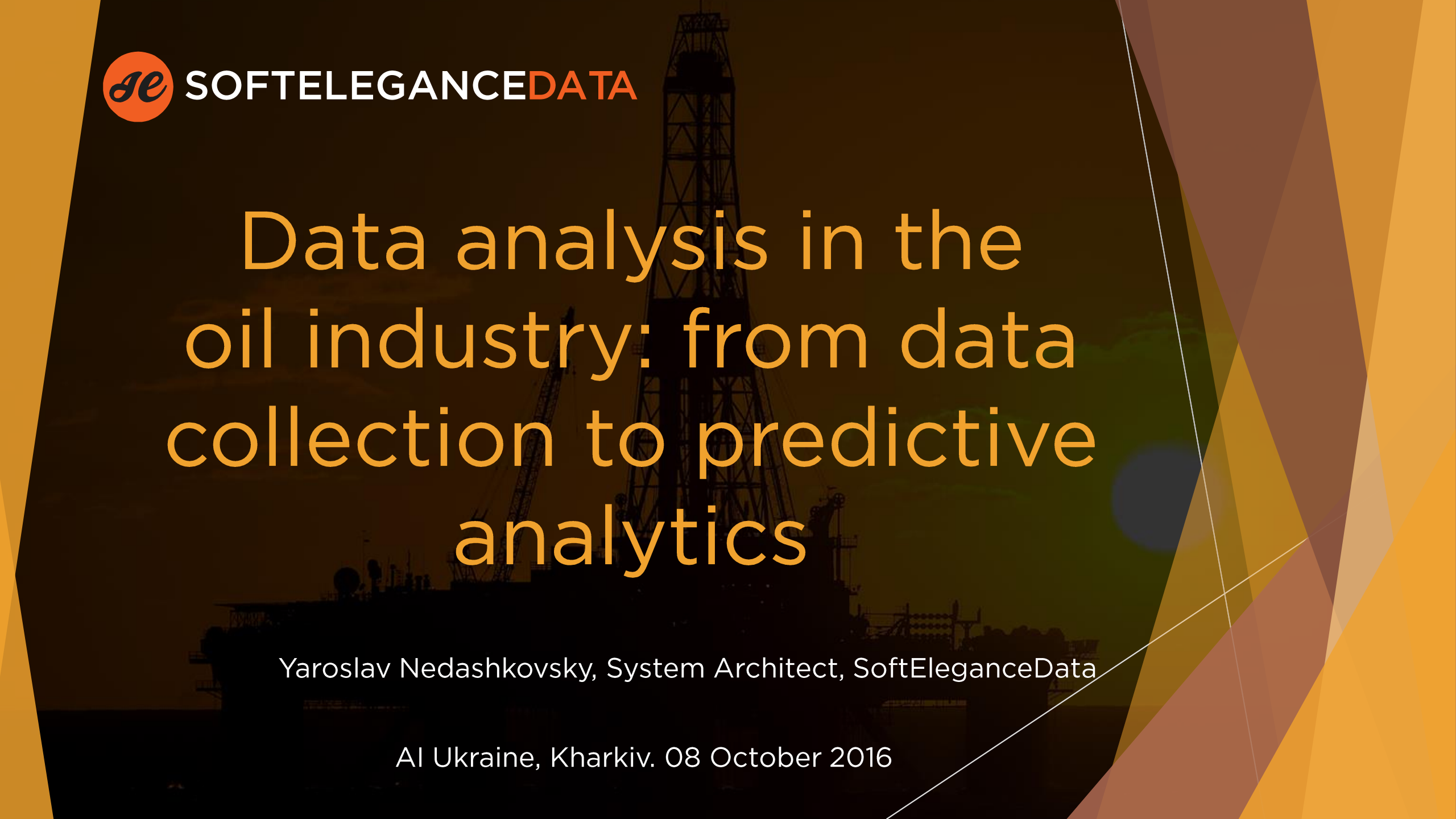


Data analysis in the oil industry: from data collection to predictive analytics

Yaroslav Nedashkovsky, System Architect, SoftEleganceData

AI Ukraine, Kharkiv. 08 October 2016

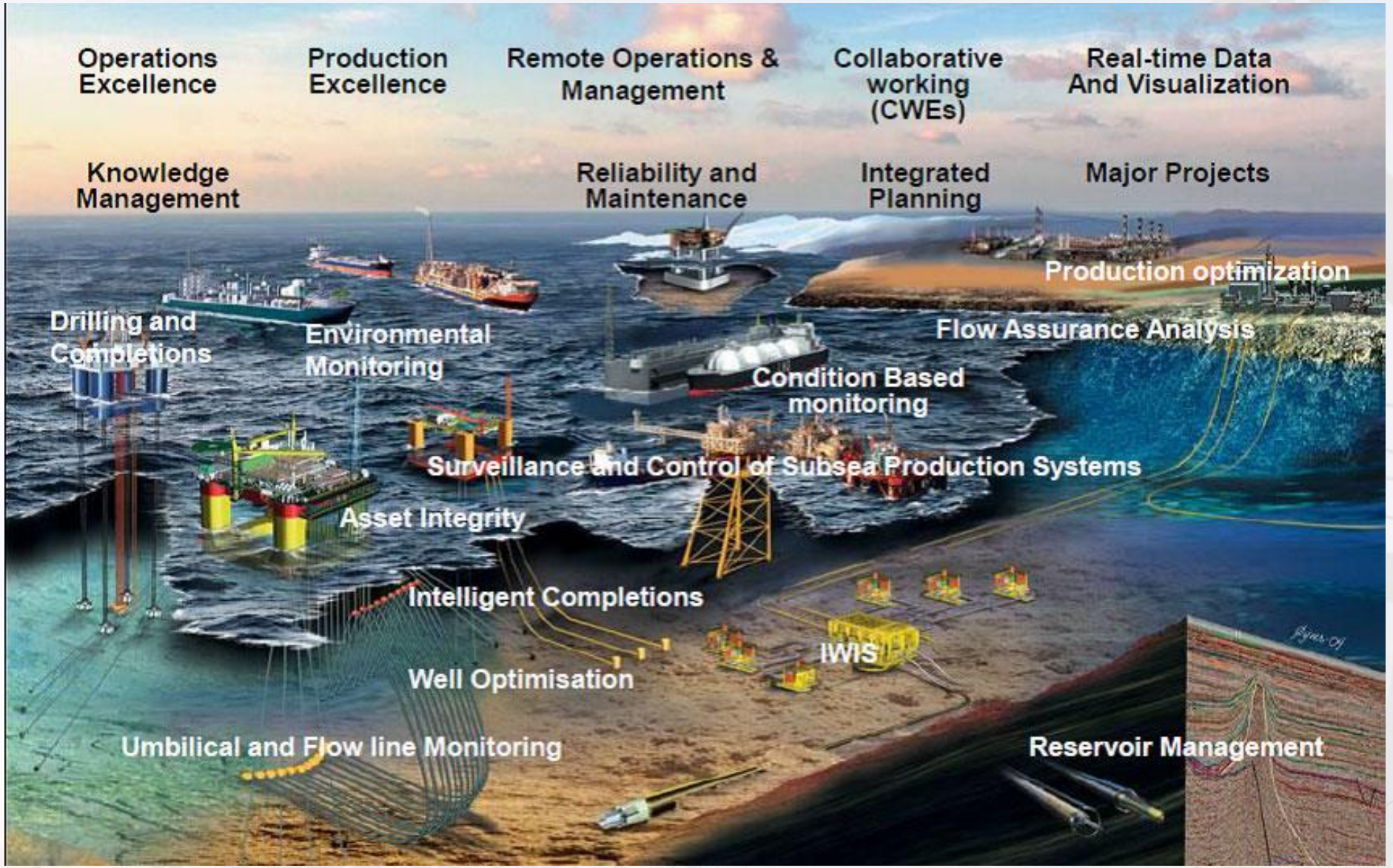
1. Oil industry overview

2. Data Lake

3. Data collection

4. Data analysis

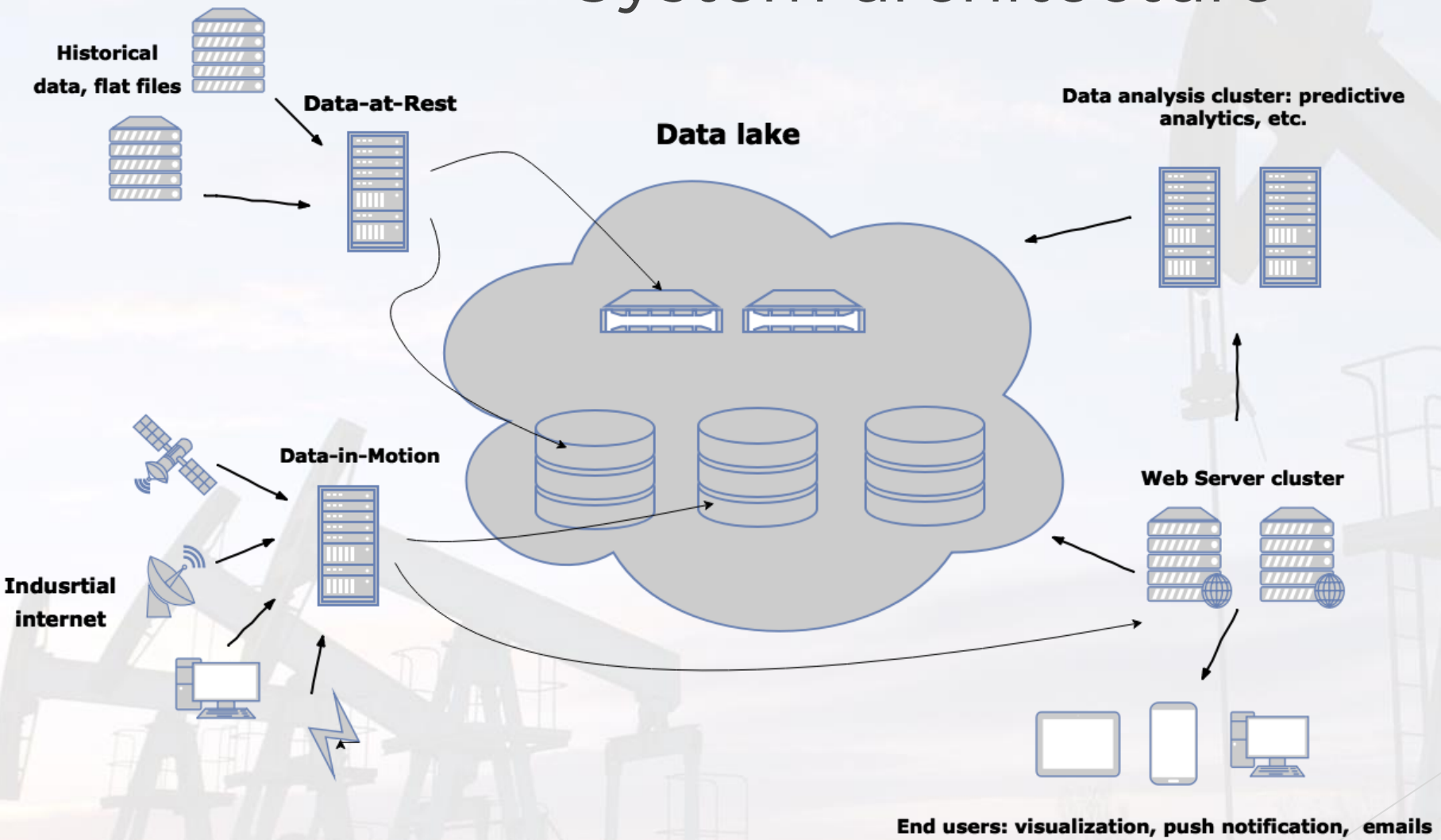
1. Oil industry overview



data.softelegance.com

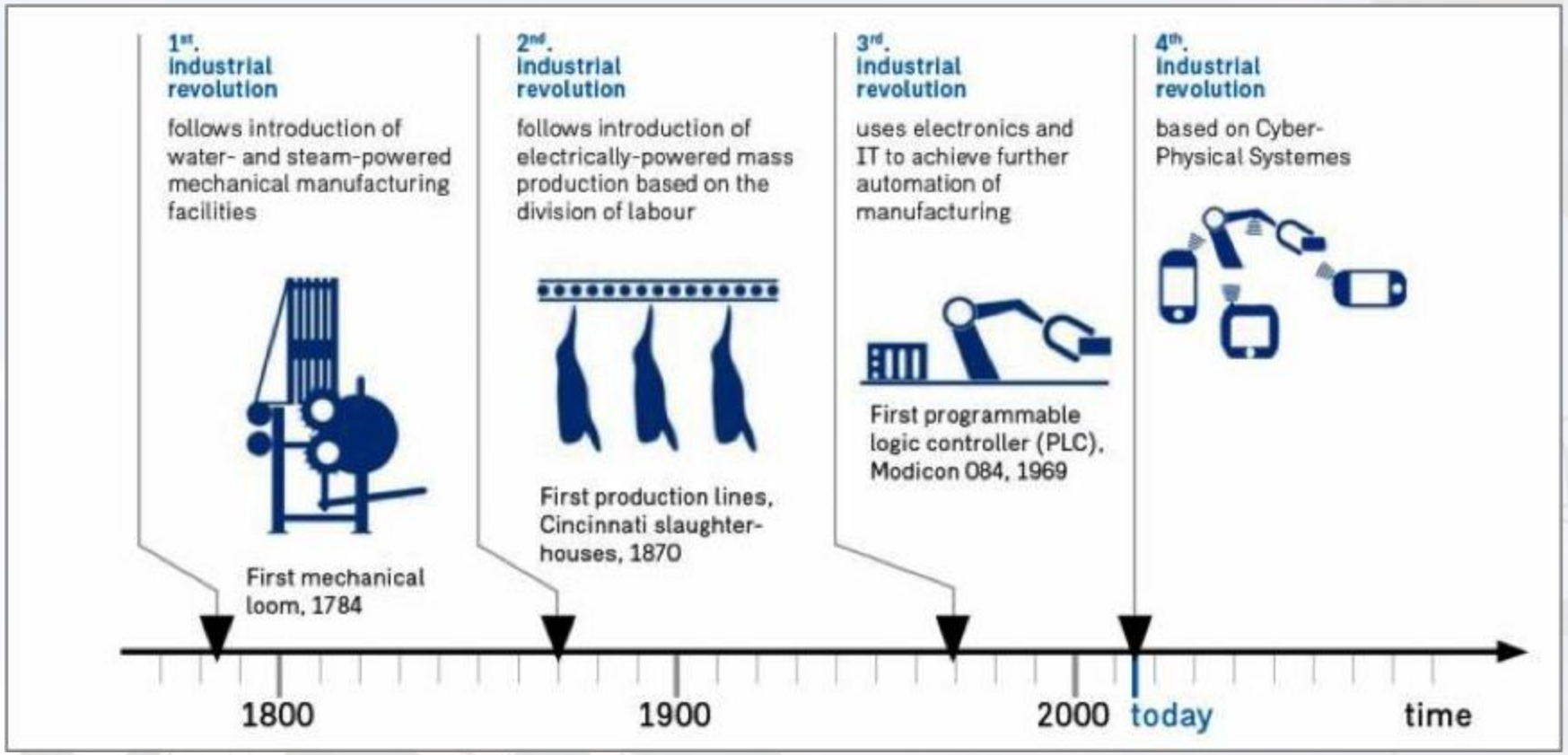
