Notes

Suspect->prospect->lead->Customer->Advocate

# mikhart@microsoft.com

# IOT-Big Data Analytics (CEC)

## Data

Types of data generated

Location

Orientation

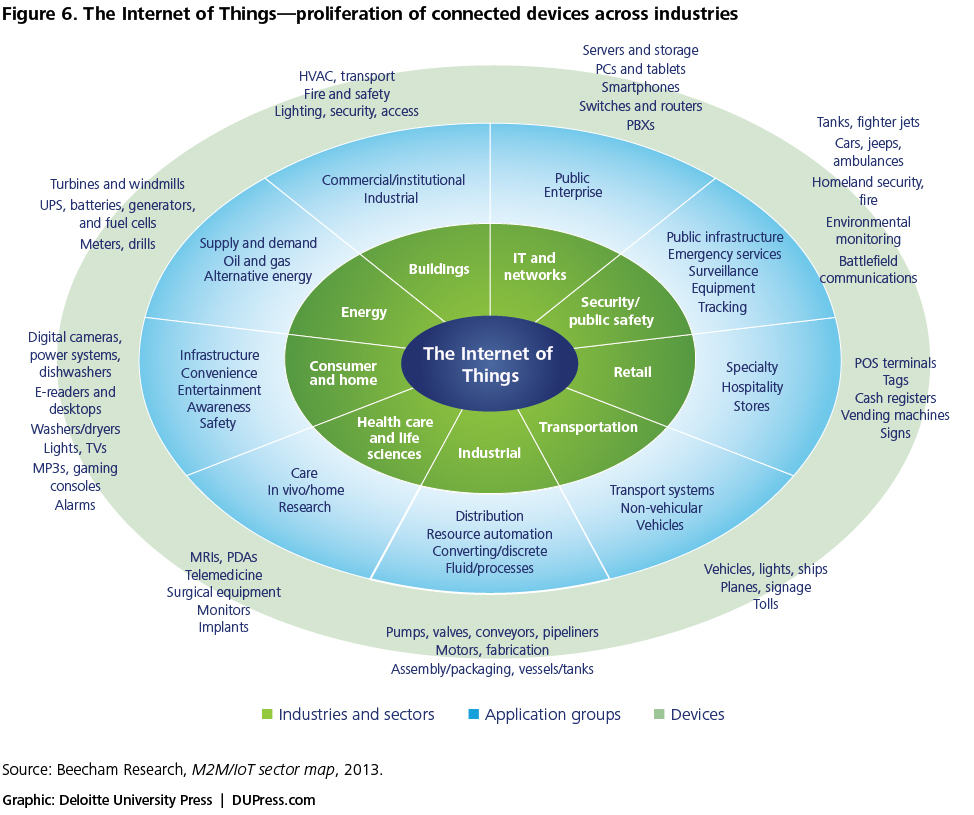
Heartrate

Events (e.g., threshold exceedance)

Status

For a thermostat, temperature and presence, settings

## Industries



## Application notes

### Infrastructure

Areas of application

Electrical infrastructure (utility and private)

Water systems

Road systems

Building systems

Gas infrastructure

* Types of data generated and consumed
  + Meter data
    - Customer premise
    - Flow (for water utilities)
  + Intermediate devices
    - Utility infrastructure (substations, generation, transmission equipment)
  + Structural status
    - Roads, bridges
  + SCADA
* Communication types
  + Cellular
    - For dispersed meters, engines, etc.
    - Special data only
  + Ethernet over powerline
  + Satellite
    - For very remote locations

### Medical

Medical devices

In care facilities

In homes

In-body (embedded)

Facility monitoring

Supplies

Devices (location and scheduling)

* Types of data generated and consumed
  + Patient data (heartrate, glucose levels)
  + Dose control information
  + Equipment location
  + Medication and supply level and location
  + Equipment health and status
* Communication types
  + Ethernet
  + WiFi
  + Cellular
  + Bluetooth

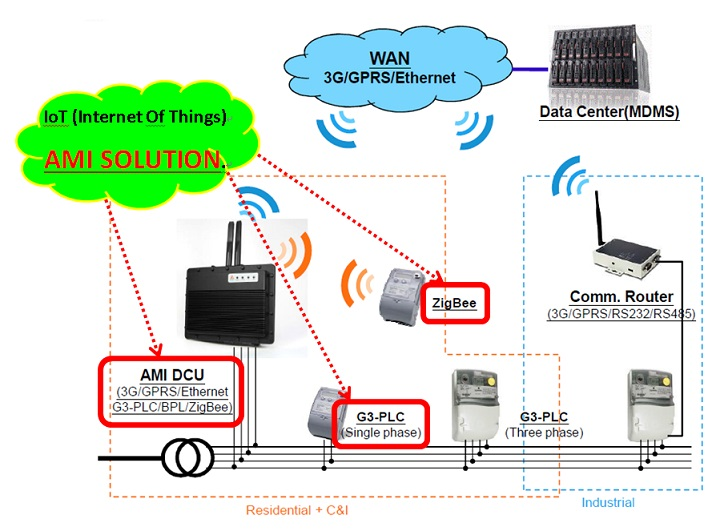
### Industrial

* + Plant equipment
    - Robots
    - CNC machines
    - Networked monitoring and control equipment
  + “Dispersed” machinery
    - Large engines (e.g., from Cummins for vehicles and power generation)
    - Monitoring equipment
  + Data Centers
* Types of data generated and consumed
  + Engine status (periodic, e.g., 5 minutes)
  + Machine actions (events) and status
  + Program in operation
  + Specific movement commands
* Communication types
  + Industrial, in plant
    - Industrial Ethernet/EtherCAT
    - Profibus
    - Modbus
    - Wireless (WiFi for industrial control)
    - Fieldbus
    - Many vendor proprietary protocols

Standards

* + EtherNet/IP (IP for Industrial Protocol)
  + CAN (Controller Area Network)
  + Modbus
  + Profibus
  + GE STRP (Industrial PLC’s)
  + Sinec H1 (Siemens)





Zigbee – small devices/sensor fabric that manages itself

Industrial bus -

## Use Case

### Product marketing

* Detect industry trends
* External factors
  + Weather
  + Competitors
* Social media
  + What is “trending”
* Current product performance

### Product Lifecycle Management (PLM)

* A growing area of product development and design
* Encompasses many of the previous use cases
* Integration of many data sources with CAE, CIM and CAD systems
* Drive product decision with data
  + Release schedules
  + Pricing

## Predictive Maintenance

* Benefits
  + Identifies key prediction factors
  + Determines likelihood of predicted outcomes
  + Optimizes decision making
    - Systematically apply institutional knowledge
  + Extending asset life
    - Uncover root causes
    - Determine optimum correction actions
    - Enhance diagnostic capabilities
  + Find trends and predict failure times
    - Proactive vs. reactive
  + Schedule maintenance and upgrades
    - Merge IoT data with other schedule information
    - Customer requirements
  + Software upgrades
    - Many products with embedded processors can be made more efficient with a software change
    - Simulate to predict improvements
    - Test against real data.

## Key enablers

Distributed computing

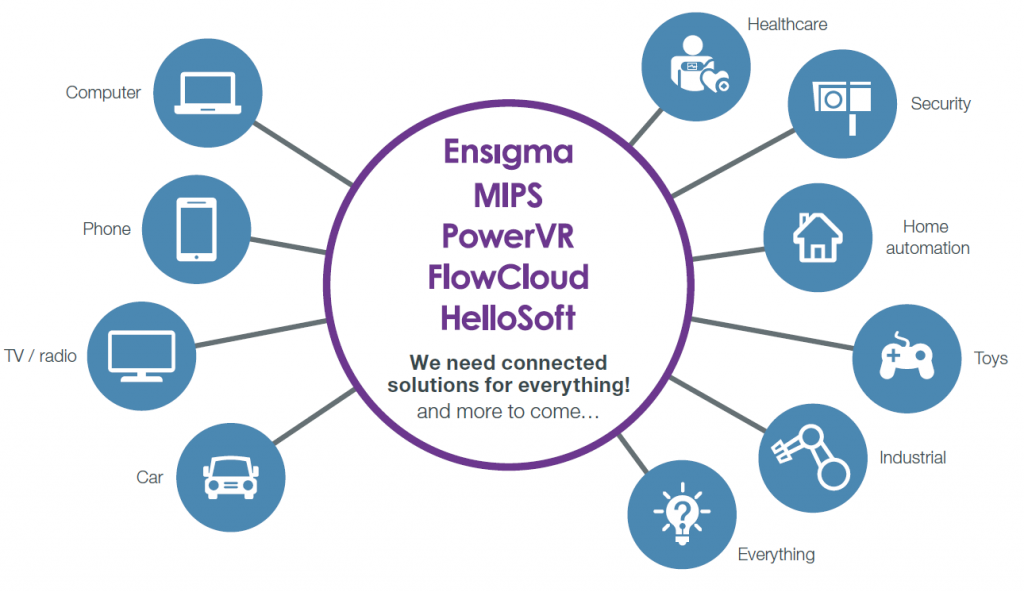
Cellular

## Architectures

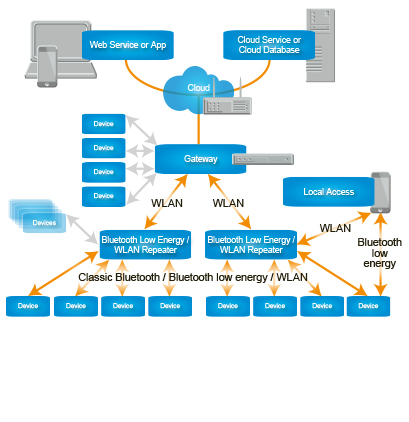


## Customer

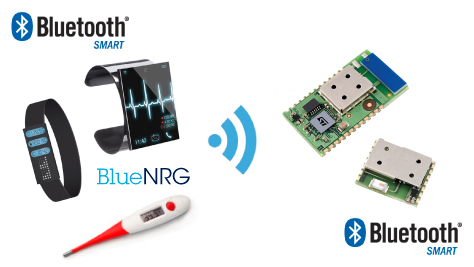
* Flexibility a key aspect
* Not generally real-time
* Cloud services



