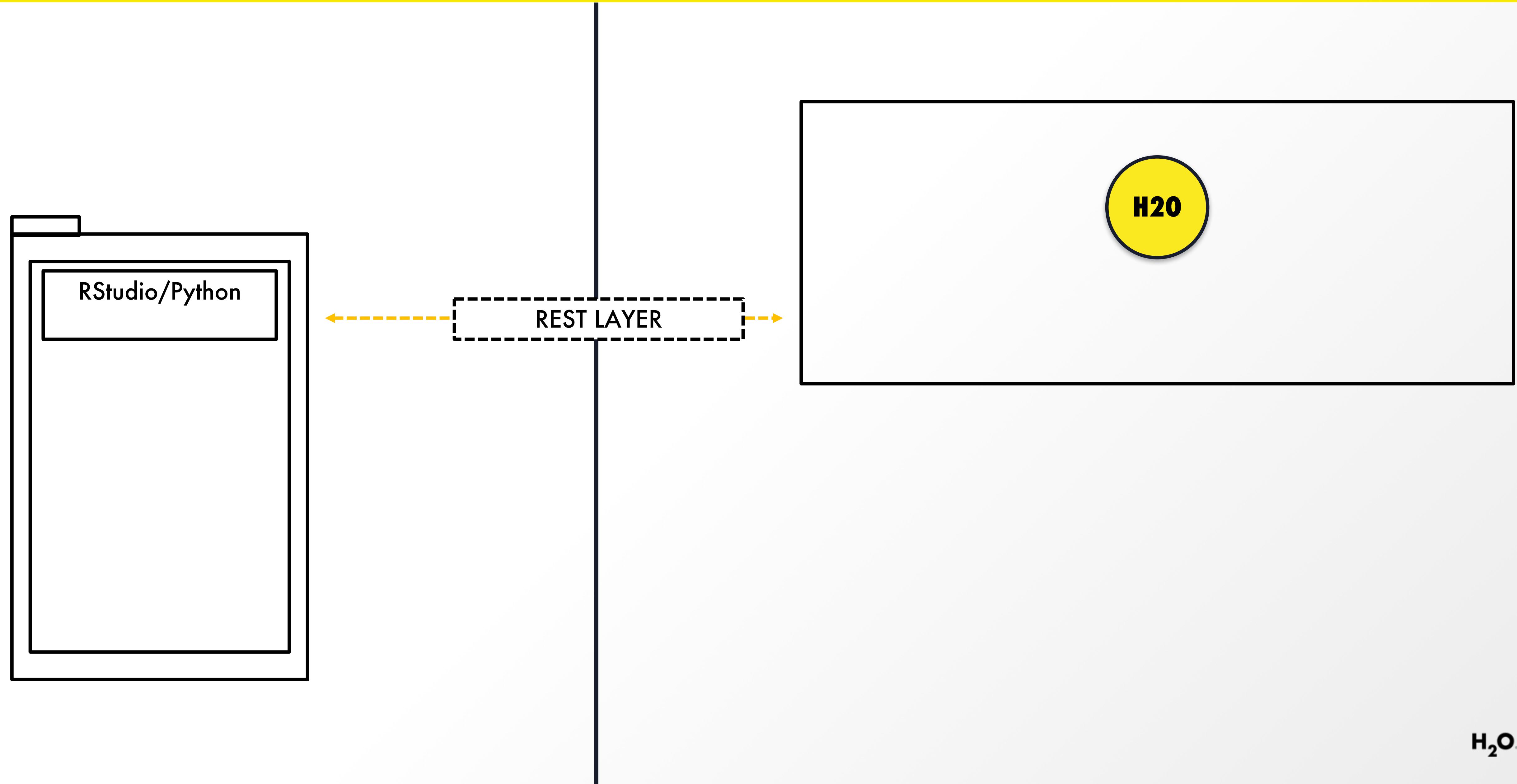
parallelism & distribution of work



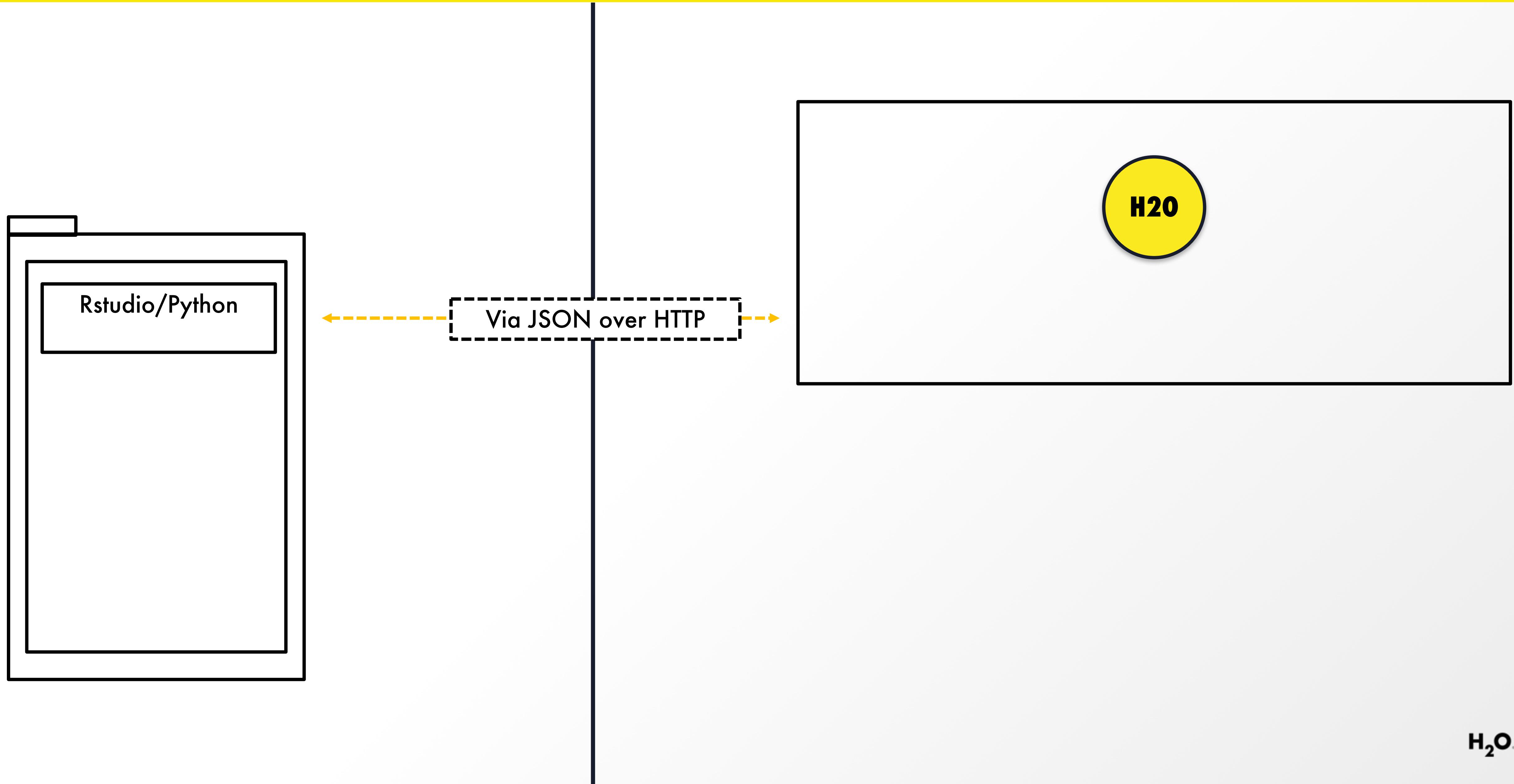
H2O API



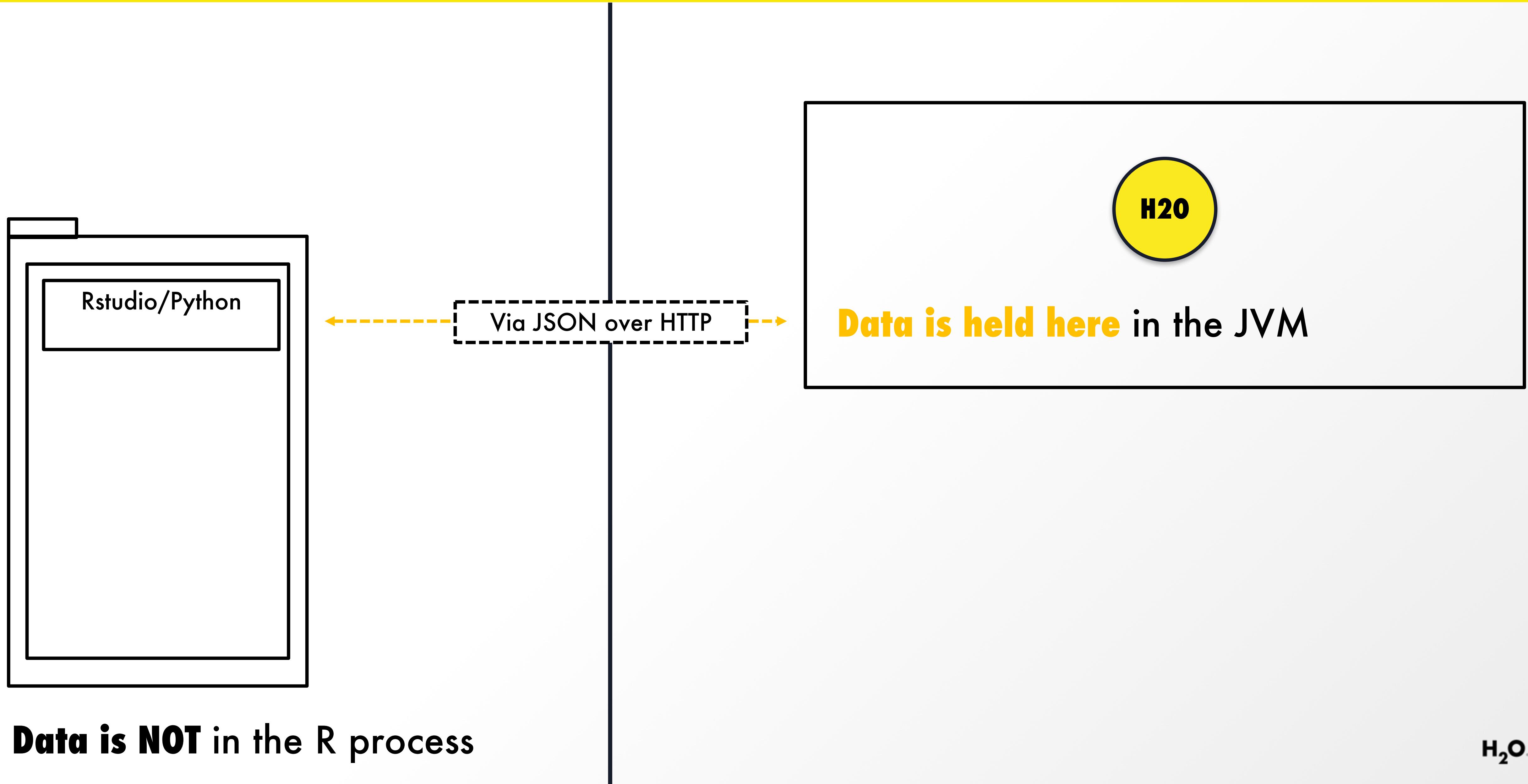
H2O API



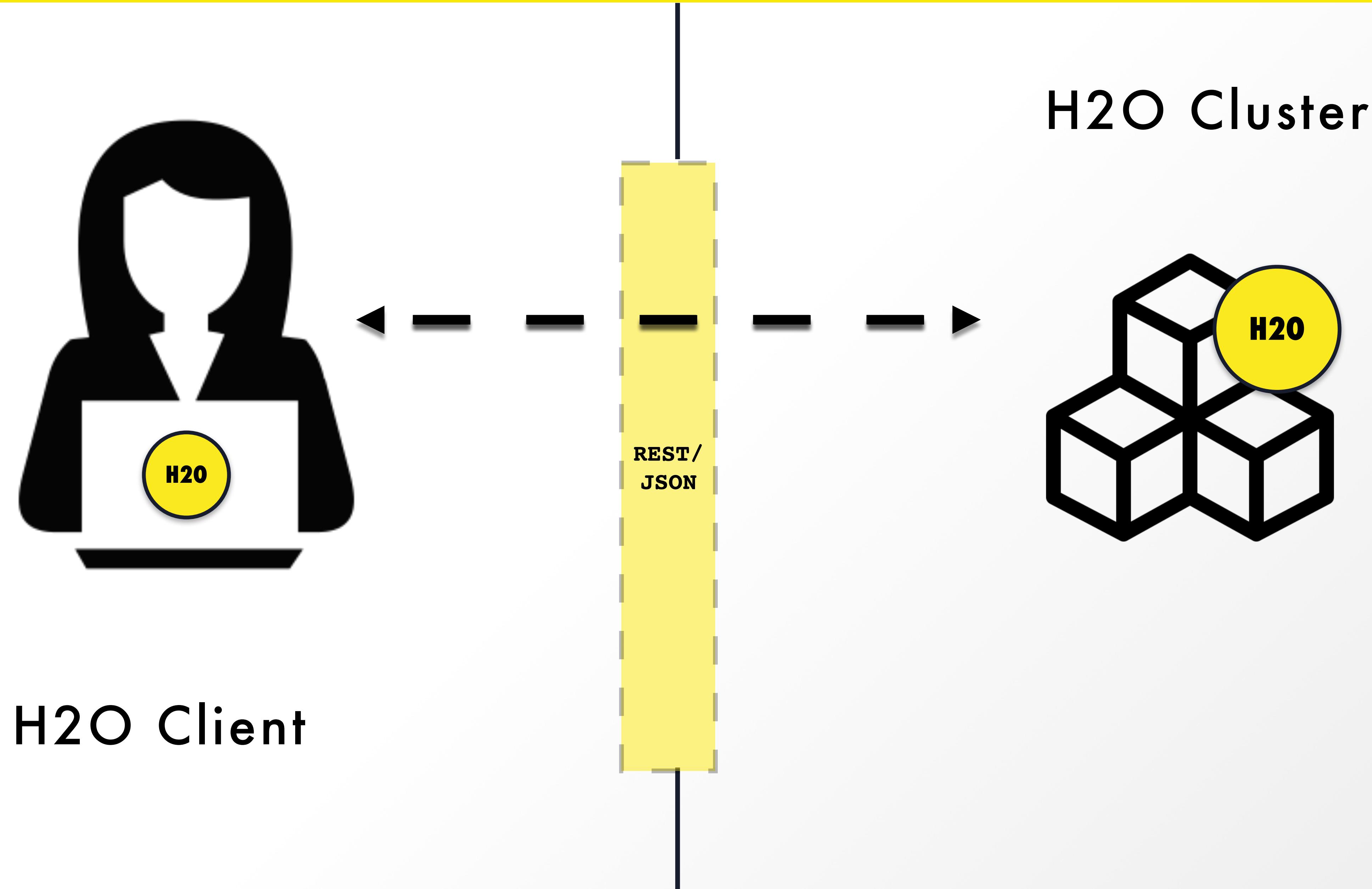
H2O API



H2O API

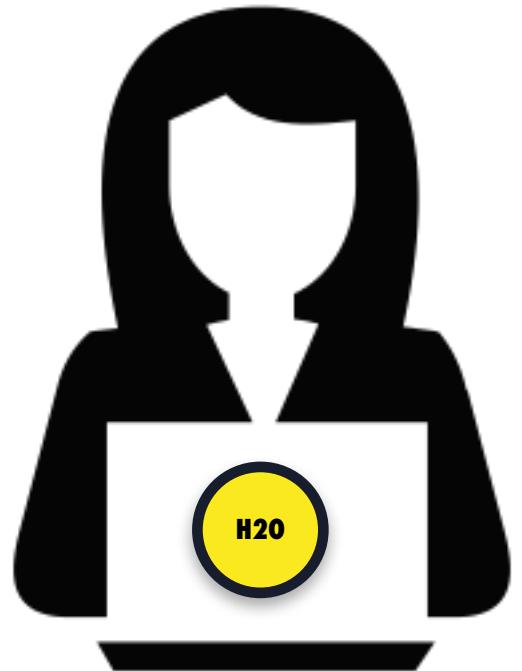


Client & Cluster Communication



Communication Layers: Interface

Standard R
Interface



the Client

RStudio Interface

```
Console | Terminal ~ /Desktop/rencontres-R-2018/h2o-deeplearning/ ~

H2O is not running yet, starting it now...

Note: In case of errors look at the following log files:
  /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.out
  /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.err

java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

R is connected to the H2O cluster:
  H2O cluster uptime:      3 seconds 970 milliseconds
  H2O cluster timezone:    America/Los_Angeles
  H2O data parsing timezone: UTC
  H2O cluster version:     3.20.0.2
  H2O cluster version age: 12 days
  H2O cluster name:        H2O_started_from_R_navdeepgill_kdm352
  H2O cluster total nodes: 1
  H2O cluster total memory: 3.56 GB
  H2O cluster total cores: 8
  H2O cluster allowed cores: 8
  H2O cluster healthy:    TRUE
  H2O Connection ip:       localhost
  H2O Connection port:     54321
  H2O Connection proxy:   NA
