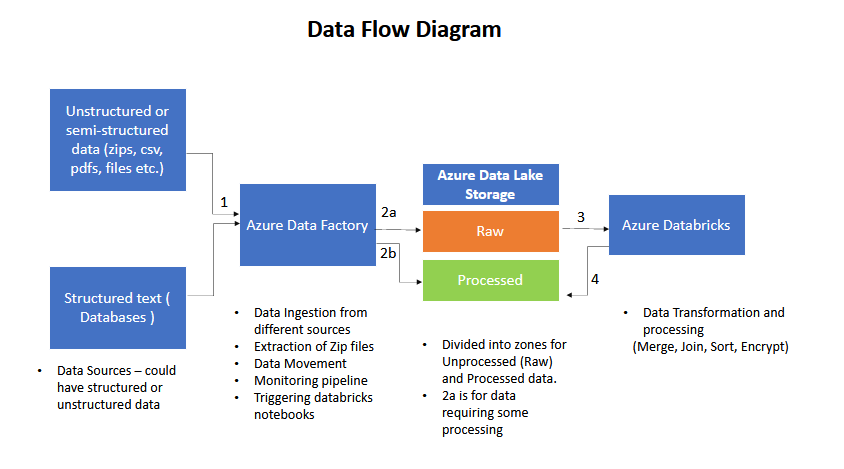
**Design Document**

* Data Ingestion- Ingestion of data from multiple sources would be performed using Azure Data Factory and Azure Databricks.
* Compressed files – The decompression would be performed either through Azure Data factory or through Azure Databricks and further processing or transformations on it would be performed with Azure Databricks.
* Scripts would be written in azure data bricks for merging, sorting , encrypting (whenever required) and performing other processing and transformations on data and the results would be saved to the data lake. These scripts would be triggered through pipeline(s) created in Azure Data Factory for automation.
* Relational data requiring regular data processing tasks would be saved in the form of Delta tables in Azure Datalake storage.
* Regular running of pipeline would be scheduled from azure data factory.
* Monitoring of pipeline would be done through the azure data factory.
* Security Aspects**:**
  + Encryption of sensitive data – Column level encryption of data in delta tables could be done for protecting personally identifiable information via databricks notebooks.
  + Role based access control of data in Azure Data Lake Storage – Azure platform’s inbuilt Azure RBACs (for controlling coarse-grained access to **all** data on specific storage accounts based on roles) and ACLs (for controlling fine grained read/write access to specific files) could be used for this purpose.
  + Table Access controls on Delta tables will be added to allow access
  + Account level should not be timed out by azure and databricks.



Azure Data Factory pipeline implementation details

* Single pipeline for all data inputs (could be separated if needed)
* Initially it will load configurations for each DataSource (relational or file based). Configurations would have list of details of each of the different DataSources.
  + Configurations could be saved on a Azure SQL Database table (Basic DTU version of Azure SQL costs 5$/month for 2GB) or configs could also be saved on a csv file and be loaded from it as an alternative
* Steps of pipeline completion could be saved to a SQL database table (azure sql db cost will add as mentioned previously, alternative way would be with a csv file in adls). Logs and Metadata details for pipeline run can be saved in ADLS in separate metadata directory.

Azure Data Lake Storage details

* Different blob containers can be created in same storage account for Staging, Processed and Curated data for easy separation.