*Data are taken from internet, including web-sites with user’ reviews and comments. Sources are provided below.*

1. FRAMEWORK AND STORAGE :
   1. **Elastic Compute Cloud (Amazon EC2):**

**Pros**: EC2 users have to increase or decrease the infrastructure resources assigned to meet their needs.

**Cons**: Expensive at on-demand rates if elasticity is not needed (or expensive upfront payment if not using server for entire purchase length).

Performance can still vary.

* 1. **Amazon Simple Storage Service (S3):**

**Pros**: A web-scale service designed to store any amount of data.

The strength of its design center is performance and scalability. Users specify which region the Bucket should be deployed in.

**Cons:** Increased latency: Storing files on Amazon S3 means you'll need to deal with additional latency because, after files are uploaded to your server, they'll still have to be forwarded to the Amazon S3 infrastructure.

* 1. **Cloudera(Apache Hadoop)**:

**Pros**: Related projects (Hive, Pig, Oozie, HBase, Flume, Sqoop, etc.) tested together and work as a cohesive system.

A number of critical bug fixes and features and the most recent stable releases, so you’re usually using a more stable and feature-rich product.

**Cons:** Security Concerns.

Vulnerable By Nature (written in Java).

Not Fit for Small Data.

Potential stability issue.

* 1. **Google Cloud Storage :**

**Pros:** Storing files or data locally presents businesses with more security concerns whereas encrypted data on online storage services prevents unauthorized use or access in an easy way.

Low costs.

**Cons:** Slow downloading and uploading large documents.

Risk of technical problems.

2. DATA PROCESSING SOLUTION (BATCH, LOT, OR REAL TIME):

2.1 **Apache Spark:**

**Pros:** Spark comes with GraphX, a distributed graph system.

Spark is ideal for iterative processing, interactive processing and event stream processing.

It is very flexible and powerful.

Apache Spark Supports Machine Learning Algorithms for Future Predictions.

Same platform for real time and Batch Processing.

**Cons:** It consumes a lot of Memory.

**2.2 Apache Kafka:**

**Pros:** High achievable ingest rates with clear scaling pattern

High resiliency via distributed replicas with little impact on throughput

**Cons:** No current framework for monitoring and configuring producers

**2.3 Amazon Kinesis (similar to Kafka, in addition** Integrates with other AWS services)

3. ANALYTICS

3.1. **AmazonRedshift:**

**Pros:**

Redshift is very fast when it comes to loading data and querying it for analytical and reporting purposes.

High Performance:Redshift gains high performance using massive parallelism, efficient data compression, query optimization, and distribution.

Redshift is horizontally scalable. Whenever you need to increase the storage or need it to run faster, just add more nodes using AWS console.

Security: various ways to handle access control, data encryption etc.

**Attractive and transparent pricing**

**Cons:**

Not fast enough for live web apps

Redshift can load only Amazon S3 or relational DynamoDB using Massively Parallel Processing

**3.2. Teradata**

**3.3. Cloudera Impala**

**3.4. Presto**

4. VISUALISATION AND DASHBOARD:

4.1. **Tableau:**

**Pros:** A large collection of data connectors and visualizations. Intuitive design.

Powerful processing engine.

Mature product.

Large community of users.

**Cons:**

Not intuitive, meaning a significant training phase for successful deployments.

**4.2. Microsoft Power BI:**

**Pros:** Very powerful tool with loads of data source connectors. Very easy to use.

Superb data visualization capabilities.

**Cons:** Divides data prep tools between desktop and web versions.

Refresh cycle is only daily on free version.

**4.3. IBM Watson Analytics:**

**Pros:** Intuitive design.

Smart guidance.

Fast analytics.

Natural language querying.

**Cons:** Can't do real-time streaming analytics but can do near-real-time with 5-second refreshes.

Sources :

<http://www.dummies.com/programming/big-data/engineering/big-data-cloud-providers/>

<https://www.quora.com/What-are-the-pros-and-cons-of-using-Amazon-EC2>

<http://www.jscape.com/blog/amazon-s3-vs-local-storage-where-should-you-store-files-uploaded-to-your-file-transfer-server>

<http://nosql.mypopescu.com/post/20582276970/what-are-the-pros-and-cons-of-running-clouderas>

<http://www.bigdatacompanies.com/5-big-disadvantages-of-hadoop-for-big-data/>

<https://hubpages.com/technology/Advantages-and-Disadvantages-of-Cloud-Storage-Services>

<http://www.hadooptpoint.org/advantages-and-disadvantages-of-apache-spark/>

<https://www.phdata.io/binary-stream-ingest-flume-vs-kafka-vs-kinesis/>

<https://hevodata.com/blog/amazon-redshift-pros-and-cons/>