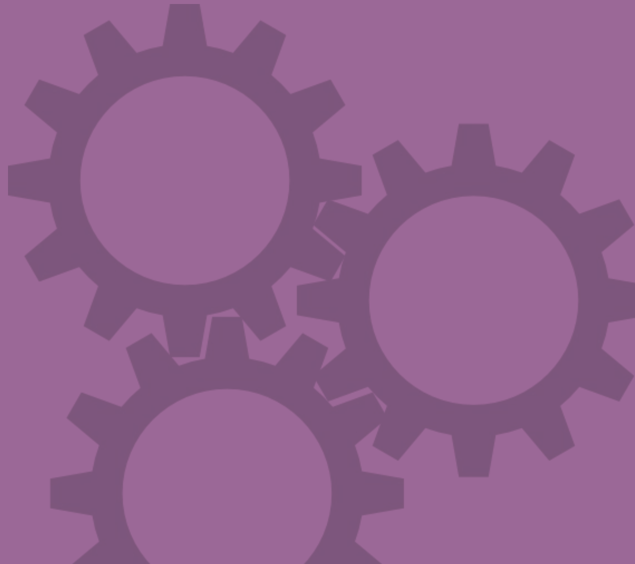
SPARK SUMMIT

PREVENTING REVENUE LEAKAGE AND MONITORING DISTRIBUTED SYSTEMS WITH MACHINE LEARNING



Flavio Clésio, Movile
Eiti Kimura, Movile

#EUai10

ABOUT US

Flávio Clésio

- Core *Machine Learning* at Movile
- MSc. in Production Engineering (Machine Learning in Credit Derivatives/NPL)
- Specialist in Database Engineering and Business Intelligence
- Blogger at *Mineração de Dados* (Data Mining) - http://mineracaodedados.wordpress.com
- Strata Hadoop World Singapore Speaker (2016)

flavioclesio

ABOUT US

Eiti Kimura

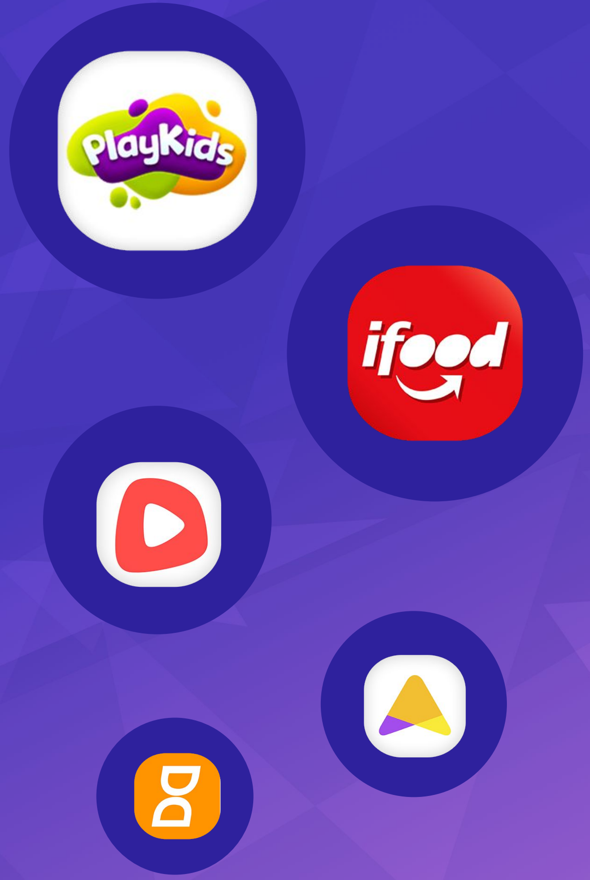
- IT Coordinator and Software Architect at Movile
- Msc. in Electrical Engineering
- Apache Cassandra MVP (2014/2015 and 2015/2016)
- Apache Cassandra *Contributor* (2015)
- Cassandra Summit *Speaker* (2014 and 2015)
- Strata Hadoop World Singapore Speaker (2016)

