Microsoft Azure for Data Engineering Notes

(Based on Microsoft Azure Data Engineering Associate (DP-203) Coursera Course)

Date: 08/20/2022

Azure Data Engineer and DP-203 exam:

- **What does Microsoft Azure Data Engineer do?**
    - Azure Data Engineer will integrate, transform, and consolidate data from many data systems into structures that are used to build analytic solutions. You'll ensure that data pipelines and stores are high-performing, efficient, organized, and reliable.
- **Who will be the ideal candidate to pursue the DP-203 certificate?**
    - If you are already familiar with data processing languages such as SQL, Python, or Scala, and understand parallel processing and data architecture patterns, then, you are the ideal candidate for the exam DP-203: Data Engineering on Microsoft Azure.
- **Why Microsoft DP-203 certification?**
    - Your Microsoft certification will be recognized globally by industries, colleagues, and employers. It is a certification that will lead to your new role as a data engineer.

- **Why DP-203 for Data Engineer?**
    - Responsibilities for the role of Data Engineer include helping stakeholders understand data through expiration and building and maintaining secure and compliant data processing pipelines by using different tools and techniques.
    - You'll use various Azure data services in languages to store and produce cleansed and enhanced datasets for analysis.
    - The world of data continues to evolve, and the advent of Cloud technologies has provided new opportunities for businesses to explore.
    - This exam helps you to learn explore:
        i. The Data Platform technologies that are available today and how a data engineer can take advantage of those technologies to an organization's benefit.
        ii. How data systems are evolving, and how the changes affect data professionals.
        iii. The differences between On-premises and Cloud Data Solutions and consider sample business cases that apply Cloud technologies.
        iv. The responsibilities of a Data Engineer.
        v. How do they relate to the jobs of other Data and AI Professionals?

vi. Common data engineering practices and a high-level architecting process for a data engineering project.

vii. About Azure technologies that analyze text and images, and relational, non-relational, and streaming data.

viii. How can Data Engineers choose the technologies that meet their business needs and scale to meet demand securely?

- **How Microsoft Learning path assists you?**
  - Microsoft gives you the ability to take a specific certification path that matches your career aspirations, while at the same time improving your skills.
  - There are different certification paths for you to choose depending on your expertise and background.
  - The certifications are structured into three expertise levels.
    i. The fundamentals certifications are targeted toward those just starting out with the technologies covered or looking to change careers.
    ii. This associate certification is targeted toward professionals that already have at least two years of practical experience working with the technologies covered.
    iii. The expert certifications are targeted toward professionals that have a minimum of five years of advanced practical experience and skills with the technologies covered.

DP-203 Skills measured

Core Microsoft Azure data platform technologies

Data Types:

Azure provides many Data Platform technologies to meet the needs of Common Data varieties. Let's take a brief review of the two broad types of data:

- Structured.
- Unstructured.

Structured Data Type:

In relational database systems like Microsoft SQL Server, Azure SQL Database, and Azure SQL Data Warehouse, Data Structure is defined at design time. The data structure is designed in the form of tables. This means it's designed before any information is loaded into the system. The Data Structure includes the relational model, table structure, column width, and data types. Relational systems react slowly to changes in data requirements because the structural database needs to change every time a data requirement changes. When new columns are added, you might need to bulk update all existing records to populate the new columns throughout the table. Relational systems typically use a querying language such as transact SQL and T-SQL.

Unstructured Data Type:

Unstructured data is stored in non-relational systems, commonly called unstructured or no SQL systems. Examples of unstructured data include binary, audio, and image files. In non-relational systems, the data structure isn't defined at design time and data is typically loaded in its raw format. The data structure is defined only when the data is read. The difference in the definition point gives you the flexibility to use the same source data for different outputs. Non-relational systems can also support semi-structured data such as JSON file formats. The open-source world offers four types of NoSQL databases. They are key-value store, which stores key-value pairs of data in a table structure. Document database stores documents that are tied with metadata to a document search. Graph database finds relationships between data points by using a structure that's composed of vertices and edges. Column database stores data based on columns rather than rows. Columns can be defined at the queries runtime allowing flexibility in the data that's returned performance. Now that you've reviewed data types, the next step is to look at common data platform technologies that facilitate the storage, processing, and querying of these data types.