2. Data Lake

System architecture



Data flow

Industry 4.0, Industrial Internet

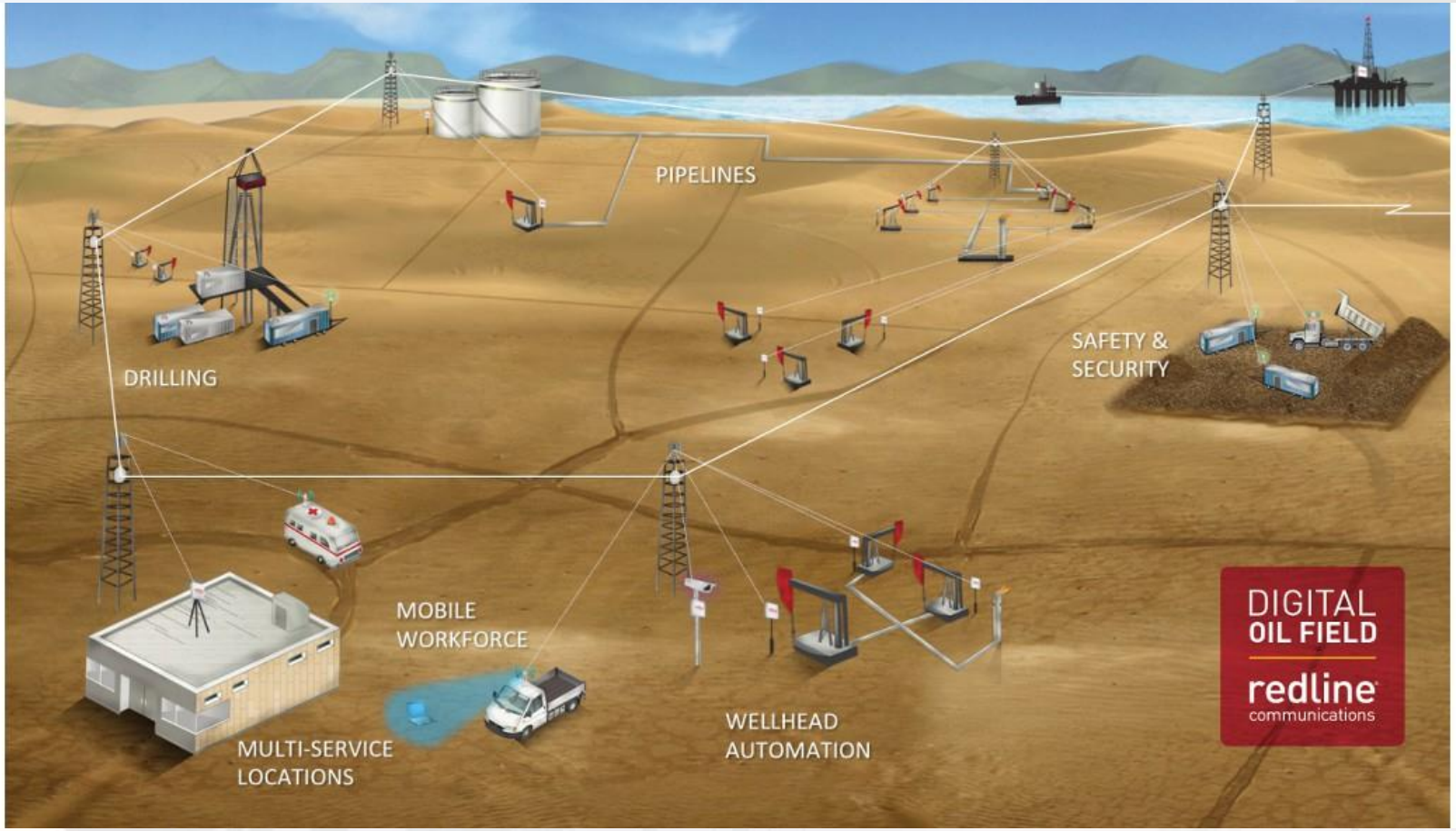


"Last year, British Petroleum connected 650 wells to the Industrial Internet. If all goes according to plan, the companies will expand the scope to 4,000 British Petroleum subsea wells around the world"

data.softelegance.com

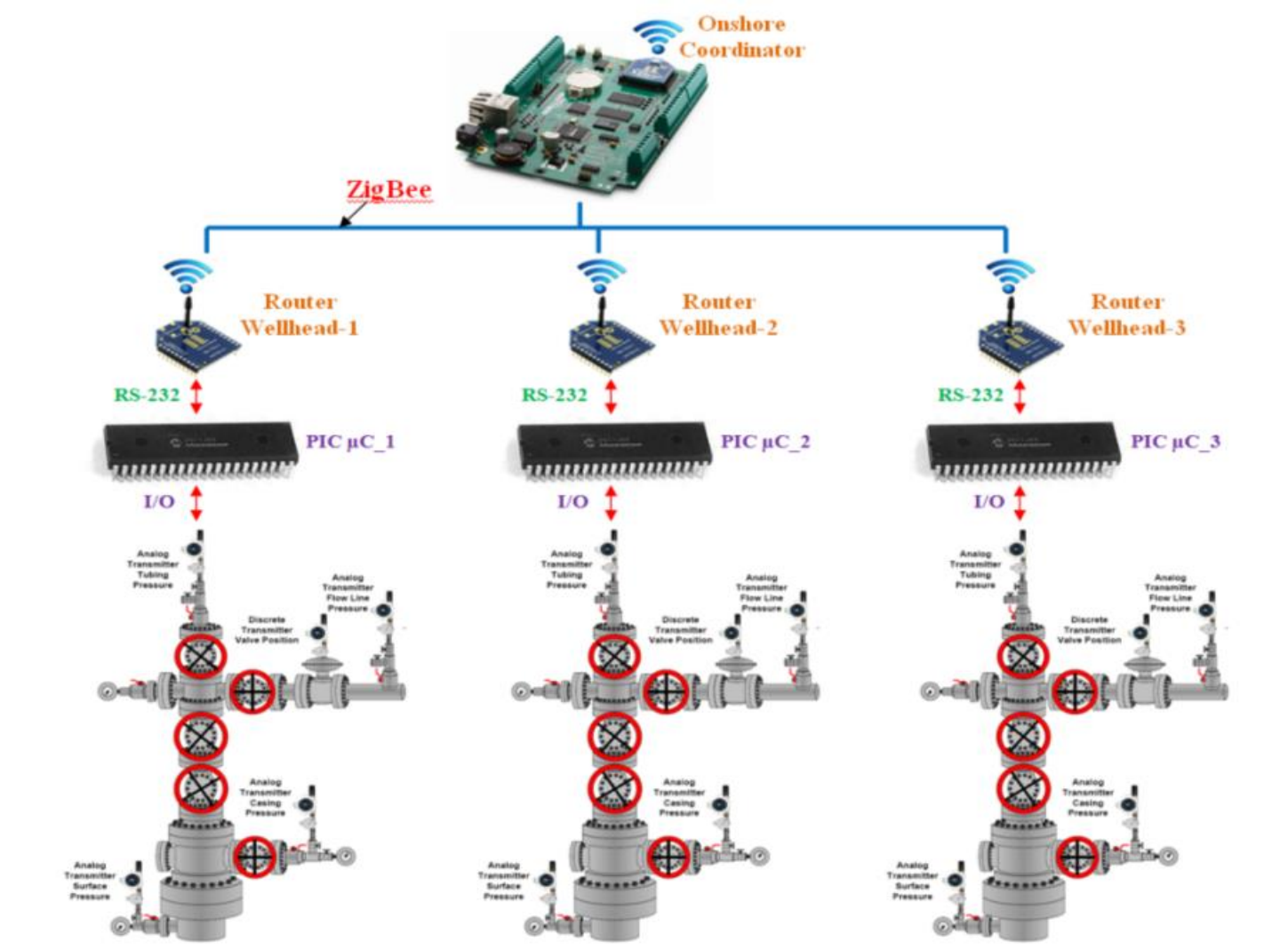
3. Data collection

Digital oil field



data.softelegance.com

Smart well



