[Training OneDrive](https://intsol291-my.sharepoint.com/:f:/g/personal/iomegalearning_intsol291_onmicrosoft_com/EgGhsOimVaJOsOYaAJX63U8BXNj-9RvFVunjmQ39cQFd1Q?e=fOpLoW)

[Data Generation](https://mockaroo.com/)

[Codebunk](https://codebunk.com/b/546337325/)

**Day #1**

Roadmap

* BI Trends
* Cloud tech. related DB, Data Analytics & AI Cognitive
* Spark – In memory processing
  + Data Bricks – Covers Spark Infrastructures and provides enterprise capabilities
    - * MLlib Graphx – Maintained by Spark Team
      * Python Scikit
      * Cognitive services – By Azure Engg. Team, exposed as services REST API
      * Data Lakes
      * Azure Data Factory
      * Visualization
    - Streaming features
      * Kafka
      * Event Hub
  + Languages
    - R
    - Python
    - Scala
* Awareness of Big Data & Hadoop tech.

BI – (BI + AI)

BI Stages

* Data Warehousing – Normalized DB store – We use ETL (Data movement – Extract > Transform > Load)
  + Data Mart – Context specific data warehouses
  + Schema First Approach – Schema is defined before data is dumped
  + RDBMs store
  + Expensive to maintain
  + Supports structured data
  + Data Lake Store – To move – ADF (Data Factory V2), Apache Nifi (Open Source)
    - Schema on Read – Data is dumped as is on write and Schema applied on Read
    - Typical file system (hierarchical)
    - Economical cost
    - Supports any form of data
* OLAP (Online Analytical Processing) Process – OLAP Databases for Analysis
  + Previously used MD (Multi-Dimensional) – Tabular Model
* Data Mining
* Analytics > OLAP Process + Data Mining
  + As opposed to Analysis, provides hidden insights and not just surface level truths.
  + Types:
    - Descriptive Analytics – What happened?
    - Diagnostic Analytics – Why happening?
    - Predictive Analytics – What will happen?
    - Prescriptive Analytics – Will happen
* Reporting – Aligned to previous generation
* Visualization Tools – Aligned to Analytics – e.g. Power BI, Spotfire
  + Data discovery
    - Stages
      * Explanatory graphs and charts
      * Exploratory dashboards
  + Data story telling
* ML
  + Deep Learning
  + NLP
* Cognitive Services – Automation – On Premises + Cloud

Azure Data Analytics

* Business Domains
  + Infrastructure related services – IT Professional
    - Compute related
    - Containerization
    - Networking
    - Storage
  + Operations – Developer + IT Professional
    - Identity management
    - Security
    - Management
    - Monitoring
  + Platform Services – Developer
    - Databases
    - Integration
    - Migration
    - Web
    - Mobile
    - Media
  + Data & Analytics - Data Professional
    - Databases
    - Analytics
    - ML – Data Scientist
    - Cognitive Services
  + Connected devices / Development
    - IoT – Internet of Things
    - IoE – Internet of Everything – Bring Analytics to where Data resides – Edge Computing
  + Other Domains
    - Block Chain
    - Device Twins
    - VR – Virtual Reality | AR – Augmented Reality | MR – Mixed Reality
    - Gaming Services
* Azure Messaging Services
  + Cloud Queue Storage
    - * Send/Receive messages without enterprise capabilities (e.g. Secured delivery, different protocol etc.)
      * Low cost
      * Simplified messaging
  + Service Bus – Enterprise messaging
    - * Type of messaging
        + Brokered messaging – Between applications SB acts as broker for TTL, Dead letter etc.

Queue

Topic – Content based Routing

* + - * + Relay messaging – Integration b/w on premises & cloud.

When application tries to access Databases On-Premises secured way. SB acts as a relay which in turn talks to an Application running On-premises interacting with the DB. Hence application is unaware of the DB details. E.g. when logic apps are required access On-Premises DB we install On-Premises Agent.

