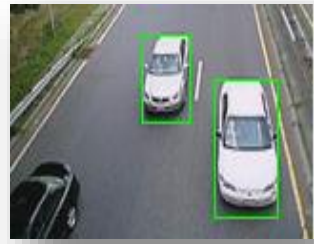
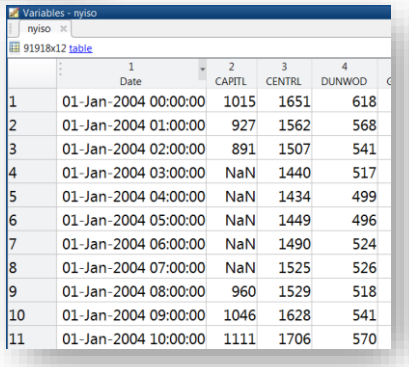
Building a Big Engineering Data Analytics System using MATLAB

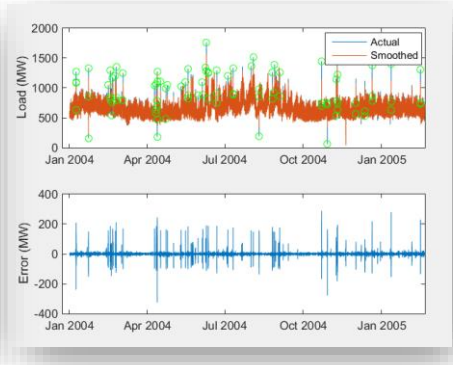
Arvind Hosagrahara
Principal Technical Consultant
Arvind.Hosagrahara@mathworks.com
Tel: 310-819-3960

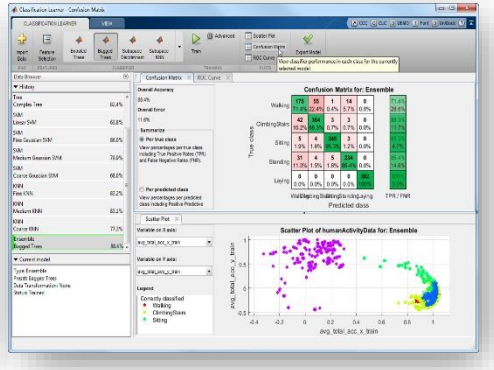
Trend: Data Economy

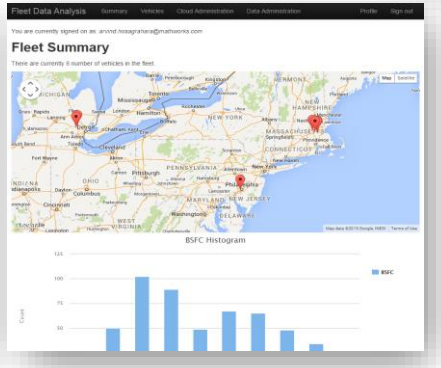
> ***"Information is the oil of the 21st century, and analytics is the combustion engine"***
>
> *Peter Sondergaard, Gartner Research*





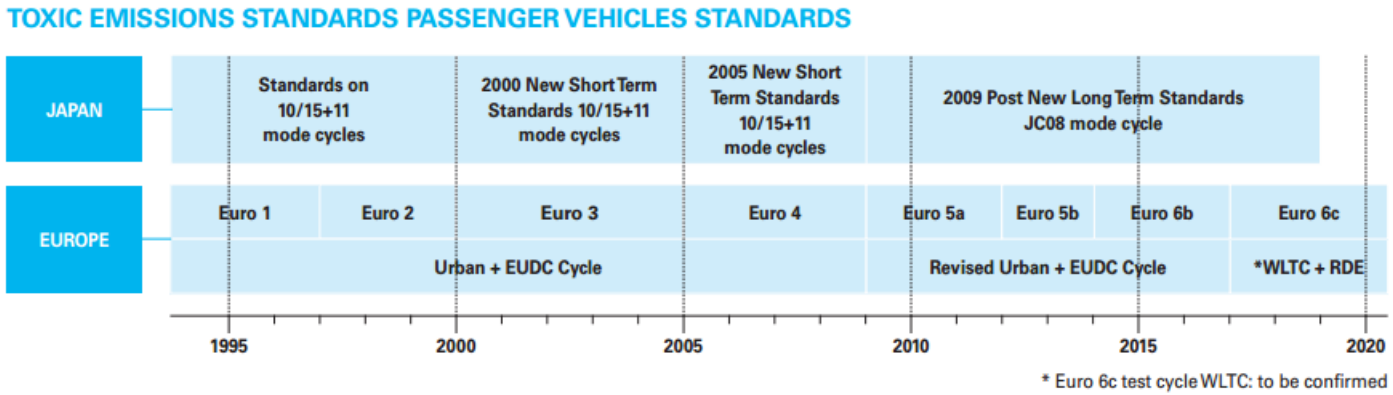


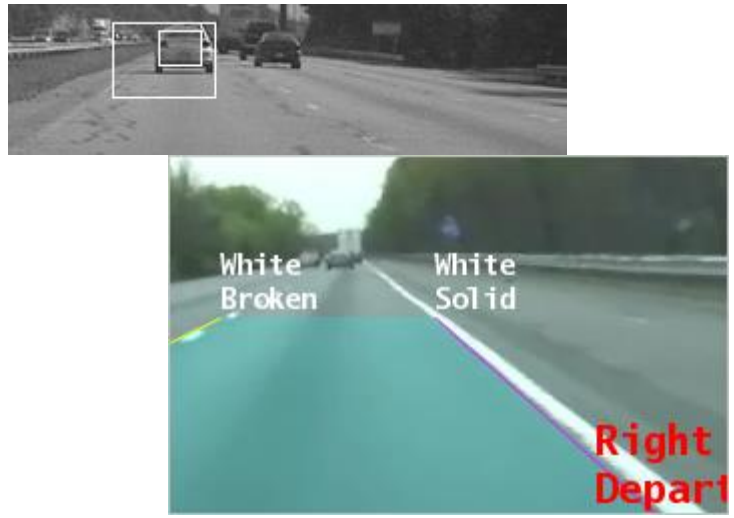




Access and Explore Data → **Preprocess Data** → **Develop Predictive Models** → **Integrate Analytics with Systems**