Source: http://www.ethanpublishing.com/uploadfile/2015/0608/20150608015338765.pdf

Data sources

- sensors readings
- flat files (for example las* file)
- legacy dataset

Majority part of injected information converted to internal formats furthermore, part of it, should be left as is (las)

Data storage

- cassandra
- distributed file system

4. Data analysis

Add-hoc query with Spark Notebook

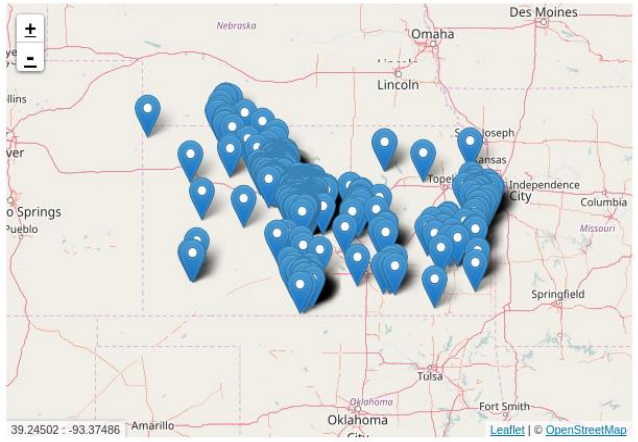
SPARK NOTEBOOK visualization (unsaved changes)