eitikimura

WE MAKE LIFE BETTER THROUGH OUR APPS

movile

Movile is the company behind several apps that makes the life easier



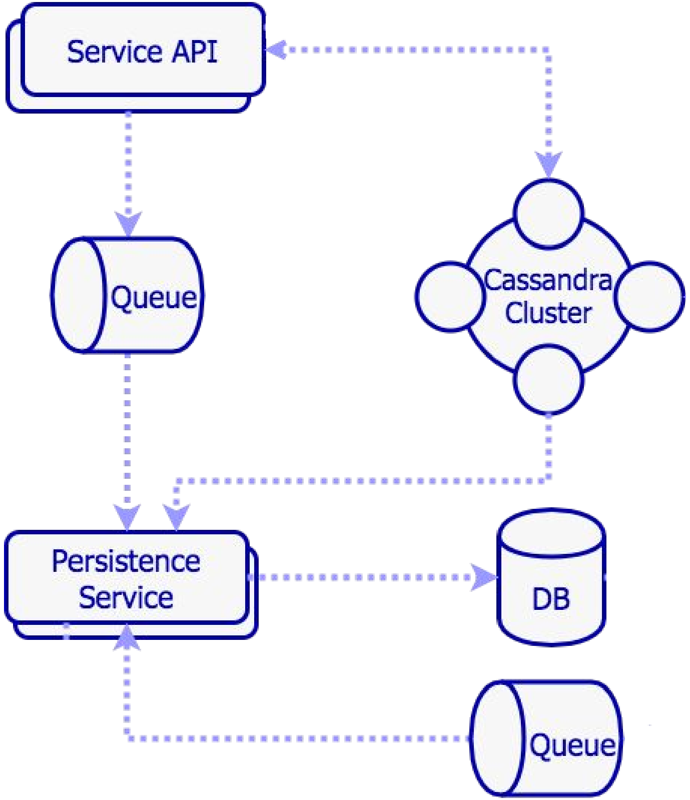
Agenda

- The Movile's Platform Case
- Practical Machine Learning Model Training
- Key Takeaways and Results