Problem Statement



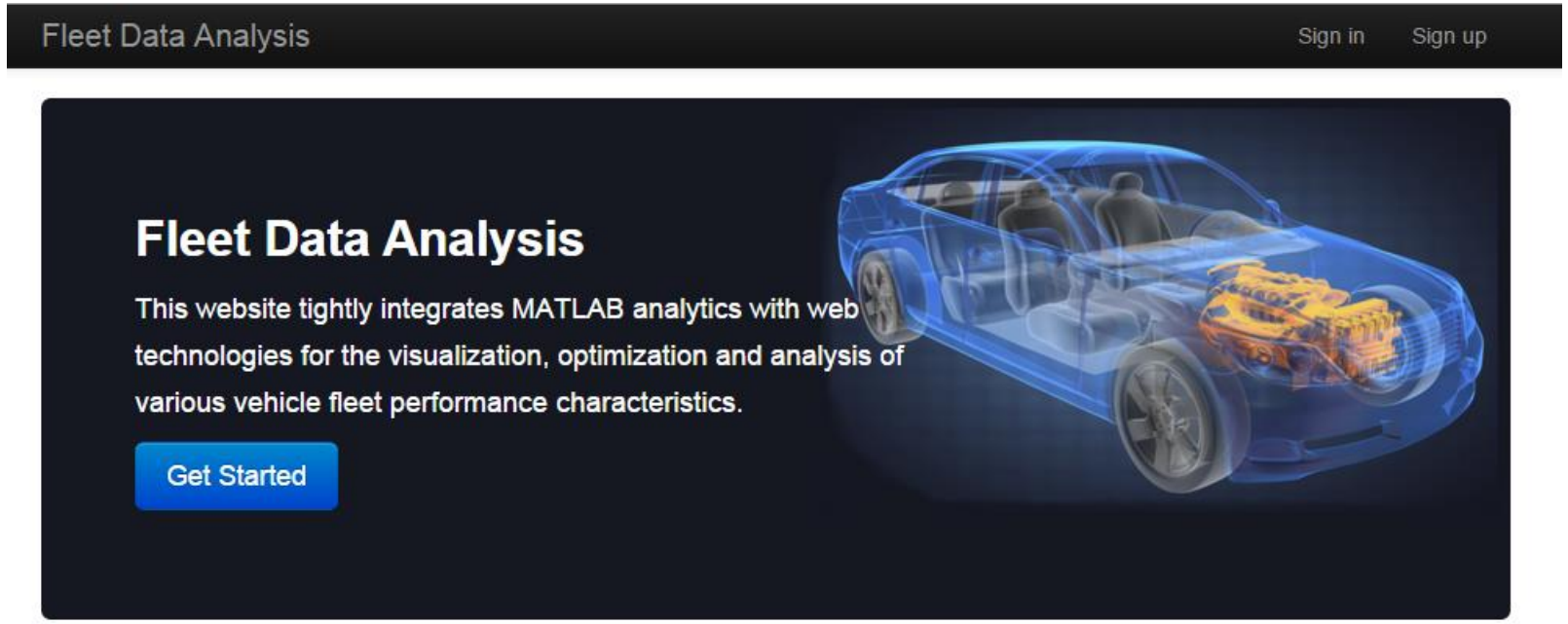


- Can MATLAB® scale up and meet increasingly demanding fleet test data analytics requirements?

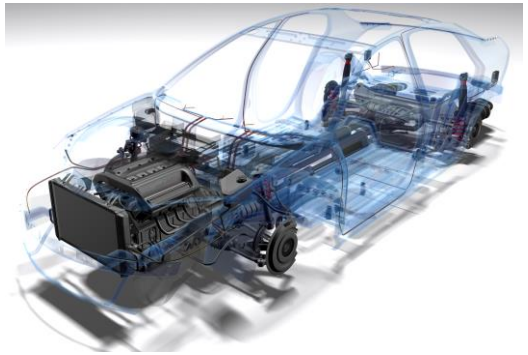
- Is it possible to build data analytics algorithms in a flexible, scalable manner and yet satisfy production requirements?

The Fleet Data Analysis System

http://goo.gl/EeE5JE



A quick recap of our connected cars



OBD2

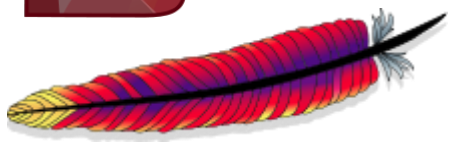
Bluetooth

4G LTE

HTTP

Amazon™ EC2™

LAMR Stack



ubuntu

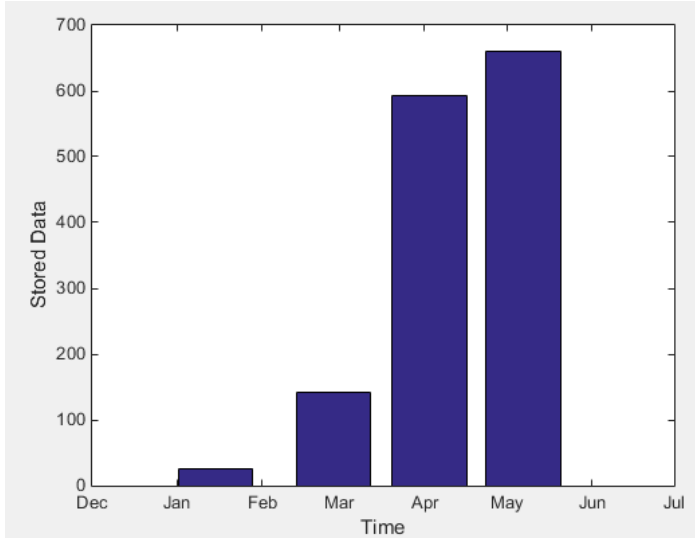
Deployed MATLAB

MATLAB Desktop

Hadoop Ecosystem

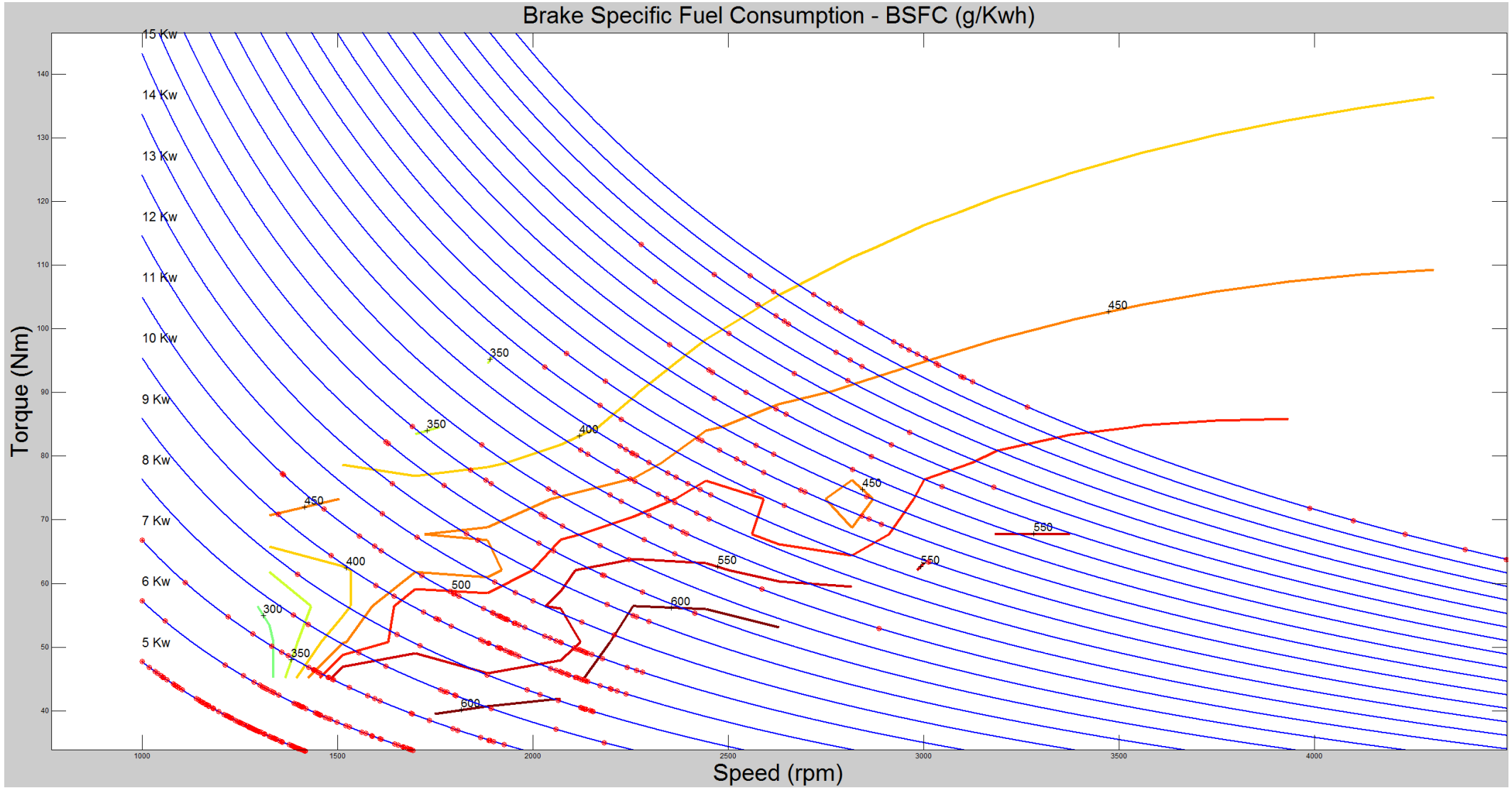
Fleet Data Analysis System (Test Bed Summary)