File Edit View Insert Cell Kernel Help | Scala [2.11.8] Spark [2.0.0] Hadoop [2.2.0] {Hive ✓}

Code | Cell Toolbar: None

```
import org.apache.spark._
import org.apache.spark.rdd._

import org.apache.spark.mllib.classification.{LogisticRegressionWithLBFGS, LogisticRegressionModel}
import org.apache.spark.mllib.evaluation.MulticlassMetrics
import org.apache.spark.mllib.regression.LabeledPoint
import org.apache.spark.mllib.linalg.Vectors
import org.apache.spark.mllib.util.MLUtils

val spark = SparkSession
  .builder()
  .appName("Wells visualization")
  .getOrCreate()

val rawDS = spark.read.
                  format("com.databricks.spark.csv").
                  option("header", "true").
                  load("/datalake/ks_wells.csv").cache()


val fDS = rawDS.filter("STATUS == 'OIL'").select("LATITUDE", "LONGITUDE")
fDS.count()
```

```
val rawData = fDS.rdd

val converter = (x:Any) => { if (x == null) 0.0 else x.toString.toDouble }
val points = rawData.map(row => (converter(row(0)), converter(row(1)))).collect()

GeoPointsChart(points)
```

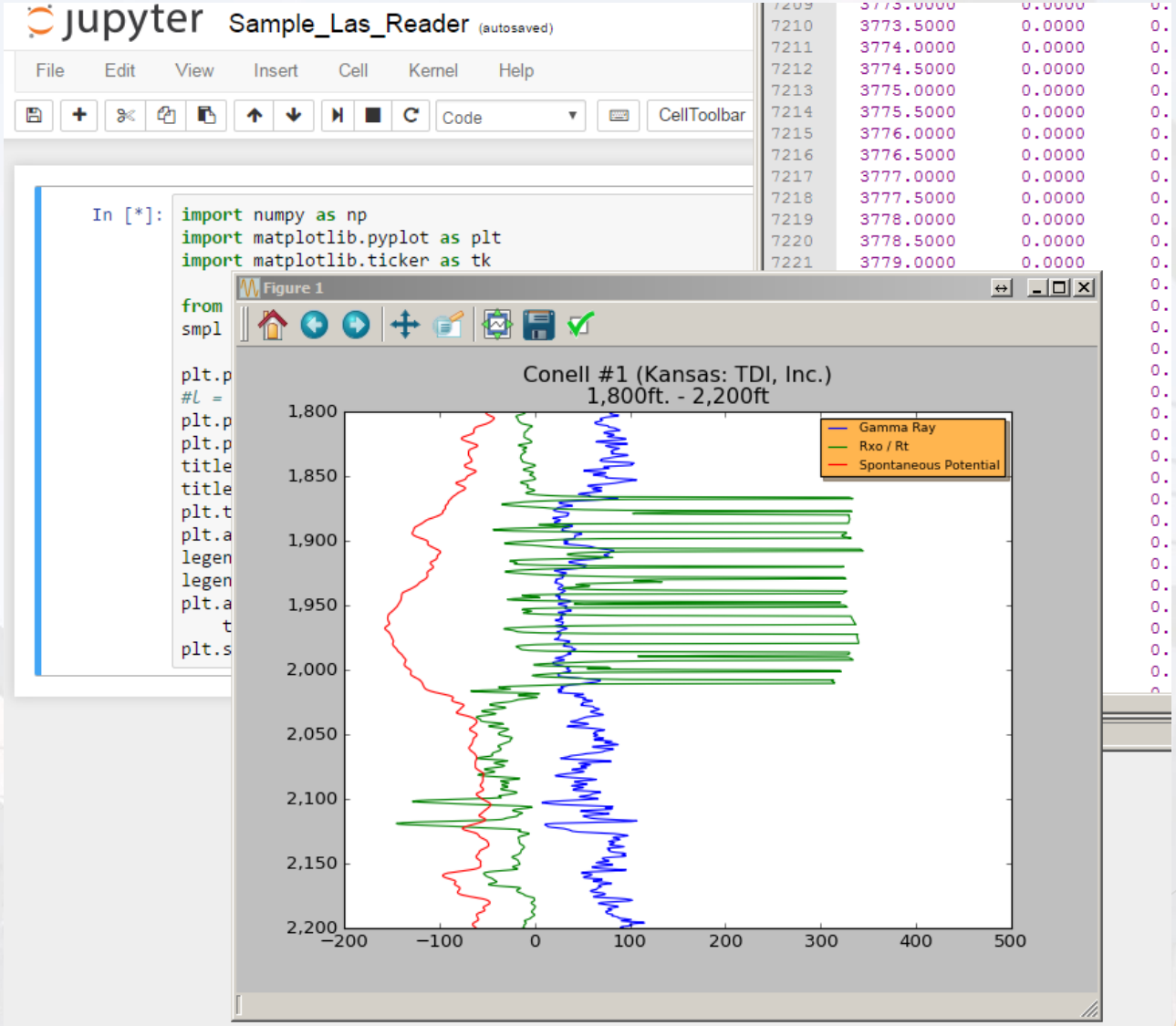


data.softelegance.com

Las file (from Wiki) - is a standard file-format common in the oil-and-gas industry to store well log information.

Well logging is used to investigate and characterize the subsurface stratigraphy in a well

Jupyter + Python for LAS file analysis

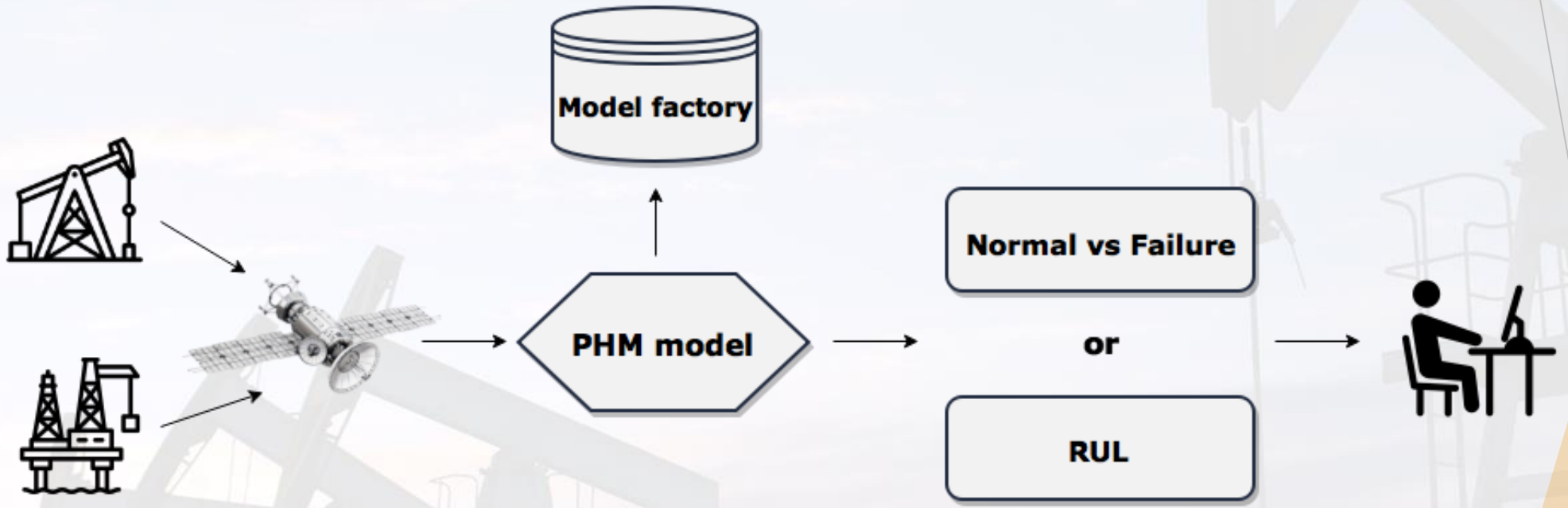


data.softelegance.com

Prognostic Health Monitoring

The implementation of an integrated software and hardware system which monitors the health, status and performance of a vehicle or system and determines remaining life of all safety and performance critical components, predicting failures before they occur