* + Event Hubs – Streamed messages – Kafka broker
  + IoT Hub
    - Streamed messaging wit Duplex channels
    - Device Management
  + Event Grid
    - Serverless Messaging
    - HTTPs
    - Rest Webhooks
    - Publish & Subscribe – If no subscribers message discarded
  + Notification Services – Not related to messaging directly
    - Notification Hub – Device independent way of notifications
    - Signal R Service – WebSocket serve hosting as a service

1. Event Hubs – Streamed messaging
2. Azure Stream Analytics – Streamed Processing
   * As in data comes into input sink, data is processed continuously and pushed result to output sink which can be another Event Hub for more processing.
   * It is a platform service as opposed to Apache Storm with Analytics which is Infrastructure service which is to be maintained.
   * Intermediate logic – Has to be in JavaScript.
   * Predictive Analytics Integration – Invoke ML capabilities
3. Azure ML Studio – Automated ML Studio
   * Has self-learning capabilities with new data.
4. Azure SQL Data Warehouse
5. Azure SQL Analysis Services
   * Tabular model DB hosting as a service
6. Power BI embedded – Power BI report rendering as a service
7. DLS Generation 2 (Data Lake Storage)
   * Enterprise storage for Analytics
8. Data Bricks – Spark in Memory Analytic engine as a platform service. For Spark direct usage we need to maintain infrastructure like VMs, Hadoop clusters.
9. HDInsight – Infrastructure insights
10. ADF V2 – Data flow management

Data Formats

* + Text file – Data ready for Presentation – No compression/partitioning etc.
  + Avro – Compressing in Binary for e.g., we use meta file to link to the file
  + Parquet file – Partitioning for parallel reading – Best format
  + ORC – Compression + Partitioning – But while reading, need to decompress hence good for Archival cases

Use Case #1 – Diagram in Note

* Event Hub Namespace, Storage Account, SQL DB, SQL Server, Stream Analytics Job
* In CMD Administrator set these environment var
  + set EventHubConnectionString=Endpoint=sb://casestudy1eventhub.servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=vIAhp24vv/gZdCt6hOWi0Mv3PUfy24IGV246Re/fe0U=
  + set EventHubPath=sensormessages
  + Then dotnet SensorTelemetrySenderApp.dll, to send messages
* Connect to SQL Server
  + sqlcmd -S casestudy1sqlsb.database.windows.net -U casestudy1sqladmin -P Test1234567 -d casestudy1sql -i SensorReferences.sql
* [ML – Github R Programs](https://github.com/iomegak12/fidelityrscripts/tree/master/Demonstrations)
* Programs in Code bunk



**Day #2**

Hadoop

* + - HDFS
    - Map Reduced Framework
  + Common technologies
    - Pig – Scripting language alternate to map reduced
    - Sqoop – Data Import Export to RDMS
    - Hive – SQL interface to HDFS data
    - HBase – Proper columnar format database
    - Flume – Provides data lake operations
    - Storm – Gives stream analytics
    - Solr – Crawling indexing operation – Design Environment
    - Elastic Search – Performance improvement (next Gen. Solr)
    - Ambari – Provides admin capabilities
    - Impala - Massive parallel processing databases
    - Mahaut - Adds ML programming capabilities for Predictive Analytics
  + Drawbacks
    - Disk I/O based processing
    - Storage & Processing Framework design is tightly coupled –

For batch processing operations since there is heavy I/O operations as data needs to be moved from HDFS to process and then again moving it back. – Map reduced operations HDFS and map reduced are tightly coupled. And modern Data lake stores need an adapter to work with Hadoop.

* + - Not designed to handle small to medium size scaled files
    - Batch processing job suitable and not for typical user interactive scenarios
    - Java language specific implementation
    - Platform OS dependent
    - Various and miscellaneous technologies required, increased operation cost – No unification of technology stack

**VS**

Spark

* + Lightweight – 70MB core
  + Open source
  + Platform independent
  + In-memory Analytic engine for ETL and Data Processing – Avoids Disk I/O
  + Without dependencies like HDFS & Hadoop – Flexibility & Extensibility
  + Unified tech stack for ML, ETL, Graph Processing, Querying
  + OOB Practices
  + User Interactive – BI & Analytics
  + Real Time Processing
  + RDD – Resilient Distributed Dataset (Distributed partitioned data table)
  + DAG – Execution Plan
  + Architecture
    - OS
    - JVM
    - Scala
    - Spark Core
      * RDD
      * DAG
    - Components
      * Data Frame – Type safe
      * Dataset – Pure Java Object, type safe, everything
      * Spark Streaming
        + File System
        + Socket Server – Opens a socket for listening
        + Kafka Streams
        + MLlib – Machine learning algorithms as Classes
        + SparkQL – SQL like interface to reach underlying storage

