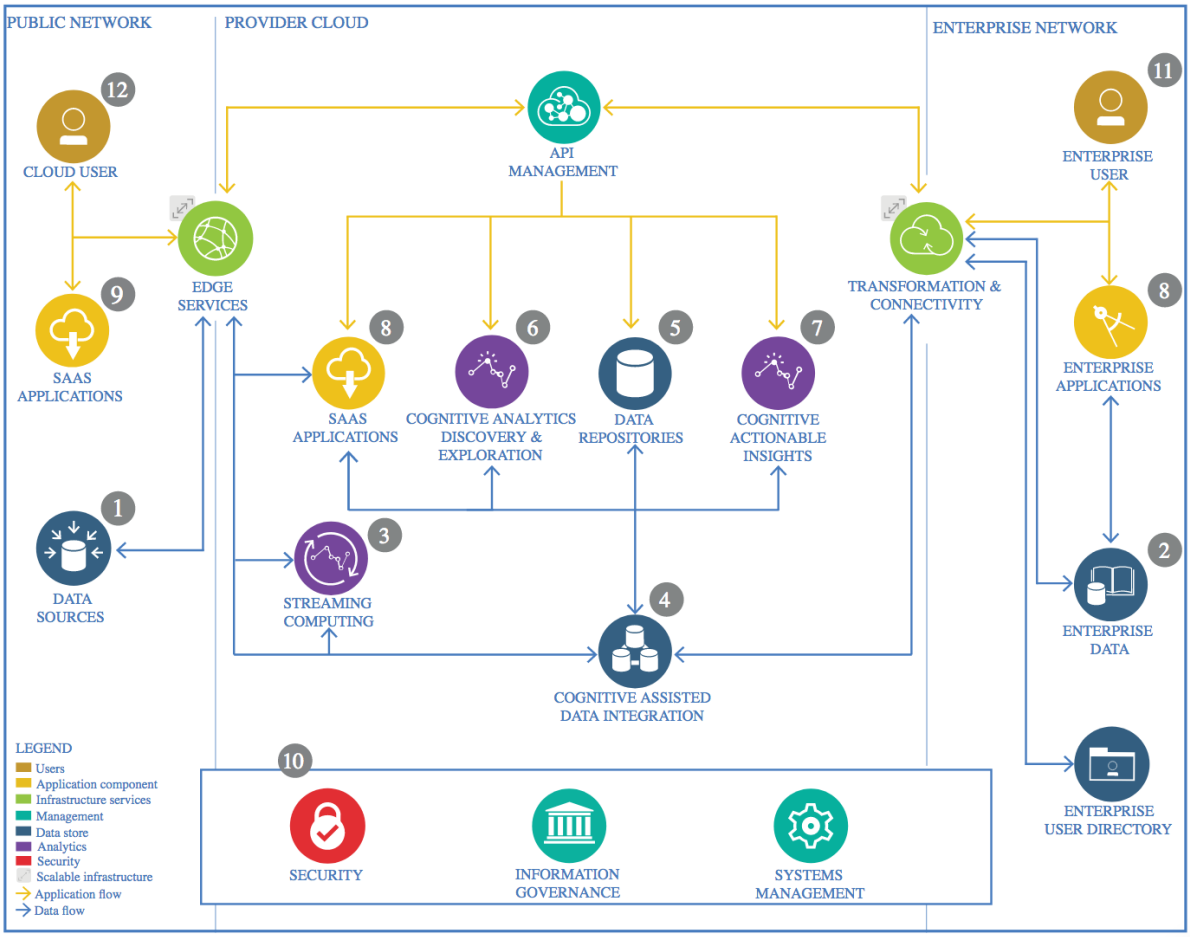
The Lightweight IBM Cloud Garage Method for Data Science
MD ROUNOK SALEHIN

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

Understanding the data is the key and most essential part of machine learning. In the use case, I am using the Kaggle competition dataset “Crimes in Chicago”. This dataset reflects reported incidents of crime (with the exception of murders where data exists for each victim) that occurred in the City of Chicago from 2001 to 2017, minus the most recent seven days. Data is extracted from the Chicago Police Department's CLEAR (Citizen Law

Enforcement Analysis and Reporting) system. The preliminary crime classifications may be changed at a later date based upon additional investigation and there is always the possibility of mechanical or human error. Therefore, the Chicago Police Department does not guarantee (either expressed or implied) the accuracy, completeness, timeliness, or correct sequencing of the information and the information should not be used for comparison purposes over time.