PHM workflow



Artificial lift

Artificial lift technologies

- Sucker Rod (walking beam/hydraulic)*
- Electrical Submersible Pump (ESP)
- Gas Lift (continuous and intermittent)
- Intermittent gas lift with plunger
- Gas lift with continuous slug injection
- Hydraulic Pump (jet/piston)
- Progressive Cavity Pump (PCP)
- Plunger Lift

There are approximately 2 million oil wells in operation worldwide. More than 1 million wells use some type of artificial lift. More than 750,000 of the lifted wells use sucker-rod pumps

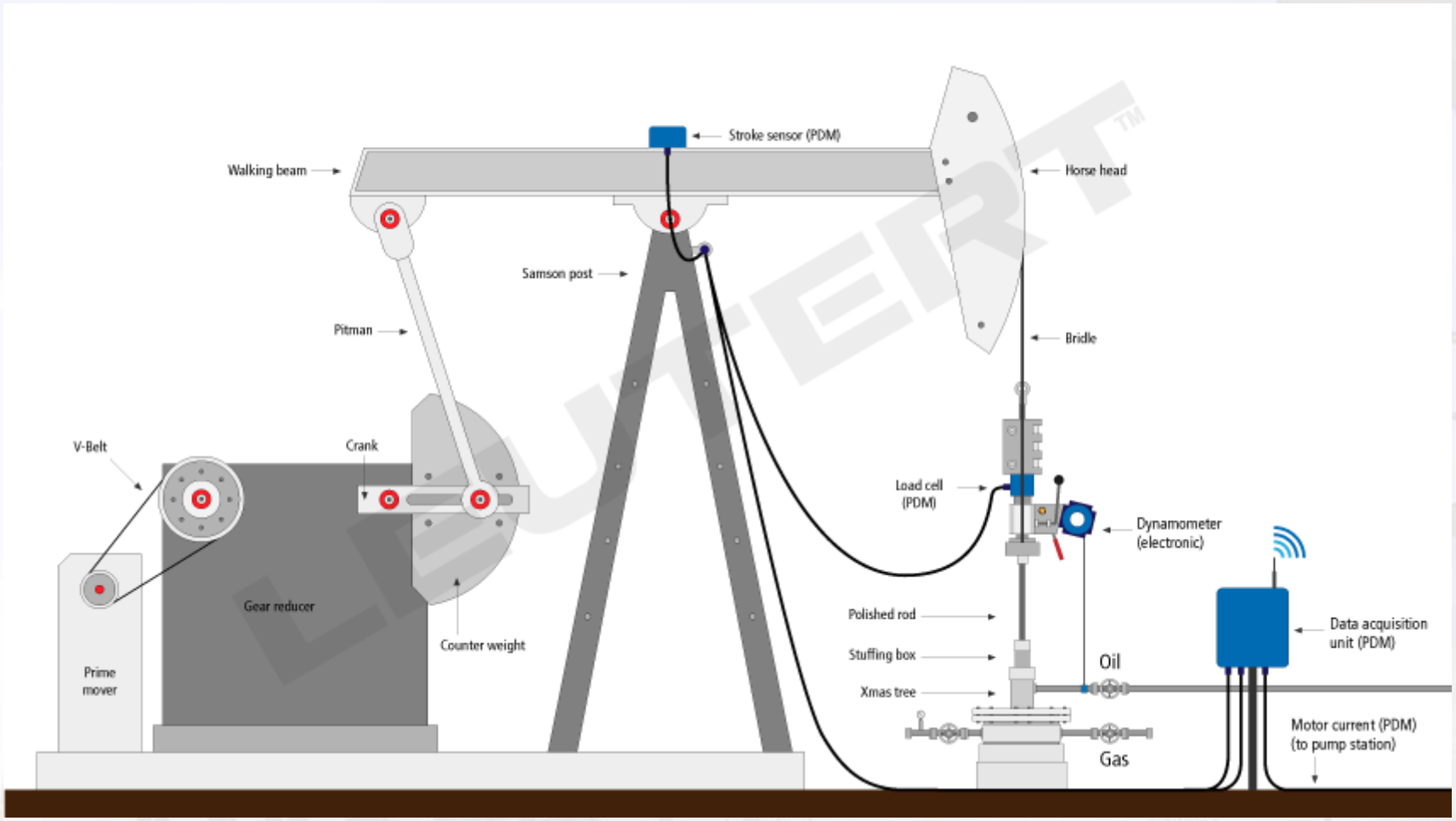
It has been estimated that just a 1% improvement in ESP (Electric submersible pumps) performance world-wide would provide over a half-million additional barrels of oil per day

Road pump failures (surface, tubing, down-hole) could lead ~ two weeks of down time for oil producing

Goal: prevent/reduce downtime (especially handle pre-failure state to help support engineers service road pump), reduce cost

Ideal: build global failure prediction model that could be scaled to all wells in each fields

Sucker rod pump



data.softelegance.com

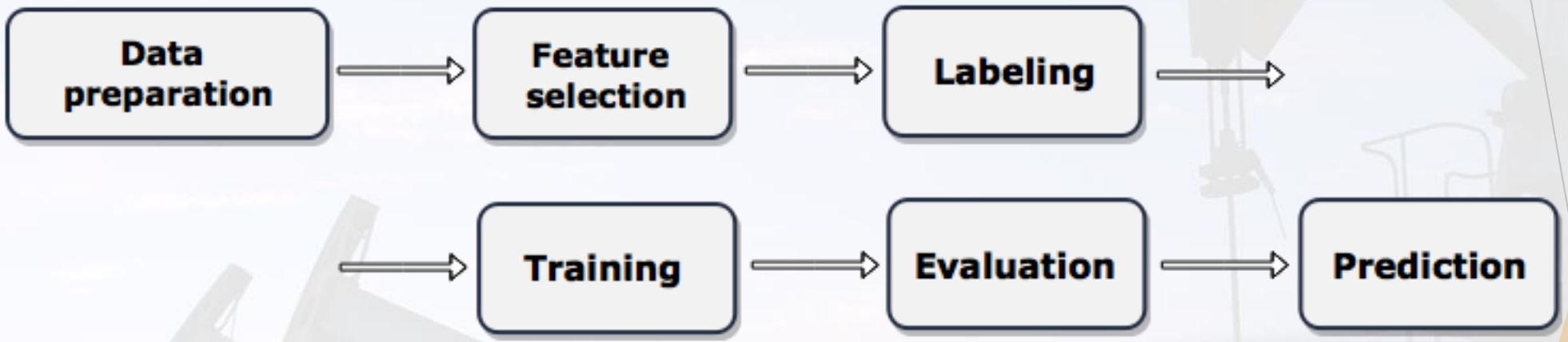
Data source - pump off controllers (POC)

Tracked parameters

- card area
- peak surface load
- minimum surface load
- daily run time
- yesterday cycles
- last approved oil
- etc