  H2O Internal Security: FALSE
  H2O API Extensions:    XGBoost, Algos, AutoML, Core V3, Core V4
  R Version:               R version 3.4.0 (2017-04-21)
```

Communication Layers: Code Script

Rstudio using H2O Package

RStudio



the Client

```
Console Terminal ~~/Desktop/rencontres-R-2018/h2o-deeplearning/ ↵

H2O is not running yet, starting it now...

Note: In case of errors look at the following log files:
/var/folders/55/rj4cny_s29q4vn1wjt_x08sm0000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.out
/var/folders/55/rj4cny_s29q4vn1wjt_x08sm0000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.err

java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

R is connected to the H2O cluster:
H2O cluster uptime:      3 seconds 970 milliseconds
H2O cluster timezone:    America/Los_Angeles
H2O data parsing timezone: UTC
H2O cluster version:     3.20.0.2
H2O cluster version age: 12 days
H2O cluster name:        H2O_started_from_R_navdeepgill_kdm352
H2O cluster total nodes: 1
H2O cluster total memory: 3.56 GB
H2O cluster total cores: 8
H2O cluster allowed cores: 8
H2O cluster healthy:     TRUE
H2O Connection ip:       localhost
H2O Connection port:     54321
H2O Connection proxy:    NA
H2O Internal Security:  FALSE
H2O API Extensions:    XGBoost, Algos, AutoML, Core V3, Core V4
R Version:               R version 3.4.0 (2017-04-21)