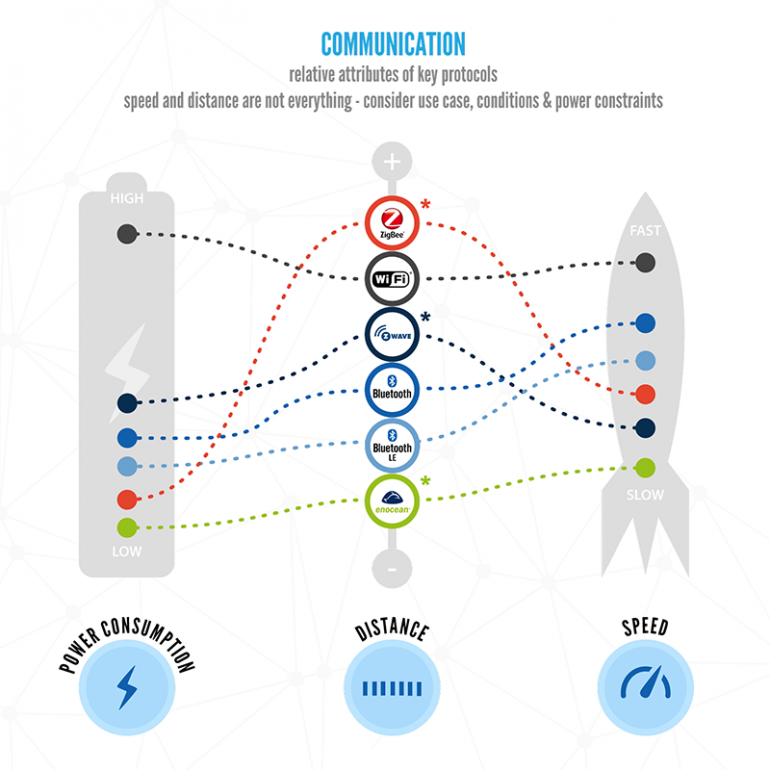
Protocols



device->wired-> gateway

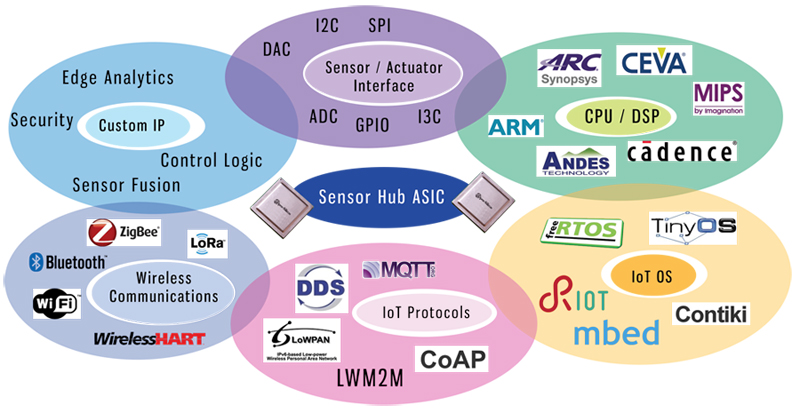






## Devices

### The Things



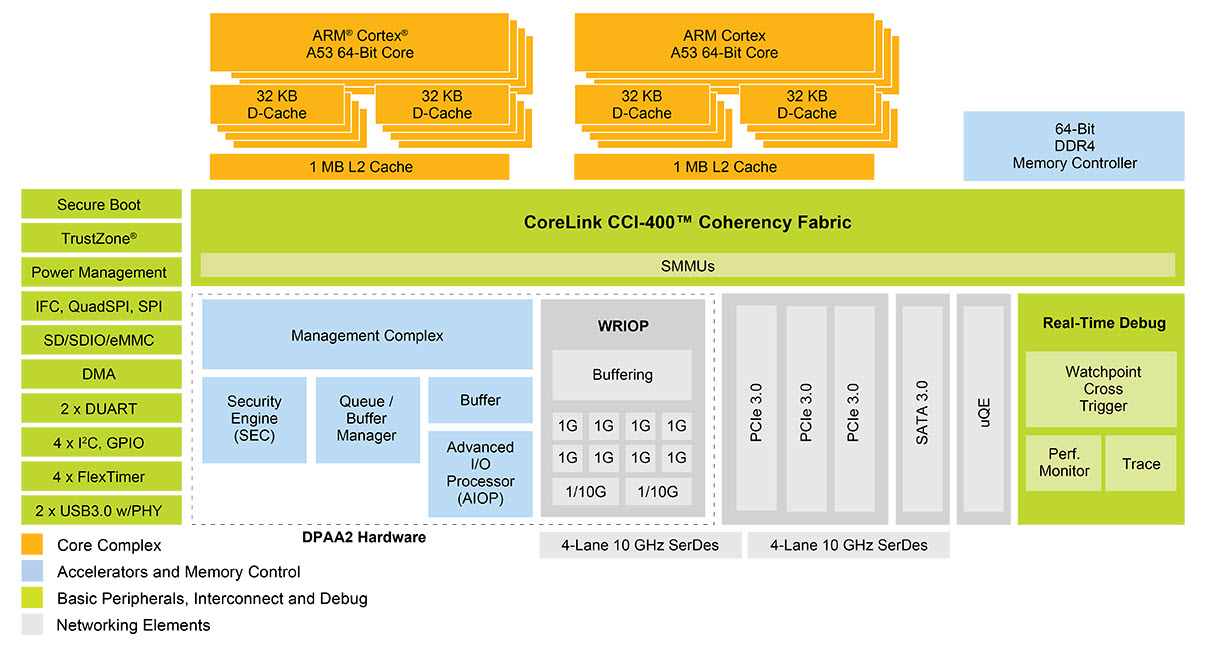
* Consumer devices
  + ARM based microcontrollers from companies such as STMicro (STM32), Texas Instruments (MSP430, C2000), NXP/Freescale (Kinetis)
* Industrial devices
  + Intel based devices
    - Quark
    - Atom
  + ARM based devices
    - Cortex R
    - Cortex M

ARM – mbed

**NXP QorIQ**

/COMmunications hub/protocol merging , handling, processing (packetizing, responding to analytics engine)