data.softelegance.com

Workflow

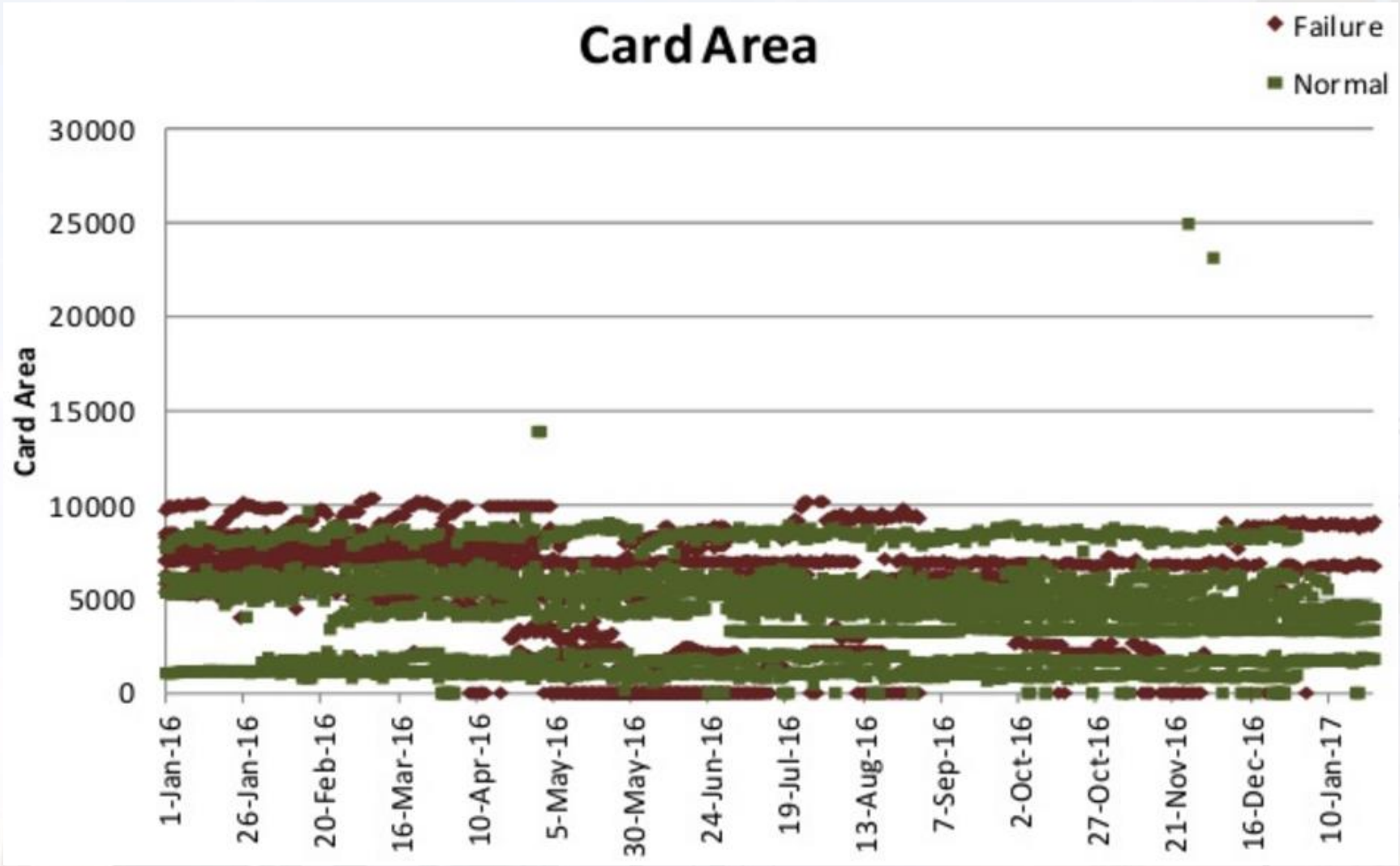

Data preparation
Feature selection
Labeling
Training
Evaluation
Prediction

Data preparation

- cleaning: equipment mistake, empty data, outliers (packets' loss due to weak radio signals or interference, etc.)

- moving median for noise reduction and trend extraction

- additional “magic”...



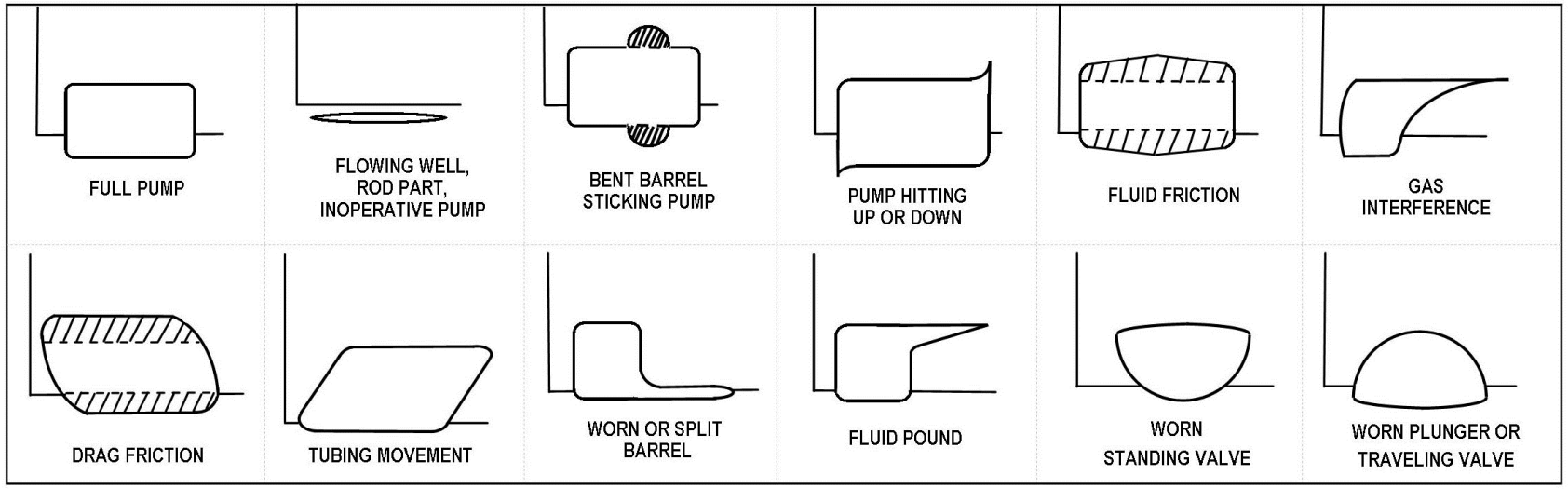
Source: Failure Prediction for Rod Pump Artificial Lift Systems

Feature selection

- select most relevant and having the highest data quality features

- features should be able to capture patterns that operator usually use to identify rod pump anomalies