> |
```

Communication Layers: A Command

Importing Big Data with R Code

R Commands



the Client

```
Console Terminal ~~/Desktop/rencontres-R-2018/h2o-deeplearning/ ~

H2O is not running yet, starting it now...

Note: In case of errors look at the following log files:
/var/folders/55/rj4cny_s29q4vn1wjt_x08sm0000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.out
/var/folders/55/rj4cny_s29q4vn1wjt_x08sm0000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.err

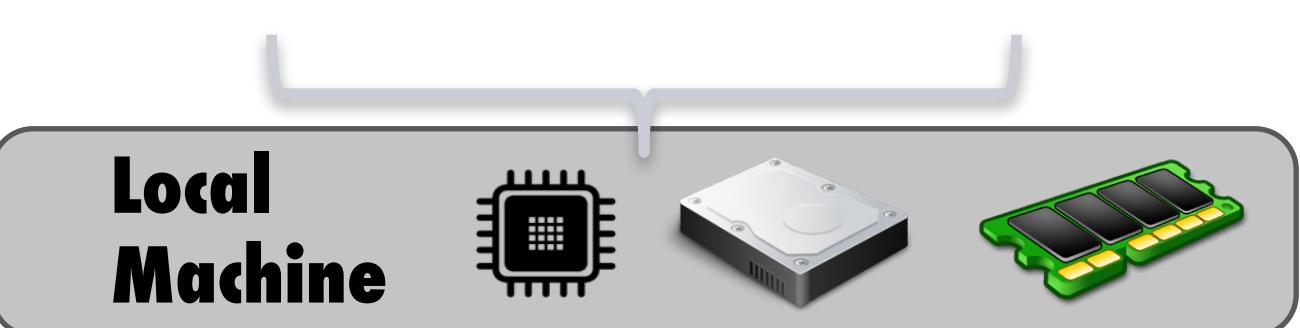
java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

R is connected to the H2O cluster:
  H2O cluster uptime:      3 seconds 970 milliseconds
  H2O cluster timezone:    America/Los_Angeles
  H2O data parsing timezone: UTC
  H2O cluster version:    3.20.0.2
  H2O cluster version age: 12 days
  H2O cluster name:       H2O_started_from_R_navdeepgill_kdm352
  H2O cluster total nodes: 1
  H2O cluster total memory: 3.56 GB
  H2O cluster total cores: 8
  H2O cluster allowed cores: 8
  H2O cluster healthy:     TRUE
  H2O Connection ip:       localhost
  H2O Connection port:     54321
  H2O Connection proxy:   NA
  H2O Internal Security:  FALSE
  H2O API Extensions:   XGBoost, Algos, AutoML, Core V3, Core V4
  R Version:              R version 3.4.0 (2017-04-21)

> |
```

h2o.importFile(...)