1.1.2 Justification

The dataset is very informative and easy to access (CSV format), having large size (2001-2017) with 23 feature attributes associated with crimes and data definition is clear to understand the domain history. I have used dataset for 2005-2017 consists of around 6M datapoints.

1.2 Data Integration

1.2.1 Technology Choice

- IBM Watson Studio jupyter notebooks, scikit-learn, pandas, matplotlib

1.2.2 Justification

1.2.2.1 Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modelling, data visualization, machine learning, and much more.

1.2.2.2 IBM Watson Studio

IBM Watson Studio provides tools for data scientists, application developers and subject matter experts to collaboratively and easily work with data to build and train models at scale. It gives you the flexibility to build models where your data resides and deploy anywhere in a hybrid environment so you can operationalize data science faster.

It’s important to notice that data integration is mostly done using ETL tools or plain SQL or a combination of both. ETL tools are very mature technology and an abundance of technologies exist. On the other hand, if streaming analytics is part of the project it is worth to check if one of those technologies fits the requirements since reuse of such a system reduces technology heterogeneity with all its advantages.

1.2.2.3 scikit-learn:

- Simple and efficient tools for data mining and data analysis with lots of open source libraries.
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib, Pandas
- Open source, commercially usable - BSD license

1.3 Data Repository

1.3.1 Technology Choice

- IBM Watson Studio jupyter notebooks, scikit-learn, pandas, matplotlib, IBM cloud object store.

There exists an extremely huge set of technologies for persisting data. Most of them are relational databases. The second largest group are NoSQL databases and file system (including Cloud Object Store) form the last one. The most important questions to be asked are:

- How does is the impact of storage cost?
- Which data types must be supported?
- How good must point queries (on fixed or dynamic dimensions) be supported?
- How good must range queries (on fixed or dynamic dimensions) be supported?
- How good must full table scans be supported?
- What skills are required?
- What's the requirement for fault tolerance and backup?
- What are the constant and peak ingestion rates?
- What's the amount of storage needed?
- How does the growth pattern look like?
- What are the retention policies?

1.3.2 Justification

For this use case data was stored in object store file system using pandas and data store APIs. One can store any kind of data, such as images, videos, documents, etc., in any format. We can upload objects up to 10TB in size. We can provision 100 buckets per Cloud Object Storage service instance. Objects can't exceed 200 MB in size when you are using the console to upload unless the Aspera high-speed transfer is installed. Using the Aspera high-speed transfer, you can upload larger size objects in the background instead of in the active browser window. In addition, the transfers can be viewed, paused or cancelled.

1.4 Discovery and Exploration

1.4.1 Technology Choice

- IBM Watson Studio jupyter notebooks, scikit-learn, pandas, matplotlib, seaborn

1.4.2 Justification

1.4.2.1 Jupyter, python, scikit-learn, pandas, matplotlib, seaborn

The components mentioned above are all open source and supported in the IBM Cloud. Some of them have overlapping features, some of them have complementary features. This will be made clear by answering the architectural questions

- What type of visualizations are needed?
    - Matplotlib/seaborn supports the widest range of possible visualizations including bar charts, run charts, histograms, box-plots and scatter plots.
- Are interactive visualizations needed?
    - Whereas matplotlib/seaborn creates static plots, pixiedust supports interactive ones.
- Are coding skills available / required?
    - Whereas matplotlib/seaborn needs coding skills, for computing metrics, some code is necessary in Python
- What metrics can be calculated on the data?
    - Using scikit-learn and pandas, all state-of-the-art metrics are supported
- Do metrics and visualization need to be shared with business stakeholders?
    - Watson Studio supports sharing of jupyter notebooks, also using a fine-grained user and access management system

1.5 Actionable Insights

1.5.1 Technology Choice

- IBM Watson Studio, Jupyter notebook, spark services, Keras model,scikit-learn

There exists an abundance of open and closed source technologies (IBM Watson Studio, Jupyter,spark service, Keras WML etc.). Here, the most relevant are introduced. Although it holds for other sections as well, decisions made in this section are very prone to change due to the iterative nature of this process model. Therefore, changing or combining multiple technologies is no problem, although decisions let to those changes should be explained and documented.