- 8-25 Mb per day
- Non-telemetric data files
- Up to 25 operators
- 660Mb of data and counting



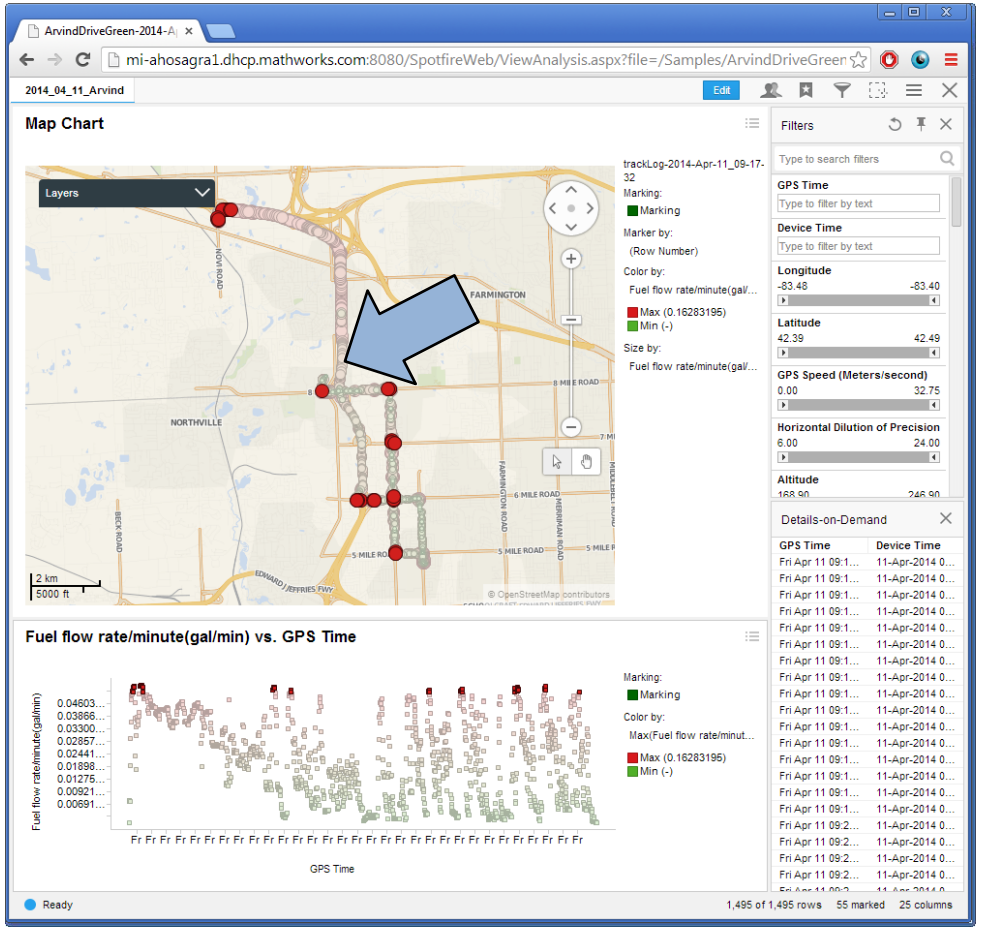
Name	Instance ID	Instance Type	Availability Zone	Instance State
Arvind - Fleet Data Analytics...	i-837e238a	m1.large	us-west-2a	running
FleetWorker	i-7e725a74	m3.large	us-west-2a	running
FleetWorker	i-7f725a75	m3.large	us-west-2a	running
FleetWorker	i-7c725a76	m3.large	us-west-2a	running
FleetWorker	i-7d725a77	m3.large	us-west-2a	running

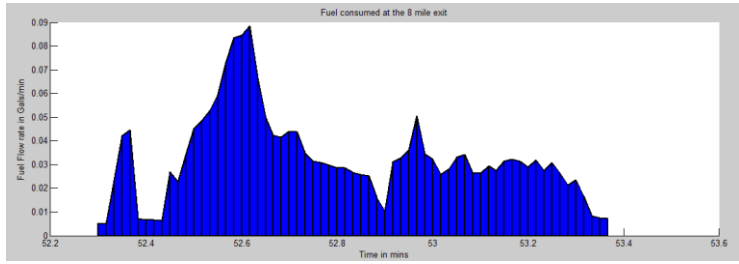
Insights (Engine Fuel Consumption and Efficiency)



Insights (8 mile traffic)

- Traffic Patterns (the case for roundabouts)





- 0.0351 Gal/car at the intersection
- 12 cars a minute on the average

- A saving of 121.3 gallons of gasoline per day if the traffic lights were replaced with a round-about.
- A rough saving of 4.5 million pounds of CO_2 per year.

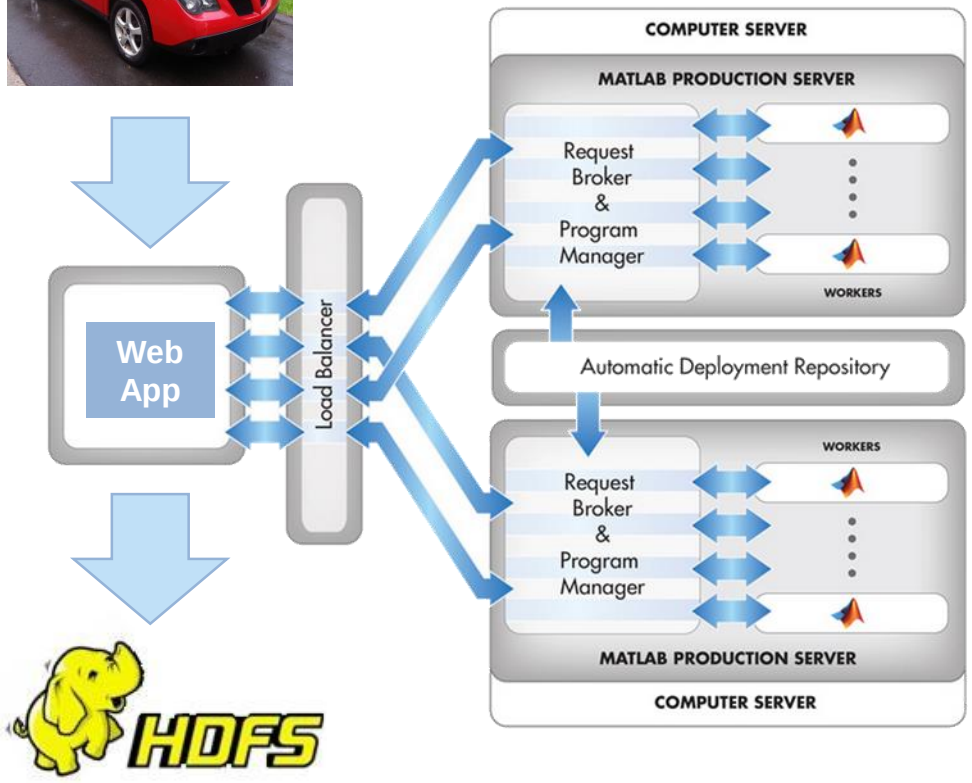
What has changed?