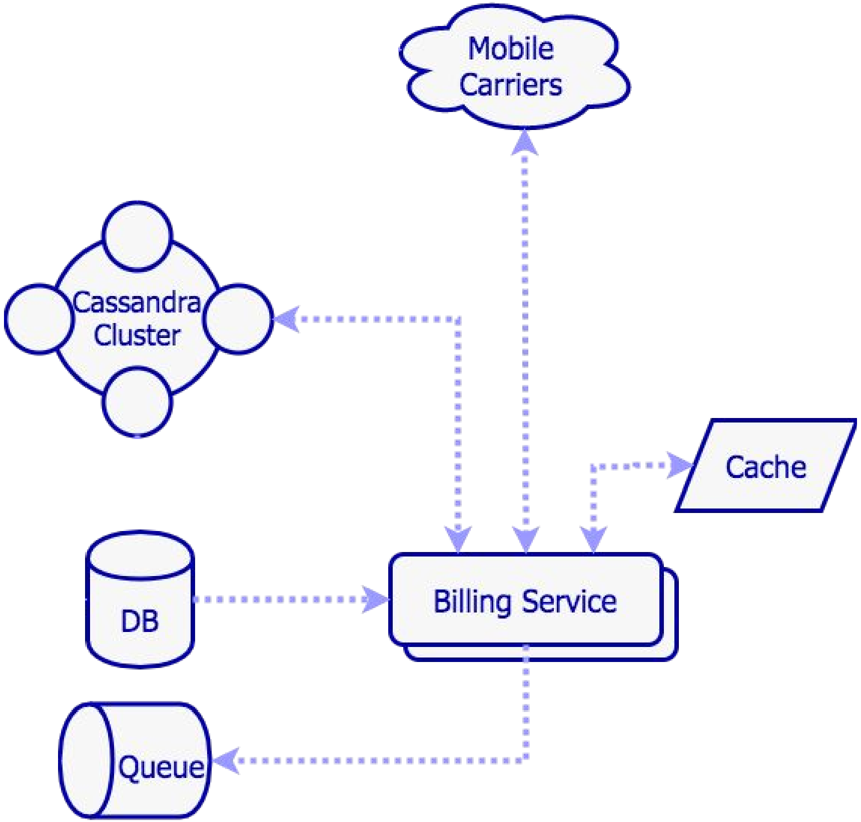
SBS

Subscription and Billing Platform

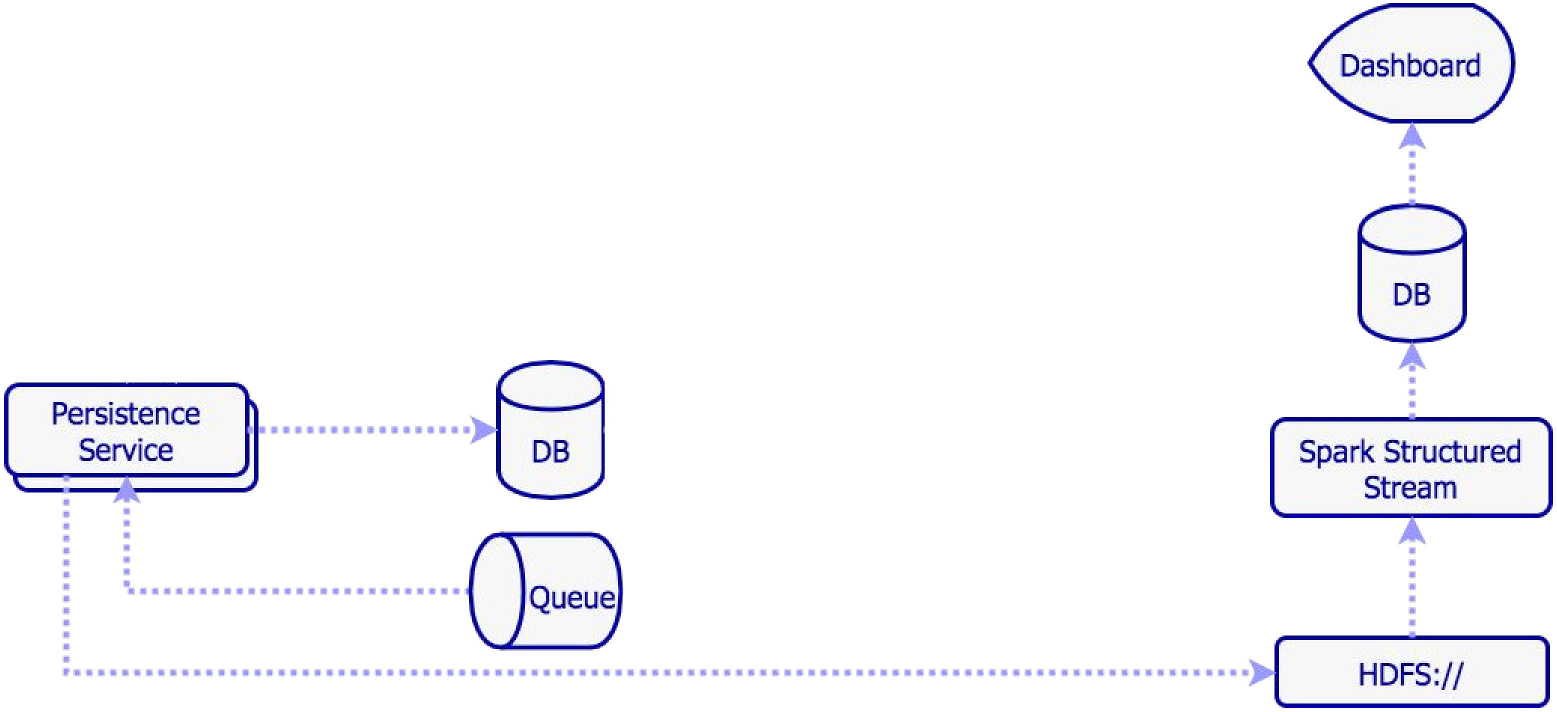
THE INPUT

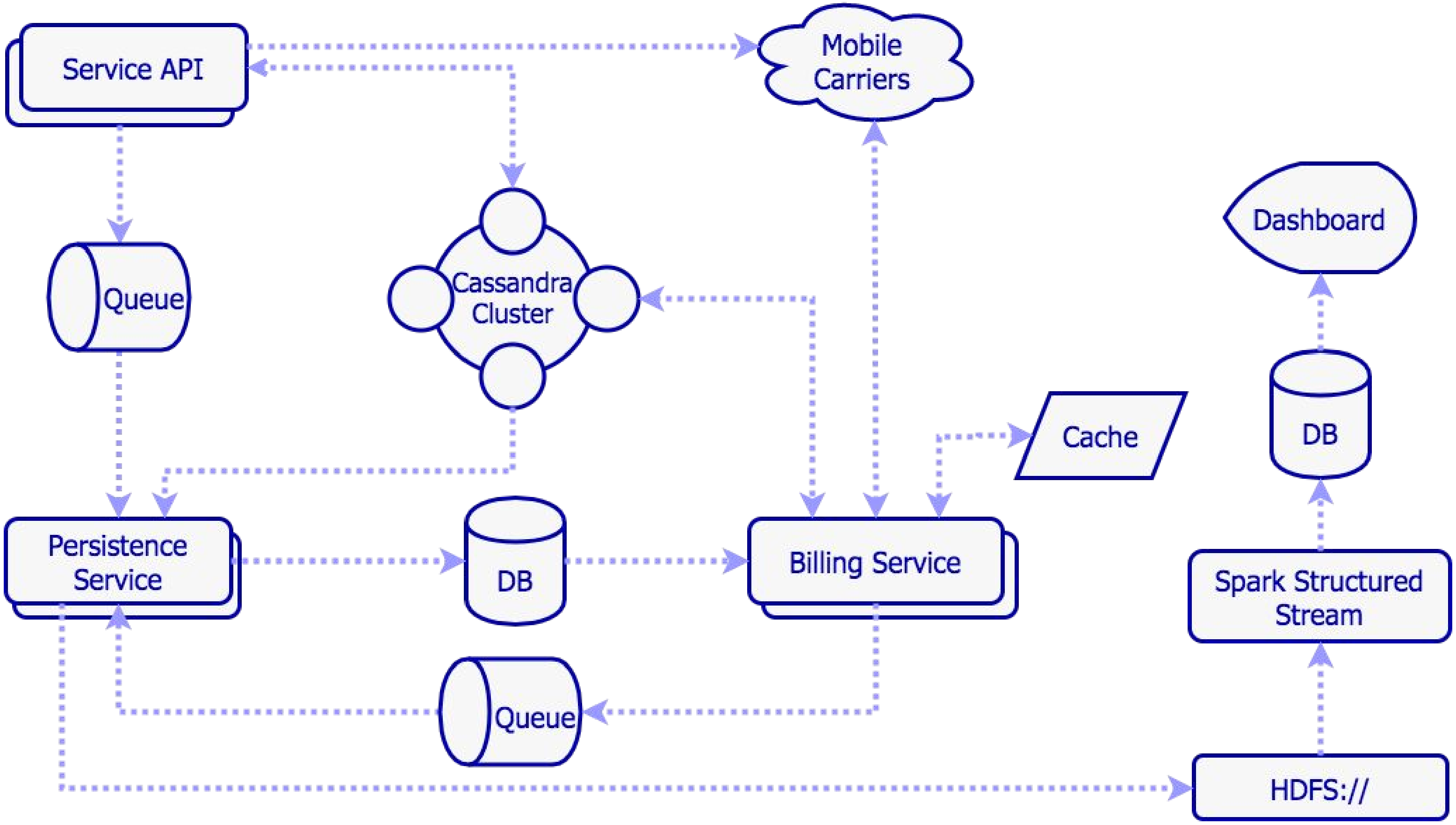

Service API
Queue
Cassandra Cluster
Persistence Service
DB
Queue

THE PROCESSING



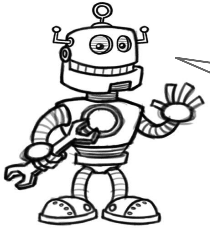
THE OUTPUT


Dashboard
DB
Persistence Service
DB
Spark Structured Stream
Queue
HDFS://


Service API
Mobile Carriers
Queue
Cassandra Cluster
Dashboard
Cache
DB
Persistence Service
DB
Billing Service
Spark Structured Stream
Queue
HDFS://

Main Problem: Monitoring

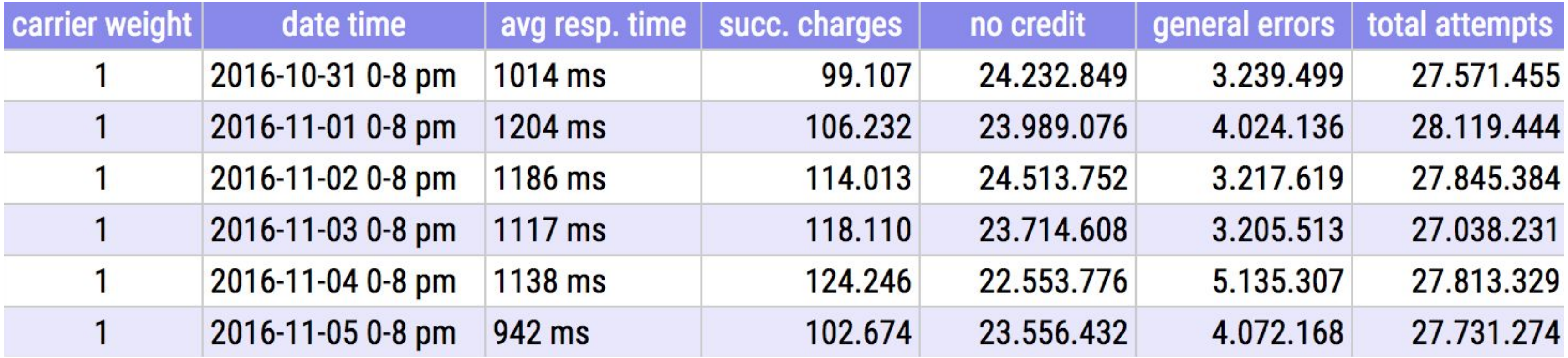
How can we check if platform is fully functional based on data analysis only?



Tip: what if we ask help to an intelligent system?

The Data Volumetry

- **236 Millions +** of billing requests attempt a day
- 4 main mobile carriers drive the operational work



carrier weight	date time	avg resp. time	succ. charges	no credit	general errors	total attempts
1	2016-10-31 0-8 pm	1014 ms	99.107	24.232.849	3.239.499	27.571.455
1	2016-11-01 0-8 pm	1204 ms	106.232	23.989.076	4.024.136	28.119.444
1	2016-11-02 0-8 pm	1186 ms	114.013	24.513.752	3.217.619	27.845.384
1	2016-11-03 0-8 pm	1117 ms	118.110	23.714.608	3.205.513	27.038.231
1	2016-11-04 0-8 pm	1138 ms	124.246	22.553.776	5.135.307	27.813.329
1	2016-11-05 0-8 pm	942 ms	102.674	23.556.432	4.072.168	27.731.274

