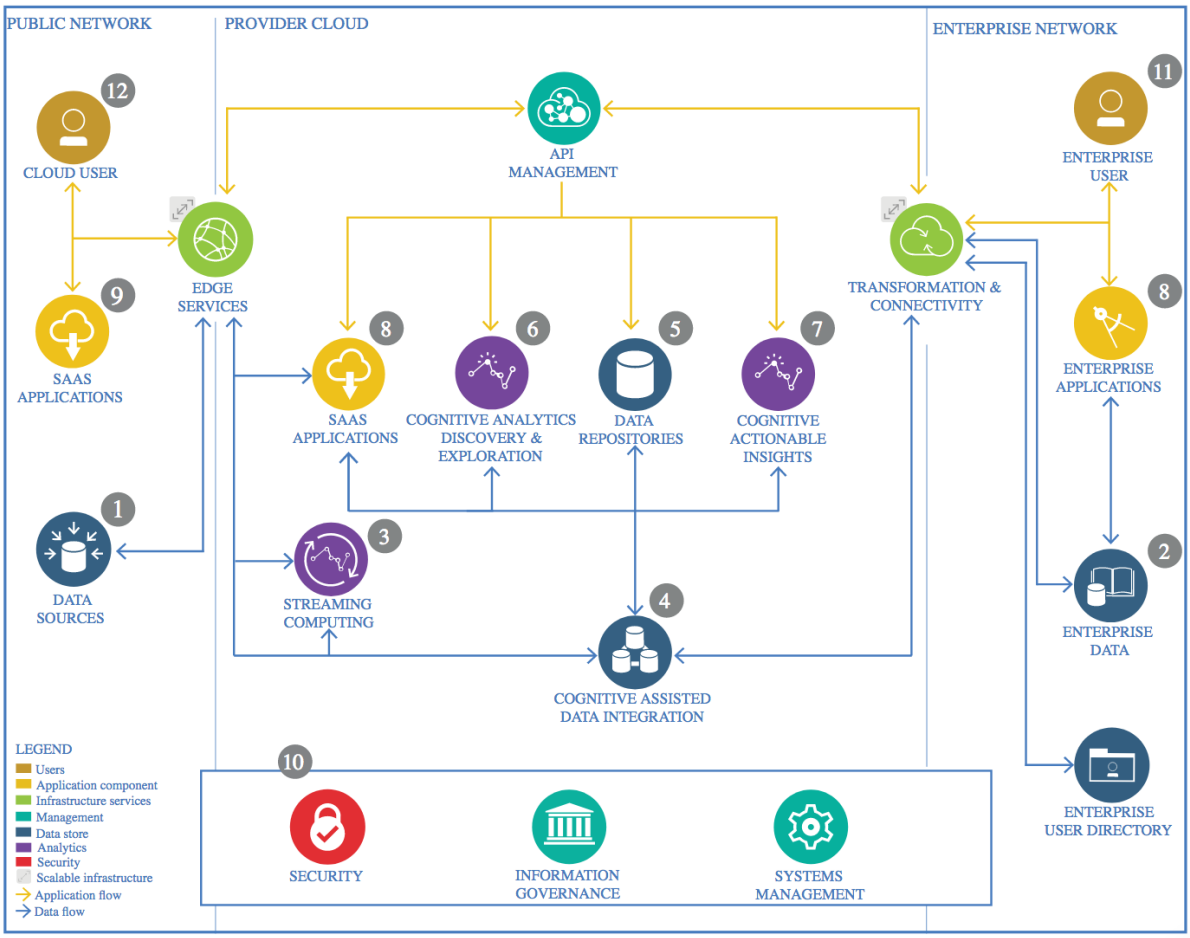
The Lightweight IBM Cloud Garage Method for Data Science

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

For the data source I utilized the Credit Card Fraud Detection from Kaggle.
https://www.kaggle.com/mlg-ulb/creditcardfraud

1.1.2 Justification

Credit Card Fraud is an interesting subject for someone who's into security operations. Also CSV files are a convenient and effective way to import and export datasets since they are universal and easy to create, read and manipulate in different ways. They can be easily updated by adding new weather records directly from online resources and updating the loaded file in the GitHub repository.