Communication Layers: A Command

Importing Big Data with R Code

R Commands



the Client

```
Console Terminal ~ /Desktop/rencontres-R-2018/h2o-deeplearning/ ~
H2O is not running yet, starting it now...
Note: In case of errors look at the following log files:
      /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.out
      /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.err

java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

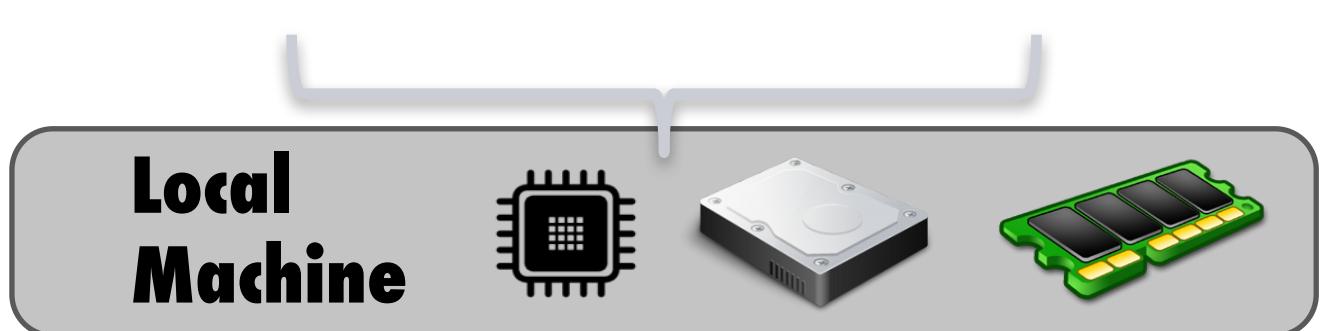
Starting H2O JVM and connecting: . Connection successful!

R is connected to the H2O cluster:
  H2O cluster uptime:      3 seconds 970 milliseconds
  H2O cluster timezone:    America/Los_Angeles
  H2O data parsing timezone: UTC
  H2O cluster version:    3.20.0.2
  H2O cluster version age: 12 days
  H2O cluster name:       H2O_started_from_R_navdeepgill_kdm352
  H2O cluster total nodes: 1
  H2O cluster total memory: 3.56 GB
  H2O cluster total cores: 8
  H2O cluster allowed cores: 8
  H2O cluster healthy:    TRUE
  H2O Connection ip:      localhost
  H2O Connection port:    54321
  H2O Connection proxy:   NA
  H2O Internal Security: FALSE
  H2O API Extensions:   XGBoost, Algos, AutoML, Core V3, Core V4
  R Version:              R version 3.4.0 (2017-04-21)

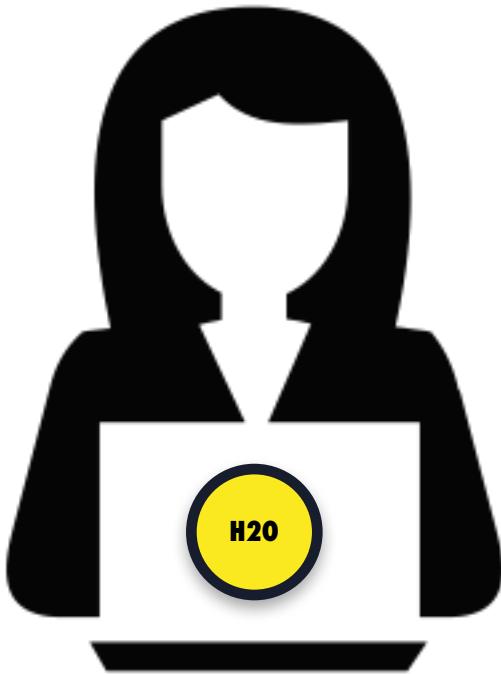
> |
```

Path to Your Dataset

`h2o.importFile(...)`



Fourth: Communicate



`h2o.importFile(...)`
requests file import

```
Console Terminal ~Desktop/rencontres-R-2018/h2o-deeplearning/ ~

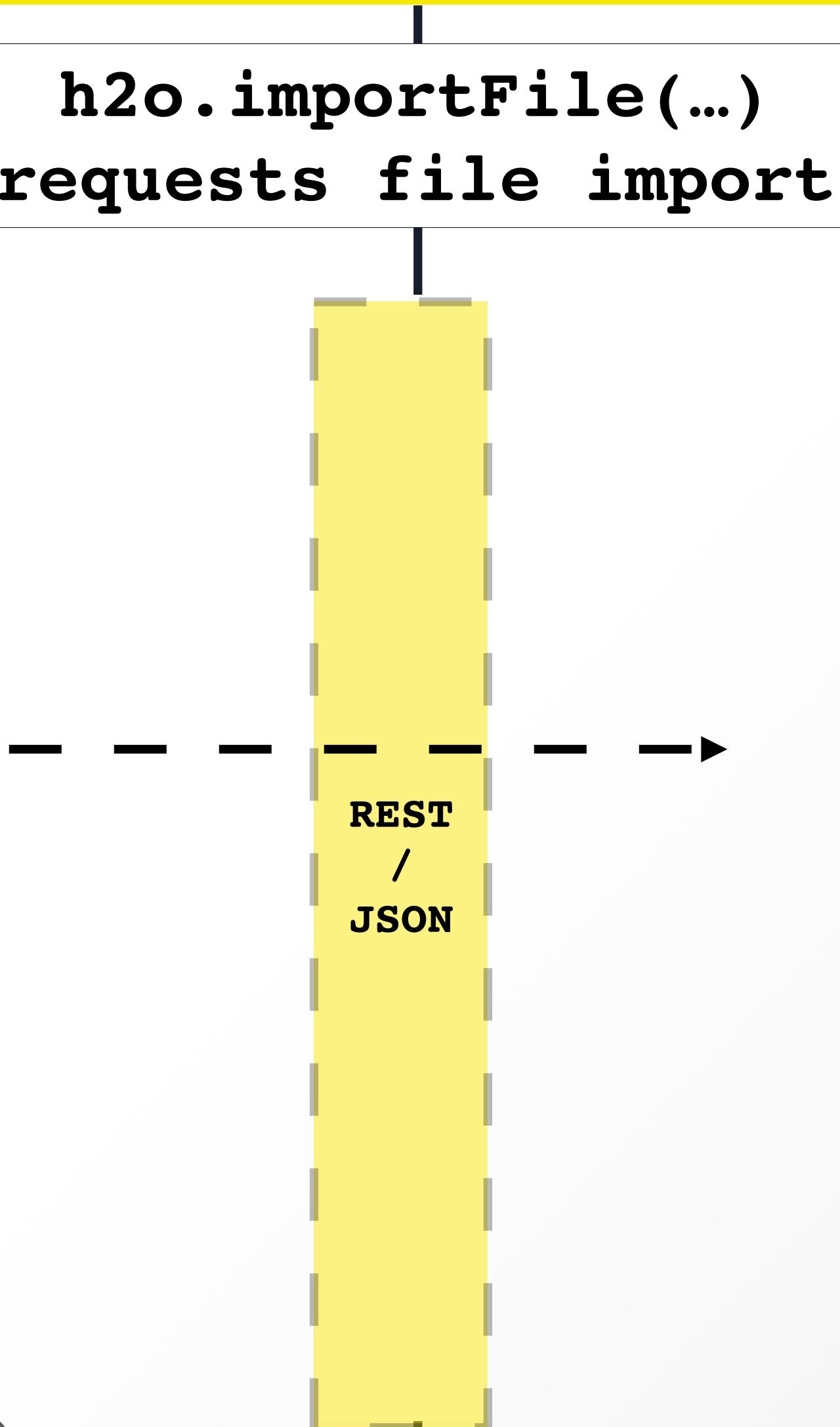
H2O is not running yet, starting it now...

Note: In case of errors look at the following log files:
  /var/folders/55/rj4cny_s29q4vn1jt_x08sm0000gn/T/RtmpH6ZkxR/h2o_navdeepgill_started_from_r.out
  /var/folders/55/rj4cny_s29q4vn1jt_x08sm0000gn/T/RtmpH6ZkxR/h2o_navdeepgill_started_from_r.err

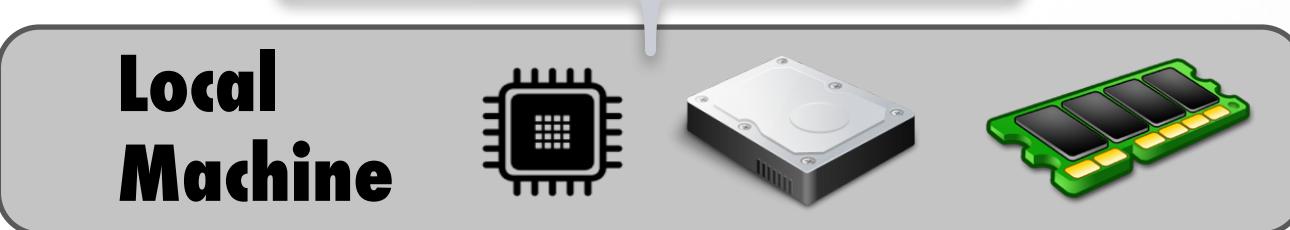
java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

R is connected to the H2O cluster:
H2O cluster uptime: 3 seconds 970 milliseconds
H2O cluster timezone: America/Los_Angeles
H2O data parsing timezone: UTC
H2O cluster version: 3.28.0.2
H2O cluster version age: 12 days
H2O cluster name: H2O_started_from_R_navdeepgill_kdm352
H2O cluster total nodes: 1
H2O cluster total memory: 3.56 GB
H2O cluster total cores: 8
H2O cluster allowed cores: 8
H2O cluster healthy: TRUE
H2O Connection ip: localhost
H2O Connection port: 54321
H2O Connection proxy: NA
H2O Internal Security: FALSE
H2O API Extensions: XGBoost, Algos, AutoML, Core V3, Core V4
R Version: R version 3.4.0 (2017-04-21)
```