Stating the problem

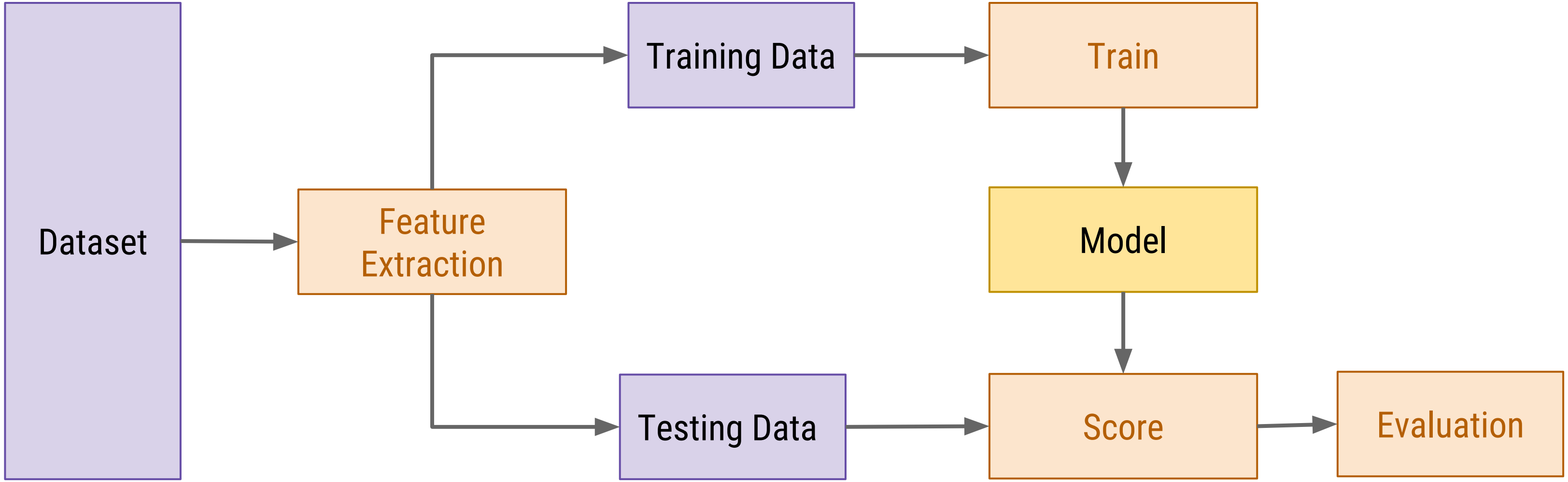
Sample of data (predicting the number of success)

# success	carrier_weight	hour	week	response_time	#no_credit	#errors	# attempts
61.083,	[4.0,	17h,	3.0,	1259.0,	24.751.650,	2.193.67,	26.314.551]
label/target		features					

SUPERVISED LEARNING

Linear Regression

The Modeling Lifecycle



KEEP
CALM
IT IS
DEMO
TIME

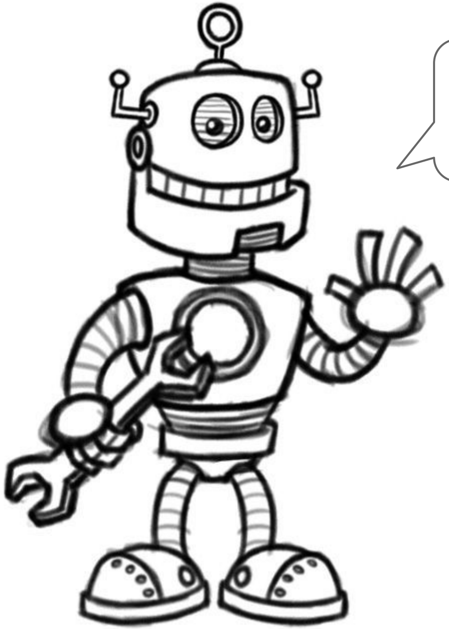
Training notebook available

github.com/fclesio/watcher-ai-samples

Evaluating Model Results

Machine Learning Tested Model	Accuracy	RMSE
Lasso with SGD Model	35%	0.32
Ridge Regression with SGD Model	87.5%	0.13
Elastic Net with SGD Model	35%	0.32
Decision Tree Model	93.4%	0.05