Data storage in Azure Storage:

Azure storage accounts are the base storage type within Azure. Azure storage offers a very scalable object store for data objects and file system services in the cloud. It can also provide a messaging store for reliable messaging, or it can act as a no sequel store.

Azure storage offers four configuration options.

- Azure blob is a scalable object store for text and binary data.
- Azure files, manage file shares for cloud or on-premises deployments.
- Azure queue, a messaging store for reliable messaging between application components.
- Azure table and no sequel store for no schema storage of structured data.

You can use Azure storage as the storage basis when you are provisioning a data platform technology such as Azure Data Lake storage and HD insight. But you can also provision Azure storage for

standalone use, for example, you provision an Azure blob store either as standard storage in the form of magnetic disk storage or as premium storage in the form of solid-state drives, SSDs.

- If you need to provision a data store that will store but not query data your cheapest option is to set up a storage account as a blob store. Blob storage works well with images and unstructured data, and it's the cheapest way to store data in Azure. It also provides rest API and SDK for Azure storage in various languages. And supported code languages include .net, java, node, python PHP ruby and go.
- Azure storage also supports scripting in Azure Power Shell and in the Azure command line interface. Data ingestion, to ingest data into your system use Azure Data factory storage Explorer, the AzCopy tool, Power Shell, or visual studio.
- If you use the file upload feature to import file sizes above two gigabytes, use Power Shell or Visual Studio. AzCopy supports a maximum file size of one terabyte and automatically splits Data files that exceed 200 GB.
- If you create a storage account as a blob store, you can't query the data directly. To directly query the data, either move the data to a store that supports queries or set up the Azure storage account for a data lake storage.
- Azure storage encrypts all data that are written to it. Azure storage also provides you with fine-grain control over who has access to your data. You'll secure the data by using keys or shared access signatures.
- Azure resource manager provides a permissions model that uses role-based access control, RBAC. Use this function to set permissions and assign roles to user groups or applications.

Data storage in Azure Data Lake Storage:

Azure Data Lake Storage is a Hadoop-compatible data repository that can store any size or type of data. This storage service is available as Generation 1, Gen 1, or Generation 2, Gen 2.

Where to use Data Lake Storage Gen2?

Data Lake Storage is designed to store massive amounts of data for big data analytics. As an example, consider Contoso Life Sciences, Accounts, and Research Center, which analyzes petabytes of genetic data, patient data, and records of related sample data. Data Lake Storage Gen 2 reduces computation times, making the research faster and less expensive. The compute aspect that sits above this storage can vary. The aspect can include platforms like HDInsight, Hadoop, and Azure Databricks.

The key features of Data Lake Storage:

Unlimited scalability, Hadoop compatibility, security support for access-control lists or ACLs, POSIX compliance and optimized Azure Blob Filesystem, ABFS, the driver that's designed for big data analytics, zone redundant storage, and geo-redundant storage.

Data Ingestion:

To ingest data into your data lake system, use Azure Data Factory, Apache Sqoop, Azure Storage Explorer, the AzCopy tool, PowerShell, or Visual Studio. To use the File Upload feature or to import file sizes above two gigabytes, use PowerShell or Visual Studio. AzCopy supports a maximum file size of one terabyte and automatically splits data files that exceed 200 gigabytes.

In Data Lake Storage Gen 1, data engineers query data by using the U-SQL language. In Gen 2, use the Azure Blob Storage API or the Azure Data Lake System, ADLS, API. Because Data Lake Storage supports Azure Active Directory ACLs, security administrators can control data access by using the familiar active directory security groups. Role-Based Access Control, or RBAC, is available both in Gen 1 and Gen 2.

Built-in security groups include read-only users, right access users, and full access, users. Enable the firewall to limit traffic to only Azure services. Data Lake Storage automatically encrypts data at rest, protecting data privacy.

Azure Cosmos DB:

Azure Cosmos DB is a globally distributed multi-model database. You can deploy it by using several API models like Sequel API, MongoDB API, Cassandra API, Gremlin AP, and Table API.

Because of the multi-model architecture of Azure Cosmos DB, you benefit from each model's inherent capabilities. For example, you can use MongoDB for semi-structured data, Cassandra for white columns, or Gremlin for graph databases. When you move your data from the sequel, MongoDB or Cassandra to Azure Cosmos DB. Applications that are built using the sequel MongoDB or Cassandra APIs will continue to operate.

