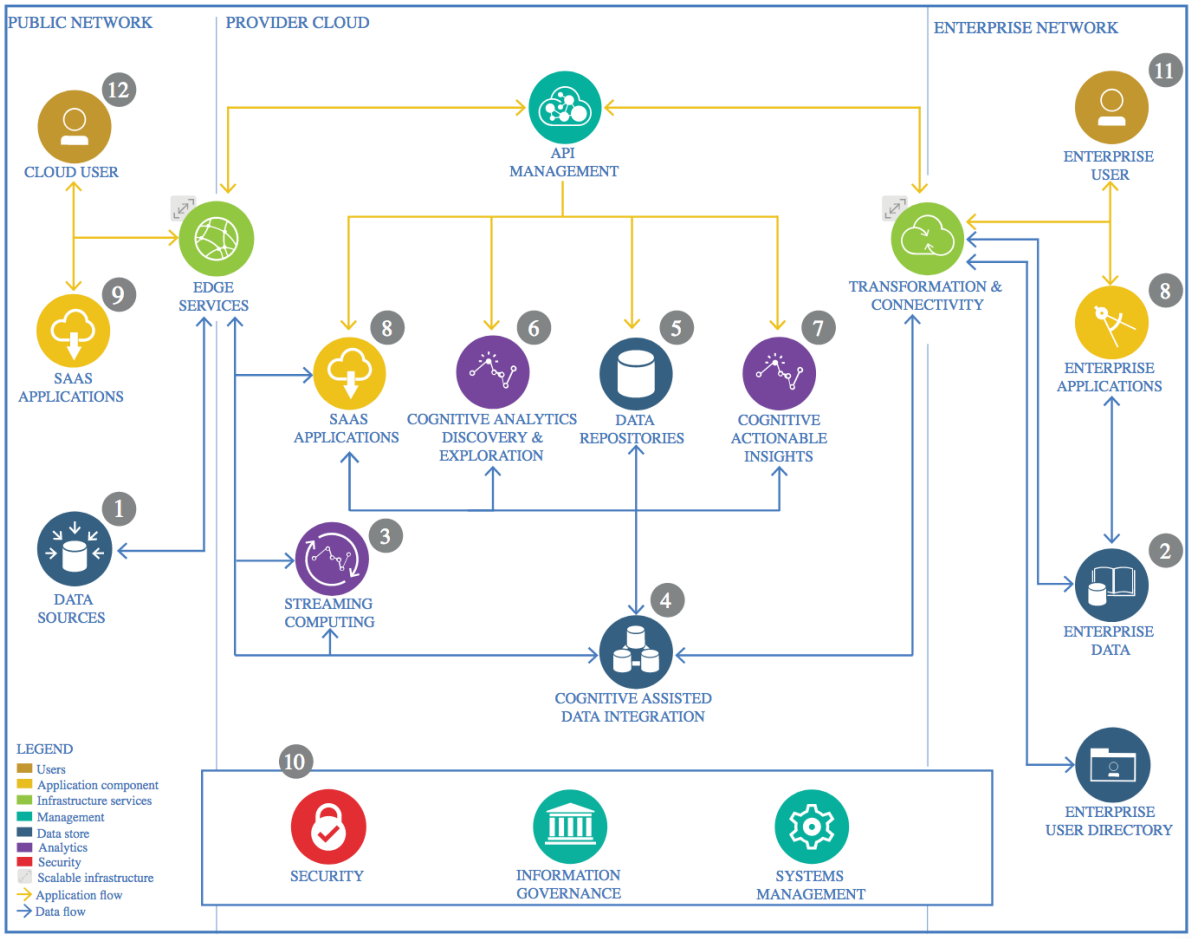
The Lightweight IBM Cloud Garage Method for Data Science

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

A 'Forest Cover Type' dataset has been sourced from Kaggle.

https://www.kaggle.com/uciml/forest-cover-type-dataset

1.1.2 Justification

The 'Forest Cover Type' dataset itself has been chosen due to the increasing pressure for environmental safeguarding. Environmental impact analysis, determination of land cover/use, and the management of natural resources are all important aspects in environment protection and could benefit greatly from partly automated and optimized geographic information systems.

By identifying tree species from simple cartographic measurements, a greater understanding of the environment can be built, allowing for improved environmental safeguarding.