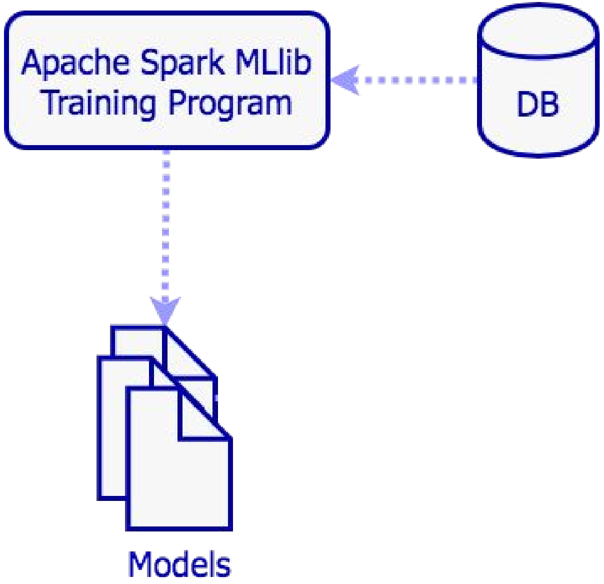
Watcher-ai Introduction



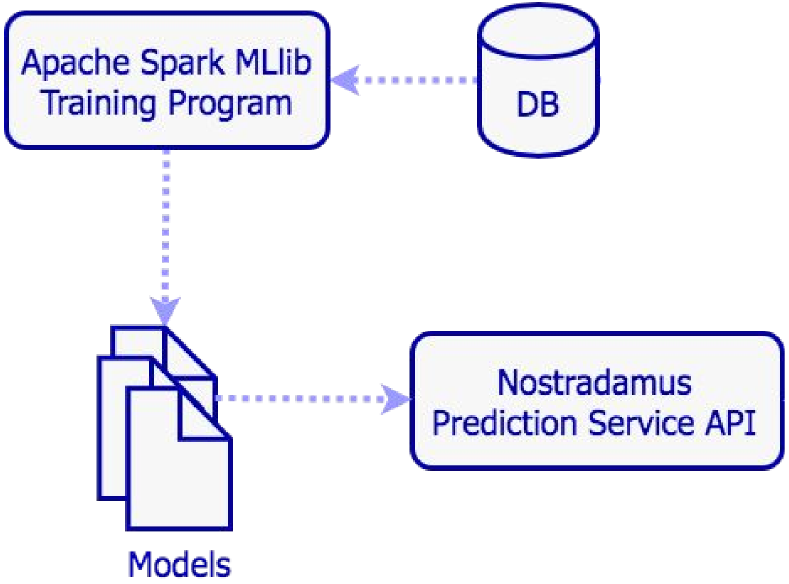
Hi I'm Watcher-ai!
It is nice to see you here

