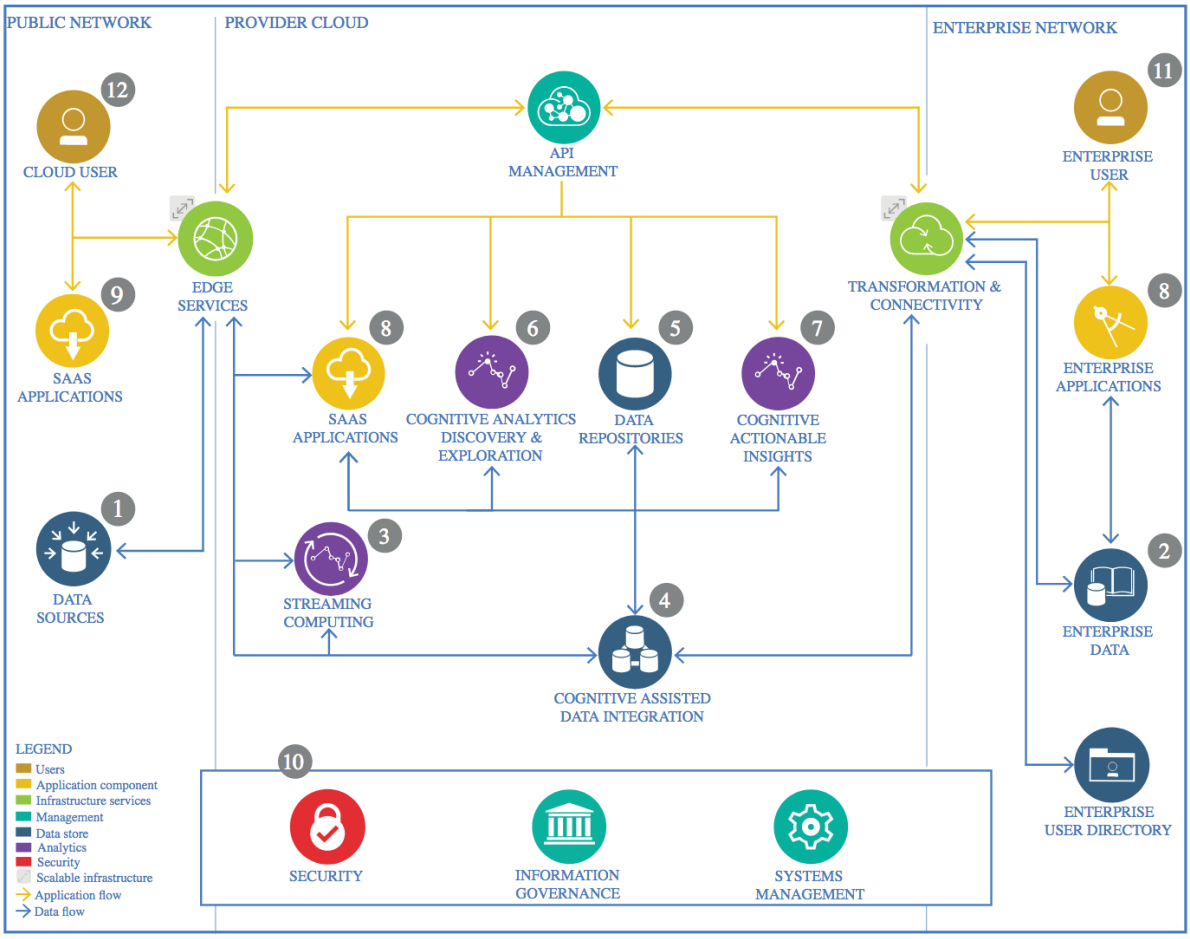
The Lightweight IBM Cloud Garage Method for Data Science

Architectural Decisions Document for project ‘Suggesting Affordable yet Safe Housing in Seattle, WA’

1 Architectural Components Overview



IBM Data and Analytics Reference Architecture. Source: IBM Corporation

1.1 Data Source

1.1.1 Technology Choice

For the data acquisition part, we use the following data sources: Wikipedia to find out Seattle's district names and coordinates https://en.wikipedia.org/wiki/Category:Neighborhoods_in_Seattle

Crime data is available from official sources of City of Seattle: https://data.seattle.gov/Public-Safety/Crime-Data/4fs7-3vj5

For prices of flats we use data set provided by Airbnb on Kaggle portal: https://www.kaggle.com/airbnb/seattle/version/2#_=_

To locate parks nearby flats, we access FourSquare API.

1.1.2 Justification

Several other sources have been investigated, but they lacked good API or broad span over all the districts of Seattle

1.2 Enterprise Data

1.2.1 Technology Choice

The data being used in the project is public. No in-house or enterprise sources were used.

1.3 Data Integration

1.3.1 Technology Choice

IBM Object Storage is used to store the scraped data from the raw sources in CSV format. The data is persistent and not being streamed.

1.3.2 Justification

All computations and modeling are done using Jupyter notebooks within IBM Cloud, therefore IBM cloud is a good fit to store unstructured data, because it can be accessed from there. Files aren't that big, and shall they grow in size, they can be naturally split 'horizontally' by the names of the districts of Seattle they belong to, ZIP codes etc.
The information in files has to be prepared and normalized first (for instance, names of the districts should be the same), so there is no point in, say, inserting this raw data into an RDBMS.

1.4 Data Repository

1.4.1 Technology Choice

Python pandas DataFrames are being used as in-memory store of the data. No Database is being used.

1.4.2 Justification

Later on, if the dataset becomes larger, apache spark can be used and the code rewritten to operate on RDD instead of pandas DataFrames.

1.5 Discovery and Exploration

1.5.1 Technology Choice

We use Python, pandas, matplotlib, seaborn and Folium to visualize all the data we have gathered and understand them better. We use it to create new features: proximity to parks score, affordability score, and criminality level score.

1.5.2 Justification

Python is great for data analysis and quick ideas prototyping. There are many powerful third-party libraries for statistical analysis which are free of charge. There are also many Python programmers available on the market.

1.6 Actionable Insights

1.6.1 Technology Choice

Looking deeper at the data provided by data.seattle.gov we've understood that we'd need additional data, because the number of crimes has to be normalized by the population size. Since we couldn't find population size on Wikipedia pages for Seattle's district, we had to search for an additional source of data. We get the population information from the portal 'Find My Seattle' https://findmyseattle.com/home and enter them by hand, because the website is hard to crawl automatically (some JavaScript most likely).

1.7 Applications / Data Products

1.7.1 Technology Choice

We use Keras to build models for the prediction of good flats.

1.7.2 Justification

Keras library is flexible, extensible and allows for model serialization. It is also high-level enough to explain the details of our model to the stakeholders.

1.8 Model Architecture

1.8.1 Technology Choice

The final model architecture was chosen to be a Feed Forward neural network with 30 input gates in the first layer and 30 epochs for training. The selected architecture has shown very good accuracy on the test data in persistent manner.

1.9 Model Training

1.9.1 Technology Choice

The data set was split into 80% for training and 20% used for testing purposes.

1.10 Model Deployment

1.10.1 Technology Choice

Keras models can be serialized and deployed later on different cloud solutions.
Google Cloud Functions was eventually chosen as deployment environment.

1.10.2 Justification

The cost factor was the main driving factor for this decision.

1.11 Security, Information Governance and Systems Management

1.11.1 Technology Choice

Once we integrate user profiles with their preferences, such sensitive information has to be protected.

1.11.2 Justification

Due to GDPR, we will strip away all relevant customer information and anonymize their preferences and choices.