used with micro and femtocells



Freescale

COmmu

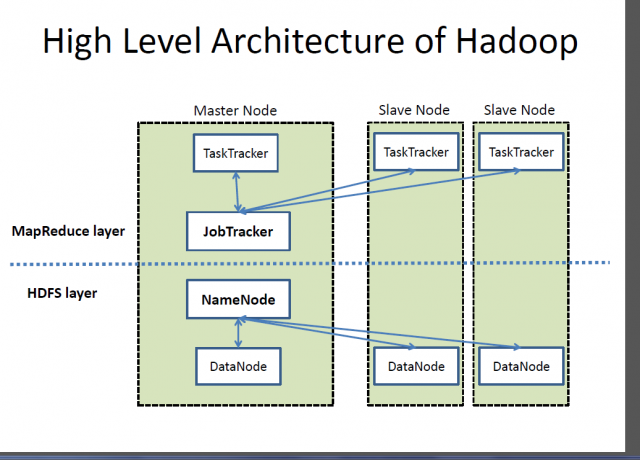
### Communications Infrastructure

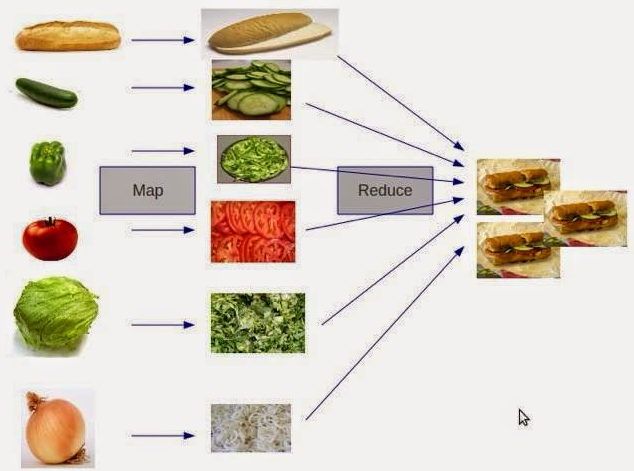
### Storage

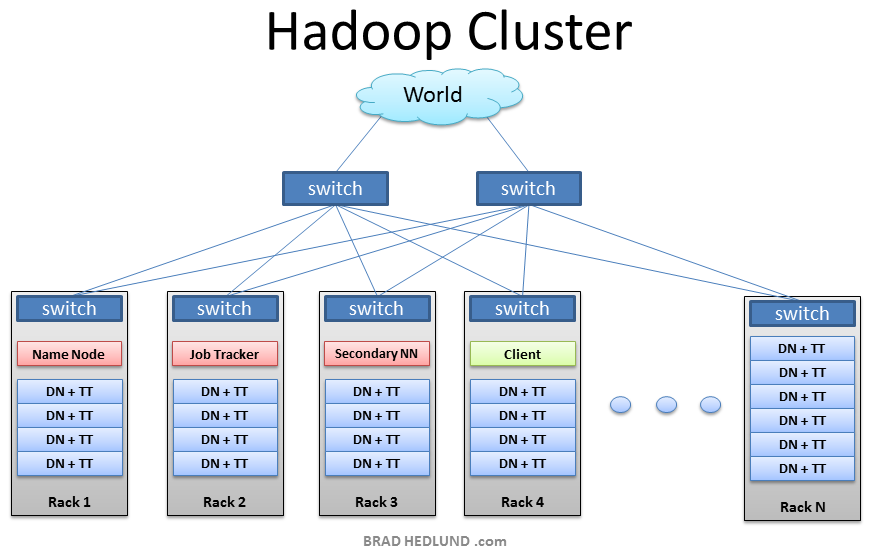
analytics techniques

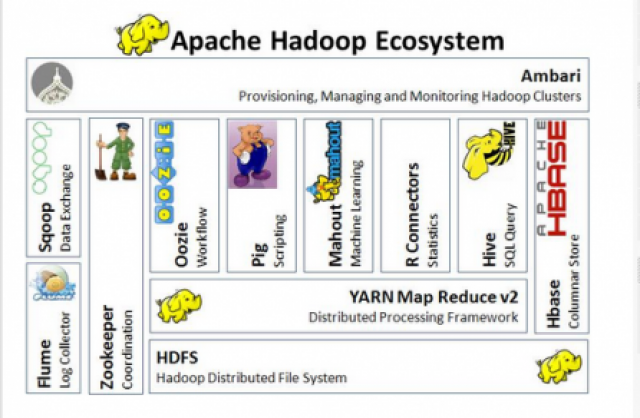
data lake

Hadoop-hdfs : distributed execution and redundancy



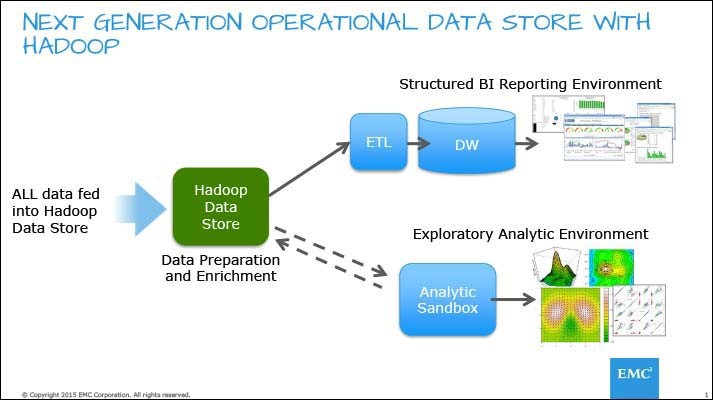






Oozie – multiple workflows

Scoop – db exchange



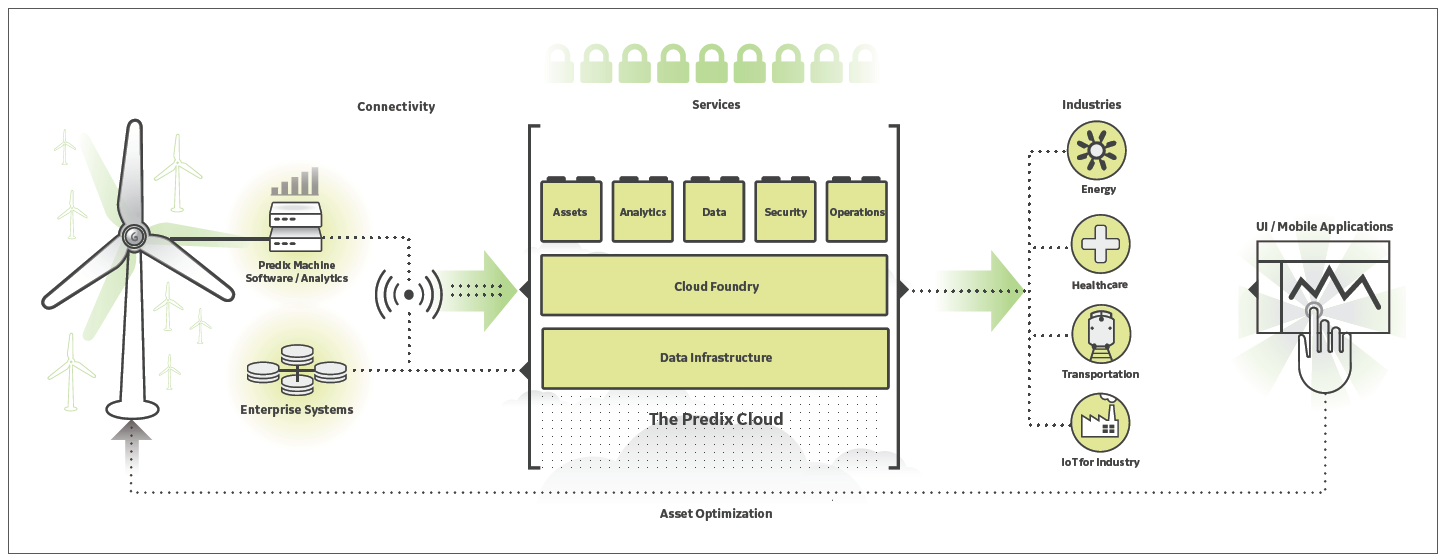
SSD – most vendors

## Platforms

### GE

BrilliantFactory

DigitalPowerPlant

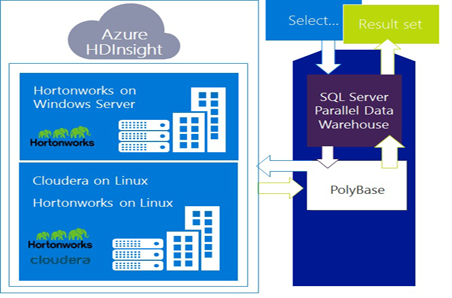


### Siemens PLM

### Thingsworx

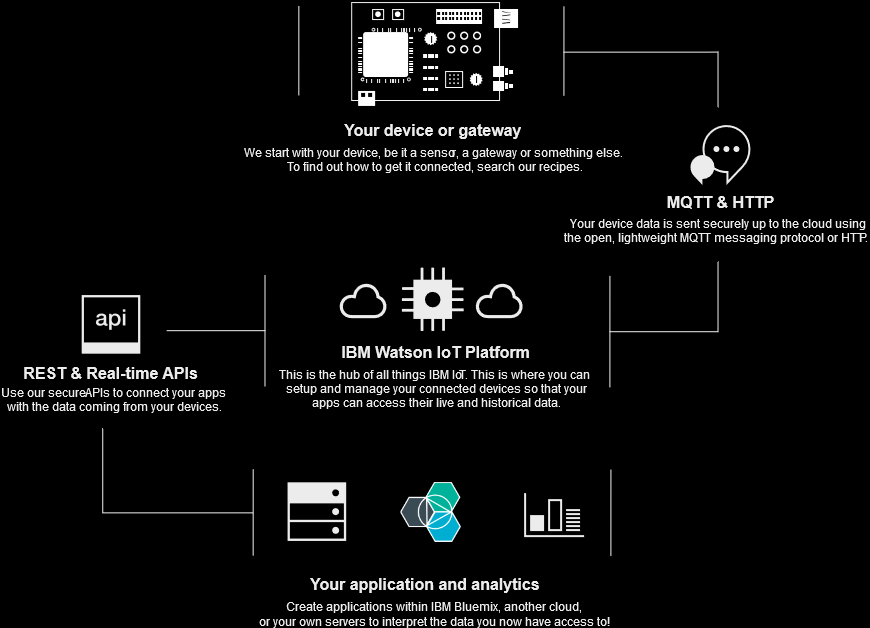
### Azure

* + - Azure HDInsight
    - Azure Machine Learning
    - Azure Data Factory



Polybase – query multiple databases

### IBM



Watson

## Analytics Techniques

Anomaly Detection

Clustering

Classification

Regression

## Software

# Big Data Stacks/ Data Engineering

## Real-time

### IBM Blue Mix

#### Message Hub

Message Hub provides multiple interfaces through which messages can be produced and consumed. Both a secure native Kafka interface using standard Kafka clients and a REST API are supported.

#### Message Connect Service/Bluemix Labs Catalog

**Attention**: Experimental runtimes and services might be unstable or change frequently, and might be discontinued at short notice.

***IBM Message Connect for Bluemix*** *is a streaming service that can connect to multiple streaming data sources, generate events and publish them to Message Hub. Message Connect supports the following connectors (with more to come in the future):*

*– Twitter*

*– SalesForce*

*– MQ Light: Connect to an on-premise MQ Light instance*

*– IBM Cloudant*

REST API: <https://kafka-rest-prod01.messagehub.services.us-south.bluemix.net/consumers/0b7344d5-2aad-4096-bf9c-1f10b38ec4ab/instances/rest-consumer-kafka-rest.7bdb550b-e6bc-11e5-ab2a-627094fadb8a-93fffc71-e5a9-4952-8ea9-02547a0b2e28/topics/twitter-spark>

X-Auth-Token: Ep8I9Tw8qcSviukFXkyGgKWQZQKmLgbbYIHLmD86HwnPnSbX

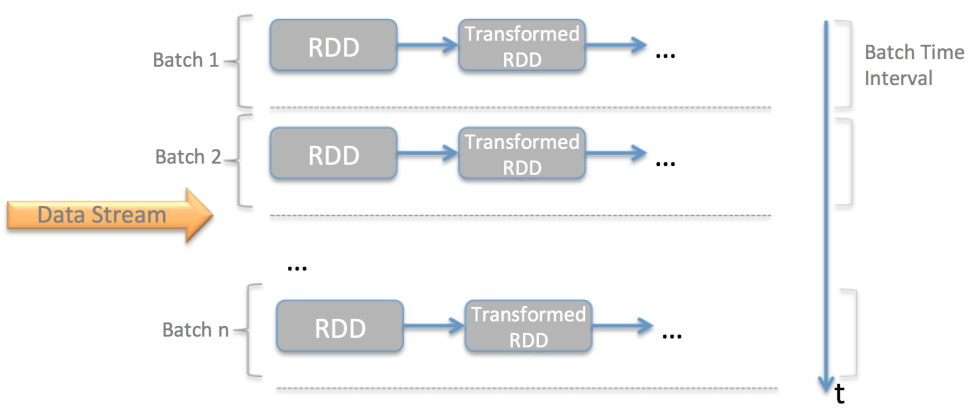
Content-Type: application/vnd.kafka.binary.v1+json

Note: This REST API will stop working after 15 minutes of inactivity, as consumer groups are periodically cleaned up. For more information on using this REST API, creating new consumer groups, and more, consult the [Message Hub documentation.](https://www.ng.bluemix.net/docs/services/MessageHub/index.html)

The Message Connect service automatically detected that a Message Hub instance was created in the same space and started publishing tweets using a topic name based on the name given to the stream. In this case, the topic is *twitter-spark* (all lowercase, spaces replaced by a dash).

we rebuild these analytics so you can run them continuously on the streaming data received from Twitter. To achieve that, we’ll use [Spark Streaming](http://spark.apache.org/docs/latest/streaming-programming-guide.html) which is an extension to the core Spark API.

Spark Streaming uses Discretized Streams (DStream) as opposed to RDDs for Spark Core. A DStream abstracts the streaming data into a continuous micro-batch of RDDs as shown in this diagram:



#### StreamingContext

When building a Spark Streaming app, the first step is to create a StreamingContext from a SparkContext and specify the batch time interval (after which a new RDD is generated by the DStream). Also, you must enable checkpointing on the StreamingContext, which lets you persist RDD metadata information periodically.

Checkpointing lets you:

1. Resume operations after a restart (normal restart or due to failure)

Enable stateful transformations between micro-batch by using the updateStateByKey API

The following code shows how to create a StreamingContext with a Batch Time Interval of 5 seconds, then set the checkpoint directory:

|  |
| --- |
| ssc **=** **new** StreamingContext( sc, Seconds(5) )   ssc.checkpoint(kafkaProps.getConfig( MessageHubConfig.CHECKPOINT**\_**DIR**\_**KEY )); |

#### Checkpoint directory

Checkpoint directory is identified by a URI that must point to a hadoop compatible filesystem, which means that the filesystem must provide an implementation of the org.apache.hadoop.fs.FileSystem class, like any of the following:

1. **file**. local filesystem
2. **HDFS**. Hadoop File System
3. **FTP**. File Transfer Protocol
4. **S3**. Amazon S3
5. **swift**. OpenStack Object Store supported by Bluemix and Softlayer.

When running this app on Bluemix, you can use the Object Storage container associated with your Spark instance as the checkpoint directory. Here’s how that works:

1. Url must have the following format:

|  |
| --- |
| 1. swift://notebook.<name>/<container> 2. where: |

* 1. <name> is an abritrary string, like **spark**, that you’ll use later in the hadoop configuration step

<container> is the name of the container or folder where all the files will live, like **ssc**.

