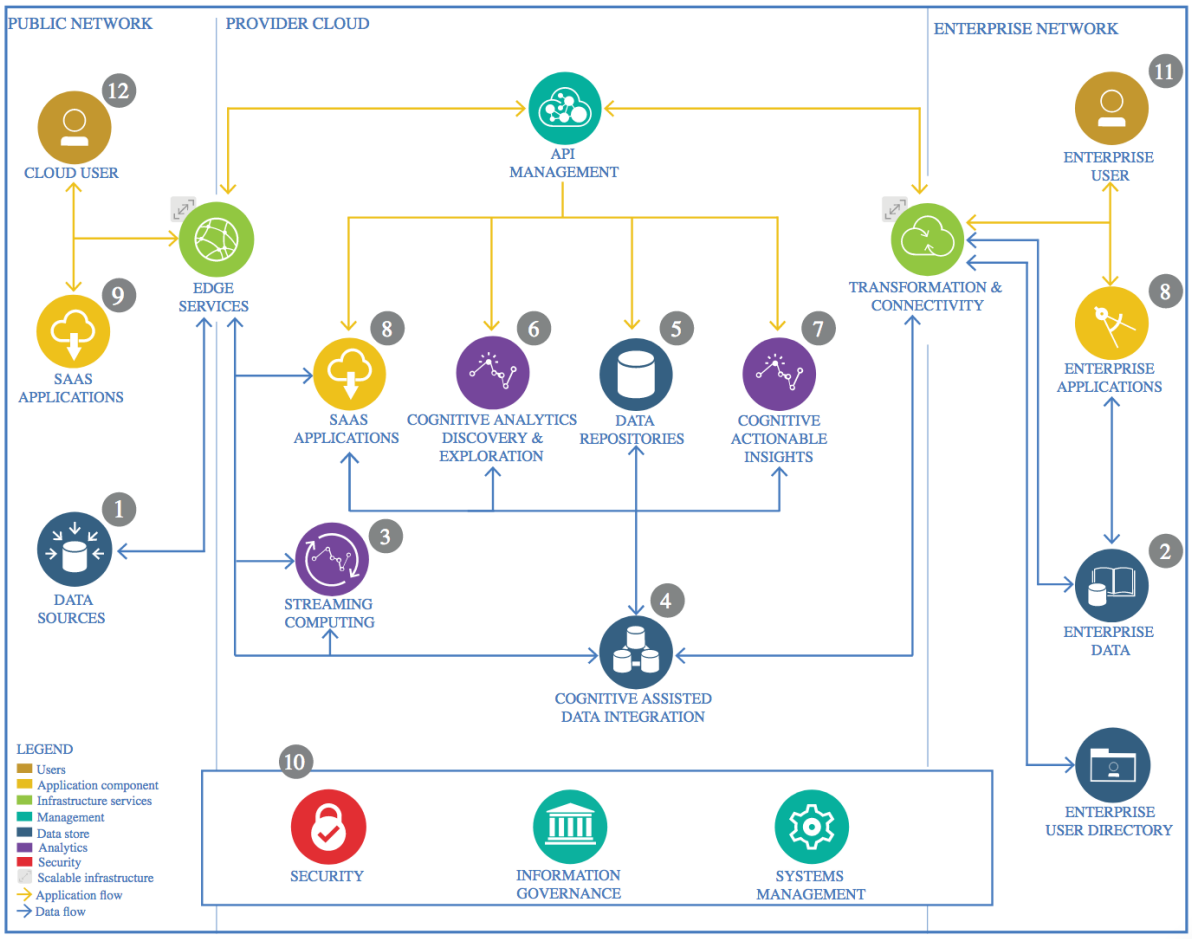
The Lightweight IBM Cloud Garage Method for Data Science

Architectural Decisions Document Template

1 Architectural Components Overview



IBM Data and Analytics Reference Architecture. Source: IBM Corporation

1.1 Data Source

1.1.1 Technology Choice

The data was downloaded from the website of the Bureau Of Transportation Statistics, U.S. Department of Transportation.

1.1.2 Justification

Primary reason to download these data was availability of huge amount of data and ease of use.

1.2 Enterprise Data

1.2.1 Technology Choice

NA

1.2.2 Justification

NA

1.3 Streaming analytics

1.3.1 Technology Choice

NA

1.3.2 Justification

NA

1.4 Data Integration

1.4.1 Technology Choice

Based on our data set choices, we can use Panda Dataframes to do most of our data integration and cleansing.

1.4.2 Justification

Easy to integrate data and prepare it for analysis.

1.5 Data Repository

1.5.1 Technology Choice

The training data is coming from static files that are imported and saved using Hierarchical Data Format (HDF5 file), which performs very well for large data sets.

1.5.2 Justification

This provides an easy way to get to work and seems to be faster than other tools such as pickle.

1.6 Discovery and Exploration

1.6.1 Technology Choice

We will use mostly Jupyter notebooks and IBM Watson Studio.

We use scikit learn for both our deep and non-deep learning models.

We tested with two non-deep learning models and one deep learning model. Our non-deep learning models were Random Forest Classifier and Support Vector Machine. The deep learning model was the Multi-Layer Perceptron Classifier.

1.6.2 Justification

IBM Watson Studio is free to use and allows us to scale up our compute power if needed. It allows us to use compute and storage infrastructure at scale without having to spend too much time on logistics, thus allowing us to focus on the problem at hand.
Bagged decision trees like Random Forest can be used to estimate the importance of features. The goal of the SVM algorithm is to find a hyperplane in an n-dimensional space (n is the number of features) that distinctly classifies the data points. One can do simple deep learning regression and classification models with the scikit-learn pacckage. However, for most real-life large-scale projects, these algorithms might not be the best choice as there is no GPU support and very limited options to tweak the parameters.

1.7 Actionable Insights

1.7.1 Technology Choice

We used Recall, Precision, Accuracy and F1 as the evaluation metrics.

1.7.2 Justification

Accuracy might be the most intuitive metric for classification tasks, but shouldn't be taken as the only solution.

1.8 Applications / Data Products

1.8.1 Technology Choice

This project might lead in the future to possible applications in the sector of aviation and mobility.

1.8.2 Justification

It is not ready for production use until it has been trained and tested on more data.

1.9 Security, Information Governance and Systems Management

1.9.1 Technology Choice

NA.

1.9.2 Justification

As this is just an academic exercise, we are not concerned with security issues.