What has improved?

What is new?

Infrastructure Changes

- Enrichment of incoming streams



- Non-telemetric data
- Drag/drop addition and management of data
- MATLAB with visualization tools
- Direct Connectivity with MATLAB
- Caching of data for performance
- Upsized compute resources
- Load balancers
- Simulation sources
  - MATLAB
  - Simulink

Location and Methodology changes

- Web enabled dashboards
- Privacy and Security
- Enabling relational query

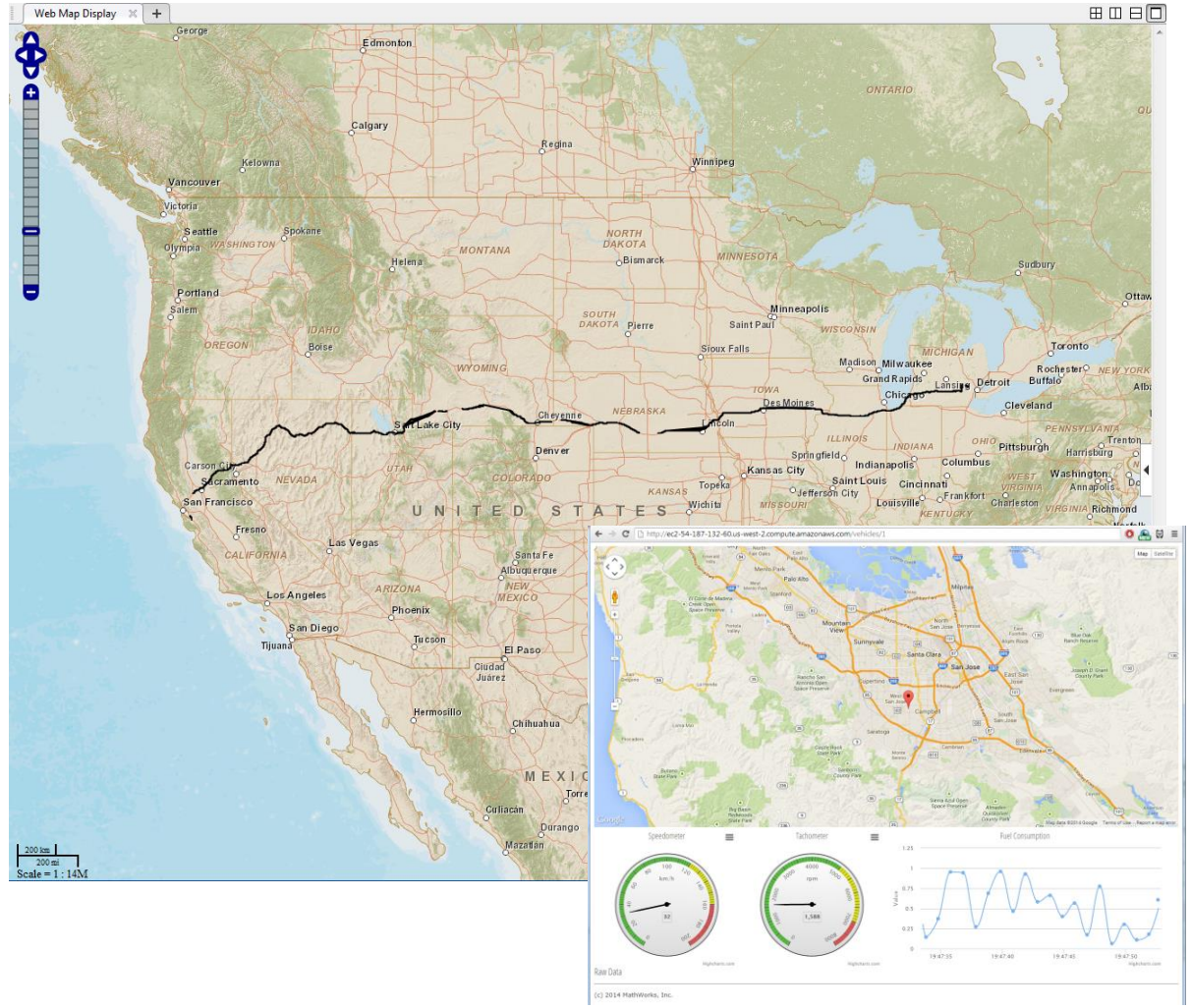
  Cloudera Impala

  Database Toolbox
  Exchange data with relational databases

- **Reference architectures** offer solutions with an emphasis on selecting the right tool for the task resulting in more efficient workflows