//Spark Streaming checkpointing configuration

(from object storage: service credentials)

projectId": "cd4931897c6e4110956d4d2bd4a2e0aa",

"region": "dallas",

"userId": "bb107f2115794d5abbd99d9d1db0b54e",

"password": "rZ~.QCV]Je]\_79.#",

#### Kafka Connector

Apache Spark already provides a Kafka connector for Spark Streaming based on Kafka 0.8, but we can’t use it here because Message Hub requires Kafka 0.9. So, I built a custom Spark Streaming receiver for Message Hub using Kafka 0.9 (see [build.sbt](https://github.com/ibm-cds-labs/spark.samples/blob/master/streaming-twitter/build.sbt) updates to point at Kafka 0.9 libraries from Maven repository)

***Tip:*** *You can find the code in* [*KafkaInputDStream.scala*](https://github.com/ibm-cds-labs/spark.samples/blob/master/streaming-twitter/src/main/scala/com/ibm/cds/spark/samples/dstream/KafkaInputDStream.scala).

To create a new receiver, you need to create a scala class that inherits from org.apache.spark.streaming.dstream.ReceiverInputDStream and override the getReceiver method which returns an instance of type org.apache.spark.streaming.receiver.Receiver

In turn, the Receiver must implement the following lifecycle methods:

1. onStart: called when the receiver is started. Starts a new Thread that will poll MessageHub for new messages and store them in Spark’s memory.

onStop: called when the receiver is stopped. Cleans up all resources and stops the Thread.

#### Build the streaming analytics

## Spark

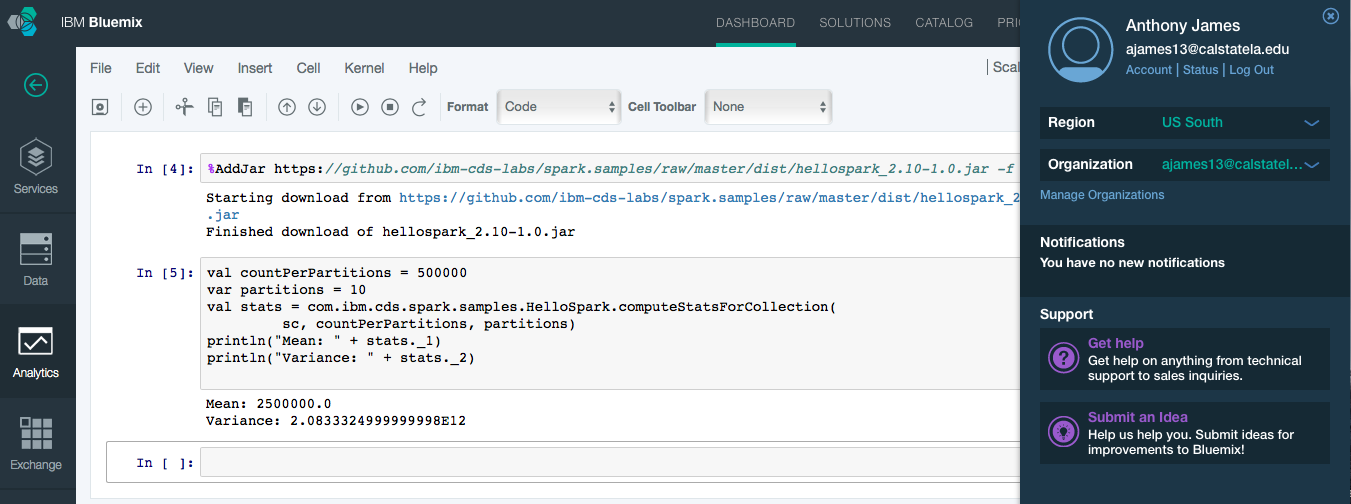
IBM Blue Mix

Add file from Github

%AddJar https://github.com/ibm-cds-labs/spark.samples/raw/master/dist/hellospark\_2.10-1.0.jar -f

-f : forces the download even if file is in the cache

Click on the face in the upper right corner so you’re e-mail shows up, when you do a screen shot



Create a new object storage service on Bluemix. Select the free tier.

Open the Spark service and continue with the tutorial at #1 – the beginning

1e

Add from existing storage (Bluemix tab).

Create a container within the storage.

### Scala

**Persist the data in a parquet file on the Object Storage bound to this Spark instance.**

fullSet.repartition(1).saveAsParquetFile("swift://notebooks.spark/tweetsFull.parquet")

**Add API configuration information**

val demo = com.ibm.cds.spark.samples.StreamingTwitter //Shorter handle

//Twitter OAuth params from section above

demo.setConfig("twitter4j.oauth.consumerKey","Do3tCRf4ZEUs5nAlKA94YiT22")

demo.setConfig("twitter4j.oauth.consumerSecret","XQXtXV2GPwPXSVejNQ7Z6P6mdfWGdcj4KtoXjNq9oAsbxwfDVI")

demo.setConfig("twitter4j.oauth.accessToken","115745248-6pfHKjwL4NiyJgwA284dPnjd1kSPapUDujgC4PcP")

demo.setConfig("twitter4j.oauth.accessTokenSecret","MaDQUIG6TTPCOfx4c10pq7JbG7c53aLgIpzkHWtUPcWYa")

//Tone Analyzer service credential copied from section above

demo.setConfig("watson.tone.url","https://gateway.watsonplatform.net/tone-analyzer-beta/api")

demo.setConfig("watson.tone.password","T4uyWlm1tNx3")

demo.setConfig("watson.tone.username","b9c5d352-8dcf-40c4-8cce-5d9bcbd66e65")

import org.apache.spark.streaming.\_

demo.startTwitterStreaming(sc, Seconds(30))

**Create a dataframe**

val (sqlContext, df) = demo.createTwitterDataFrames(sc)

**Create a dataframe**

val (sqlContext, df) = demo.createTwitterDataFrames(sc)

**Query**

angerSet.show()

### PySpark

# Import SQLContext and data types

from pyspark.sql import SQLContext

from pyspark.sql.types import \*

# sc is an existing SparkContext.

sqlContext = SQLContext(sc)

parquetFile = sqlContext.read.parquet("swift://notebooks.spark/tweetsFull.parquet")

print parquetFile

parquetFile.registerTempTable("tweets");

sqlContext.cacheTable("tweets")

tweets = sqlContext.sql("SELECT \* FROM tweets")

print tweets.count()

tweets.cache()

#create an array that will hold the count for each sentiment

sentimentDistribution=[0] \* 9

#For each sentiment, run a sql query that counts the number of tweets for which the sentiment score is greater than 60%

#Store the data in the array

for i, sentiment in enumerate(tweets.columns[-9:]):

sentimentDistribution[i]=sqlContext.sql("SELECT count(\*) as sentCount FROM tweets where " + sentiment + " > 60")\

.collect()[0].sentCount

#Plotting data

**%**matplotlib inline

**import** matplotlib

**import** numpy as np

**import** matplotlib.pyplot as plt

ind**=**np.arange(9)

width **=** 0.35

bar **=** plt.bar(ind, sentimentDistribution, width, color**=**'g', label **=** "distributions")

params **=** plt.gcf()

plSize **=** params.get\_size\_inches()

params.set\_size\_inches( (plSize[0]**\***2.5, plSize[1]**\***2) )

plt.ylabel('Tweet count')

plt.xlabel('Tone')

plt.title('Distribution of tweets by sentiments > 60%')

plt.xticks(ind**+**width, tweets.columns[**-**9:])

plt.legend()

plt.show()

1. Enter the following to compute the top 10 hashtags contained in the tweets. This code uses RDD transformations (flatMap, filter, etc…) to massage the data that will be used by the visualization code. [Read more about the RDD APIs](http://spark.apache.org/docs/latest/programming-guide.html#transformations) .

|  |
| --- |
| 1. **from** operator **import** add **import** re tagsRDD **=** tweets.flatMap( **lambda** t: re.split("\s", t.text))\     .filter( **lambda** word: word.startswith("#") )\     .map( **lambda** word : (word, 1 ))\     .reduceByKey(add, 10).map(**lambda** (a,b): (b,a)).sortByKey(False).map(**lambda** (a,b):(b,a)) top10tags **=** tagsRDD.take(10) |

Enter this visualization code to plot the data as a pie chart:

**%**matplotlib inline

**import** matplotlib

**import** matplotlib.pyplot as plt

params **=** plt.gcf()

plSize **=** params.get\_size\_inches()

params.set\_size\_inches( (plSize[0]**\***2, plSize[1]**\***2) )

labels **=** [i[0] **for** i **in** top10tags]

sizes **=** [int(i[1]) **for** i **in** top10tags]

colors **=** ['yellowgreen', 'gold', 'lightskyblue', 'lightcoral', "beige", "paleturquoise", "pink", "lightyellow", "coral"]

plt.pie(sizes, labels**=**labels, colors**=**colors,autopct**=**'%1.1f%%', shadow**=**True, startangle**=**90)

plt.axis('equal')

plt.show()

# Databases/ Data Engineering

## Data Platform

**Seamless Integration**

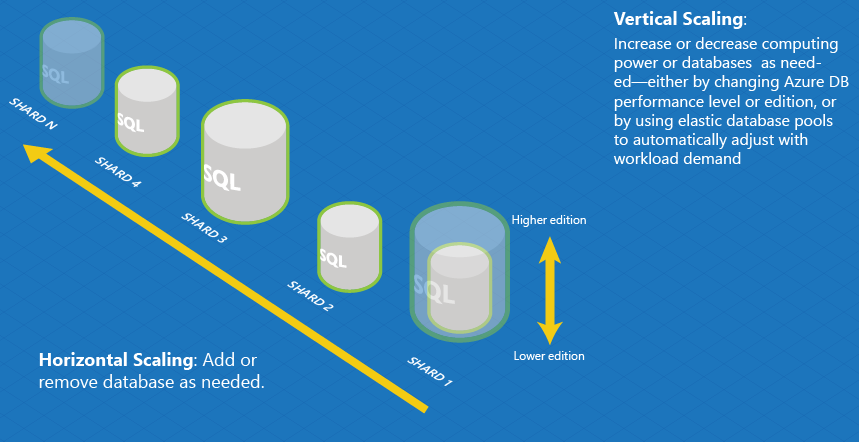
Visualiza & Decide – Natural Language Query ++

Transformation & Analyze – Cep, ML ++

Capture+manage – Relational, Non-relational, NoSQL, Streaming, Internal & External

Elastic database pool – query a pool using a single query/ use the virtually unlimited database resources of **Azure SQL Database** to create solutions for transactional workloads, and especially Software as a Service (SaaS) applications

SaaS



**Azure SQL Database V12**

**DTU – database throughput unit – computer, writes, reads, memory**

**Security – row-level, auditing, dyamic data masking, filter by location, dynamic data masking**

IN-memory column store

Basic – 5

Standard - 10 to 100 DTUs | line of business

Premium – P1 to p3 (125 to 1000) | high enterprise

**SQL Server on Azure VM**

Saas

Elastic

99.99% availability

Geo-rep

Compatible with SQL 2014

**AzureDocumentDB – eg. like Mongo**

Auto indexing

Fast reads

**Azure Search**

**Azure ML**

(Smarter experience)