Applied Machine Learning to solving problems

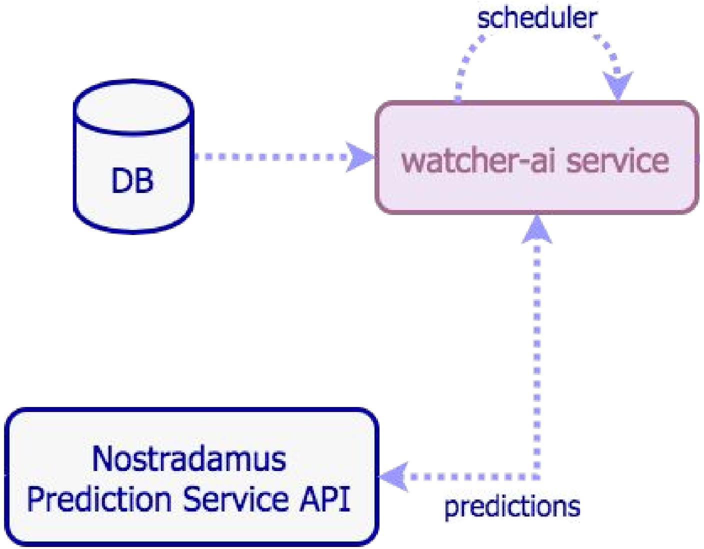
Watcher-ai Training



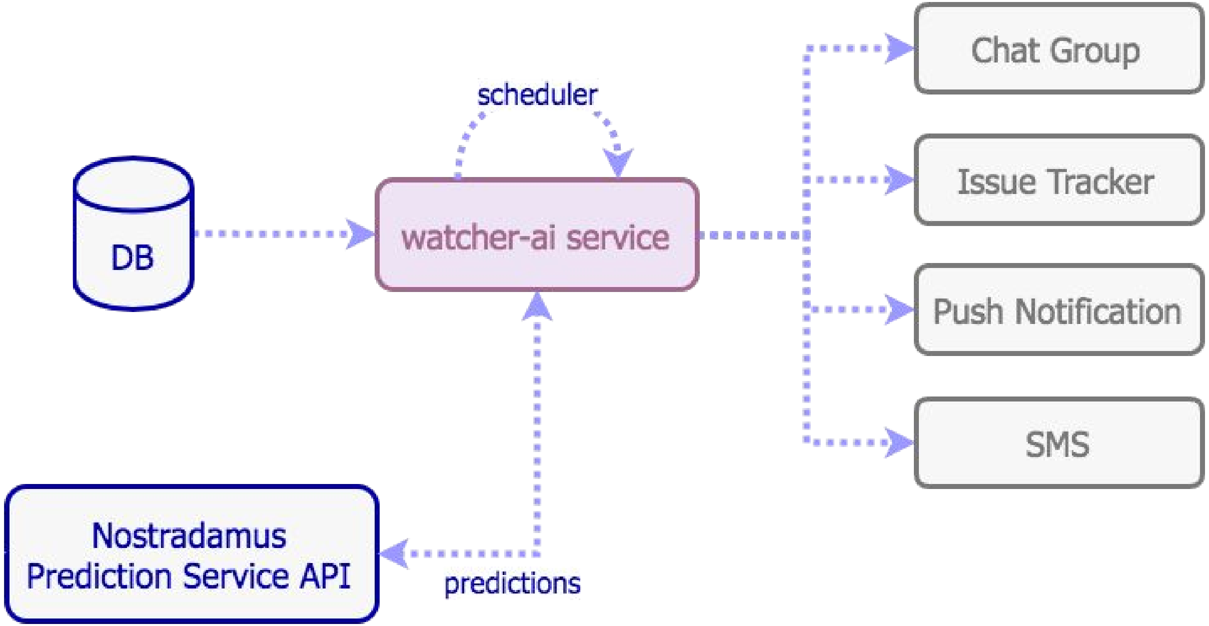
Watcher-ai using models


Apache Spark MLlib Training Program
DB
Models
Nostradamus Prediction Service API

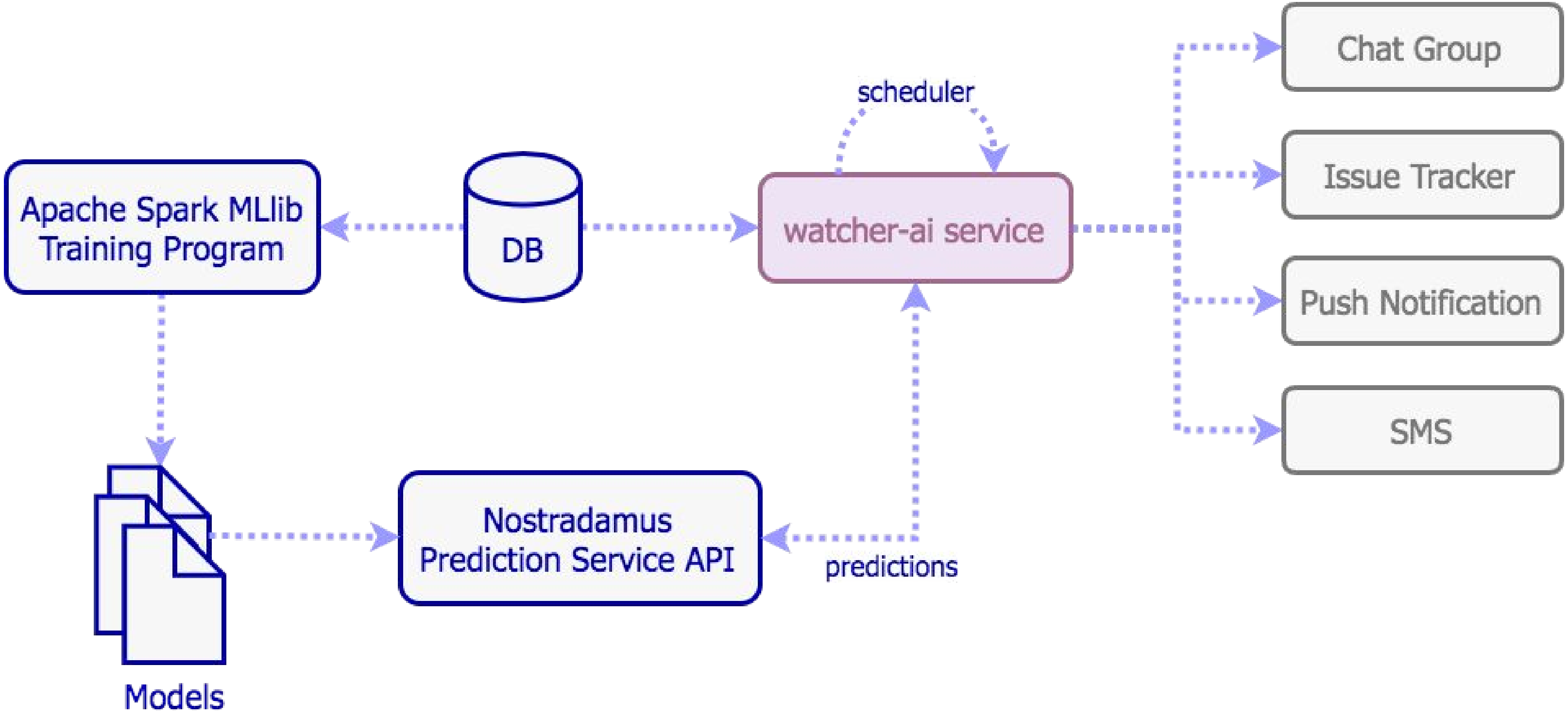
Watcher-ai request predictions



Watcher-ai notification



Watcher-ai Architecture



Lessons Learned

Empirical observations about this kind of problem

Regularization and Linear Methods

- Regularization doesn't fit so well with our low dimensional data

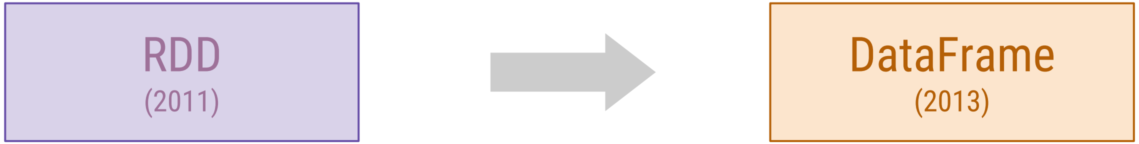
- Linear Methods are good for extrapolation but Decision Trees are more suitable for interpolation problems

The Timeseries Thing

- Time Series with thresholds didn't work in the past because we have several exogenous factors that make the regular algorithms behaving badly.

- We avoid ~~(totally removed)~~ fixed thresholds based on standard deviations

Why we changed from RDD to Dataframe?