The host (Kaggle) been chosen due to their typical reliability and high-quality of datasets.

1.2 Enterprise Data

1.2.1 Technology Choice

Not used.

1.2.2 Justification

No enterprise data is used as the only required data is sourced from Kaggle.

1.3 Streaming analytics

1.3.1 Technology Choice

Not used.

1.3.2 Justification

The objective of this project is to create an 'environmental image' of areas from previous observations of the Roosevelt National Forest in Colorado. For the purpose of this project live streamed cartographic data would be expensive to set up and not necessary to build up a picture of the environment within the observed area.

1.4 Data Integration

1.4.1 Technology Choice

The data hosted by Kaggle can be downloaded in CSV format, to integrate this IBM Watson Studio Jupyter notebooks (used throughout the development process), the 'Insert to Code' function is used.

Initial data cleansing and storage back into IBM Watson Studio ObjectStorage is performed with IBM Watson Studio Jupyter notebooks with Python.

1.4.2 Justification

The ‘insert to code’ function allows for a straight-forward upload of locally stored data to IBM Watson Studio. As the dataset was stored locally and was required to be uploaded to the IBM Cloud to be used within the IBM Watson Studio Jupyter notebooks required for development, the ‘insert to code’ function was deemed the fastest and easiest approach.

Python is used for data cleansing and formatting as it is a very readable and versatile language with significant support for data import/export and cleaning.

1.5 Data Repository

1.5.1 Technology Choice

The source data is stored in ObjectStorage within the IBM Watson Studio ‘ibm_capstone’ project.

1.5.2 Justification

Storing within the project cloud space allows for the data to be imported into Jupyter notebook spaces quickly and easily. ObjectStorage is used due to its ease of use, the data is stored as an object and is given a unique ID and HTTP URL, meaning the Jupyter notebooks can quickly access the data. ObjectStorage is also known for its fast data retrieval and recovery and cost-effectiveness.

1.6 Discovery and Exploration

1.6.1 Technology Choice

IBM Watson Studio Jupyter notebooks with Python, using Pandas, Numpy, Matplotlib and Seaborn.

1.6.2 Justification

IBM Watson Studio allows for flexible processing power and cost-effective power usage, meaning the discovery and exploration script can be run remotely with little setup time.

Python is used as a very readable language with significant support for ad-hoc data analysis. In particular the Pandas, Numpy, Matplotlib and Seaborn libraries are used extensively. Pandas for easy data manipulation and processing. Numpy for is used for statistical analysis and further data manipulation. Matplotlib and Seaborn are used together for data visualization.

1.7 Actionable Insights

1.7.1 Technology Choice

IBM Watson Studio with Python, using Keras, Pandas, Numpy, Scikit-Learn and Matplotlib.

1.7.2 Justification

IBM Watson Studio allows for quick, cost-effective and scalable parallel processing ideal for machine learning applications.

Jupyter notebook allows for flexible and readable document to be created combining code with text and images crucial for understanding the process.

Keras is a high-level machine learning library and in this instance uses a Tensorflow backend. Keras is highly readable and excellent for quick model development and performance analysis.

Pandas and Numpy are both used for data manipulation and statistics.

Scikit-Learn is used for further in-depth analysis of model performance post-training. This is coupled with Matplotlib to allow for easy visualization of performance metrics.

1.8 Applications / Data Products

1.8.1 Technology Choice

Jupyter notebook is used to contain code and model, the model is stored using ObjectStorage on IBM Watson Studio.

1.8.2 Justification

Jupyter notebooks are used to contained the model prediction code, this allows for easy to follow instructions to be implemented alongside the code. Adjustments to the code are easily made, this is also enabled by the Python language used, renown for readability.

IBM Watson Studio hosts the Jupyter notebook and stores the serialized model using the HDF5 format in ObjectStorage. This means that the user can quickly access the model and load correctly into a usable Keras model using the Jupyter notebook provided.

1.9 Security, Information Governance and Systems Management

1.9.1 Technology Choice

The IBM Cloud allows for access to the IBM Watson Studio project to be restricted as much as required

1.9.2 Justification

The IBM Cloud allows for easy and clear permissions access to be setup for the IBM Watson Studio project. This solution is well developed with a strong focus on security, therefore making the platform very secure.