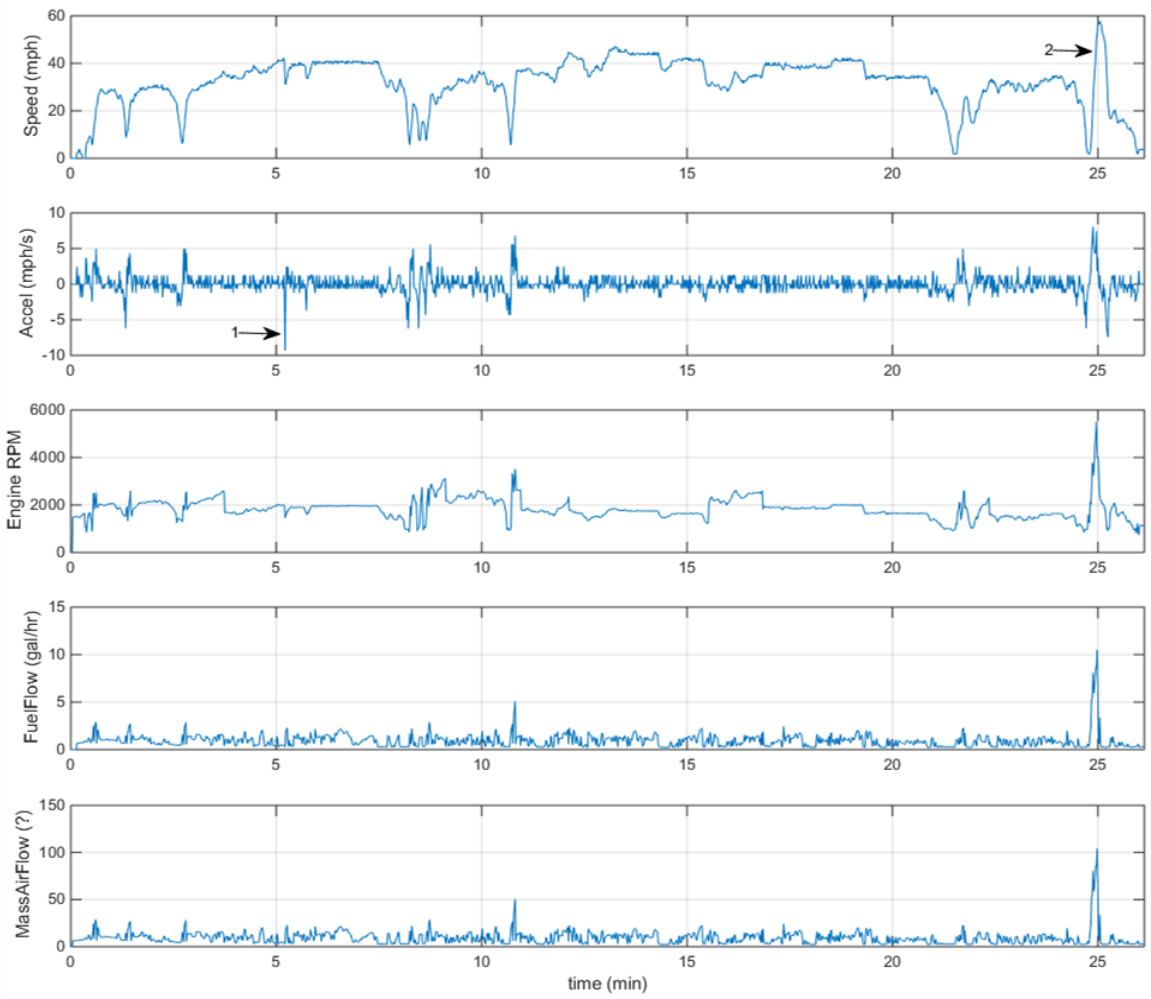
TIBCO Spotfire

BigQuery

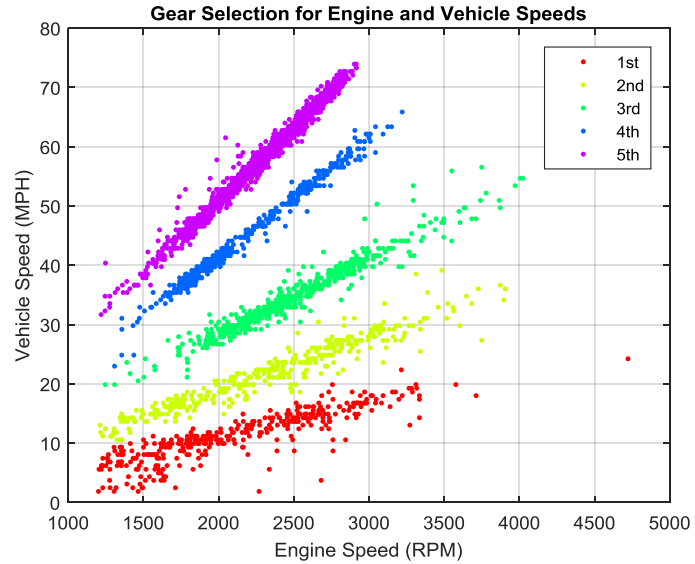


Insights from the field

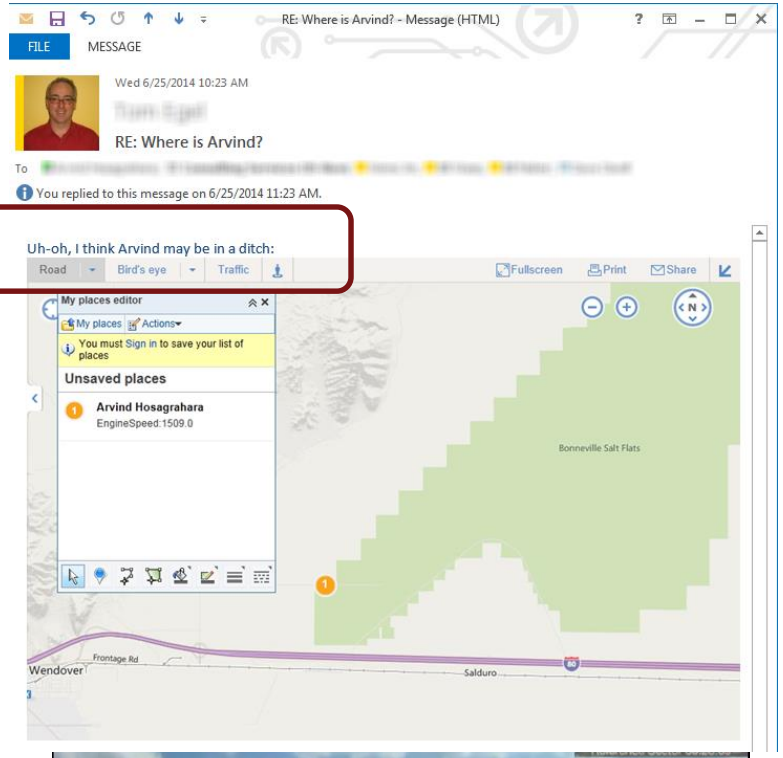
Study of Driving Patterns

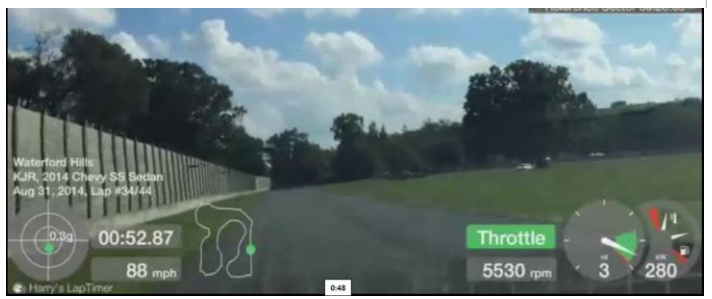


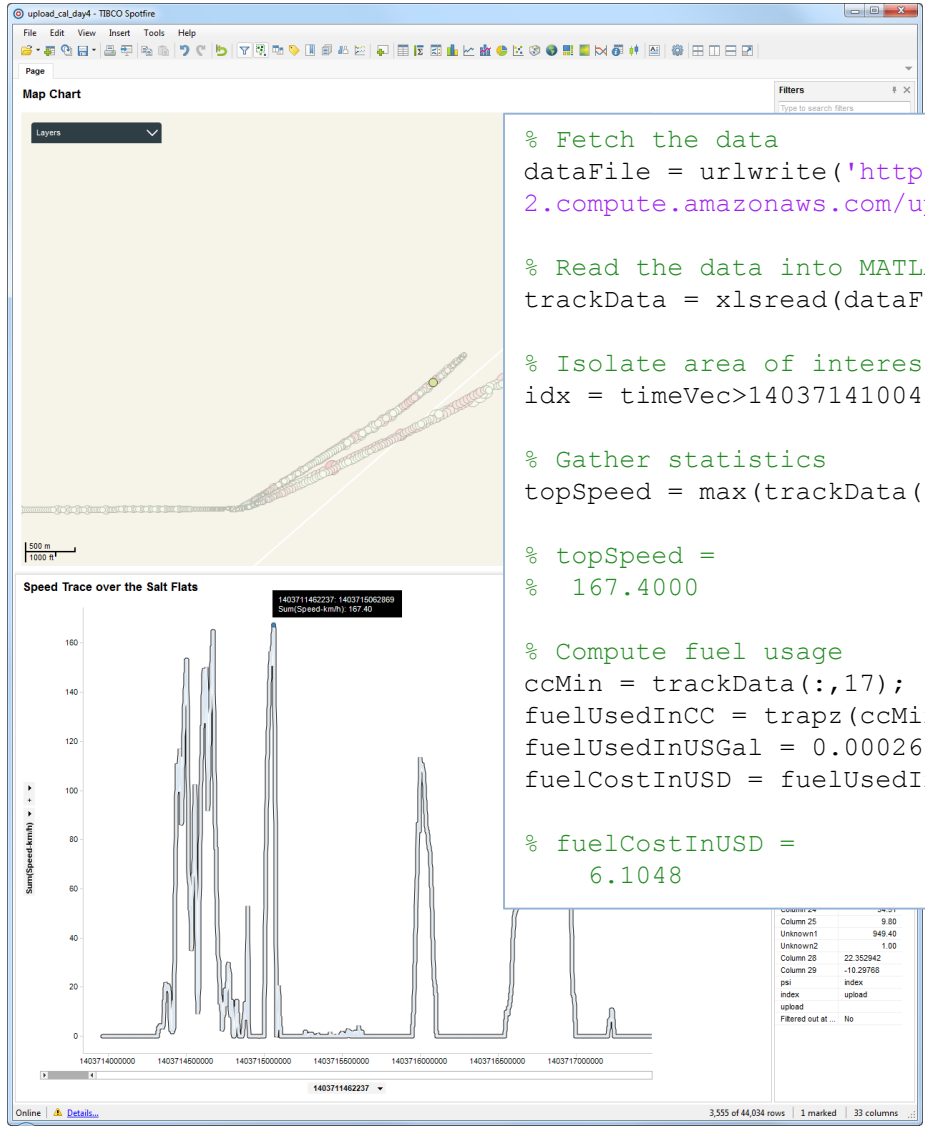
"By the way, this is pretty neat data to look at in MATLAB. Even without the GPS, I can pinpoint (1) the moment I nearly clobbered a deer this morning and (2) the merging onto and off of route 9…"