H2O Cluster



Fifth: Cluster Does Heavy Lifting



```
Console Terminal ~~/Desktop/rencontres-R-2018/h2o-deeplearning/~

H2O is not running yet, starting it now...

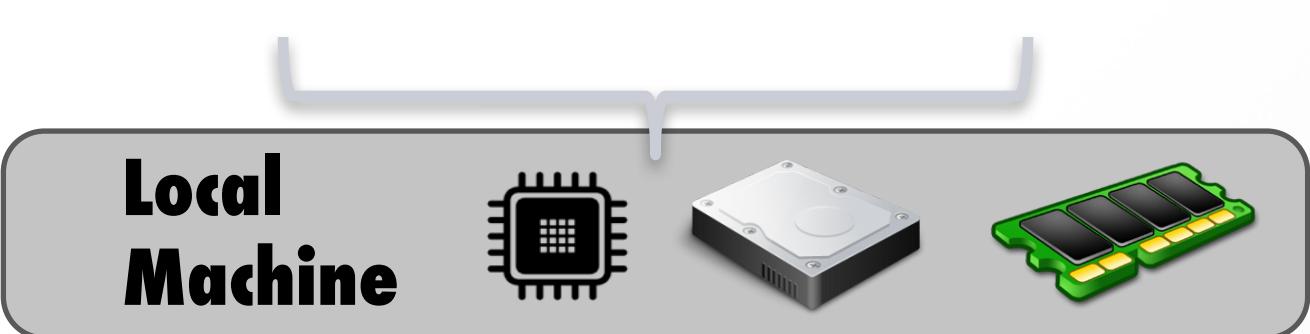
Note: In case of errors look at the following log files:
  /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZlxR/h2o_navdeepgill_started_from_r.out
  /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZlxR/h2o_navdeepgill_started_from_r.err

java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

R is connected to the H2O cluster:
H2O cluster uptime: 3 seconds 970 milliseconds
H2O cluster timezone: America/Los_Angeles
H2O data parsing timezone: UTC
H2O cluster version: 3.20.0.2
H2O cluster version age: 12 days
H2O cluster name: H2O_started_from_R_navdeepgill_kdm352
H2O cluster total nodes: 1
H2O cluster total memory: 3.56 GB
H2O cluster total cores: 8
H2O cluster allowed cores: 8
H2O cluster healthy: TRUE
H2O Connection ip: localhost
H2O Connection port: 54321
H2O Connection proxy: NA
H2O Internal Security: FALSE
H2O API Extensions: XGBoost, Algos, AutoML, Core V3, Core V4
R Version: R version 3.4.0 (2017-04-21)

> |
```



H2O Cluster

Fifth: Cluster Does Heavy Lifting



```
Console Terminal ~/Desktop/rencontres-R-2018/h2o-deeplearning/ ~

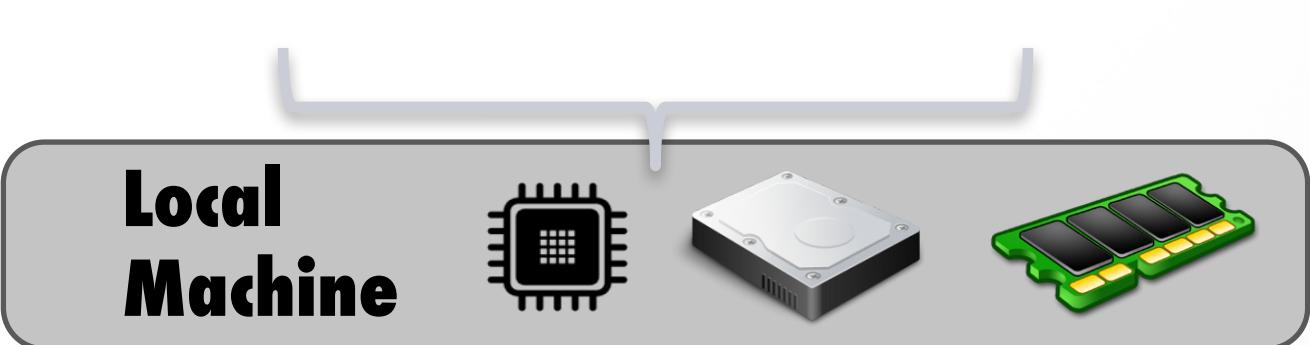
H2O is not running yet, starting it now...

Note: In case of errors look at the following log files:
  /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.out
  /var/folders/55/rj4cny_s29q4vn1wjt_x08sm000gn/T//RtmpH6ZkxR/h2o_navdeepgill_started_from_r.err

java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

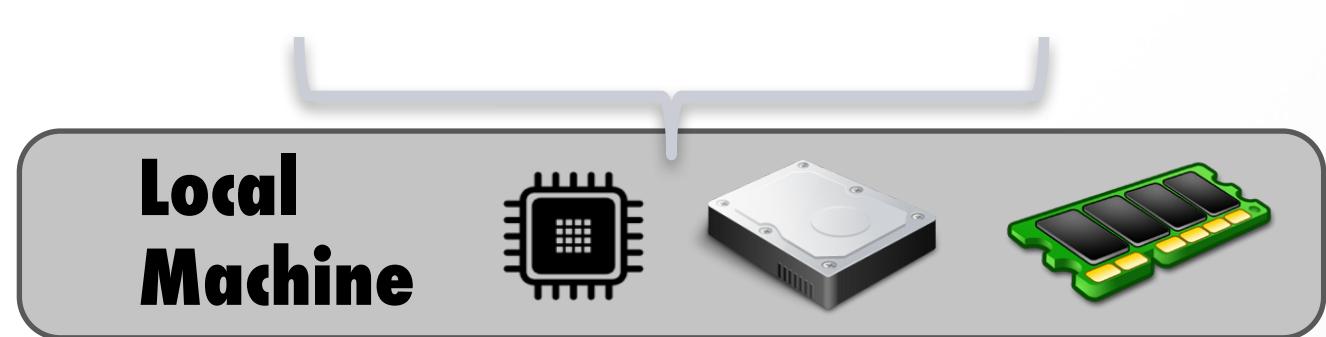
R is connected to the H2O cluster:
  H2O cluster uptime:      3 seconds 970 milliseconds
  H2O cluster timezone:    America/Los_Angeles
  H2O data parsing timezone: UTC
  H2O cluster version:     3.20.0.2
  H2O cluster version age: 12 days
  H2O cluster name:        H2O_started_from_R_navdeepgill_kdm352
  H2O cluster total nodes: 1
  H2O cluster total memory: 3.56 GB
  H2O cluster total cores: 8
  H2O cluster allowed cores: 8
  H2O cluster healthy:     TRUE
  H2O Connection ip:       localhost
  H2O Connection port:     54321
  H2O Connection proxy:    NA
  H2O Internal Security:  FALSE
  H2O API Extensions:    XGBoost, Algos, AutoML, Core V3, Core V4
  R Version:              R version 3.4.0 (2017-04-21)
```



