

## White Paper / Azure Data Platform: Ingest

### Contents

White Paper / Azure Data Platform: Ingest .....	1
Versioning .....	2
Meta Data .....	2
Foreword.....	3
Prerequisites .....	3
Azure Data Platform.....	4
Flowchart Guidance .....	5
Flowchart .....	6
Appendix 1: Next Steps and Considerations.....	7
Maximum Size.....	7
Cost .....	7
Ingest Speed.....	8
Data Model .....	8
Processing Type .....	8
Coupled or Decoupled data system.....	8
Public Access.....	9

## Versioning

1.0: 2017-07-13 – First Version

1.1: 2017-07-18 – Iteration based on Feedback

1.2: 2017-09-06 – Removal of port opening for SQL Server – this is not possible. Removal of swim lanes. Iteration based on feedback.

## Meta Data

*Length:* 2000 words or less.

*Audience:* Data Professionals.

*Purpose:* Fill in an informational/thought leadership gap around choosing how to ingest data (particularly with Azure Data Factory) onto the Azure Data Platform.

*Author:* Phil Harvey. CSA-P for Data Platform, Analytics, ML/AI and IoT. OCP UK. Microsoft.

## Foreword

The Azure data platform (Both Cortana Intelligence Suite and IoT Suite) provides tools for handling a wide range of traditional data warehouse, modern data warehouse, advanced analytics and machine learning scenarios. For this reason, it is an ideal choice for modernising your data operations in the cloud.

As with any other data work, be that full scale ETL or something personal in Excel or PowerBI, the first step is to get hold of the data. This can be a simple one off process for an experiment or as complex as a full platform migration. In either case, as you work through what is required you will need to make decisions about different data sets and how to access them.

The Azure Data Platform provides tools for many common scenarios. This document hopes to guide you through the decision-making process to choose the right tools for the job.

## Prerequisites

This document assumes that you know where your data is. Especially the data you wish to use in a currently defined piece of work. You will need an understanding of the types of system that contain the data and can answer questions about the possibilities for interacting with the wider system that contains those sources. At the very least you know who you can ask questions related to data sources and the wider system.

If you are at the stage of defining the scope of the work and are looking for a place to start, then you could consider a scoping process called 'Data Landscaping' (<http://online-behavior.com/analytics/data-landscape>) to help define the work.