Card area patterns

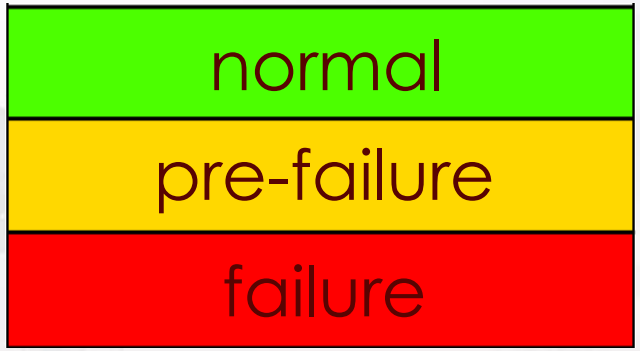


Result feature set

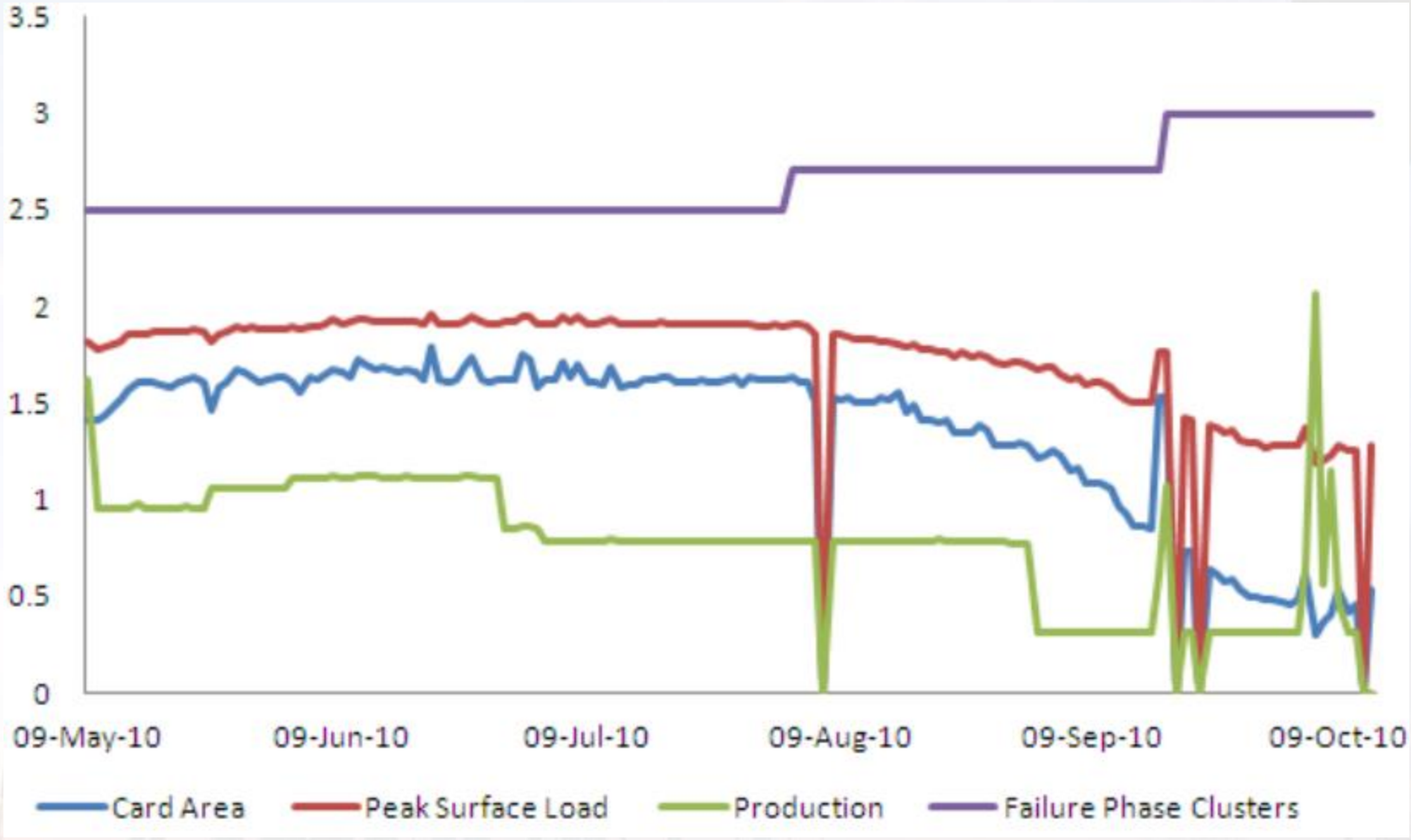
- card area
- peak surface load
- minimum surface load
- daily run time
- few calculated features ...

Labeling

- raw dataset contains normal and failure data
- from semi-supervised to supervised
- expectation maximization (probabilistic clustering)
- additional “magic”...



Failure clusters



Source: Failure Prediction for Rod Pump Artificial Lift Systems

Training

- Logistic regression
- SVM
- Naive Bayes
- Decision Tree

Evaluation

- k-fold cross validation

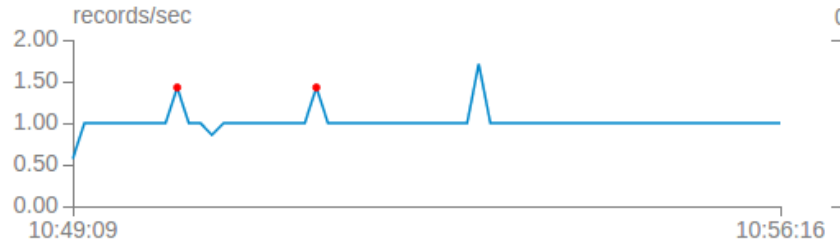
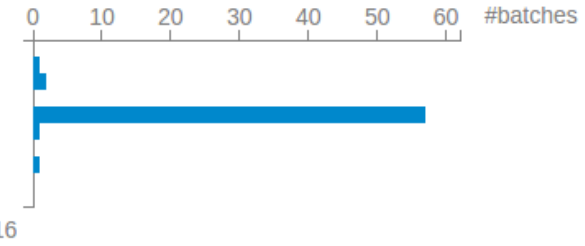
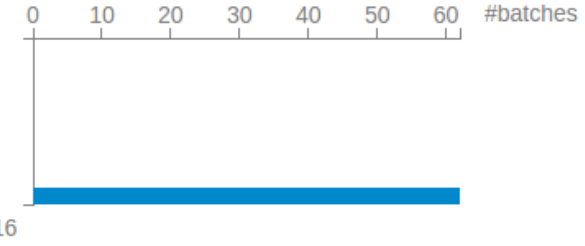
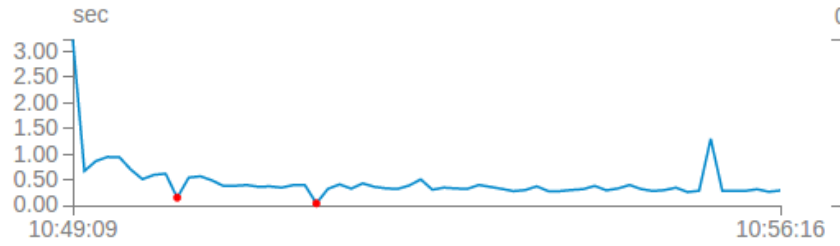
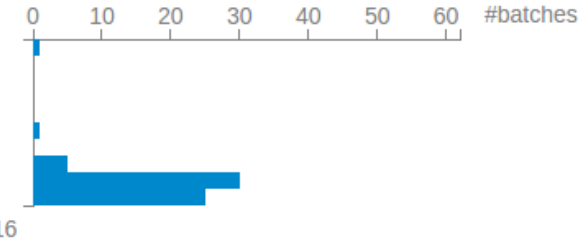
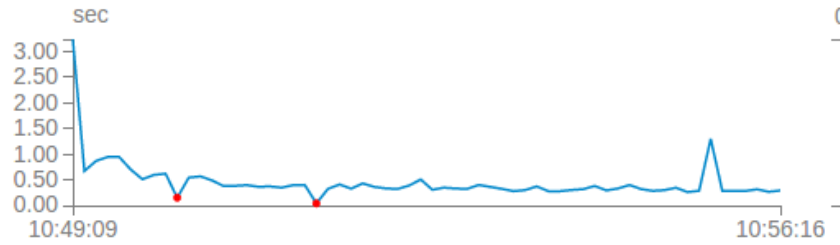
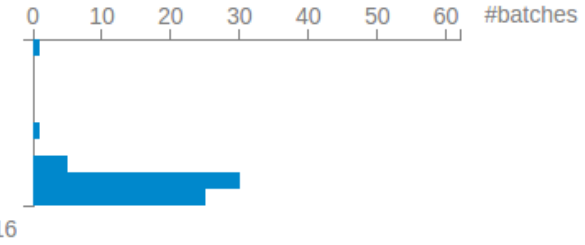
- precision, recall (very important minimize false positive rate)

Predict

Spark streaming - data transformation at runtime

```
//at driver - load model to broadcast variable
val alSrpPredictionModel = {
  val model = *Model*.load("/datalake/models/al_srp/")
  ssc.sparkContext.broadcast(model)
}

//at client – make predictions
val predictions = alSrpPredictionModel.value.transform(inputDataset)
predictions.select("id", "time", "prediction")
  .collect()
  .foreach { case Row(meterId: String, time: Long, prediction: Double) =>

    //failure
    if (prediction != NORMAL) {
      session.execute(s"INSERT INTO meters.failures (time, meter_id, type)
                                VALUES ($time, '$meterId', $prediction)")

      //send notification about failure
    }
  }
```

Running batches of **7 seconds** for **7 minutes 21 seconds** since **2016/09/24 10:48:55** (**62** completed batches, **441** records)

	Timelines (Last 62 batches, 0 active, 62 completed)	Histograms
Input Rate Receivers: 1 / 1 active Avg: 1.02 records/sec		
Scheduling Delay (?) Avg: 0 ms		
Processing Time (?) Avg: 451 ms		
Total Delay (?) Avg: 451 ms		

And what's about global model?