Fifth: Cluster Does Heavy Lifting



```
Console Terminal ~/Desktop/rencontres-R-2018/h2o-deeplearning/ ~  
H2O is not running yet, starting it now...  
  
Note: In case of errors look at the following log files:  
/var/folders/55/rj4cny_s29q4vn1wjt_x08sm0000gn/T//RtmpH6ZlxR/h2o_navdeepgill_started_from_r.out  
/var/folders/55/rj4cny_s29q4vn1wjt_x08sm0000gn/T//RtmpH6ZlxR/h2o_navdeepgill_started_from_r.err  
  
java version "1.8.0_101"  
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)  
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)  
  
Starting H2O JVM and connecting: . Connection successful!  
  
R is connected to the H2O cluster:  
H2O cluster uptime: 3 seconds 970 milliseconds  
H2O cluster timezone: America/Los_Angeles  
H2O data parsing timezone: UTC  
H2O cluster version: 3.20.0.2  
H2O cluster version age: 12 days  
H2O cluster name: H2O_started_from_R_navdeepgill_kdm352  
H2O cluster total nodes: 1  
H2O cluster total memory: 3.56 GB  
H2O cluster total cores: 8  
H2O cluster allowed cores: 8  
H2O cluster healthy: TRUE  
H2O Connection ip: localhost  
H2O Connection port: 54321  
H2O Connection proxy: NA  
H2O Internal Security: FALSE  
H2O API Extensions: XGBoost, Algos, AutoML, Core V3, Core V4  
R Version: R version 3.4.0 (2017-04-21)  
> |
```



Fifth: Cluster Does Heavy Lifting



```
Console Terminal ~/Desktop/rencontres-R-2018/h2o-deeplearning/ ~

H2O is not running yet, starting it now...

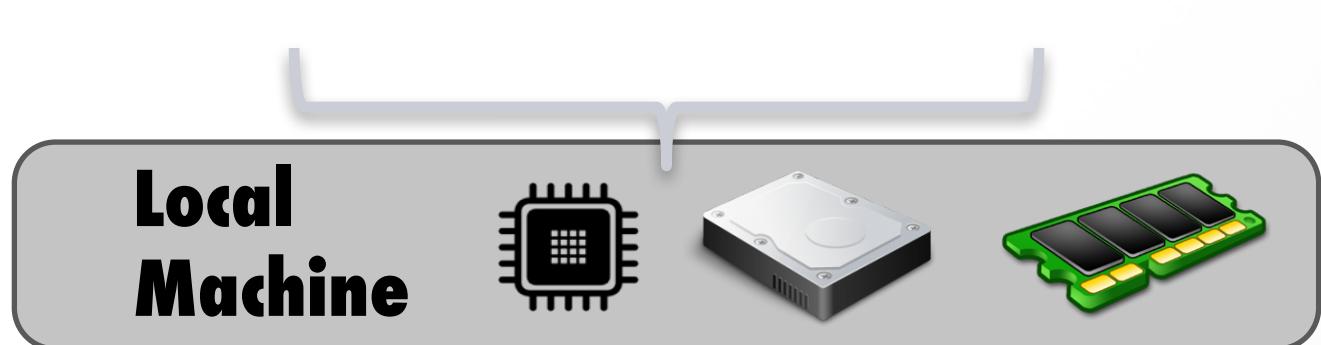
Note: In case of errors look at the following log files:
  /var/folders/55/rj4cny_s29q4vn1wt_x08sm0000gn/T/RtmpH6ZlxR/h2o_navdeepgill_started_from_r.out
  /var/folders/55/rj4cny_s29q4vn1wt_x08sm0000gn/T/RtmpH6ZlxR/h2o_navdeepgill_started_from_r.err

java version "1.8.0_101"
Java(TM) SE Runtime Environment (build 1.8.0_101-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.101-b13, mixed mode)

Starting H2O JVM and connecting: . Connection successful!

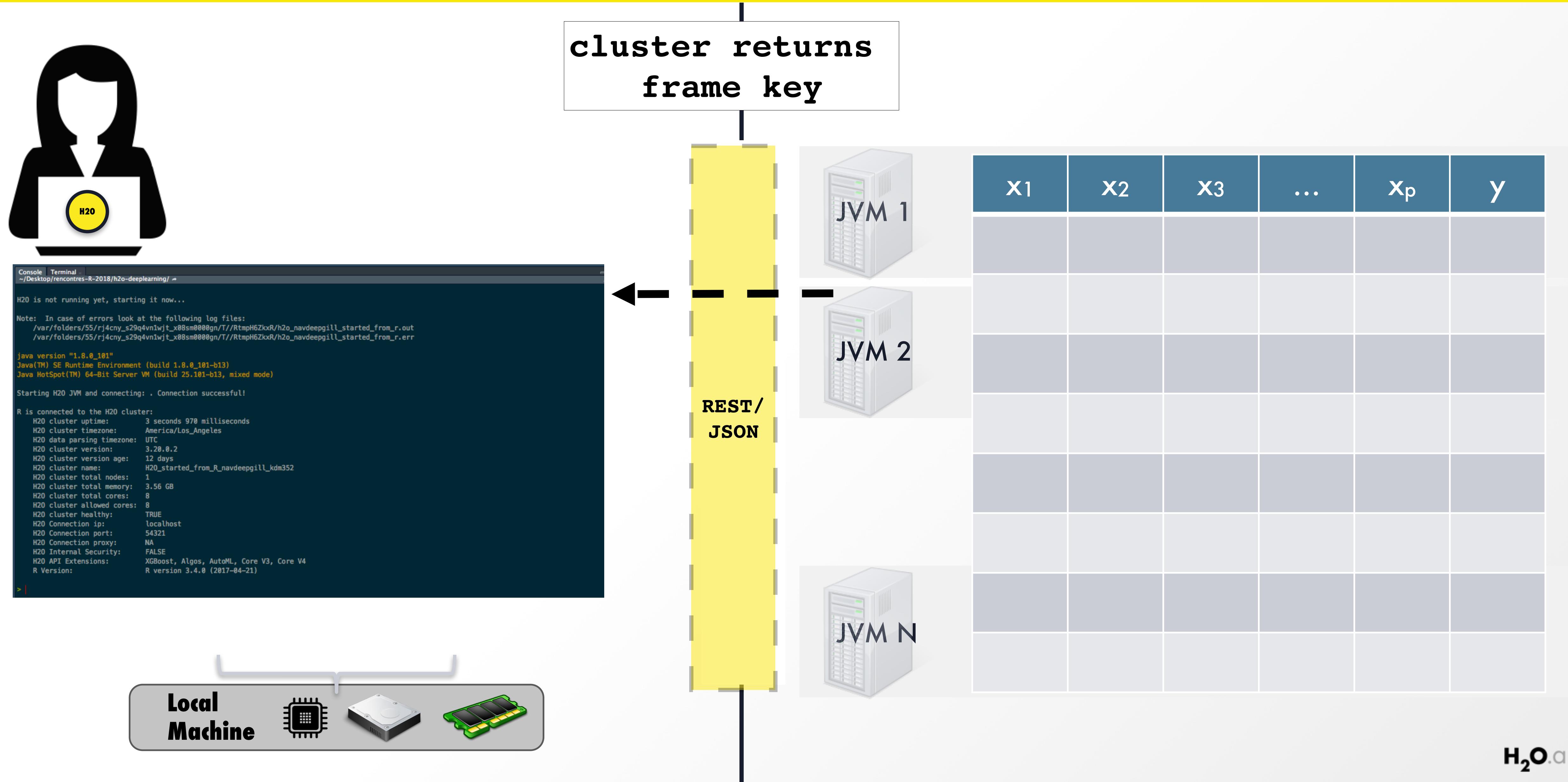
R is connected to the H2O cluster:
  H2O cluster uptime: 3 seconds 970 milliseconds
  H2O cluster timezone: America/Los_Angeles
  H2O data parsing timezone: UTC
  H2O cluster version: 3.28.0.2
  H2O cluster version age: 12 days
  H2O cluster name: H2O_started_from_R_navdeepgill_kdm352
  H2O cluster total nodes: 1
  H2O cluster total memory: 3.56 GB
  H2O cluster total cores: 8
  H2O cluster allowed cores: 8
  H2O cluster healthy: TRUE
  H2O Connection ip: localhost
  H2O Connection port: 54321
  H2O Connection proxy: NA
  H2O Internal Security: FALSE
  H2O API Extensions: XGBoost, Algos, AutoML, Core V3, Core V4
  R Version: R version 3.4.0 (2017-04-21)

> |
```