RDD
(2011)

DataFrame
(2013)

distributed collection of JVM objects

functional operators like (map, filter, etc)

Distributed collection of Row objects

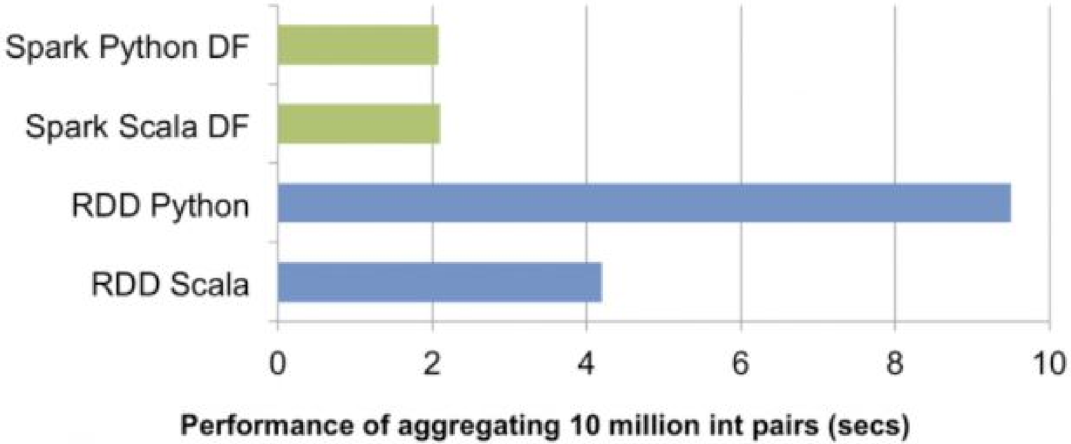
Expression-base operations and UDF

Logical plans and optimizer

Fast/efficient internal representation

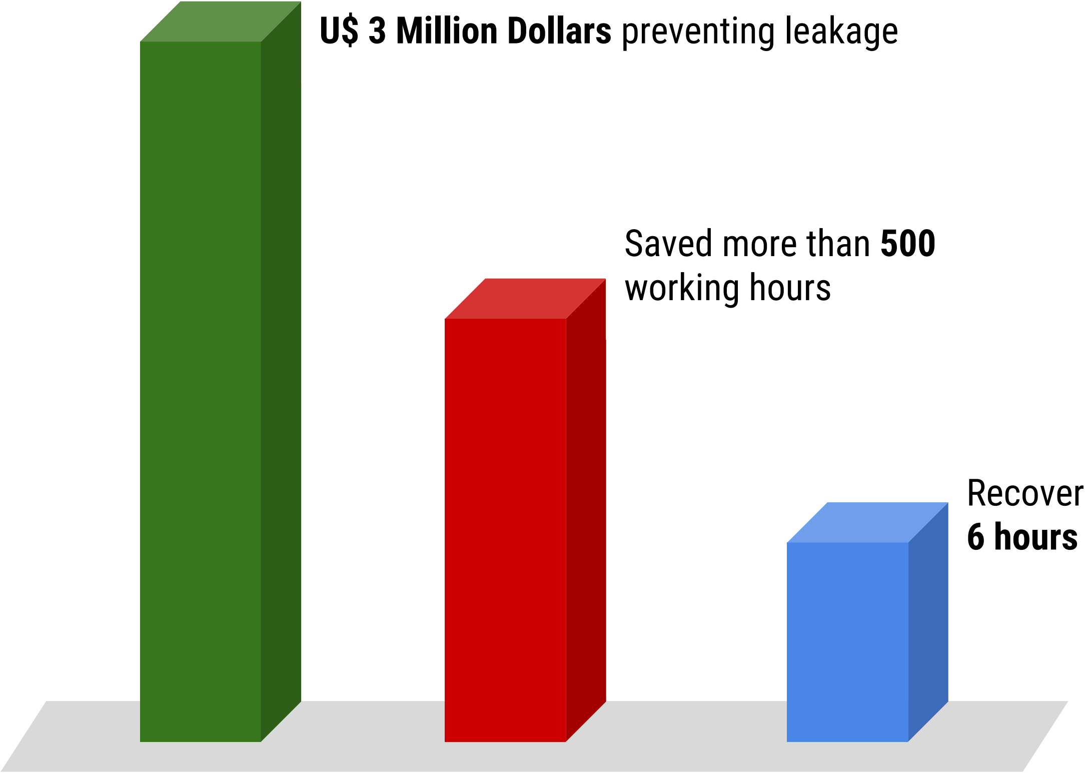
Why we changed from RDD to Dataframe?

- A good way to perform Grid-Search in our models
- Simpler and cleaner code, better to debug



Final Results

movile



Avoid to Lose more than
U$ 3 Million Dollars preventing leakage

Saved more than **500**
working hours

Recovery Time drops from
6 hours to **1 hour**

Our Goals

- Able to prevent revenue loss
- The main monitoring system
- Successful case of applied Machine Learning
- Simple solution with Apache Spark

SPARK SUMMIT

THANK YOU!

github.com/fclesio/watcher-ai-samples

eitikimura

flavioclesio

eiti.kimura@movile.com

flavio.clesio@movile.com

movile