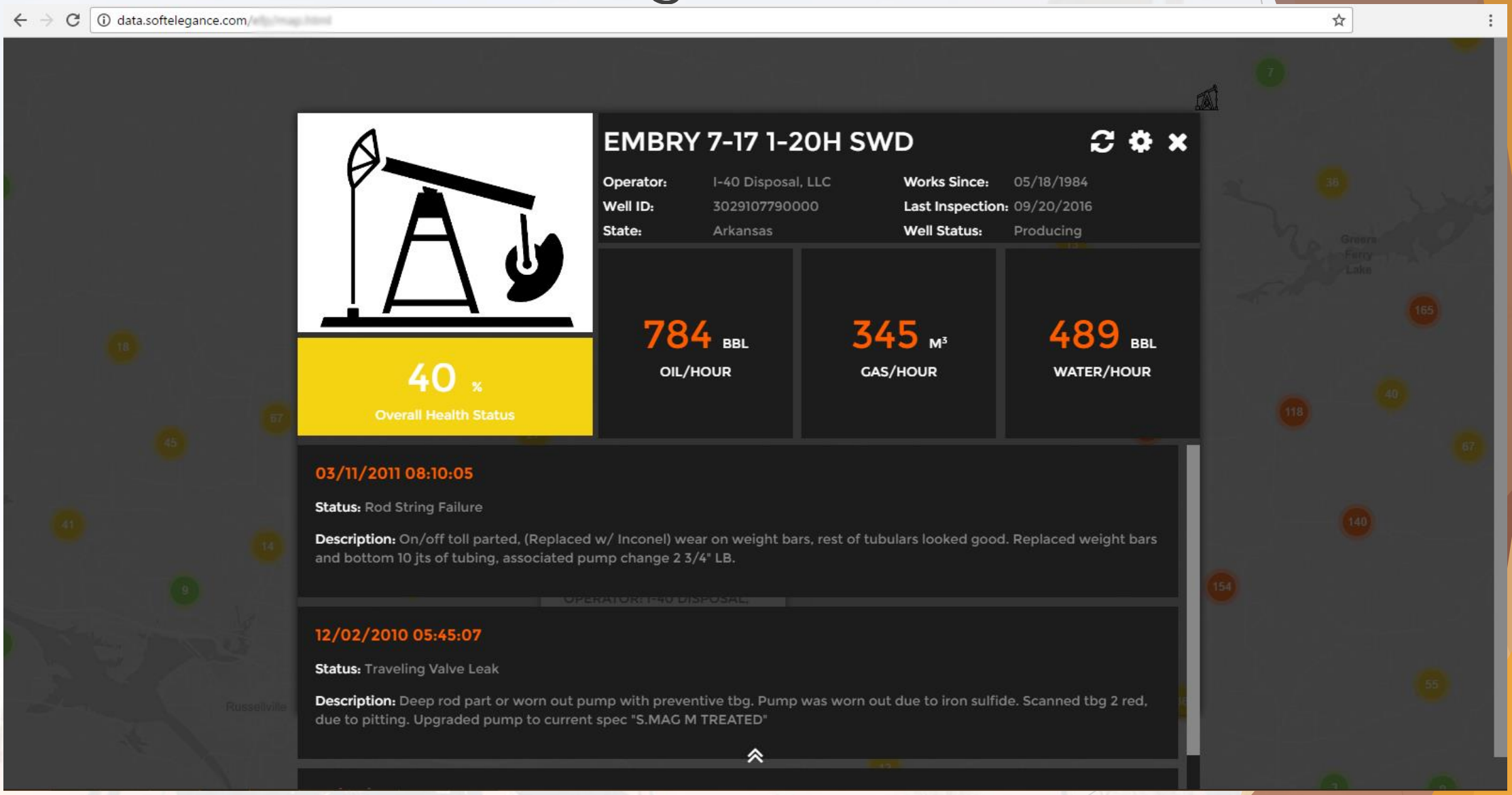
Regularized multi task learning

"Failure Prediction for Rod Pump Artificial Lift Systems by Yintao Liu"

Another approach: fast shapelets for ESP

"Predicting Failures from Oilfield Sensor Data using Time Series Shapelets
Om Prasad Patri, Anand V. Panangadan, Charalampos Chelmis,
University of Southern California,
Randall G. McKee, Chevron U.S.A. Inc., and
Viktor K. Prasanna, University of Southern California"

Intelligent Assistant



data.softelegance.com

References

1. Predicting Failures from Oilfield Sensor Data using Time Series Shapelets - http://www-scf.usc.edu/~chelmis/pubs/spe14.pdf

2. Failure Prediction for Rod Pump Artificial Lift Systems - http://cdm15799.contentdm.oclc.org/cdm/ref/collection/p15799coll3/id/323256

3. Expectation maximization - http://ai.stanford.edu/~chuongdo/papers/em_tutorial.pdf

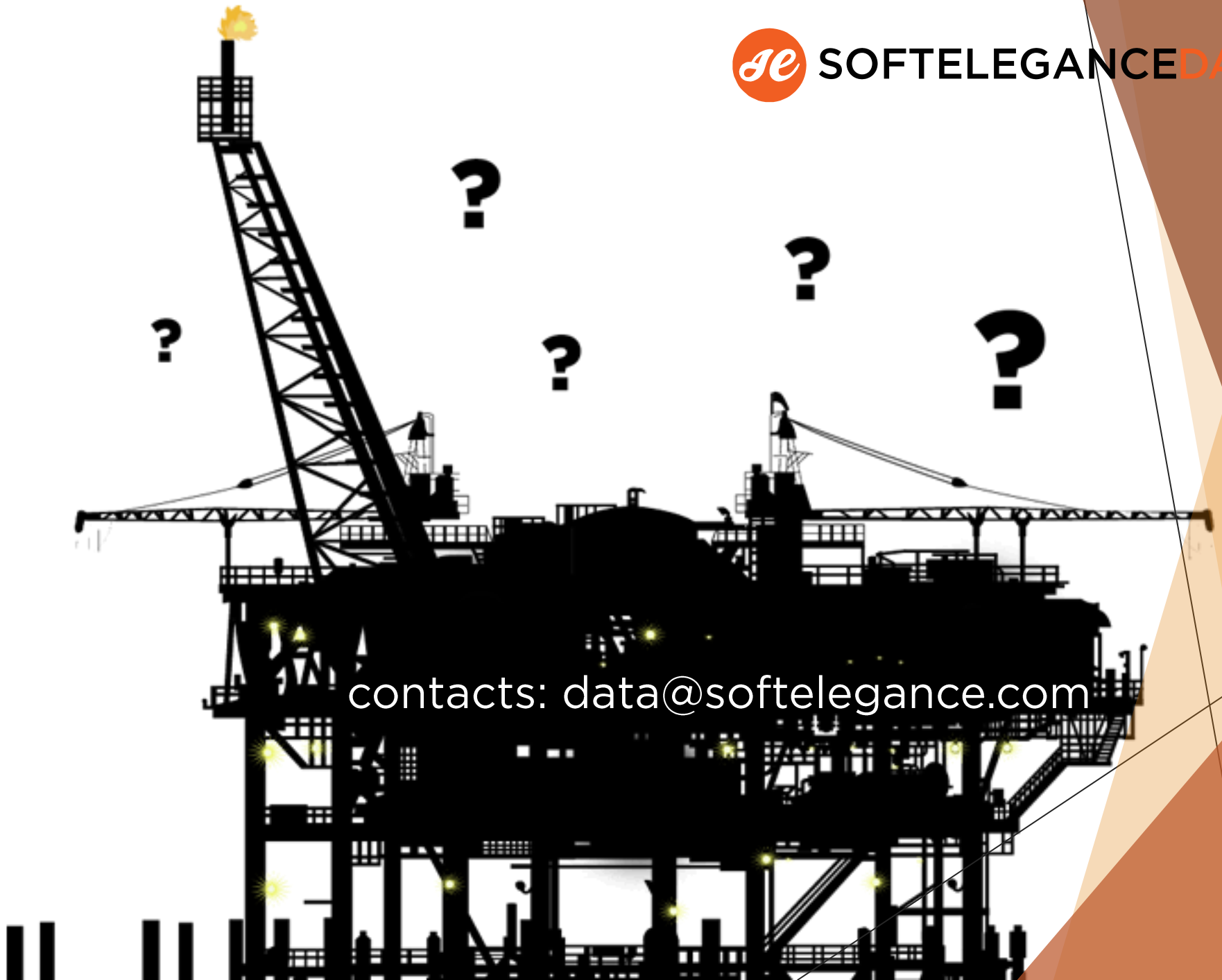
4. Using ZigBee for Wireless Remote Monitoring and Control - http://www.ethanpublishing.com/uploadfile/2015/0608/20150608015338765.pdf

Summary

- digitalization, collection and data analysis in the oil industry help monitor the entire infrastructure, equipment and perform predictive analysis, which leads to a reduction in production downtime and reducing costs

- approaches in building such kind of system could be scaled to another type of equipment, for gas industry and for industrial wireless sensor networks in general

SOFTELEGANCEDATA



contacts: data@softelegance.com