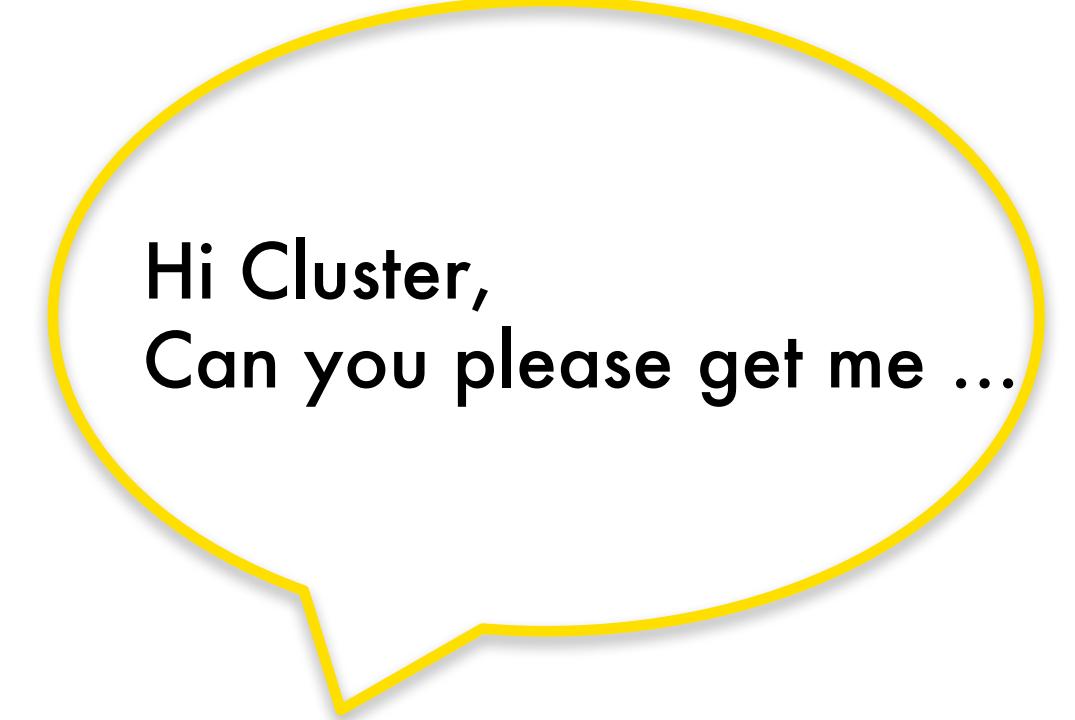
H2O Cluster

H2O Client



What does the Client Do?

Clients Only Tells the Cluster What to Do

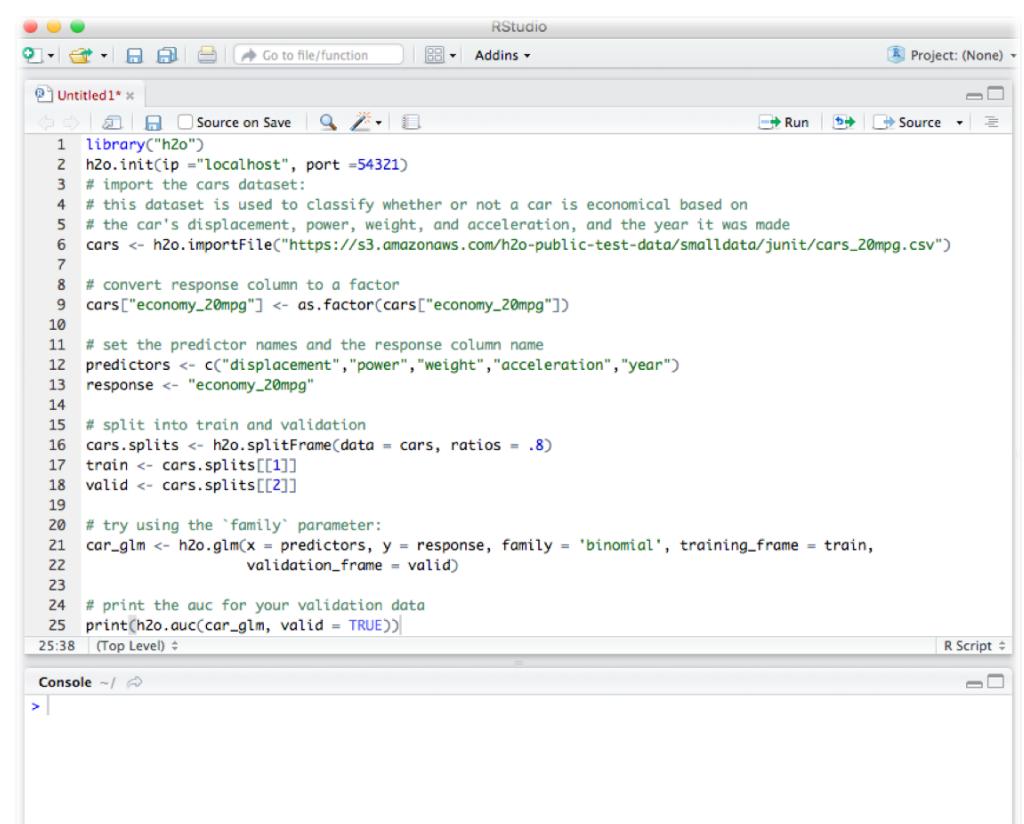


```
library("h2o")
h2o.init(ip = "localhost", port = 54321)
# import the cars dataset:
# this dataset is used to classify whether or not a car is economical based on
# the car's displacement, power, weight, and acceleration, and the year it was made
cars <- h2o.importFile("https://s3.amazonaws.com/h2o-public-test-data/smalldata/junit/cars_20mpg.csv")
# convert response column to a factor
cars["economy_20mpg"] <- as.factor(cars["economy_20mpg"])
# set the predictor names and the response column name
predictors <- c("displacement","power","weight","acceleration","year")
response <- "economy_20mpg"
# split into train and validation
cars.splits <- h2o.splitFrame(data = cars, ratios = .8)
train <- cars.splits[[1]]
valid <- cars.splits[[2]]
# try using the `family` parameter:
cor_glm <- h2o.glm(x = predictors, y = response, family = 'binomial', training_frame = train,
validation_frame = valid)
# print the auc for your validation data
print(h2o.auc(cor_glm, valid = TRUE))
```

Hi Cluster,
Can you please get me ...