1.2 Enterprise Data

1.2.1 Technology Choice

For the enterprise data I chose to store all of my data as a parquet file on IBM Cloud Storage.

1.2.2 Justification

The dataset isn't large, therefore I fit within IBM's Coursera Program boundaries.

1.3 Streaming analytics

1.3.1 Technology Choice

The dataset is static. So, N/A

1.3.2 Justification

The dataset is static, so no streaming analytics were required.

1.4 Data Integration

1.4.1 Technology Choice

I chose Apache Spark with Python 3.11 and PySpark to integrate the data.

1.4.2 Justification

The entire size of the data set was only 60 MB which made it extremely easy to run Spark jobs with a single worker on the basic Node that IBM provides. The data types where simply float64 types and the source system was just a CSV file. To code all the model's extensive knowledge in Python 3.11, the Pandas, PySpark, Scala, Keras, Seaborn, and Matplotlib libraries was required. I turned some of the .csv data tables into Spark DataFrames which allowed for SQL queries to be run on the data set in the Spark Session.

1.5 Data Repository

1.5.1 Technology Choice

For my persistent storage I utilized IBM's Cloud Object Store.

1.5.2 Justification

Cloud Object Store is implemented within the IBM Watson development environment.

- How does this impact cost of storage?
    - Given that I'm under the Coursera program, it's free. But In production, it is not too expensive.
- Which data types are supported?

    - Resembles a file system, any datatype is supported.
- How good must point queries be supported?
    - RDBMS is very good at point queries because an index can be created on each column.
- What skills are required?
    - Apache Spark, Python, SQL.
- How good must full table scans be supported.
    - Full table scans are just bound by I/O bandwidth of the OS
- What is the amount of storage needed?
    - Will use only 80MB of storage.
- Growth and scaling?
    - Fully elastic on Cloud Object Storage.

1.6 Discovery and Exploration

1.6.1 Technology Choice

IBM Watson Studio, Jupyter Notebook, Python, Apache Spark (PySpark, PySpark ML), Scikit-learn, Pandas, Matplotlib, Seaborn.

1.6.2 Justification

- What type of visualizations are needed?
    - Matplotlib and Seaborn supports the widest possible visualizations including run charts, histograms, boxplots, and scatter plots.
- Are interactive visualizations needed?
    - No
- Are coding skills available/required?
    - Yes.
- Do metrics and visualizations need to be shared with business stakeholders?
    - The notebook can be shared through Jupyter notebooks.

1.7 Actionable Insights

1.7.1 Technology Choice

To identify fraudulent credit card transactions, three models were created, each with different selections in technology. Scikit-learn for a single node Logisitic Regression model, PySpark ML's Logistic Regression model, and Keras+Tensorflow backend sequential feed forward neural model made of Dense layers. All of which were implemented with Python, Pandas and Apache Spark.

1.7.2 Justification

- What are the available skills regarding programming language?
    - Python.
- What is the cost of skills regarding the programming language?
    - Costs are usually low as Python is open source and easy to use.
- What are available skills regarding frameworks?
    - Pandas and Scikit-learn are both clean and easy to learn, skills are widely available.

- What are the costs regarding frameworks?
    - Python is open source, so cost is low. Keras and Tensorflow skills however are could be much more expensive.
- Is model interchange required
    - Scikit-learn, Keras/Tensorflow can be serialized, and we can save Keras model and export/load them as need be.
- Is parallel or GPU based training or scoring required?
    - Not applicable in this case but in Apache Spark this can very quickly be integrated by loading a trained Keras model into Apache Spark. It is though recommended to use a GPU for more complex models, either locally or through Kaggle and Google Collab.
-

1.8 Applications / Data Products

1.8.1 Technology Choice

The data produced for this project is four .ipynb files from the IBM Watson Studio' Jupyter Notebook environment.

1.8.2 Justification

N/A

1.9 Security, Information Governance and Systems Management

1.9.1 Technology Choice

N/A

1.9.2 Justification

Given the fact that the Dataset is available publicly and that the repository is open, there is no need for Security, Information Governance and Systems Management.