## Azure Data Platform

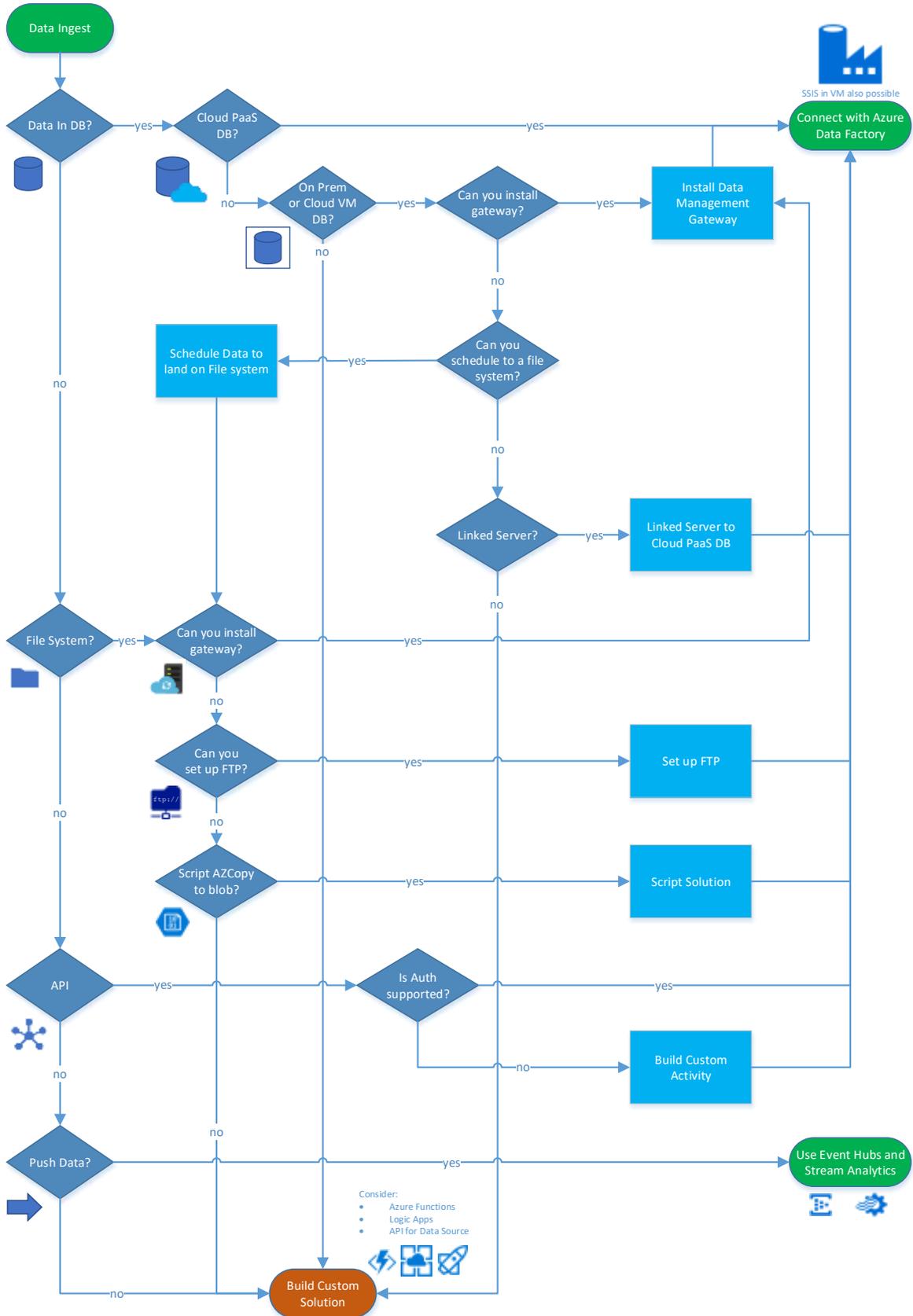
Data Factory		<a href="https://azure.microsoft.com/en-gb/services/data-factory/">https://azure.microsoft.com/en-gb/services/data-factory/</a>
SQL Data Warehouse		<a href="https://azure.microsoft.com/en-gb/services/sql-data-warehouse/">https://azure.microsoft.com/en-gb/services/sql-data-warehouse/</a>
SQL Database		<a href="https://azure.microsoft.com/en-gb/services/sql-database/">https://azure.microsoft.com/en-gb/services/sql-database/</a>
Blob Storage		<a href="https://azure.microsoft.com/en-gb/services/storage/blobs/">https://azure.microsoft.com/en-gb/services/storage/blobs/</a>
CosmosDB		<a href="https://azure.microsoft.com/en-us/services/cosmos-db/">https://azure.microsoft.com/en-us/services/cosmos-db/</a>
Data Lake Store		<a href="https://azure.microsoft.com/en-gb/services/data-lake-store/">https://azure.microsoft.com/en-gb/services/data-lake-store/</a>
Event Hubs		<a href="https://azure.microsoft.com/en-gb/services/event-hubs/">https://azure.microsoft.com/en-gb/services/event-hubs/</a>
Stream Analytics		<a href="https://azure.microsoft.com/en-gb/services/stream-analytics/">https://azure.microsoft.com/en-gb/services/stream-analytics/</a>

## Flowchart Guidance

When following this flowchart start at 'Data Ingest' and follow the 'yes/no' paths based on the questions given. The dark blue diamonds represent binary questions you need to ask to navigate the data ingest decision making process. The light blue boxes represent activities you will need to perform in the path chosen. You will see that there are three possible end points:

1. Connect with Data Factory. If you reach this option you are ready to use Azure Data Factory to bring data onto the Azure Data Platform.
2. Use Event Hubs and Stream Analytics. If you reach this option you have 'push' data and can accept the data from the source using Event Hubs.
3. Build Custom Solution. If you reach this option then you will have a situation that is not straight forward, and you will need to look deeper into how to move your data. There are opportunities to use Custom Activities, Azure Functions, Logic Apps as well as building a custom API for these data sources.

Flowchart  
Start here...



## Appendix 1: Next Steps and Considerations

The flowchart process helps you to understand what you need to do to ingest data onto Azure. However, the choice of where the data goes once on the platform also takes thought. This is a topic for another white paper but following are some examples of the type of constraints you need to consider:

### Maximum Size

As with an on-premise data store, a cloud data store in most cases, has a capacity limit. Often these capacity limits are tiered against cost. Due to the evolving nature of cloud technologies, this capacity limit can rise over time. However, at any point you will need to research and be aware of the current limit. For example:

**Azure SQL DB** currently has a top limit of 4TB for Premium tiers.

**Azure SQL DW** currently has a top limit of 240TB.

**Azure Blob Storage** currently has a maximum blob size of just over 4TB with some configurations.

**Azure CosmosDB** currently has a top limit of 10GB per partition. But unlimited partitions.

**Azure Data Lake Store** currently has no known top limit for file size.

If the above stores do not suit your needs, then you will need to consider a data store in IaaS. Such as SQL Server 2016/17.