**Azure Data Factory**

Orchestration and moving data (Pipeline)

**Azure DataLake**

Schema on Read

**Active Directory Connect**

(Active Director-local <-> Active directory-cloud)

## SQL Server 2016

Database - 46% market share (Microsoft), Beat Oracle in Magic Quadrant

High availability workloads - 60% (Microsoft)

98% said 1 hour downtime = 100,000 or more (upper mid-market companies)

Cloud Co. Spending

I-Infrastructure – 31%

II-Security -21%

III-Application hosting -27%

**IV\_Managed Services – 22**

**Managed Service**

- Professional Services (DBA, migration services, solution architecture)

- IT & Infrastructure Automation

- Managed security

- Managed Applications

- DevOps

**Application Hosting**

(Database)

1)Database as a Service

2)In-memory

3)High Availability

4)

-redundancy/disaster recovery

1 | Introduction

2 | Server Virtualization

3 | Software-Defined Storage

4 | Software-Defined Networking

5 | Introducing Nano Server

6 | Introducing Windows Server and Hyper-V Containers

7 | Automation in Windows Server 2016

## SQL Server 2016

([sarahb@microsoft.com)](mailto:sarahb@microsoft.com))

Public cloud-Azure, AWS, ….Google

Private cloud – Azure Stack (on-premise)

Cloud – legacy, dying hardware

Database as a Service (SQL Server) – spread costs, bigger than SQL Azure

Virtualization – 6% hit on performance

Hybrid Cloud

**Use Case 1 - Scale out reads , Can read from secondary’s**

**License @ host level (2 hosts (A-primary- writes, B-secondary -reads); also B2-primary, and A2-secondary**

**Case 2 - Production + Analytics**

**Primary – production, Secondary – Analytics**

**Backup** – to a URL/Azure Storage (no cost for incoming) //stored procedure

EXEC smart\_admin.sp\_set\_db\_backup

@database\_name=”TestDB’,

@storage\_name=’TestDB’

……

**Disaster Recovery** – Log shipping, Always on Availability groups – 2 nodes/Replicas (2 local, 3rd geographical – can be in Azure)

Secondary - use for backup/pay for full version of enterprise version; reporting/BI apps

VPN tunnel

**Cloud bursting (Burst into Azure)**

**Multi-server management/**

SQL Server **Management studio** (own release cycle): eg. Like Toad tools

Registered servers

Queries on a many machines at once

Central Server

**Policy based Management (Management studio)**

**High Availability**

* + - * **SQL Server Standard –** log shipping, database mirroring, clustering (has a problem- not good for critical e.g. .life support)

**SQL Server Enterprise (Premium) –** AlwaysOn Failover Cluster Instance **(need SAN vendor to**

**build out AlwaysOn)**, Always on availability groups (SQL Server Azure), online maintenance

**Temporal Tracking of data –** data audit, repair, time travel

**Security** – in-fligth and at rest, even certain column, PCI compliance, social security, revenue, salary,

**JSON**

**Polybase-access any data eg. HDFS**

**Datazen – mobile app design**

**l**

**In Memory**

Good/bad candidate for in-memory

10 x faster (20x 2014->2014)

Machine compiled stored procedures

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | In-Memory OLTP |  | designed to run both OLTP and OLAP (on-line analytics processiong |
|  | you can define part of the application data as being specifically for transactions -- typically, the high-speed, intensive reads and writes that come with market segments like retail and banking. |  |  |
| low latency and high overall performance | defined parts of the database are kept in memory, |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name | **Microsoft SQL Server**[**X**](http://db-engines.com/en/system/SAP+HANA) | **SAP HANA**[**X**](http://db-engines.com/en/system/Microsoft+SQL+Server) | | | | | | | | | | | |
| Description | Microsofts relational DBMS | In-memory, column based data store. Available as appliance or cloud service | | | | | | | | | | | |
| Database model | [Relational DBMS](http://db-engines.com/en/article/RDBMS) | [Relational DBMS](http://db-engines.com/en/article/RDBMS) | | | | | | | | | | | |
| |  |  | | --- | --- | | [DB-Engines Ranking](http://db-engines.com/en/ranking) |  | | [Trend Chart](http://db-engines.com/en/ranking_trend/system/Microsoft+SQL+Server%3BSAP+HANA) | |  | | Score | | 1136.49 | |  | | Rank | | #3 | |  | |  | |  | | Overall | |  | | #3 | |  | |  | |  | | Relational DBMS | | Score | | 39.99 | |  | | Rank | | #19 | |  | |  | |  | | Overall | |  | | #12 | |  | |  | |  | | Relational DBMS | | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Website | [www.microsoft.com/­sqlserver](http://www.microsoft.com/sqlserver/) | [www.saphana.com](http://www.saphana.com/) | | | | | | | | | | | |
| Technical documentation | [www.microsoft.com/­sqlserver/­en/­us/­default.aspx](http://www.microsoft.com/sqlserver/en/us/default.aspx) | [help.sap.com/­hana](http://help.sap.com/hana) | | | | | | | | | | | |
| Developer | Microsoft | SAP | | | | | | | | | | | |
| Initial release | 1989 | 2010 | | | | | | | | | | | |
| Current release | SQL Server 2014, April 2014 | 1.0 SPS11, November 2015 | | | | | | | | | | | |
| License | commercial | commercial | | | | | | | | | | | |
| Database as a Service (DBaaS) | no | no | | | | | | | | | | | |
| Implementation language | C++ |  | | | | | | | | | | | |
| Server operating systems | Windows | Appliance or cloud-service | | | | | | | | | | | |
| Data scheme | yes | yes | | | | | | | | | | | |
| Typing | yes | yes | | | | | | | | | | | |
| XML support | yes | no | | | | | | | | | | | |
| Secondary indexes | yes | yes | | | | | | | | | | | |
| SQL | yes | no | | | | | | | | | | | |
| APIs and other access methods | OLE DB  Tabular Data Stream (TDS)  ADO.NET  JDBC  ODBC | JDBC  ODBC | | | | | | | | | | | |
| Supported programming languages | .Net  Java  PHP  Python  Ruby  Visual Basic |  | | | | | | | | | | | |
| Server-side scripts | Transact SQL and .NET languages | SQLScript, R | | | | | | | | | | | |
| Triggers | yes | yes | | | | | | | | | | | |
| Partitioning methods | tables can be distributed across several files (horizontal partitioning); sharding through federation | yes | | | | | | | | | | | |
| Replication methods | yes, but depending on the SQL-Server Edition | yes | | | | | | | | | | | |
| MapReduce | no | no | | | | | | | | | | | |
| Consistency concepts | Immediate Consistency | Immediate Consistency | | | | | | | | | | | |
| Foreign keys | yes | yes | | | | | | | | | | | |
| Transaction concepts | ACID | ACID | | | | | | | | | | | |
| Concurrency | yes | yes | | | | | | | | | | | |
| Durability | yes | yes | | | | | | | | | | | |
| In-memory capabilities |  | yes | | | | | | | | | | | |
| User concepts | fine grained access rights according to SQL-standard | yes | | | | | | | | | | | |

**Security and Compliance**

SQL Server 2014 – no vulnerabilities

1)Standard – user defined roles, server-define roles, ++

2)2016 – masking, encrypting ++

Separation of duties (admin but not see the data – SQL Server 2014, can get around – view/doesn’t work)

**(see slides)**

**Migration Tools**

# Systems Engineering

# Software Engineering

# Business Analytics

Fraud – fit a particular statistical distribution

# Shadow IT’

# (Sort)

Meetup- Big data, security notes

Networking notes: Security seminar/webinar

Ngrep

Certificate strip tls setting http a:// start tls

Enforce tls 1.2

Start tls + pinning policies -- ideal

End to end

Php email encryption

Header message not encrypted

Gpg

End to end chrome extension

<HTTPS://got> [big.com/](http://big.com/) yahoo/end to end

E2e

Webcrypto

Content security policy - limit scripts and embedded

Packs king process not audit able from injected vulnerabilities

Networking: Big data-Splunk notes

Download [splunk.com](http://splunk.com)

Splunk

Bi over unstructured and structured

Da taming with auto schema auto key value pairs

Customer experience based on all sources and union base on IP address

Time to insight without schema - real- time actions

Track customer process on website

Seminar Notes/export

Cisco data virtualization information server

Sql query-driven

Virtualization layer

Shorten data engineering cycle

Reduces Data quality issue in Etl

More time on analysis

Integration software like informatics Etl golden date cdc (consolidation), tibco ( event driven), but is on demand

Cache a data set or view to boost performance

Jdbc, odbc, rest or json service call

Networking: Spark IBM notes

Rrdd Rudd for r

**Error! Filename not specified.**

Need spark:

A parallel programming

Complex computation

Large scale

Interactive

System ml open source - more algorithms

Educational training free - big data university

Spark fun

Adv spark development

Ds fund.

[Www.spark.tc/beta](http://Www.spark.tc/beta)

Sem modeling - causal relationship

Oscp osce security test

Training

Defect management

No vulnerabilities

Time to remediate

Metrics and accountability

Top bugs out-focus on

Preventative testing 67% useful

Dynamic interactive testing 54%

Compliance reviews audits 49%

Firewall rasp run/time app sec protection

manual code review

threat modeling 34%

Static code testing 31%

Metrics-benchmark/your results

Average days open

Avg no open

Remediation ratefirstvtime pass rate

Application security capability maturity // bsimm out of 12 practices

Verizon

Trust wave

Owasp top 10

Cigital

Veracode

PwC

Ponemon

Gardner

Sans institute.