When to use Azure Cosmos DB?

Deploy Azure Cosmos DB when you need a no sequel database of the supported API model at planet scale and with low latency performance. Currently, Azure Cosmos DB supports five nine uptimes, which is 99.999%. It can support response times below ten milliseconds when it's correctly provisioned.

Consider this example where Azure Cosmos DB helps resolve a business problem, Konta which is an e-commerce retailer based in Manchester UK. The company sells children's toys after reviewing power bi reports, Contessa's managers notice a significant decrease in sales in Australia. Managers review customer service cases in dynamics 365 and see many Australian customer complaints that their sights shopping cart is timing out. Contessa's network operations manager confirms the problem. The issue is that the company's only data center is located in London. The physical distance from Great Britain to Australia is causing delays. Contessa applies a solution that uses the Microsoft Australia East Datacenter to provide a local version of the data to users in Australia. Contessa migrates their on-premises sequel database to Azure Cosmos DB by using the sequel API this solution improves performance for Australian users. The data can be stored in the UK and replicated in Australia to improve throughput times.

The key features of Azure Cosmos DB:

- It supports 99.999% uptime. You can invoke a regional failover by using programming or the Azure portal an Azure Cosmos DB database will automatically failover if there's a regional disaster.
- The multi-master replication in Azure Cosmos DB Can often achieve a response time of less than one second from anywhere in the world. Azure Cosmos DB is guaranteed to achieve a response time of fewer than 10 milliseconds for reads and writes.
- To maintain the consistency of the data in Azure Cosmos DB, your engineering team should introduce a new set of consistency levels that address the unique challenges of planet-scale solutions. Consistency levels include strongly bonded stillness, session, consistent prefix, and eventual.

Data Ingestion: To ingest data into Azure Cosmos DB using the Azure Data factory, create an application that writes data into Azure Cosmos DB through its API upload Jason documents, or directly edit the document. As a data engineer, you can create stored procedures, triggers, and user-defined functions UDFs or use the JavaScript query API.

You'll also find other methods to query the other APIs within Azure Cosmos DB, for example, in the Data Explorer component, you can use the graph visualization pane. Azure Cosmos DB supports data encryption type, firewall configurations, and access from virtual networks. Data is encrypted automatically user authentication is based on tokens, and Azure Active Directory provides role-based security. Azure Cosmos DB meets many security compliance certifications, including hip hop, fed ramp, SOCS, and high trust.

Azure SQL Database:

Azure SQL Database is a managed relational database service. It supports structures such as relational data and unstructured formats such as spatial and XML data. SQL database provides online transaction processing, OLTP, that can scale on demand. You'll also find comprehensive security and availability in Azure Database Services.

Azure SQL Database is the PaaS database offering. The local Microsoft SQL Server or those within an Azure Virtual Machine, VM are similar, the configuration of benefits of Microsoft SQL Server differ from those of Azure SQL Database. Use SQL database when you need to scale up and scale down OLTP systems on-demand. SQL database is a good solution when your organization wants to take advantage of Azure security and availability features. Organizations that choose SQL databases also avoid the risks of capital expenditures and of increasing operational spending on complex on-premises systems. SQL database can be more flexible than an on-premises SQL server solution because you can provision and configure it in minutes. Even more, the SQL database is backed up by the Azure service level agreement, SLA.

The key features of the Azure SQL Database:

- SQL database delivers predictable performance for multiple resource types, service tiers, and compute sizes.
- Requiring almost no administration, it provides dynamic scalability with no downtime,
- Built-in intelligent optimization, global scalability, and availability
- Advanced security options.

With these capabilities of Azure SQL Database, focusing on rapid app development and on speeding up your time to market, you do not have to devote precious time and resources to managing virtual machines and infrastructure.

SQL database can ingest data through application integration from a wide range of developer STKs. Allied programming languages include.Net, Python, Java, and Node.js. Beyond applications, you can also ingest data through Transact-SQL, T-SQL, techniques, and from the movement of data using Azure Data Factory.

Use T-SQL to query the contents of a SQL database. This method benefits from a wide range of standard SQL features to filter, order, and project the data into the form you need.