Has a meta store with Table definition, Views etc. pointing to underlying storage

* + - * + GraphX – Graph processing
  + Programming Decision chart
    - Dev -> RDD Programming Scala
    - Business Analyst -> SQL
    - Data Scientist -> R or Python
  + Drawbacks
    - No support for maintaining diagnostics information/job execution history.
    - No cluster management support out of the box. Clusters are kept running and jobs are submitted
    - No support for credentials managements like KeyVault.
    - Collaborative authoring not supported.
    - Does not provide any visualization or dashboarding of analyzed data
  + Architecture
    - Notepad

ADB

* Spark infrastructure as a platform service with enterprise capabilities
* Designed to leverage public CSP (AWS & Azure) features



**Day #3**

* DLS – Data Loke Store
  + File System like storage for warehousing purposes for heavy I/O applications/workloads
  + Gen2 = DLS Gen1 + Best of bob storage features and enhancements
  + Gen 1 Features:
    - Hierarchical namespace for storage
    - OAuth2 protocol for Authentication
    - Authorization – Granular control over access on namespace, folders and files
    - Azure data catalog integration for meta data catalog
    - Enterprise search integration
    - External tools | APIs | SDK support
    - Encryption at REST
  + Gen 2 Features:
    - Unified storage for any types of or any size of data.
    - Best blended features of Blob and Gen 1
    - CDM integration – Open standard
      * Meta data store
      * Standardization
      * Search
  + Gen 2 Missing Features:
    - No support for ADLA integration
    - PowerShell support for DLS Gen 2 data movement not supported
    - External visualization tools not supported connecting to storage
    - Explorer tool in browser not supported
  + Design Considerations:
    - Why should we use?
      * Innovative way of storing any type and size of data
      * Unification of technologies especially in storage
      * Speed
      * Self Service
      * Optimization of operations capital cost
    - Physical Storage of data
      * Exceptional scalability
      * High durability - Redundancy | Access type | Storage type
      * Independence from fixed schema
      * Support for structured/semi-structured/unstructured data storage
      * Separation of computing resources
    - Format of the file we want to dump
      * Independent of storage services – Same performance for any file size
      * Data Formats:
        + Raw data – readable/binary form (CSV, JSON, XML, TSV) – Suitable for visualization clients
        + Binary data – Avro, ORC, Parquet

Avro

Storage Format – Row based

Suitable For – Data Warehousing

Preferred operation – Write heavy

Type – Binary

Partition type – Not partitioned (Manual required)

Good for – Data ingestion of wide column & bulk records

Compression support - No

Parquet

Storage Format – Columnar based

Suitable for – Data Warehousing

Preferred operation – Read heavy & processing

Type – Binary

Partition Type – Partitioned

Good for – Data Analytics

Compression support – No

ORC

Storage Format – Columnar based

Suitable for – Data Warehousing

Preferred operation – Read heavy & processing

Type – Binary

Partition Type – Partitioned

Good for – Archiving

Compression support - Yes

* + - Option regarding governance
      * Policy definition w.r.t. defining meta data at the time of data ingestion
      * Data access policy in terms of read and write can be constrained
      * Meta data store security policy
    - Enterprise security aspects
      * Encryption @ REST inbuilt certificates, Azure Key Vault certificates
      * Network Access Security
        + Firewall
        + Secure VNet to be used to access DLS
        + ACL
    - Meta data cataloging and search features
      * CDM – Open standard for defining meta data as part of data store itself
        + Folder Structure

Storage A/C

Hierarchical File System

Business use case/unit

Folder

Use case

Model.json

Entity1

.CSV

Parquet

Entity2

.CSV

Entity3

JSON file

* + - * Azure search
      * Automation of meta data creation
      * Prioritize cloud native solutions to meta data itself
    - How to access and mine the lake
      * Schema on read
      * Data processing operations – Ready heavy operations
      * Data warehousing workloads – Write heavy operations
      * Data exploration & ML workflows
      * Data presentation & dashboarding