WhiteHat

day marketing

Robert wood Nuna is c

Blaze rule management

Encrypt longer string as varbyte aes256

Tokenization store same space

Masking changing at run time what is being displayed

Obfuscation always convert Jon to mark

De identification method identify part of the field to protect the record

Apply protection to data

Format preserving encryption encrypt and compress back to size

Fibbs approval protection method

Referential integrity keep the field

Tokenization more sensitive to data quality issues

80% of analytics can be performed on protected data

Audit trails for viewers to a remote log

Course/file level protection

Course grain for unstructured datA

CTO

Lean startup

MPD DATA SALES

Manage data quality

Snowball AWS STORAGE ACCOUNT

[Ian@datascience.com](mailto:Ian@datascience.com)

Dynamically create content

Asking the right questions

Blaze rule management

Encrypt longer string as varbyte aes256

Tokenization store same space

Masking changing at run time what is being displayed

Obfuscation always convert Jon to mark

De identification method identify part of the field to protect the record

Apply protection to data

Format preserving encryption encrypt and compress back to size

Fibbs approval protection method

Referential integrity keep the field

Tokenization more sensitive to data quality issues

80% of analytics can be performed on protected data

Audit trails for viewers to a remote log

Course/file level protection

Course grain for unstructured datA

CTO

Lean startup

MPD DATA SALES

Manage data quality

Snowball AWS STORAGE ACCOUNT

[Ian@datascience.com](mailto:Ian@datascience.com)

Dynamically create content

Asking the right questions

Attribution modeling

Owasp

Ami and docker Images

Strip down

Stay away from standard images

Strip down ami images

Apply iam roles to take advantage of least price ages

Laying with iam containers

Automation configuration chefs and puppet

Have deployment code reviewed

S3 redshift etc has to be configured

Data ex filtration

Apple security groups on top of s3 etc

S3 buckets

Role based decryption, denitrification of data

Iam policies and security group

Setup network flow logs

Log access box flow logs splunk data dog

Duo push dual authentication key into Non Sudo accounts

Lock away hardware root key

Protect root account

Terraform you can deploy to digital ocean race pace or azure

Threat model early

Embrace platform provider controls

 Understand the doomsday scenarios

Malicious insider

Signal science monitoring - proxy or Apache server extension, cloud flair, Emilio and splunk forwarders

Ip whitelist

Hd more internet scanning

Application controls

Jenkins run all the security tests automated pen tests queks

Security monkey

Dependency check

Grr tool

Cloud formation -cloud deployment

Laptops - chef

Odpi kernel

<http://pivotal.io/platform>

Securing micro services

Securing micro services with red hat KEYCLOAK

authentication server - distributed, fault tolerant, scale independent

Storm path, azure service

FOSS

KEYCLOAK, auth service

Get a token

Stores password

two factor authorization

Supports Saml v2, openid connect (gets credentials)

Clint adapters know how to talk to :

Java EE spring JavaScript and node is

SPI is a plugin to KEYCLOAK

I'd brokering - openid Saml xml github google Twitter

Micro services

Martin fowler

Discrete functional services

Independent release

Scaled independently

Http resource api

Pick the best technology

distributed app

1 function per service

(Shopping cart, search...

Uses Ali and messaging device

Contract at service levels

Eg. Load balancer to 3 micro services

Deploy in dockers- lighter than VMs

SPi

Jenkins - automation and delivery

Build infrastructure

Developer tools - Kira, confluence

Etc

NGIos monitoring

Puppet enterprise

Pager duty

Data dog

Servers spec

Infrastructure

-Resources

-configuration management

Continuous delivery = build archive test stage deploy

1, Unit testing r-spec puppet also include in chef cookbooks

2. Acceptance testing - [serversspec.org](http://serversspec.org)

Pipeline - Daiseycahinbchange

jenkins invokes as a

Deploy using! R10k

[Tyler@linux.com](mailto:Tyler@linux.com) r Tyler Cory

<Http://Jenkins>. Workflow examples

CLASSIC HARDENING

PRINCIPLE OF LEAST PRIVELEGDE

keep it simple

Apply patch

Layers of defense

Good logging/ audit trails

Encrypt data and networking protocols

DONT

Obscurity change port number

Attacker-generated firewall rules (fail2ban, etc.

Port knocking

Avoid networking software that doesn't support encryption etc TLS

Avoid complexity

SSH SERVER

Sshd\_config

Permittootlogin NO

Only use protocol 2

Disable password authentication (use ssh keys)

Limit crypto Options

Eliminate insecure ciphers

SSH CLIENT

 Eg

Strong key

Ssh - keychain -t rosa -b 4096

Ssh - key gem -t ed255

Avoid copying private keys

Use password protected SSH keys

Don't cache forever - ssh-add to cache password for limited time

Ssh -add -t 3h

Pay attention to host key warnings

SSH 2fa

Requires additional factor before login

Some use TOTP, others SMS/Phone, or both

Goggle has wide support of Duo

Install google authenticator Debian Libyan-google-authenticate or from source.

Etc. get slide

ROOT AND SUDO

DISABLE root/group accounts and use Sudo

Restrict NOPASSWORD Sudo to daemon role accounts

REUSE PUPPET CERT

If you use puppet masters, you have internal trusted CA

CLOUD HARDENING

delete default admin account

Don't store secrets in user data script

Try to generate the secrets on the host

Limit access even within security groups

Encrypt internal communication

Store sensitive data on a non-root, disk

Use config management with con figs checked into source control

Encrypt secrets checked into source control

Use orchestration software

Use /Devyn/sum to store sensitive files

Consider logging all new network connections

Set up remote logging

Ssh into internal servers via bastion host

Restrict access to your networks VPN

Enable TLS between web services

Cloud lock down IPv6 if not used

[Kyle@getfinal.com](mailto:Kyle@getfinal.com)

@kylerankin

[Greenfly.org/talks/security/simple\_hardening.html](http://Greenfly.org/talks/security/simple_hardening.html)

ANSIBLE - one click deployment

Application requirements

infrastructure

Configuration management requirements

ANSIBLE

Cloud provisional ng

IaaS orchestration - add /remove load balancer

Configuration manager

Valance

Hack admin credentials-

Targeted spear phishing-admin cred

Winrar break up data

Several instances of ps exec

Find installed tools

Ps kill to reboot systems

Log tool exe use

Malware use

Old

1-Diagnose

File and is system audit

Assess

2-Host memory analysis

Network Los audit

3-Collect/ all end points

Hard disk forensics

Network forensics

Cylance

Backend data

Looking for:

Don't deploy another agent

Don't ignore other information-splunk, siem, av, aim, nics

Need time stamps

10pm and 5am

Collect-file system objects

System services and settings

Schedules and tasks

Parse with parsers,sql server, forensic tools

Encase, Ftk

Apply smart methods

Develop timeline and determine additional

1?data loss and sabotage winwar Mozilla in the right context

2User profile propagation - time stamp

3Lateral movement - correlated user profiles and

4 Malware and iOS

5 Build and application inconsistencies

Nation state or hacktivismhackwrs

Droppers and downloaded - phishing & water holing, malware in users pace zero single-day exploits that lead to...

Malware as a service

-Blackdoor Trojan rats

-

-

 Old xp based systems

Oss have back doors

Social engineering sys admin

Apt

Most common seen x4 see pics

Role base access violations

Malware identifiers

I'd files that are running as system

Find and contain the malware

Book - hacking exposed same technics smaller timeframe

Carbon black

# Archived:

Sql saturday

## ##Mdm issues

Challenges - politics, governance ownership

3 months to one year to build a golden record

Build a single source dw system first

Customer analysis

Top customers

Receivables

Sales- by industry, creditworthiness, can't many size, type of company, etc.

I'd opportunities based on attributes of existing customers

Determine credit line

Cleaning process

Address cleansing

Fill data gaps - missing, incomplete

De-duplicating/match - need a unique identifier;-fuzzy grouping not 100% correct// also use dqs

Grouping by- industry, geographical

Unique ids

Customers- ssn,tax I'd

Company - tax I'd, duns #,

(Use cccm

Provide matching confidence

Enrich data

Write sssis component to auto mate clean

data

1People-Melissa data, email, phone, demographics, geography, social media

2.Company-Melissa data-match and append services; duns and broadsheet / all companies in the world financial

## ##Streaming analytics

Event hub (in only-data queuing) subset of service bus (in and out)

Streaming analytics - > aggregate and push to Powerbi real-time

Service bus - messages

Resource group

Use-case/ exception handling, alerts

 - equipment / vehicle me motoring and analysis

- energy

- medical device

- manufacturing

- call centers

Business cases from sys archit.

Saql value:

Temporal and windowing functions (hopping- overlap; slidding window for the last x minutes )

Sql job has the same group by eg window functions

Select

Into  eg output eg warehouses  wsPowerbi - old azure other options

From

Group

Nosql collections, nested data, fast reads

Sql

## ##Optimize SQL Server enterprise data warehouse for pulling data?

Use surrogate keys

Parallel thread adjustment

Avoid explicit conversions

Physical and logica model alignment - pull from one source

Leverage hekaton - in memory use for hotspots

Enterprise features

Compression settings- fact table/page vs row level : page is best/ rebuild index with page compression 80-90% if data is numeric ; if at 80% cup don't compress.

Partitioning visual studio/gui or XMLA, Olap

- logical model partitioning/ analysis services- query

- partition alignment to physical dw - don't need indexing to support logical partitioning; etc./ partition range right for partitioning

OR shard tables for standard edition

Process add/ incremental processing/ XMLA query only

Use clustered index

Clustered index to ordered gate key

It satisfaction / shadow it

Db systems

Db analytics - kpi

Db renewable energy real estate system design

Review

How to build a cube

Bi and analytics examples

Data science notes and books

# Mobile

## Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Feature description | Testing | Technologies |
| Capturing users – | Capturing users – |  |  |
| Registered user- Login/Authentication (elevated use-storage, bandwidth)  -guest access (less resources) |  |  | Amazon Cognito |
|  |  |  |  |
| Authorize |  |  | Amazon Cognito |
| Sync user preferences across multiple devices |  |  | Amazon Cognito sync |
| Download and deliver media /content delivery |  |  | S3-transfer utility (object storage) |
| Analytics |  |  |  |
|  | (user behavior) – active users, engagement and stats | (injections, storage, etl , presentation) | Mobile analytics package |
|  | Send real-time events |  | Kinesis Recorder |
|  | Batch analytics |  |  |
|  | -Revenue |  |  |
|  | Real-time analytics  -A/B testing |  |  |
| Push notifications |  |  | SNS Mobile Push |
| Scalable back end storage |  |  | Dynamo NoSQL key value database (Dynamo db mapper) |
| **Design** |  |  |  |
| Mobile first |  |  |  |
| Personalized Experience |  |  |  |
| Engaging |  |  | Vibration/wearables ? |
|  |  |  |  |
|  |  |  |  |
| Prototyping |  |  | Google Experience Design  -artboard |
| **User** |  |  |  |
| Get the job done quickly |  |  |  |

## User experience design requirements

Research

Planning

Reiteration

Wireframes

test (outside into larger user community)

Usability – organization, flow, simplicity (high color-contrast, legible fonts)

Front end – new items /alert

## Accessiblity Requirements

Can’t rely on native features for accessibility features (older versions of phones)

User scenarios

-Learning disabilities/cognitive impairment

strobing/blinking/flashing can cause seizures;

low-distraction interface;

use in an upredictable environment (noise, glare, etc.,

-break up long blocks of text-dyslexia )

-Cognitive impairment /

-Deaf /hearing impairment

50% of 18 in US has an hearing impairment

visual cues for sounds

captions for video

transcripts for audio

-Dexterity/physical impairment

keyboard issue – carpel tunnel syndrome,

voice dictation

-Low vision, color blindness

magnification tools (buildt in to apple products)

color contrast of 4.5:1 for standard text – also for low vision /color contrast checker

problem color combinations

1 out of 12 men are color blind (85)

