Introduction

The E-textiles capture person's vitals via sensors and motors. Once that data is captured it is then transferred to MongoDB which is a non-relational database that collects data in real time and saves the information into binary or JSON files. The current task at hand is to find a cost-effective relational database that allows the extraction of data from MongoDB for more permanent storage.

Thesis

There are five types of relational data: ETL, ELT, OLAP, OLTP and data warehousing. I will be discussing all of them, their applications, their purposes and technologies that can be used.

*** Extract, Transform and Load (ETL)**

ETL is used to collect data from multiple sources, transform the data and then load into a predefined target. The transformation part usually consists of: sorting, filtering, reduplicating and joining raw data. Then the modified information can be loaded into a target database for storage.

Azure Tools:

-Azure Data Factory

Other Tools:

-SQL Server Integration Services

*** Extract, Load and Transform (ELT)**

ELT differs from ETL in where the transformation takes place. In ELT the transformation occurs in the target data store instead of using a separate transformation engine to filter the data. ELT is mainly used in big data analysis where large amounts of data need to be processed. The transformation happens directly in the storage unit which skips the data copy step which can save a lot of time and computational resources when working with large data sets.

ELT is generally more efficient than ETL and the data may be partitioned into smaller chunks and distributed among multiple machines for maximum efficiency.

Azure Tools:

-Azure Synapse

-HDInsight with Hive

-Azure Data Factory v2

-Oozie on HDInsight

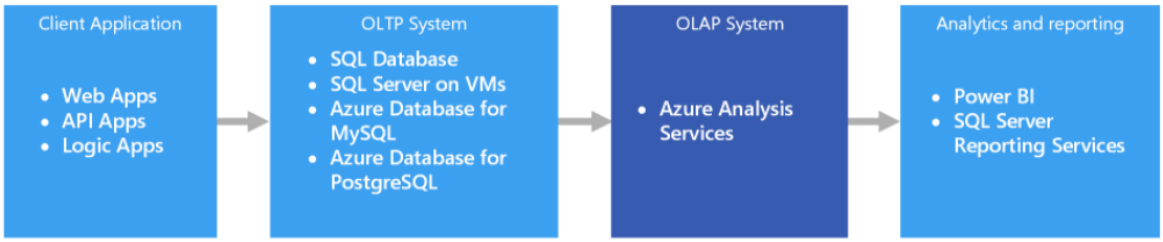
Other Tools:

-SQL Server Integration Services (SSIS)

*** Online Analytical Processing (OLAP)**

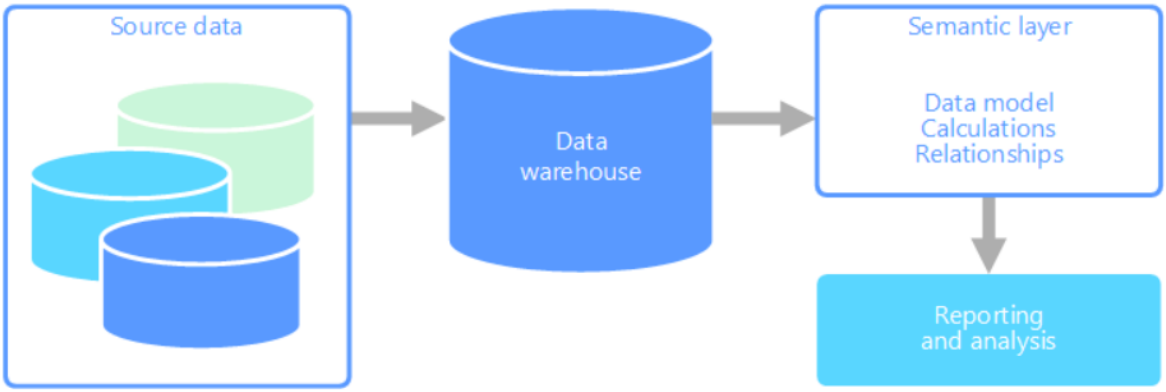
OLAP databases are optimised for heavy read, low write workloads. That makes them perfect for data analysis. The typical OLTP databases are not designed for analysis which makes data retrieval costly in terms of computational power and time. This is why OLAP databases are designed to help extract valuable information from OLTP.