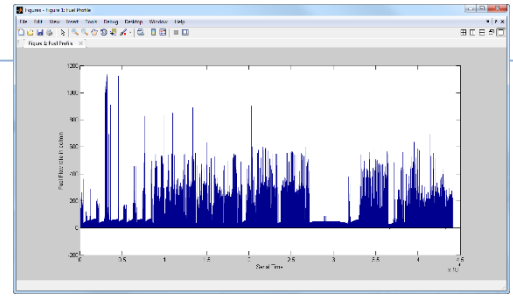
Performance



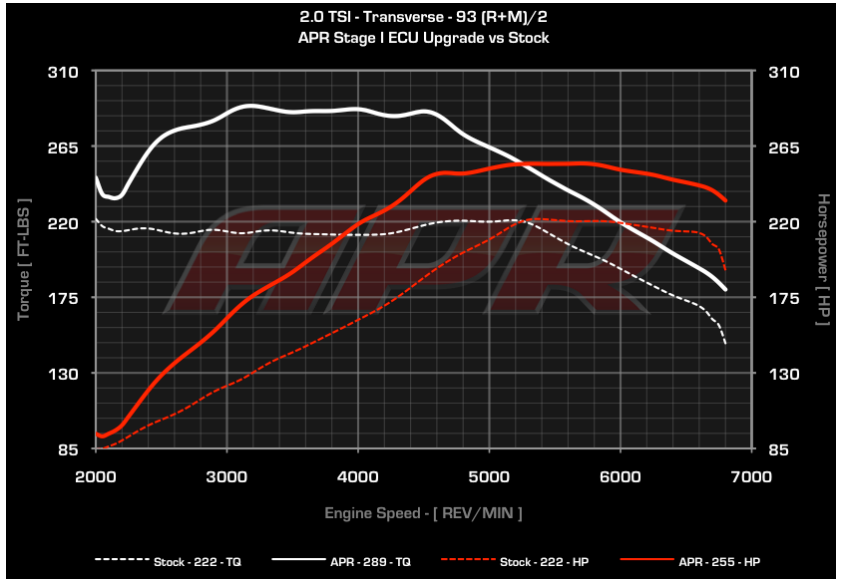





```
% Fetch the data
dataFile = urlwrite('http://ec2-54-187-132-60.us-west-
2.compute.amazonaws.com/upload.csv','trackdata.csv');

% Read the data into MATLAB
trackData = xlsread(dataFile);

% Isolate area of interest
idx = timeVec>1403714100426 & timeVec<1403717061373;

% Gather statistics
topSpeed = max(trackData(idx,7)); % in kmph.

% topSpeed =
%  167.4000

% Compute fuel usage
ccMin = trackData(:,17);
fuelUsedInCC = trapz(ccMin(idx))/(60);
fuelUsedInUSGal = 0.000264172052*fuelUsedInCC;
fuelCostInUSD = fuelUsedInUSGal*3.91;

% fuelCostInUSD =
    6.1048
```



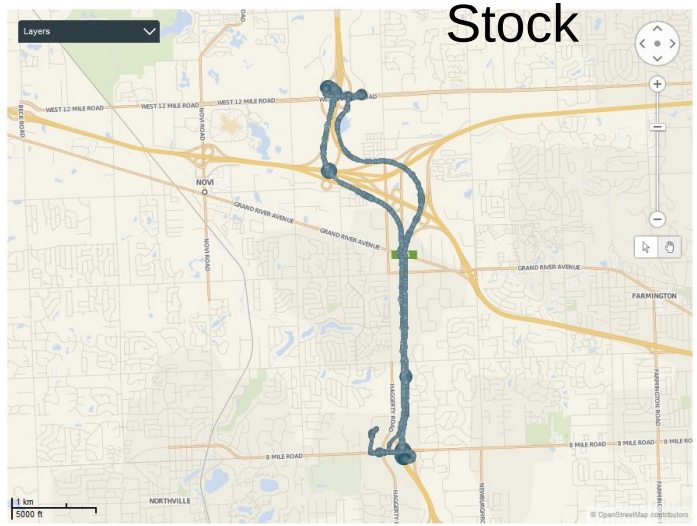
Discovering problems in veracity of collected data

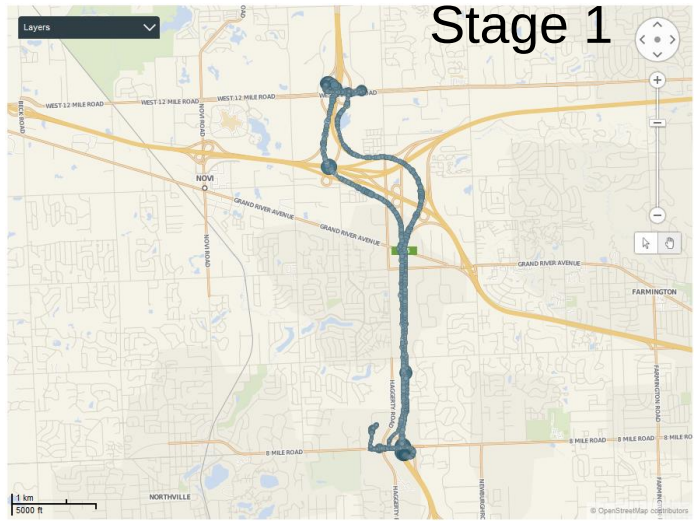


APR Stage I & II+ ECU Upgrade Calibration Report

	Stock *As reported by VAG*	Stock *As measured by APR*	Stage I 93 Octane (R+M)/2 or 98 Octane RON	Stage I *with APR Intake* 93 Octane (R+M)/2 or 98 Octane RON	Stage II *with APR Intake & Exhaust* 93 Octane (R+M)/2 or 98 Octane RON
Speed Limit	Limited	Limited	Unlimited	Unlimited	Unlimited
Peak Horsepower (HP)	200 HP	222 HP	255 HP	269 HP	273 HP
Peak Torque (FT-LBS)	207 TQ	222 TQ	289 TQ	295 TQ	300 TQ
Max HP Increase over Stock*			+58 HP @ 4,450 RPM	+66 HP @ 4,200 RPM	+71 HP @ 4,250 RPM
Max FT-LBS of Torque Increase over Stock*			+76 TQ @ 3,150 RPM	+83 TQ @ 4,200 RPM	+87 TQ @ 4,200 RPM

