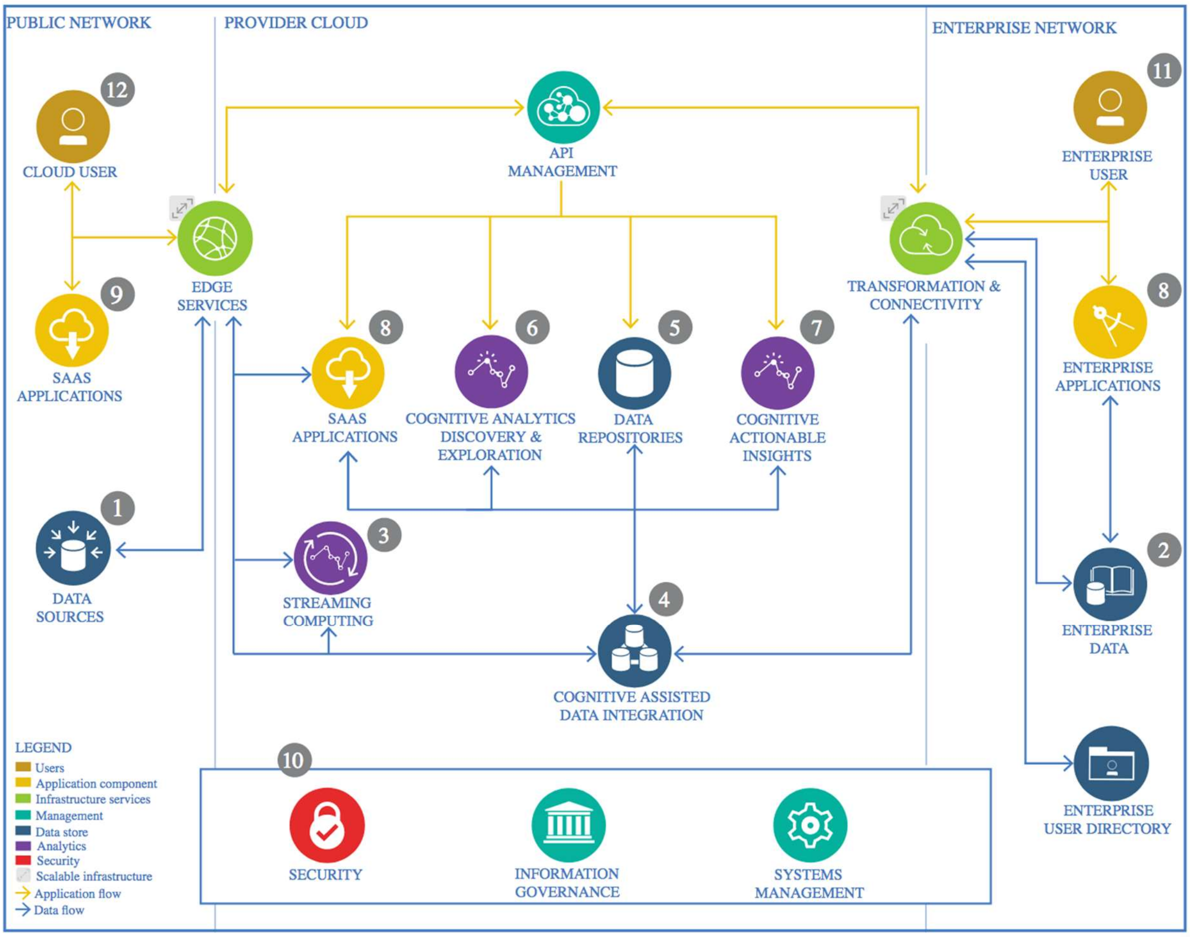
1 Architectural Decisions Document

This document describes the technical solutions used in the capstone project.

In this business problem, we are handling sensitive personal data (although pseudonymized) such as income and applicants' social benefits. However, we are not using any data that can be used to identify individual persons. We are thus able to process the data in IBM public cloud.

2 Architectural Components Overview



IBM Data and Analytics Reference Architecture. Source: IBM Corporation

2.1 Data Source

2.1.1 Technology Choice

No public data sources are used.

2.1.2 Justification

Detailed housing benefit data is not available in public data sources.

2.2 Enterprise Data

2.2.1 Technology Choice

Enterprise data is fetched from my company's Data Warehouse staging area (Netezza solution). Pseudonymized housing application and decision data can be fetched with SQL statements in "Aginity Workbench for PureData System for Analytics" to CSV files from DW.

2.2.2 Justification

This is the best available data for the business problem outlined.

2.3 Streaming analytics

2.3.1 Technology Choice

Apache Spark in IBM Public Cloud environment – if needed. It may be that we may not need to process data on multiple servers.

2.3.2 Justification

This is easily available, and does the job if needed.

2.4 Data Integration

2.4.1 Technology Choice

Apache Spark / Jupyter / Python (pandas, numpy). All data is coming from a single source. We may need to handle many separate data sets and combine these in Jupyter Notebook environment. We will certainly have to data encoding and scaling and other forms of preparation for downstream processing.

2.4.2 Justification

These are nicely available and do the job.

2.5 Data Repository

2.5.1 Technology Choice

Netezza for relational data, csv files, in-memory dataframes and Cloud Object Storage in IBM Cloud.

2.5.2 Justification

We are handling between some tens of thousands to some millions of rows of data. These solutions can easily manage this kind of load.

2.6 Discovery and Exploration

2.6.1 Technology Choice

Jupyter notebook, Python (numpy, pandas, matplotlib).

2.6.2 Justification

These are readily available in IBM Cloud environment and will probably do the job.

2.7 Actionable Insights

2.7.1 Technology Choice

Python, pandas, scikit-learn, Keras, TensorFlow. We will do classification with some algorithms available in scikit-learn (SVC, DecisionTree, RandomForest, AdaBoost...) and, if needed, with feed-forward neural networks with help of Keras and TensorFlow.

If needed, we will scale processing to multiple servers using Apache Spark and SystemML. We'll start with single servers and see whether we fit in main memory of one server and can do fitting in reasonable time.

2.7.2 Justification

These are available in IBM Cloud and provide a plenty of possibilities for getting insight into data.

2.8 Applications / Data Products

2.8.1 Technology Choice

We will be producing a PDF document of the results. We are not planning on using the model in production as of yet. If we would, we could do this either by importing Keras model to DL4J and running in a web application server (WebSphere, Tomcat, Jetty) environment (we are running like 400 Java applications in production and have a mature infrastructure for this) or within ICP4D as Python (with a rest layer on top of model and API management via our normal API management solution).

2.8.2 Justification

At the moment we interested in knowing whether decisions could be automated with machine learning algorithms. No plans for production yet. PDF document does the job best.

2.9 Security, Information Governance and Systems Management

2.9.1 Technology Choice

Running in web application server (WebSphere, Tomcat, Jetty) in our in-house 24/7 managed server environment would be the most natural choice. This would provide is needed management infrastructure out-of-the box. We are not planning on doing this, however.

2.9.2 Justification

At the moment we interested in knowing whether decisions could be automated with machine learning algorithms. No plans for production yet.