-Blind/screen meter

Jaws-windows

VoiceOver utility-iOS

## Mobile accessibility

Cross pltofrom/interoperable (Phone/Cordova)

Updates on phones

Best strategy: use W3C Mobile Accessabilty docuemnt

### Apple

VoiceOver utility

Text-to-speech

(accessibility menu)

UIKit Accessilbity API

Accessability guilde on apple developer.com (make accessible function)

### Android Talkback/Smartphones and embedded devices

(aware of adding accessilibty afterwards)

Text to speech

Gesture to command

Google “Accessibility Scanner” app

### Windows

Jaws – screen reader

### PhoneGap/Cordova

Accessibility API

### Testing (All phoses)

User testing: Not a developer

Any user-unfamiliar perspective/ and filaiilar perspective

Individual unit testing

Ipmrotant Level A WCAG checkpoints – 13 AA items (that require most human judgement)

Tests

* Onsitent navigation
  + Same menu on each page H!/Top page to secondary pages
* color contrast
  + 4 or 5 different color blindness
  + high contrasts for variables environments /eg. sun
  + snook.ca tool – shoot for WGA AA/triple AAA is more restrictive
  + juicystudio too
* Double size text
  + Make 200% using control + CMD to see
  + Allow for user customization /with disability software and reasons
* Logical structure
  + Headings/H1-one , to H6 (screen reader lists these out first), and not too many headings/use for structure only
  + labels/every field has a label /
  + Emphasize text using strong or EM tags. /text should be use for visual effects
* Visible focus (desktop)
  + Keyboard user tabs through the page
  + Button/cursor location should be obvious
* Captions (level A)
  + All synchronized video
  + Convey all sounds in captions
  + You tube audio quality
* Images
  + Must have Alt=”” attribute in the tag (Alt text)
    - Decorative – leave blank
    - Button (action for buttons)
    - Link
* links (3:1) if no underline

### Accessability specifications

accessibility presentations: lnicks.github.io/p/cftm-access

Dcp.ucla.edu

www.w3.org /WAI/EO/Drafts/eval/checks

wave.webaim – run apage through it

validators – [www.w3.org/QA/Tools#valiators](http://www.w3.org/QA/Tools#valiators)

Snook studio color contrast

Juicy studio color contrast

Accessability guidelines

[www.w3.org/TR/mobile-accessibility-mapping/](http://www.w3.org/TR/mobile-accessibility-mapping/)

Google relateses accessibility scanner app for andoid

Google releases accessability scanner app for iOS

Apple – speech

-Voice over (for OS and Aps) oney reader app eg.

-repuposing gestures

-size

-color

-reduce motion

Icons

Icon fonts-font awesome

Sketch

The noun project – royalty free icons

Google-material design

UI Kits – IoS, Google

Start with sample file

Protoypting

Cognito-AWS security best practices, 1)identity pool 2)Identites: authenticalte, unauthenticated; ): 3)IAM (roles/permissions-resource inheritance)

Services>cognito>

Create pool>

Guest users-enable ap option

Authentication/social media > register to get App ID

Role summary> (2 unauth/authenticate users)

Getting started>sample code

Program--Gettting credentials

Import statements – regions, credentials

Initiale d the amazon credentials provider

Security – get OAuth (social media) /Open ID– Access token> <cognito ID

(delete user id if user ap has been compromised@ add analytics)

## Technology Feature mapping

Cognito Sync

-import cognito package

- intialze /contructor

Datasets/1MB storage (20 MG per user)

dataset.synchronize(syncCallback); (run every time the ap starts/ conflict resolutions with datasets)

Cognito post processing

Push sync/datasets to all devices

Events – execute a backend functions eg. Verify guest acess

Streams – data changes sent to kenesis

S3 Connector

-upload, download, pause, auto, retries, resume, cancel

-instantiate and S3 client, and region

(pass in key, local file, bucket)

-Track transfer process (loading bar,100%)

Mobile Analytics

* + - * Out of the box metrics /standard
      * Access data (pay $)
      * Intiatize mobile analytics manager
      * Track custom events – how many times, etc.
        + Create event with, record, and submit event
* 100mm events can store free, $1 for every additional m

Push Notifications/SNS

* Sent to gateway (auto select Apple, google baidu, amaxon, windows)
* Sent to Topic:->> have multiple subscribers to a topic (scale push notifications)

Dynamo DB Connector

* Map client-side classes to amazon table
* Create getters and setters
* Build object, save book object to dynamo DB, update item and save object again, load antoher high score
* Dynamo db as a service for HIPPA

Mobile Hub>Mobile Hub (beta)

* Sampel app to anhance and leverage
* Test with AWS device form
* Monitor
* Resource tabl – add new features

Steps:

* Create mobile ap
* create project>
  + configure features: sign-in, data storage, etc.
    - sing-in
      * OAuth
      * Facebook for developers dashboard – get app ID
      * Management roles created (service rool, auth, unatho)
    - App analytics
    - User data storage
    - Push notifications
      * Google Cloud Platform(GCM Mesaging) – create a Sms for android devices (ap id/API key & project number/sender ID)
      * Apple
      * Create topics to bradcast
    - Store app/app content delivery /CLoudfront service
      * Single locations/Global CDn
    - Cloud logic/web services <-serverless
      * Enable lamda functions
      * Create lambda functions
        + Blupreints

Simple mobile backend

Eg. Write to dynamo db

Node, java, python

Zip .java, .py and upload

Select roles

* + - * + Can be triggered dynically

Event triggers

Load

Click

* + - * + print to console console.log(‘Loading function”)
        + Test

Configure input test event

JSON

{ device: Android

}

log output : everything that was passed, & memory used

* + Build/develop
    - download and simulate
      * user files :private (sing-in required) & public (anyone can see it)

sample ap UI

amaxonmobile – helper code /scaffolding

AWSConfiguration -> user agents created (IDs stored ); ARN-services created

* + add to existing ap
    - copy helper code folder to exister ap
    - update Gradle Build – add line
    - merge code
    - add persisions
    - adjust manifest
  + AWS device farm/test applciations
    - Point to .apk (android) to upload
    - Configure test //show devices that are compatible with
    - Select devices
    - Specifiy device state eg. NFC, GPS, Bluetooth, WiFi
    - Look for issues @ UI, device resource use (threads, CPUs’

AWS Mobile blog parse aps,

AWS Mobile services

AWS Mobile SDK /Xamrin

Mobile Developer guides – Unity, Ios, Android

Git hub sample code – github.com/1) aws services 2) amazon web services - lab type in :cognito /amazon web sources

/serverless web site

( List of service that are HIPA compliant – AWS HIPAA – what services /lambda is running on EC2 )

Set up account under school agreement/use school e-mail:

* + - * + Free tier-

5G storage

20,000 get request

2,000 put request

15GB data transfer out

Cloud watch alert-costs generated

Mobile ap –

Manage normal health

After event – heart attack

Prevent onset of new

Needs-

* Exercise
* Can’t open phone
* Food
* Bath
* Psycho/stimul

Features

* Mange by
  + Self
  + Family
* Log-in by 3rd party
* Share screen/notes – writable
* E-mail access
* Alerts to self/or family or /provider
* Pain – feel good(milk), doc provided/home remedies /api

## Mobile Design Tools

Google material design (Photoshop document for download)

Ios Human Interface Guidelines

Mobile-patterns.com

Brad Frost – Atomic Design and Responsive Patterns

WIreframing and prototyping

Adobe experience Design CC moqups.com

Adobe Illustrator CC balsamiq.com

Adobe Photoshop CC // art board

## SDK

AWS Mobile SDK – Unity (C#, multiple platforms), Xamrin

Native – lower level abstractions

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

# Physical Connected Devices

Connected scale Blood pressure

Bluecose monitor

Sp02 Test – oxygen saturation

Resmed – how well you sleep /Sonar

Dexcom G5 – glucose levels

Apple watch –

# Healthkit –

database/data integration

Health – dashboard (app to see the data) – body measurements, fitness, nutrition, results, sleep, Vitals

Encrypted at rest

Encypted backups

Write data out (Mayoclinic) or to app (Nike App)

Epic-mychart, integrated healthkit

Doctors can set up alerts

Remotely monitoring

Read objects

Write objects

# Research kit

Manage data

Wider research

Use sensors

List of devices ?

1.informed consent

2.Active Tasks -

3. Surveys -

create and verify accounts

secure app

ORKSample – every possible

raywenderlich.com (example)

command B – build

xcodeporject file

Embedded binaries <- ResearchkitFramework (add to ) -- Bring in research kit framework

Add consent document in swift

Add consent task

Workflow-task

View Controller

1. Draw a line to source code /function

New action /ObtainConsent

ViewController.swift

Import ResearchKit

Let taskToRun = ConsentTask

//Preper a view controller

Let raksViewContrleer

Let taskController = ORKTaskViewController

Task:taskToRun,

taskRun

??

Delegate method – to be called when a task view finishes,

??

1. Survey

Add button, create empty function

Add surveytask.swift to directory

Let taksToRun = SurveyTask

#call delegate methods – here is the data

print(“Results: “,taskViewController.result) --optional



Deploy

Quicktime recording – attach phone to computer

@top screen change simulator to actual device @ top of windows (Demo)

ORK Catalog-download open and build

ORK Sample-download and build

Charts

Download healthkit app

Capabilities menu in main project folder

Add healthkit

Import healthkit

Read and write data (Create a set of attributes0

Get healthauthorization

Call function to access the data

Then read data

<data is stored in abstract numbers>

formatters-convert abstract quantity

data – ns time (systems routines)

Save UTC on server

hk

factor screens to different languages: English, Spanish, german …..