Snippet from the Microsoft Documentation on OLAP:



Usually to make sure that the data is serialized and easy to access it is placed in a data warehouse before being passed to the OLAP service. That ensures that the data will be the same for all users that try to query that data. We need to keep in mind that the purpose of data warehouses is to collect data from multiple sources, serialize it and keep it until it is required from other OLAP tools for data analysis or data mining.

Snippet from the Microsoft Documentation on OLAP:



Azure Tools:

-SQL Server with Columnstore Indexes

-Azure Analysis Services

-SQL Server Analysis Services (SSAS)

*** Online Transaction Processing (OLTP)**

OLTP is optimised for workloads that require high amount of writing. It is generally used for storing data such as transactions and payments made however it can be optimized for more abstract types of data. OLTP is usually used when received data must be immediately made available to client applications and delay in processing might have a negative impact on the operations of the company.

However OLTP has few drawbacks. Analytics against the data are very resource intensive and can be very slow to execute. When conducting reports on the data, the queries tend to be very complex which is very resource consuming. That is why when needed to analyse or access the data, an OLAP tool is used that is optimised for heavy reading duty. Also storing too much data for too long slows query performance. That is why it is a good idea to store the data on the OLTP database only for a specific period of time (for example a year) and after that offload historical data to a data warehouse.

Azure Tools:

-Azure SQL Database

-SQL Server in a Virtual Machine

-Azure Database for MySQL

-Azure Database for PostgreSQL

*** Data Warehousing**

”A data warehouse is a centralized repository of integrated data from one or more sources. Data warehouses store current and historical data and are used for reporting and analysis of the data.”- Microsoft Documentation on Data Warehousing. Data can be loaded into a warehouse from multiple sources. As the data is moved it can be transformed using an ETL tool. The transformation can consist of: formatting, validation, sorting and etc. At the end, the data warehouse becomes a permanent data store for reporting and analysis.

Data warehouses are useful when massive amounts of data must be made into a format that is easy to understand. They also make it easy to access historical data from multiple points by providing one centralized location using common formats and keys. Data warehouses help a user analyse data without the need of a data developer or a data scientist. Because data warehouses are optimised for reading it is generally faster to access the data. Data warehouses can be considered as an analytical data store layer and its purpose is to satisfy queries issued by different analytical and report tools.

There are different tools for implementing a data warehouse in Azure but there are two main categories: symmetric multiprocessing (SMP) and massively parallel processing (MPP). SMP involves a computer with many processors where all processors share one main memory and have full access to all input, output devices and are controlled by the same operating system. While in MPP the workload is distributed among multiple machines to simultaneously perform a set of computations. In general SMP-based warehouses are best suited for small or medium sets of data while MPP is used for big data.

SMP Tools:

-Azure SQL Database

-SQL Server in a Virtual Machine

MPP Tools:

-Azure Synapse Analytics

-Apache Hive on HDInsight

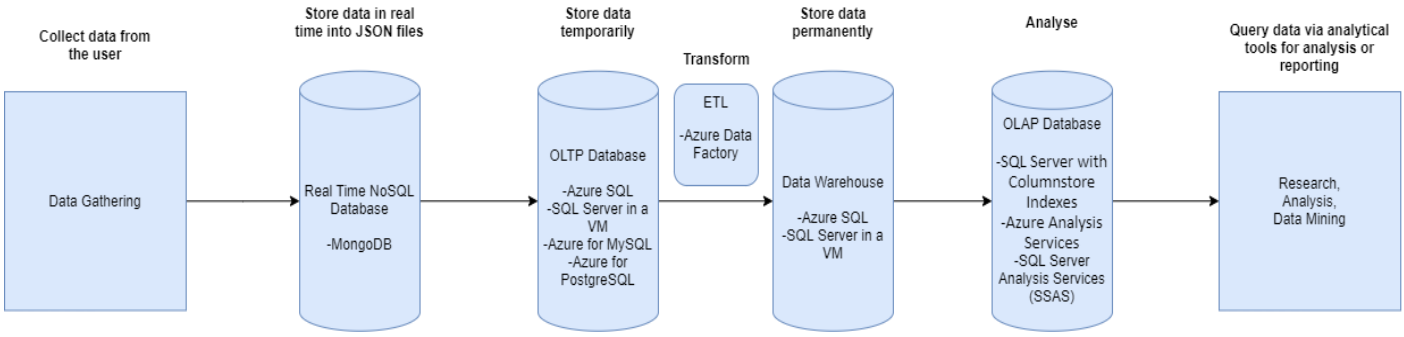
-Interactive Query on HDInsight

Conclusion

In conclusion I would say that the data type we are interested in is OLTP. We are already capturing data from the user in real time and storing that data into a non-relational database which at the moment is MongoDB. Right now we are looking for a more permanent type of storage because non-relational databases are not meant to hold data for a long periods of time.

The general pipeline process would look like this: capture data from the customer and then store it in MongoDB, move that data to an OLTP type of database and offload historical data to a data warehouse for permanent storage. We can use ETL tool such as Azure Data Factory to filter our data before storing it in a data warehouse. Then at the end we can use the data warehouse as our permanent data analysis layer and query data from there via OLAP tools for analysis and reports.

Diagram of the pipeline:



As we can see Azure SQL and SQL Server VM can be used both as an OLTP database and as a warehouse. The reason for that is that databases such as MySQL and PostgreSQL are not optimised for heavy reading duty. The usual approach to overcome that is to transfer data to a data warehouse after a specific time window and then query data from the data warehouse for analysis.

Azure SQL and SQL Server VM are already optimised for heavy reading duty which means that using them not only cuts cost because a separate OLTP database is not require but also saves time on computation.

I think the database that would best suit our needs would be Azure SQL or SQL Server in a virtual machine because as I said it cuts cost and saves time. As an ETL tool to filter our data before storing it we can use Azure Data Factory which is very intuitive and easy to use.

Main differences between Azure SQL Database and SQL Server in a Virtual Machine:

SQL Server in Azure VM	Azure SQL Database
You access a VM with SQL Server	You access a DB
You manage SQL Server and Windows: High Availability, Backups, Patching	DB is fully managed: High Availability, Backups, Patching
You can run any SQL Server version and edition	Runs latest SQL Server version, based on Enterprise edition
Full on-premise compatibility	Incomplete on-premise compatibility (e.g. no jobs, linked servers, FileStream)
Different VM sizes: A0 (1 core, 1GB mem, 1TB) to A16 (16 cores, 112GB mem, 16TB), D-Series (with SSDs)	Different DB sizes: Basic (2GB, 5tps) to Premium (500GB, 735tps)
VM availability SLA: 99.95%: Can achieve higher availability (~99.99%) configuring AlwaysOn	DB availability SLA: 99.99%
Reuse on-premise infrastructure (e.g. Active Directory)	

When using SQL Server in a VM we need to build our virtual machine first and then configure our SQL server. The choice between Azure SQL and Azure SQL in a VM entirely depends on our needs. Azure SQL in a VM is more used for migrating existing apps and supports only few SQL servers. While Azure SQL is used for building new apps and supports hundreds of databases.

Pricing of Azure SQL per month:

This diagram shows the different prices for the different vCore packages in Azure SQL. The storage space remains constant of 1 TB and the number of vCores corresponds to the computational speed of the server.

References

Microsoft Documentation: https://docs.microsoft.com/en-us/documentation/

Azure Pricing Calculator: https://azure.microsoft.com/en-gb/pricing/calculator/