Azure SQL Database helps your application meet security and compliance requirements with a range of built-in features:

- Advanced threat protection,
- SQL database auditing,
- Data encryption,
- Azure Active Directory authentication,
- Multi-Factor authentication, and
- Compliance certification.

KNOWLEDGE CHECK:

S.No	Questions	Answers
1.	**In relational database systems like Microsoft SQL Server, Azure SQL Database, and Azure SQL Data Warehouse.**	**Structured Data is used as the data structure.**
2.	**Nonstructured data is stored in nonrelational systems, commonly called unstructured or NoSQL systems. List the types of NoSQL databases.**	**Column Store Database, Key-value store, a Graph database.**
3.	**Which is a scalable object store for text**	**Azure Blob.**

	and binary data and is the cheapest option for this type of storage?	
4.	**Azure Data Lake Storage Gen 2 is designed to store massive amounts of data for big-data analytics. List the features of Azure Data Lake.**	**Unlimited scalability, Security support for access control lists (ACLs), and Hadoop compatibility.**
5.	**Which is a globally distributed, multi-model database that can offer sub-second query performance?**	**Azure Cosmos DB.**

Technologies that support the core data platform technologies:

Azure Synapse Analytics:

Azure Synapse Analytics is a Cloud-based data platform that brings together enterprise data warehousing and big data analytics. It can also process massive amounts of data and answer complex business questions on a limitless scale.

Some of the common use cases of Azure Synapse Analytics:

- Data loads can increase the processing time for on-premises data warehousing descriptive analytic solutions. Organizations that face this issue might look to a Cloud-based alternative to reduce processing time and release business intelligence reports faster. But many organizations first consider scaling up on-premises servers.
- As this approach reaches its physical limits, they look for a solution on a petabyte scale that doesn't involve complex installations and configurations. The SQL pool capability of Azure Synapse Analytics can meet this need. The volume and variety of data that is being generated provide opportunities to perform different types of analysis on the data. This can include techniques such as exploratory data analysis to identify initial patterns or meanings in the data. It can also include conducting predictive analytics for forecasting or segmenting data. The big data analytics capability of Azure Synapse Analytics will accommodate this.

  The key features of Azure Synapse Analytics:

  - SQL pools uses massively parallel processing or MPP to quickly run queries across petabytes of data. Because the storage is separated from the compute nodes, you can scale the compute nodes independently to meet any demand at any time.

- In Azure Synapse Analytics, the Data Movement Service or DMS, coordinates and transports data between compute nodes as necessary. But you can use a replicated table to reduce data movement and improve performance.
- Azure Synapse Analytics supports three types of distributed tables: hash, round-robin, and replicated. Use these tables to tune performance.
- Most importantly, Azure Synapse Analytics can also pause and resume the compute layer. This means you pay only for the computation you use. This capability is useful in data warehousing.

Azure Synapse Analytics uses the Extract, Load, and Transform or ELT approach for bulk data.

SQL Professionals are already familiar with bulk copy tools such as BCP and the SQL bulk copy API. Data engineers who work with Azure Synapse Analytics will soon learn how quickly PolyBase can load data.

PolyBase is a technology that removes complexity for data engineers. They take advantage of techniques for big data ingestion and processing by offloading complex calculations to the Cloud.

Finally, developers use PolyBase to apply stored procedures, labels, views, and SQL to their applications. You can also use Azure Data Factory to ingest and process data using the PolyBase tool.

Queries:

As a data engineer, you can use the familiar transact SQL to query the contents of Azure Synapse Analytics. This method takes advantage of a wide range of features, including the where, order by, and group by clauses. You can load data fast by using PolyBase with additional transact SQL constructs such as create table and select.

Azure Synapse Analytics supports both SQL Server Authentication and Azure Active Directory. For high-security environments, set up multifactor authentication. From a data perspective, Azure Synapse Analytics Support security at the level of both columns and rows.

Azure Stream Analytics:

Data streams: Applications, sensors, monitoring devices, and gateways broadcast continuous event data known as data streams. Streaming data is high volume and has a lighter payload than non-streaming systems. Data engineers use Azure stream analytics to process streaming data and respond to data anomalies in real-time. And take note that you can use stream analytics for the internet of things or IoT monitoring, Weblogs, remote patient monitoring, and point of sale. Also known as POS systems.