1.5.2 Justification

1.5.2.1 Python, pandas and scikit-learn

Python is a much cleaner programming language than R and easier to learn therefore. Pandas is the python equivalent to R dataframes supporting relational access to data. Finally, scikit-learn nicely groups all necessary machine learning algorithms together. It’s supported in the IBM Cloud via IBM Watson Studio as well.

- What are the available skills regarding programming languages?
    - Python skills are very widely available since python is a clean and easy to learn programming language.
- What is the cost of skills regarding programming languages?
    - Because of python's properties mentioned above, cost of python programming skills is very low
- What are the available skills regarding frameworks?
    - Pandas and scikit-learn are very clean and easy to learn frameworks, therefore skills are widely available.
- What is the cost of skills regarding frameworks?
    - Because of the properties mentioned above, cost of skills are very low.
- Is model interchange required?
    - All scikit-learn models can be (de)serialized. PMML is supported via 3rdparty libraries.
- Is parallel or GPU based training or scoring required?
    - Neither GPU nor scale-out is supported, although scale-up capabilities can be added individually to make use of multiple cores.
- Do algorithms need to be tweaked or new algorithms to be developed?
    - Scikit-learn algorithms are very cleanly implemented. They all stick to the pipeline's API making reuse and interchange easy. Linear algebra is handled throughout with the numpy library. So, tweaking and adding algorithms is straightforward.

1.5.2.2 Keras:

Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. It helps to quickly build and test a neural network with minimal lines of code, and you can build simple or very complex neural networks within a few minutes. The Model and the Sequential APIs are so powerful that one can do almost everything you may want.

1.6 Applications / Data Products

1.6.1 Technology Choice

- IBM Watson Studio, IBM Watson machine learning service, Jupyter notebook, scikit-learn.

1.6.2 Justification

1.6.2.1 IBM Watson machine learning service

As a data product I have deployed the model in IBM cloud using IBM Watson machine learning service in Watson Studio. It had client API can be imported to Jupyter notebook

easily and deploy and monitor model details on the fly. You also have the option to choose the deployment option, in our case it is web service, that exposes the end point to use via any applications.

1.7 Security, Information Governance and Systems Management

1.7.1 Technology Choice

- IBM cloud object store policies, IBM Watson studio policies.

1.7.2 Justification

All the project files (data, model data, model etc.) stored in IBM cloud object store file systems and adhere to the policies driven the platforms and services. Some examples as follows:

- Information stored in IBM Cloud Object Storage is encrypted and dispersed across multiple geographic locations. This service makes use of the distributed storage technologies provided by the IBM Object Storage System.
- Initially, only the bucket and object owners have access to the cloud object storage service instance they create. The service supports user authentication to access data; we can use access control mechanisms such as bucket policies to selectively grant permissions to users and applications. We can securely upload/download your data via SSL endpoints using the HTTPS protocol.
- If you need extra security, we can use the Key Protect Service or the Server-Side Encryption (SSE-C) with Customer-provided Keys option to encrypt data stored at rest. IBM Cloud Object Storage provides the encryption technology for both Key Protect and SSE-C. Both of these options provide server-side encryption.
- We can use Identity and Access Management (IAM) to access controlling mechanisms in order to secure your data. IAM policies enable organizations with multiple employees to create and manage multiple users under a single IBM Cloud account. With IAM policies, companies can grant IAM users control of their Cloud Object Storage service instance, buckets, etc.
- The data that is loaded into Spark service and notebooks is secure. Only the collaborators in project can access your data or notebooks. If we need to share notebook with the public, then it has option to hide your data service credentials in the notebook.