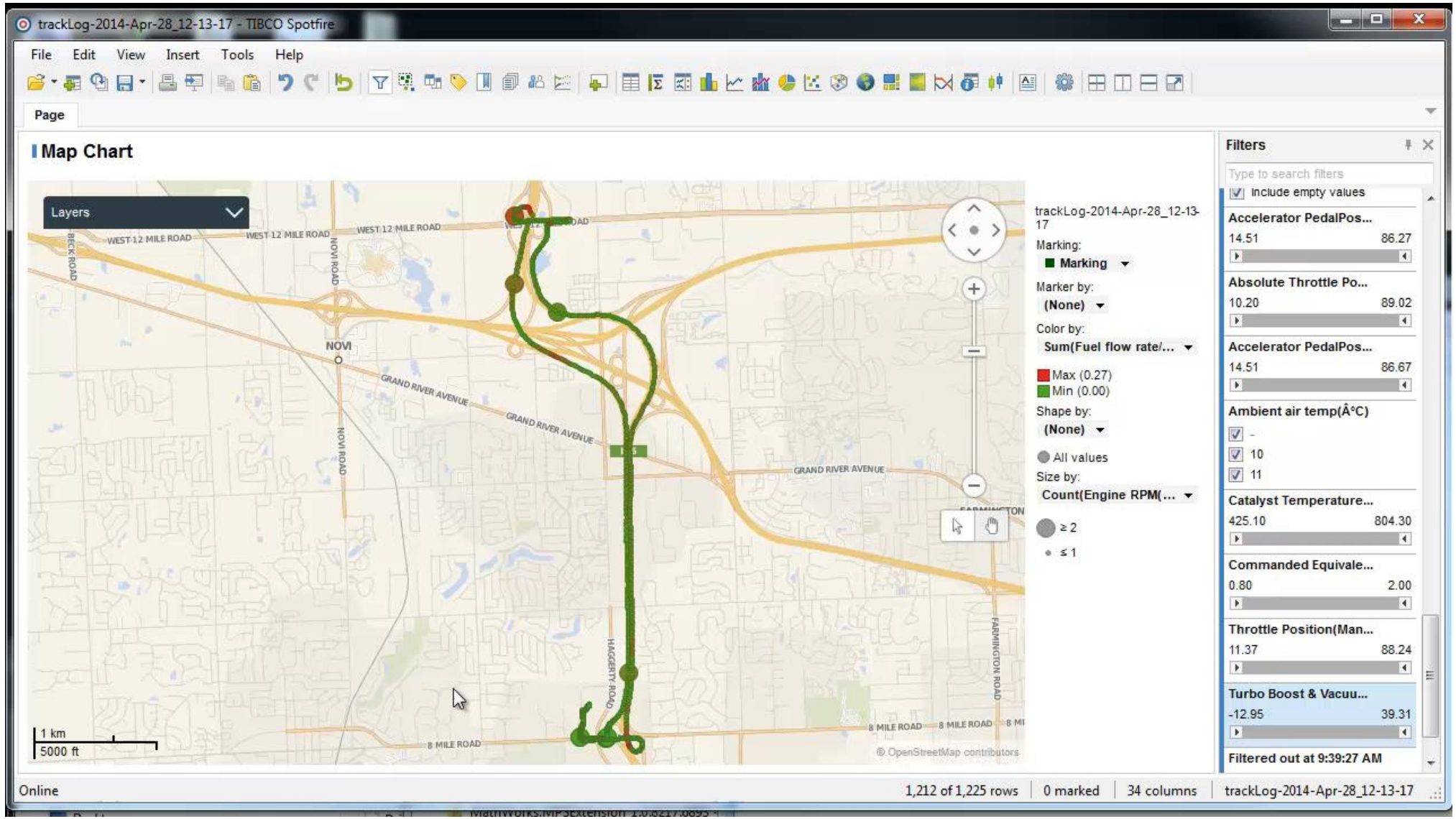
*Max increases are based on APR's actual measured stock values on a 2.0 TSI CCTA Engine and not those reported by VAG.





- Software modifications to give higher peak ft-lbs of torque
- Increased boost pressure and optimized ignition timing
- Aftermarket software was conditioning the data to ensure proper operation

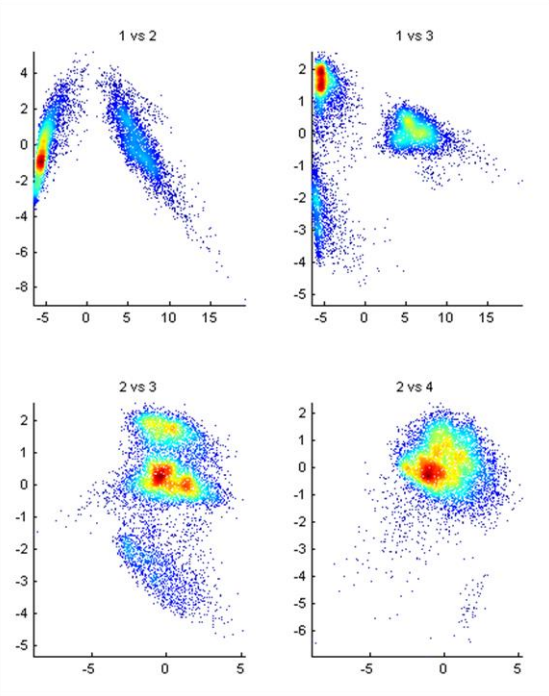
Operationalization of Analytics



Product Changes (1) - Big Data Capabilities in MATLAB

Memory and Data Access

- 64-bit processors
- Memory Mapped Variables
- Disk Variables
- Databases
- **Datastores** R2014b



Programming Constructs

- Streaming
- Block Processing
- Parallel-for loops
- GPU Arrays
- SPMD and Distributed Arrays
- **MapReduce** R2014b

Platforms

- Desktop (Multicore, GPU)
- Clusters
- Cloud Computing (MDCS on EC2)
- **Hadoop** R2014b

Product Changes (2) – Datastores and MapReduce

Datastore

- Incremental data processing feature for reading collections of data containing *tabulartext* or *keyvalue* pairs.

```
Example: 'file1.csv'
Example: '../dir/data/file1.csv'
Example: {'C:/dir/data/file1.csv','C:/dir/data/file2.csv'}
Example: 'C:/dir/data/*.mat'
Example: 'hdfs://myserver:7867/data/file1.txt'
```

- Enables access to relational databases using Database Toolbox™

MATLAB® MapReduce

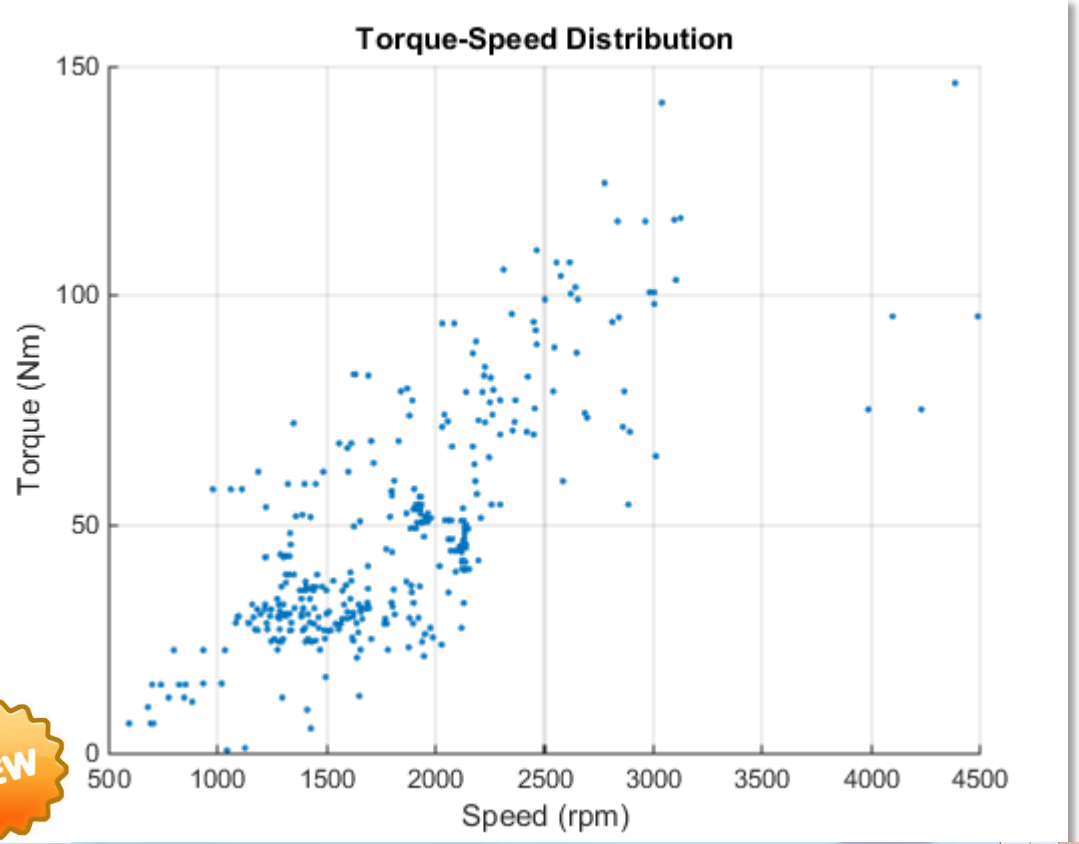
- Allows analysis of out-of-memory data.
- Deployment of mapreduce algorithms to:
  - Serial Mapreduce using local workers
  - Parallel Computing Toolbox™
  - MATLAB® Distributed Computing Server™
  - Hadoop® using the MATLAB Compiler