Using streaming analytics: If your organization must respond to data events in real-time or analyze large batches of data in a continuous time band stream, stream analytics is a good solution. In real-time, data is ingested from applications or IoT devices and gateways into an event hub or IoT hub. The event hub or IoT hub then streams the data into stream analytics for real-time analysis, then visualization products such as real-time dashboards in Power BI can be used for analysis. Don't use batch systems for business intelligence systems that can't tolerate the predefined interval. For example, an autonomous vehicle can't wait for a batch system to adjust its driving and similarly, a fraud detection system must decline a questionable financial transaction in real-time.

Ingesting data: As a data engineer set up data ingestion in stream analytics by configuring data inputs from first-class integration sources. These sources include Azure event hubs. Azure IoT hub and Azure Blob storage.

An IoT hub is the cloud gateway that connects IoT devices. IoT hubs gather data to drive business insights and automation. Features in the Azure IoT hub enrich the relationship between your devices and your back-end systems. And bidirectional communication capabilities mean that while you receive data from devices you can also send commands and policies back to devices.

Key advantages include:

- to update properties or
- invoke device management actions
- take note that the Azure IoT hub can also authenticate access between the IoT device and the IoT hub.

Azure event hubs provide big data streaming service. It's designed for high data throughput allowing customers to send billions of requests per day. And event hubs use a partitioned consumer model to scale out the Azure data stream. This service is integrated into the big data and analytics services of Azure. These include Databricks, Stream Analytics, Azure Data Lake Storage, and HDInsight. Event hubs provide authentication through a shared key. You can use Azure storage to store data before you process it in batches. To process streaming data set upstream analytics jobs with input and output pipelines. Inputs are provided by event hubs, IoT hubs, and Azure storage. Stream analytics can root job output to many storage systems. These systems include Azure Blob, Azure SQL database, Azure Data Lake Storage, and Azure Cosmos DB. After storing the data, run batch analytics in Azure HDInsight or send the output to a service like event hubs for consumption. And use the Power BI streaming API to send the output to Power BI for real-time visualization. To define job transformations, use a simple declarative stream analytics query language. The language should let you use simple SQL constructs to write complex temporal queries and analytics. And the stream analytics query language is consistent with the SQL language. If you're familiar with the SQL language, you can start creating jobs. Stream analytics handles security at the transport layer between the device and the Azure IoT hub. Streaming data is generally discarded after the windowing operations finish. Event hubs uses a shared key to secure the data transfer. Finally, if you want to store the data, your storage device will provide security.

Azure HDInsight:

Azure HDInsight is a low-cost Cloud solution that provides technologies to help you ingest, process, and analyze big data. It supports batch processing, data warehousing, IoT and data science. It includes Apache Hadoop, Spark, HBase, Kafka, Storm, and Interactive Query.

Hadoop includes Apache Hive, HBase, Spark, and Kafka. Hadoop stores data in a file system or HDFS, and Spark stores data in memory.

The difference in storage makes **Spark** about 100 times faster.

HBase is a NoSQL database built on Hadoop. It's commonly used for search engines. HBase offers automatic failover, and Kafka is an open-source platform that's used to compose data pipelines. It offers message queue functionality which allows users to publish or subscribe to real-time data streams.

Storm is a distributed real-time streamlining analytic solution. It supports common programming languages like Java, C-sharp and Python.

Finally, **Interactive Queries** allow you to query the state of your stream processing application without needing to materialize that state to external databases or storage.

Data engineers use hive to run ETL operations on the data you're ingesting or orchestrate hive queries in Azure Data Factory. In Hadoop, you use Java and Python to process big data. Mapper consumes and analyzes input data. It then emits tuples that reducer can analyze. Finally, reducer runs summary operations to create a smaller combined result set. Spark processes streams by using Spark Streaming. For machine learning, it uses the 200 pre-loaded Anaconda libraries with Python. It uses GraphX for graph computations.

Developers can remotely submit and monitor jobs from Spark. Storm supports common programming languages like Java, C-sharp and Python. For running queries, Hadoop supports Pig and HiveQL languages and in Spark, data engineers use Spark SQL. For security Hadoop supports encryption, Secure Shell or SSH, shared access signatures and Azure Active Directory security.