**Design Buildines for Researchkit**

**Add a watch kit extension**

**Researckit App has to be IRB approved**

**Healthkit App doesn’t have to be approved**

* + - * **there are rules x10**

**Test-flight**

**Enterprise – private distribute aps to enterprises – devices that you control**

# Carekit

Take an active role in the health care

Algorithms – real-data

Systematic development process/engineering (great products)

Computer science /programming (Fortran)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | measure |  |
|  |  |  |  |  |
| Physical |  |  |  |  |
|  |  |  |  |  |
|  |  | Move |  |  |
|  |  | Stand |  |  |
|  |  | Exercise |  |  |
|  |  |  |  |  |
| Diet |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Biological |  |  |  |  |
|  | Weight |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Biological (Critical) |  |  |  |  |
|  | Diabetes |  |  |  |
|  | Heat Desesase |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  | Parkinson’s | Not curable |  | Delay onset |
|  | Epilepsy |  | Detection |  |
|  |  |  |  |  |
| Disabilities |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| Disabilitating |  |  |  |  |
|  | Motor skills |  |  |  |
|  |  |  |  |  |

# Carekit

# IoS Tutorials

<https://www.raywenderlich.com/category/ios>

# Respsonsive Modern Web – Mobile+]]

Espressive semantics – human and machine comprehensive

Responsive to human needs – change, shift without reloading

Rick components

|  |  |  |
| --- | --- | --- |
| **URL Routing** |  |  |
| **Templating** |  |  |
| **MVC** |  |  |
| **Laocalizational Data** |  |  |
| **App Logic** |  |  |
| **Caching** |  |  |
| **CSS Styling** |  |  |
| **CSS Transitions** |  |  |
| **Dyanic Image Generation** |  |  |
| **App Logic** |  |  |
| **App Data** |  |  |

|  |  |  |
| --- | --- | --- |
| **Web APIs** |  |  |
| **Access Control** |  |  |
| **Service Logic** |  |  |
| **Orchestration** |  |  |
| **Service data** |  |  |
| **Service calls** |  |  |

**Building the modern web - eb.io/ghp/p-building-modern-web**

**Netbeans, code (ex folder)**

**Postgres – relational and JSON datatypes, and bind to redis**

**Twitter bootstrap responsive grid**

**Rich interface components**

**Responsive Web Design – fully-cable browsers (Ethan marquet)**

1. Flexible grid – target /context =result: Change widths to percentages (%), 960/1024=93.75%
   1. Issues small screen, constricts content
2. Media queries
   1. @media (min-width: 481px //Mobile first
   2. @media min-resolutions
   3. max/min height
   4. max aspect ratio
   5. @orientation: landscape
   6. @media (light-level) CSS4
   7. @media (pointer/hover ..)
   8. @supports (display: flex) do you support flex?
   9. @media (in-width: 40. Em) em – scale by # of pixels
3. Responsive media - Image – max width =100%
4. Viewport tag –specify other than 960/i.e. number of pixels on devic
   1. <meta name> viewport …

Responsive interaction

1. scroll/
2. INDIEUI – input agnostic //not support yet

Battery status API

Black uses less color/users know which aps burn power

Network information API

Network sampling – network properites

Resource Timing API/user timing

Can figure out compute power (CPU, GPU?)

Storage availability

Web, ap cahce, Fileystem API, indexed database

Detect storage – to enable/disable featers

Partial loading (not in HTML, could use AJX, or iFrame)/HTML import (scripts/styles, )

Javascrip ES6 modules/module loader

Template/doesn’t get rendered/not in Dom tree (MUST WRAP in polyfills because not supported in the browsers.//may not be stable?

Shadow DOM – shadow root/point in tree that you cannot extend (MUST WRAP in polyfills because not supported in the browsers. .//may not be stable?

## Web Components

Webcomponets.org

Custeomelements.io

Examples github.com/ebollens/p-building-modern-web/tree/gh-pages/ex

Soap, REST

Flash&sliverlight

Javascript

Html5

Interactive features

Web componetns

Sematics should be : expressive, encapsulated, and reuseagle

1. Custom elements-create, attach, detach, detech attribute changes
   1. Register element(must have a hypen)
      1. Var XFoo =document.regiserElements(‘x-foo);
   2. Use withi DOM
   3. Instantiate the element
2. Extend an element ; must use I”is syntax <button is=”mega-button”> ; and style
   1. For convenience add setters and getters

2.HTML templates

# Deploy a algorithm in production

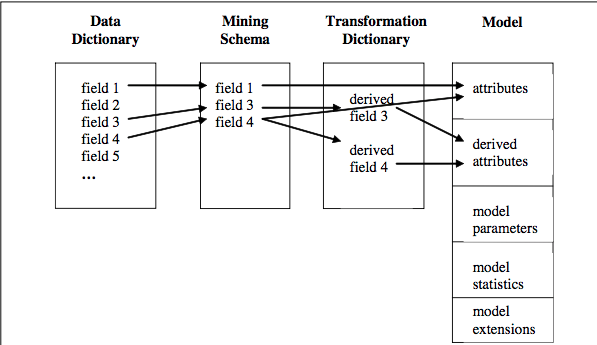
|  |  |
| --- | --- |
| **Data Scientist** | **Engineer** |
| R, SPSS, Python | Java, .NET, C, SQL |
|  |  |
|  |  |
|  |  |

Sometimes you might want to use a library that was originally written in C++ in your objective C app.

<https://www.raywenderlich.com/62989/introduction-c-ios-developers-part-1>

PMML – Lingua Franca

Pre-processing, post-processing, and predictive model



Azure ML - limited to datasets less than 10GB/recommender/SMOTE/scripting – R/python

deployed in operational systems where the models must be often be re-coded using C++, Java or other language.

<http://accord-framework.net> data, and audio video processing libraries completely written in C#. It is a complete framework for building production-grade computer vision, computer audition, signal processing and statistics applications even **for commercial use**.\

<https://github.com/accord-net/framework/wiki/Sample-applications>

converted a R model into PMML, using r2pmml.

pmml4net is a small library written in C# that provide usefull class to read and consume models in PMML file.

PMML to C++ source code translation using JPMML libraries. It's much-much more convenient than working with low-level DOM APIs.

You can use .NET XDocument or XmlDocument classes

You can also generate c# classes from your PMML XSD by using xsd.exe, which is bundled in Visual Studio.

Develop an application using UIMA/PML based serivces

KNIME

Deploy R

IBM’s Unstructured Information Management Architecture (UIMA) -unstructured data

PMML – interchange for structured data, first class object

|  |  |  |
| --- | --- | --- |
|  | Learning | Scoring/classifying |
|  | PMML producer  processes a data set to produce  a PMML model. | A PMML consum  er i) inputs a  PMML model and then ii) using the  model can process one or more data  records to produce scores |

input data (fileds used in the model), tuning parameters, derived data, model statistics (descriptive), tuning parameters (PMML 3.0 version 3.0 regression models, cluster models, trees, neural networks, Bayesian models, association rules, sequence models, support vector machines, rule sets, and text models.

UIMA – unstructured

UIMA/PMML –

<http://docplayer.net/5766885-Pmml-and-uima-based-frameworks-for-deploying-analytic-applications-and-services.html>

[JRI](http://rforge.net/JRI/) - run R inside Java applications as a single thread. Write Javascript UI code on the client side, and on the server side you can have some Java code wrapping around your R solutions.

Deploy models on RServer (Java) - <http://rforge.net/Rserve/doc.html>

1. **REngine Java client** - located in src/client/java-new is a full client suite that allows any *Java* application (JDK 1.4 or higher) to access an **Rserve**. The suite is written entirely in *Java*. It provides automatic type translation for most objects such as int,double,arrays,String or Vector and classes for special **R** objects such as RBool, RList etc. For sample code and a small tutorial see [**Examples**](http://rforge.net/Rserve/example.html). *JavaDoc* documentation of the **REngine**-classes is available online [**here**](http://rforge.net/Rserve/doc/).
2. Compared with previous Java clients (see bloew) this new client API is more flexible, with better design, has better exception handling and is aimed to support both JRI and Rserve transparently.
3. **JRclient** - located in src/client/java-old is an older Java API that was used in Rserve 0.4 and earlier. It is deprecated and neew project should use the new client above (based on REngine classes).
4. **C++ client** - it is located in src/client/cxx directory in the **Rserve** source package and provides basic interface to Rserve from any C++ program.
5. <li><b>Rcli.c</b> - a lightweight client that demonstrates how to connect to <b>Rserve</b> from C language. It does not go beyound a proof of concept, because R provides C API, so there is usually no need for a C client. (It is available in early Rserve versions, now replaced by the C++ client).<br>&nbsp;</li>
6. **R client** - a small client directly in the Rserve package. As of Rserve 0.5-0 is uses special, direct communication with Rserve. As of Rserve 1.7 it has been moved to a separate [**RSclient**](http://rforge.net/RSclient) pacakge.
7. There are several other clients written by 3rd parties:

* **Python: pyRserve** Python client - see [**http://pypi.python.org/pypi/pyRserve/**](http://pypi.python.org/pypi/pyRserve/)
* **.NET/CLI: RserveCLI** .NET/CLI client - see [**http://rservecli.codeplex.com/**](http://rservecli.codeplex.com/)
* **C#: RserveLink** C# client - see [**http://sourceforge.net/projects/rservelink/**](http://sourceforge.net/projects/rservelink/)
* **Ruby: rserve-client** Ruby client - see [**http://rubygems.org/gems/rserve-client**](http://rubygems.org/gems/rserve-client)

Standard Model –PMML

# Microsoft/Spark

@@ cross-validation/reduce bias and variance

cheer picking

random sampling and partitioning

stratified

k-fold

20\*80

@@

Driver program

Executor node (process in parallel)

Spark primitives = Vector (primary data type/normally double)

Primary data type=vector

Features = Vector

Vectors can be Sparse or Dense

|  |  |  |
| --- | --- | --- |
|  | python | scala |
| Read in file |  | val rdd = sc.textFile(“wasb:///iris.csv”) |
|  | input.count() |  |
| *Shape of data* | input.first()  *// Get the first line of data* |  |
| *Group by and Count #1*  Split by “,” response variable  Count each type of response variable | **val** splitter **=** input.map(line **=>** line.split(','))  **val** num **=** splitter.map(line **=>** (line(4), 1)).reduceByKey((a, b) **=>** a **+** b);  //reduce function to count |  |
| *Count* | **Num.take(3)** |  |
| *//*  *Create datatype class 🡪 Convert to dataframe* | **import** sqlContext.implicits.**\_**  **case** **class** Iris(SepalLength : Double, SepalWidth : Double, PetalLength : Double, PetalWidth : Double, Species : String)  **val** splitter **=** input.map(line **=>** line.split(',')).map{ line **=>**  Iris(line(0).toDouble, line(1).toDouble, line(2).toDouble, line(3).toDouble, line(4))  }  **val** df **=** splitter.toDF() |  |
| *Group by and Count #3*  *// Group by species* | df.groupBy("Species").count().show()  In [ ]:  // Register temp table to enable a query  df.registerTempTable("iris")  sqlContext.sql("SELECT Species, COUNT(Species) FROM iris GROUP BY Species").show()  x |  |
| *Group by and Count #3*  *// Register temp table to enable a query* | df.registerTempTable("iris")  sqlContext.sql("SELECT Species, COUNT(Species) FROM iris GROUP BY Species").show() |  |
|  |  |  |
| Modeling |  |  |
|  |  |  |
| Decision Tree |  |  |
| -train | val model = DecisionTree.trainClassifier(trianingData, numClasses, categoricalFeaturesInfo, impurity, maxDepth, maxBins) |  |
| -predict | val labelAndPreds = testData.map{ point=>val prediction =model.predict(point.features)(point.label, prediction)  } |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*// Convert to dataframe*

In [ ]:

In [ ]:

​

Audio analysis - Spectrogram ..Peak frequency, pitch, sound pressure, Fast Foureir Transform