### Cost

When moving to a cloud first data system you should consider cost in a very different way. Your first experience of this will likely be from a '3-year investment in hardware and licenses' to a 'monthly spend based on usage'. While this transition of cost model may be a challenging for some businesses the underlying benefits of increased agility and flexibility coupled with reduced total cost of ownership (TCO) mean that it is worthwhile.

Your second experience will likely be in relation to the separation of data storage from compute. This depends on the services you target but savings and benefits can be gained here as well.

**Azure SQL DB.** Pay for 'DTU' – a combined measure of resources needed.

**Azure CosmosDB.** Pay for 'RTU' – a combined measure of resources needed.

**Azure SQL DW.** Pay for Compute and Data Storage *separately*. With the ability to pause compute.

**Azure Data Lake.** Data Lake Store (cost per GB/ingest/egress) and **Data Lake Analytics** (cost per query) are charged *separately*. The cost per GB is very low per month.

**Azure Blob Storage.** Cheapest storage available on Azure. Includes Hot and Cold options.

**Azure HDInsight.** A compute engine that can be scaled at need or switched off to save cost. Data can be stored in the cluster, in Data Lake Store and in Blob Storage depending on need.

## Ingest Speed

Depending if your data is coming in via Batch (cold path) or Stream (hot path) you may need to consider the speed at which a data store can accept and turn around data for query. It is recommended that you test different stores for your scenario.

You could choose to upload data to your chosen data store at human speed through the **Azure Portal** using the browser based interface there. This both relies on a person doing the task and the bandwidth of their connection.

**Azure CosmosDB**, for example, with its variable consistency model and guaranteed latency can outperform **Azure SQL DB** when it comes to writes.

If your data is Pushed towards the system, such as in an IoT scenario then the speed of ingest is of critical importance. As such, **Event Hubs** and **Stream Analytics** are available to handle these scenarios. **Azure CosmosDB** would be ideal to support these in high speed scenarios.

## Data Model

Different data stores are optimised for different data models. When choosing which store you will need to choose one with the most appropriate data model. This is one aspect that influences many technologist's decision to use 'polyglot persistence' (storing data in several different data stores).

**Azure Data Lake Store** or **Azure Blob Storage** would be a valid choice for your low cost 'landing zone' for any data. The incoming format could be Binary, Document or Table.

**Azure Cosmos DB** would be a valid choice for data in a Document, Graph or Table.

**Azure SQL DB** or **DW** would be a valid choice for data in a Table or Relational model.

## Processing Type

Similarly, different data stores are optimised for different kinds of data processing. This is a second aspect that influences a decision to use polyglot persistence.

**Azure SQL DB** is optimised for OLTP workloads. In the higher tiers, there are options to use in memory column store technology, but this would be considered as additional functionality to the core OLTP workload. Large analytics workloads can negatively impact the performance of the operational OLTP system

**Azure SQL DW** is, as a MPP Column store, optimised for large analytical workloads such as large aggregates. It is suboptimal for OLTP type workloads.

**HDInsight** or **Data Lake Analytics** are optimised for 'Big Data' where horizontal scaling is needed.

## Coupled or Decoupled data system

In traditional data systems designs encouraged coupling and systematic fragility for the purposes of strict data control. For example, an Extract, Transform, Load (ETL) system couples source systems and their schemas through a highly controlled set of movements and transformations to target database systems and schemas. When data volumes are small, and content or schemas are unchanging this can be a highly efficient way to work.

**Azure SQL DB** or **DW** support this kind of coupling and workflow.

In big data systems, it is likely that data will be 'unknown' or changing and so the coupling mentioned above can lead to a lack of agility and flexibility when working. As such a decoupled and flexible data system would be desired. As data moves towards applicability to application it's content and schema becomes more controlled and those systems mentioned above can be used.

**Azure CosmosDB** provides *implicit* schema support in its document and graph model. This provides a flexible approach to schemas that is often applicable to global scale application developers.

**Data Lake Store/Analytics HDInsight** are all systems that support 'schema on read' where there is no coupling through schemas until the last minute. It is even possible under these systems to do 'schema discovery' to work out what schema or format a data has.

### Public Access

Finally, when considering where to land your data after ingest, the accessibility of the store to the 'public internet' is a consideration. All the discussed data stores have enterprise security features, such as encryption at rest and firewalls. However, in some scenarios it is still desirable to lock a resource to a private network. As such IaaS solutions (Such as SQL Server 2016/17 installed on a virtual machine in a VPN) must be considered.

Virtual machine based services, such as HDInsight, can use VPN Injection to make sure all processing and storage is done within the right closed network and using Express Route, Azure SQL DB can be peered to the network. Please review roadmap and documentation websites for updates in possibilities here.