Client Only Passes Requests

Big Data Never Flows Through The Client
Unless **Explicitly** Asked



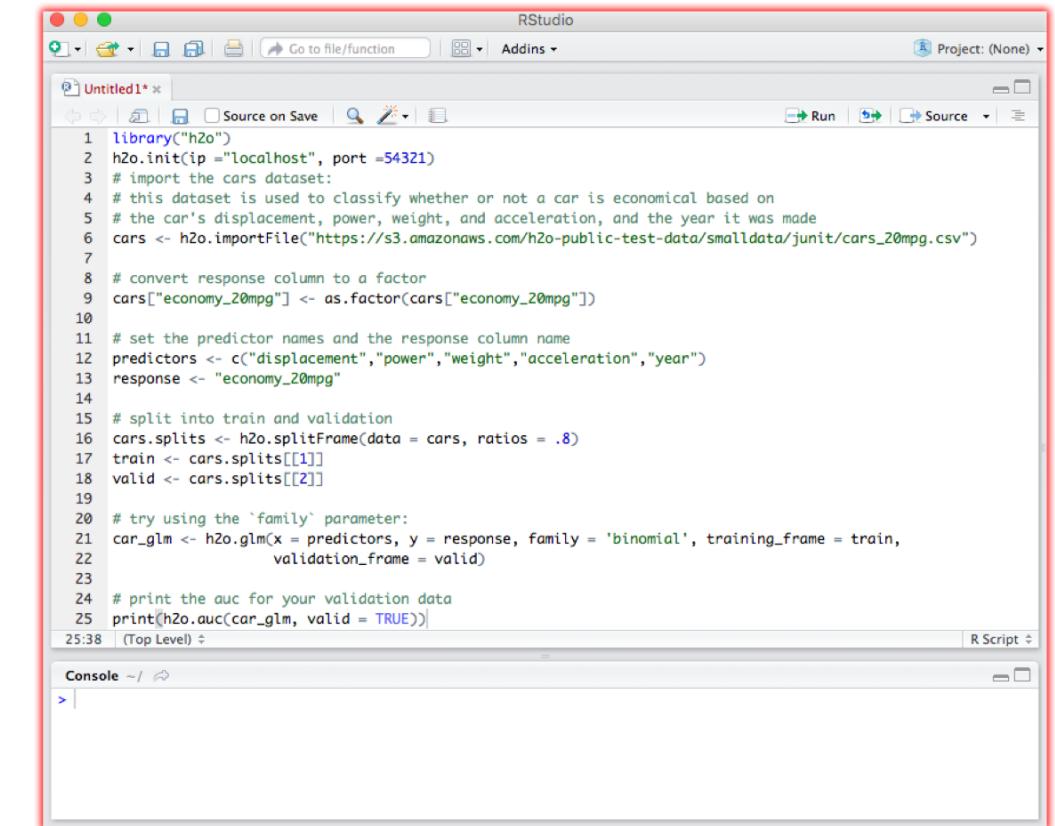
```
library("h2o")
h2o.init(ip = "localhost", port = 54321)
# import the cars dataset:
# this dataset is used to classify whether or not a car is economical based on
# the car's displacement, power, weight, and acceleration, and the year it was made
cars <- h2o.importFile("https://s3.amazonaws.com/h2o-public-test-data/smalldata/junit/cars_20mpg.csv")
# convert response column to a factor
cars["economy_20mpg"] <- as.factor(cars["economy_20mpg"])
# set the predictor names and the response column name
predictors <- c("displacement","power","weight","acceleration","year")
response <- "economy_20mpg"
# split into train and validation
cars.splits <- h2o.splitFrame(data = cars, ratios = .8)
train <- cars.splits[[1]]
valid <- cars.splits[[2]]
# try using the 'family' parameter:
car_glm <- h2o.glm(x = predictors, y = response, family = 'binomial', training_frame = train,
validation_frame = valid)
# print the auc for your validation data
print(h2o.auc(car_glm, valid = TRUE))
```

No, you take care of the
heavy lifting

What if?

Pulling Big Data into R Can Overwhelm Your Session

as.data.frame(my_big_dataframe)



A screenshot of an RStudio interface. The top bar shows 'RStudio' and 'Untitled1*'. The main area is a code editor with the following R code:

```
1 library("h2o")
2 h2o.init(ip = "localhost", port = 54321)
3 # import the cars dataset:
4 # this dataset is used to classify whether or not a car is economical based on
5 # the car's displacement, power, weight, and acceleration, and the year it was made
6 cars <- h2o.importFile("https://s3.amazonaws.com/h2o-public-test-data/smalldata/junit/cars_20mpg.csv")
7
8 # convert response column to a factor
9 cars["economy_20mpg"] <- as.factor(cars["economy_20mpg"])
10
11 # set the predictor names and the response column name
12 predictors <- c("displacement","power","weight","acceleration","year")
13 response <- "economy_20mpg"
14
15 # split into train and validation
16 cars.splits <- h2o.splitFrame(data = cars, ratios = .8)
17 train <- cars.splits[[1]]
18 valid <- cars.splits[[2]]
19
20 # try using the `family` parameter:
21 car_glm <- h2o.glm(x = predictors, y = response, family = 'binomial', training_frame = train,
22                      validation_frame = valid)
23
24 # print the auc for your validation data
25 print(h2o.auc(car_glm, valid = TRUE))
```

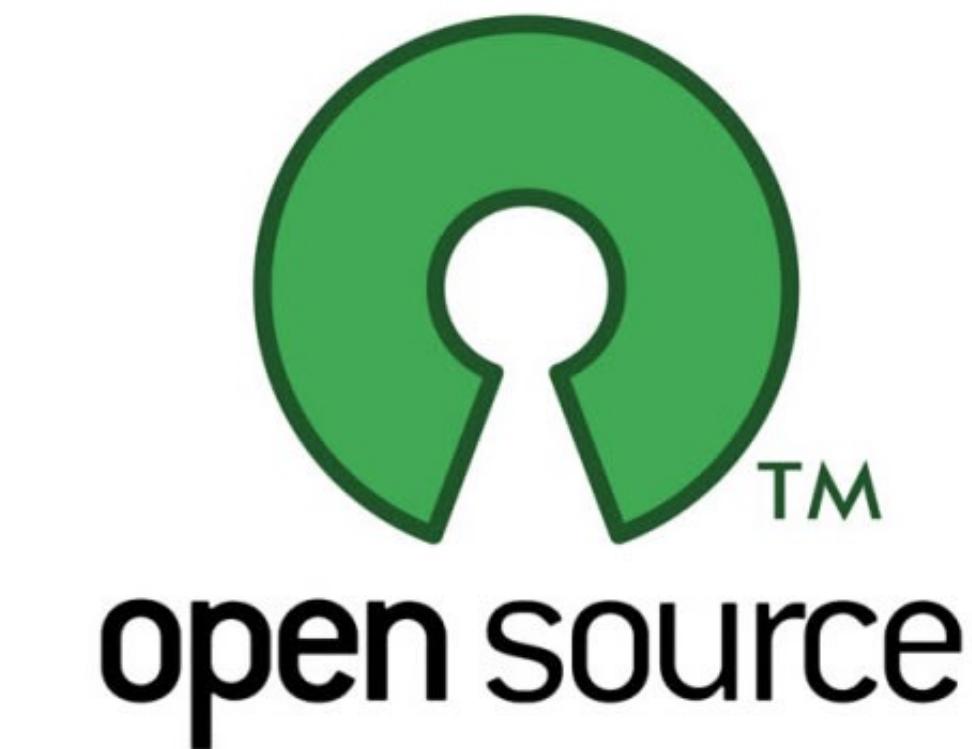
The bottom panel is the 'Console' window, which is currently empty.

AH!

H2O Resources

- Documentation: <http://docs.h2o.ai>
- Tutorials: <https://github.com/h2oai/h2o-tutorials>
- Slidedecks: <https://github.com/h2oai/h2o-meetups>
- Videos: <https://www.youtube.com/user/0xdata>
- Stack Overflow: <https://stackoverflow.com/tags/h2o>
- Google Group: <https://tinyurl.com/h2ostream>
- Gitter: <http://gitter.im/h2oai/h2o-3>
- Events & Meetups: <http://h2o.ai/events>

Contribute to H2O!



Get in touch over email, Gitter or JIRA.

<https://github.com/h2oai/h2o-3/blob/master/CONTRIBUTING.md>

D e m o

https://github.com/navdeep-G/jsm-2018/blob/master/h2o_airlines.R



Thank you!

@navdeep-G on Github

@Navdeep_Gill_ on Twitter

navdeep@h2o.ai