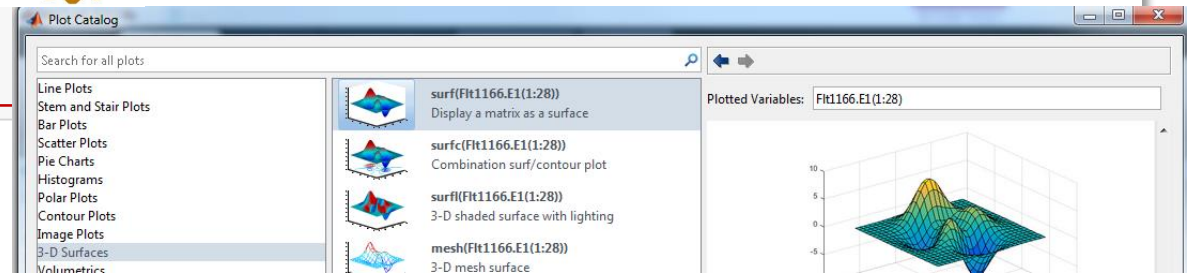
Out of memory processing using Datastores

Visualize Torque Speed Data from Morning Commute

This script will visualize the torque speed scatter using the datastore.

```
% Auth/Revision:  Arvind Hosagrahara
%                 Copyright 2014 The MathWorks Consulting Group
%                 $Id$

% Load our logged data
fleetDS = datastore('tracklog*.csv');
%fleetDS = datastore('hdfs://ec2-54-186-203-200.us-west-2.compute.amazonaws.com:7867/data/OBD2/Aztek/track

% Select columns of interest and chunk the data
fleetDS.SelectedVariableNames = {'Torque_Nm_','EngineRPM_rpm_','FuelFlowRate_minute_gal_min_'};
fleetDS.RowsPerRead = 1000;

% Preview the data to inspect for correctness
preview(fleetDS);

% Read the data and prep for plotting
plotData = read(fleetDS);
plotArray = table2array(plotData);
plotArray=strrep(plotArray,'-','NaN');

% Visualize the data
scatter(cellfun(@str2num,plotArray(:,2)),cellfun(@str2num,plotArray(:,1)),'.');
xlabel('Speed (rpm)');
ylabel('Torque (Nm)');
title('Torque-Speed Distribution');
grid on;
```





Studying ride quality using MapReduce

```
function SimpleMapReduceDemo(varargin)

% Auth/Revision:  Arvind Hosagrahara
%                 Copyright 2014 The MathWorks Consulting Group
%                 $Id$

% Load our logged data
fleetDS = datastore('trackLog-2014-Apr-11_09-17-32.csv');
%fleetDS = datastore('hdfs://ec2-54-186-203-200.us-we

% Select columns of interest and chunk the data
fleetDS.SelectedVariableNames = {'G_x_','G_y_','G_z_'
fleetDS.RowsPerRead = 1000;

% Run the MapReduce on serial MATLAB
rideDS = mapreduce(fleetDS, @RideMapFun, @RideRe
rideTable = readall(rideDS) %#ok<NASGU>
end
```

Mapper Function

```
function RideMapFun(data, ~, intermKVStore)
        % Assemble the ride data and compute ride metric based on G sensors
        rideData = [data.G_x_ data.G_y_ data.G_z_];
        rideLenValue = [sum(rms(rideData,2)), size(rideData,1)];

        % Store the key value for the reducer
        add(intermKVStore, 'rideG', rideLenValue);
```

tion

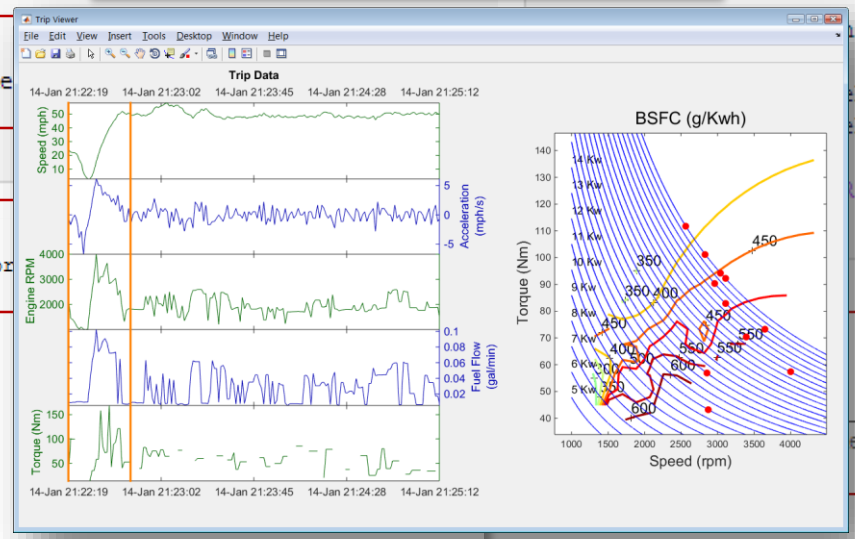
```
(~, rideLenIter, outKVStore)
0];
LenIter)
Len + getnext(rideLenIter);

ide G Sensor Mean', rideLen(1)/rideLen(2));
```

```
Trial>> mapreducer(parpool)
Starting parallel pool (parpool) using the 'local' pr
Trial>> SimpleMapReduceDemo
Parallel mapreduce execution on the local cluster:
Map Stage 0% complete
Map Stage 100% complete
Reduce Stage 0% complete
Reduce Stage 100% complete
rideTable =
            Key                 Value
    'Ride G Sensor Mean'    [5.7838]
Trial>>
```

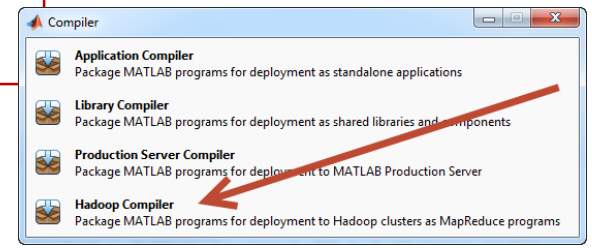
Analyzing Fleet Test Data Using MATLAB

4:00–5:00 p.m.

Seth DeLand, MathWorks



Key	Value
'Ride G Sensor Mean'	[5.7838]



Conclusion

- Single stack, open and extensible analytics toolset that plays well with other technologies
- New features support clean workflows for use in production systems and with Big Data
- A capacity for complexity