Industrial Big Data Analytics for Wind Turbines

UseR! 2015 Aalborg
Sven Jesper Knudsen - sjknu@vestas.com
Kim-Emil Andersen & Martin Qvist
Vestas Performance & Diagnostic Centre (VPDC)

Total Vestas installed base: 

+60 GW

Vestas Installed base:

+60 GW*

Worldwide installed wind capacity:

+318 GW**

*Vestas Performance & Diagnostic Centre, 31/12/13
**Global Wind Statistics 2013 (Year end) GWECC

Total monitored of Vestas installed base:

47 GW*

26,000+ turbines are monitored

55,000 date

Vestas Wind Turbine Data Science

- A Decade of data driven Asset Management
- Predictive Maintenance
- Condition Based Monitoring

Lost Production Factor (LPF)

Date

Wind. It means the world to us.”
Vestas UseR!

Desktop

RStudio is on our Vestas Software portal
Markdown reports to clients
TIBCO Spotfire + R Apps users
SQL, Py, .NET

Vestas’ HPC, FireStorm

15000 cores, 2.5+ petabyte, 40Gb infiniband interconnected
mesoscale climate model, petabyte storage
Computational Fluid Dynamics, high performance

Today focus on HADOOP+HIVE+R

Open Source and Open Standards
CFD Wind Flow
Meso Scale Thunder Storm
Industrial Big Data at Vestas

Volume  
50,000 turbines, ca 150 terabyte  
Vestas Climate Lib, ca petabyte

Variety  
500+ sensors, vibration ‘audio’, event logs  
image, lab test, service text

Velocity  
Real time, near time, batch

Varacity  
Sensitive and diverse sensors, parameters, humans …

Wind. It means the world to us.”
Data Platform

Vestas Apps

Internal App

Vestas Data Centre
RDBMS

Data Warehouse  Application DB

Vestas Performance & Diagnostic Centre

Data Science

MS SQL Server: 150tb SAN
+ It works
+ Business penetration
+ Transactional & relational

- Costly
- Staging areas, re computation
- Where is R

Wind Power Plant

Machines & Sensors
Meteorological Spatial layout
Laboratory
Unstructured Logs Reports

Wind. It means the world to us.
Value Driver

Time is Money – Downtime is Expensive
Downtime Lost Wind Energy Production

Corrective

Damage

Production loss

Inspect damage Order spare parts Repair work

Predictive

Pr(Damage) > threshold

Inspect component Order spare parts Repair work
Predict generator bearing failure

- Training data:
  Vibration 10min values, several parks filter down to turbine in full operation

- Changepoint algorithm on raw or residuals of a machine learned model
- Tune using confusion matrix: best precession, accuracy, sensitivity False Alert vs Wasted Service Visit

package: changepoint, caret
Vestas Turbine Monitor
Full Business Support for Process & Information Flow

Vestas Turbine Monitor
Automatic Notification To Service Before Failure

Work Order (ERP)

Monitor Act Plan

Mitigation & work instructions

Plan Service using Vestas Forecasting™ to Minimize Production Losses Maximize Safety

Vestas Global Advisor

Improve Mitigation Action, Monitor Accuracy

Wind. It means the world to us.”
Measurable Business

Lost Production Factor (LPF)

Date


0% 1% 2% 3% 4% 5% 6%
**HIVE – Hadoop Interactive**

**Hadoop** allows distributed processing of **large data sets across clusters**
Called Google's data model (Map Reduce)

**HIVE provides SQL** like queries over **Hadoop**

```sql
Select
turbineNumber,
    avg(WindSpeed)
from Turbine10minData
group by turbineNumber
```

Faster than indexed MS SQL Server
Much faster if grouped by non indexed column, like temperature bin
HADOOP Map Reduce (MR)

Map()  
- Distributed Data  

Shuffle  
- Are collected and sorted  

Reduce()  
- Distributed Compute  

http://blog.sqlauthority.com

Wind. It means the world to us.™
HIVE Custom Map Reduce - inject R, Py, Octave...
Inject script s using HIVE Transform

FROM (  
   SELECT * FROM TurbineData  
   DISTRIBUT BY turbineId
) as map

SELECT  
   TRANSFORM ( * ) USING 'wrapper.sh someAlgorithm.R'  
   as (turbineId,ttimestamp, propbilityOfFailure);

Map step: DISTRIBUT data to compute nodes BY tubine

HIVE: Inject script based computations

FROM (  
    SELECT * FROM TurbineData  
    DISTRIBUTED BY turbineId  
) as map  
SELECT  
    TRANSFORM ( * ) USING 'wrapper.sh someAlgorithm.R'  
as (turbineId,timestamp,propbilityOfFailure);

wrappe.sh loads R and !Rscript  
Data is piped to R using stdin  
Data piped back using stdout

Hive gives total control to R – please mind  
• Compute speed  
• Memory consumption  
• Error handling

Useful HIVE resource settings:  
set mapred.reduce.tasks=200;
set mapreduce.reduce.memory.mb=10240;
#!/usr/bin/env Rscript

.libPaths('/gpfs01/R/packages/bmk/v1.0.0')
suppressMessages( library(data.table) )
suppressMessages( library(fasttime) )
suppressMessages( library(vbmk) )

outDt <- tryCatch( {

  # fast read of piped stdin
  inDt <- fread('file://dev/stdin', sep ="\t", showProgress=FALSE)

  # fast convert to POSIXct
  inDt <- inDt[,timestamp:=fastPOSIXct(ttimestamp, tz = "UTC")]

  # do your thing
  bmkProcess(inDt)

},
  error = bmkErrorHandler )

if (!is.null(odata)) {
  write.table(odata, quote=FALSE, sep="\t",
              col.names=FALSE, na = " ", row.names=FALSE)
}

Summary

HIVE for large batch / interactive jobs
  Proven more reliable with version 0.14
  Spark is lurking for interactive analysis

Familiar to users and cost efficient – no brainer
  Need to understand how to MapReduce the problem

Need to manage your Resources
  Fast readers (fread) make a difference
  tweek HIVE resources settings to get job completion

ValueBrings more data to table
References

Vestas and Big Data

R:
http://www.r-project.org/
http://www.revolutionanalytics.com/what-r

Python:
https://www.python.org/

Apache HADOOP
http://hadoop.apache.org/

Apache HIVE
https://hive.apache.org/

Apache Spark
https://spark.apache.org/
Thank you for your attention
Hive setup

- Distribution
  - Hortonworks
  - HDP 2.2
  - Hive 0.14
  - 20 nodes

- Traditional setup runs 1/10GbE
- Disks on each node, HDFS running on top

- Our setup runs on Vestas' HPC
- Has its own parallel filesystem (GPFS)
- Fast (40Gb) Infiniband interconnect
- Very high performance with few nodes
- Very high utilization per node