1.your ingestion tools

Rdbms -> sqoop is using in our company

web sources -> flume ,kafka,strom ,and spark steaming flink streaming

2.what are the tools available in streaming

flume, kafka, strom and spark streaming ,flint streaming

3.waht are the visual tools available

tomcat, clickview ,tablue, Zippeline and many but in our company **QlikView**

4 how log are genarated

online webservice, online businesses , online markieting sites,search engine , vendars

A transaction log is a file (i.e., log) of the communications (i.e., transactions) between a system and the users of that system,[[2]](https://en.wikipedia.org/wiki/Log_file#cite_note-2) or a data collection method that automatically captures the type, content, or time of transactions made by a person from a terminal with that system.[[3]](https://en.wikipedia.org/wiki/Log_file#cite_note-3) For Web searching, a transaction log is an electronic record of interactions that have occurred during a searching episode between a Web search engine and users searching for information on that Web search engine.

5 where will log stores

web log file will store web servers

The Log files in different web servers maintain different types of information. the basic

information present in the log file

6 when we should use hbase

hbase mainly use for OLTP purpose

7 what are the NOsql are there

hbase , cassandra mangodb and phenies

8.what is the deffernce between hadoop 1.x and hadoop 2.x

Major is

SPOF, Job LIMIT,Append, YARN,OS window also

9 what is your data format

ORC format

10 why are you using orc format

The Optimized Row Columnar (**ORC**) file **format** provides a highly efficient way to store Hive data. It was designed to overcome limitations of the other Hive file **formats**. Using **ORC** files improves performance when Hive is reading, writing, and processing data

Speed. Reading data from well partitioned, indexed, aggregated file will return very fast results.

Using ORC files improves performance when Hive is reading, writing, and processing data in HDFS. In terms of raw speed, it can be up to 3x as fast as raw text. It claims to be the smallest columnar storage format available for Hadoop — and after roughly 8 months of experimenting with it in a number of pipelines and situations, that claim might very well be true.

11. hive optimisation

12. active node down then standby will become active then who will do that

automatically it becomes active within 2 seconds by QJM( Quarrel Journal Node Manager)

13. how active node standby node connect to QJM(quorum Journal Node Manager)

|  |  |  |
| --- | --- | --- |
| **asis of Difference** | **Tableau** | **QlikView** |
| Capabilities | Real-time visualization | Strong visualizations with BI reporting |
| Suitable for | Dash boarding and Rapid visualization | Departmental and enterprise-wide BI solution |
| Product range | Diverse products for varied use | Single product for entire BI solution |
| Maps | Excellent in maps | Weaker in maps |
| Tables | Weaker in tables | Very good for tables |
| Speed | Speed depends on RAM and source database | Speed depends on RAM |

**When should you use HBase?**

After going through the above sections, we have acquired some idea about HBase. We also know about the supporting services and the key considerations for HBase deployment. As a NoSQL DB, HBase offers lot of good functionalities, but it is still not a **‘Fit for All’** solution. Following are some of the key areas to be considered before finalizing HBase for your application.

**Data volume:** The volume of data is the most common point to be considered. You should have peta bytes of data to be processed in a distributed environment. Otherwise, for a small amount of data, it will be stored and processed in a single node, keeping other nodes idle. So, it will be a misuse of technology framework.

**Application Types:** HBase is not suitable for transactional applications, large volume MapReduce jobs, relational analytics, etc. It is preferred when you have a variable schema with slightly different rows. It is also suitable when you are going for a key dependent access to your stored data.

**Hardware environment:** HBase runs on top of HDFS. And HDFS works efficiently with a large number of nodes (minimum 5). So, if you have good hardware support, then HBase can be a good selection.

**No requirement of relational features:** Your application should not have any requirement for RDBMS features like transaction, triggers, complex query, complex joins etc. If you can build your application without these features, then go for HBase.

**Quick access to data:** If you need a random and real time access to your data, then HBase is a suitable candidate. It is also a perfect fit for storing large tables with multi structured data. It gives ‘flashback’ support to queries, which makes it more suitable for fetching data in a particular instance of time.

Apart from the above points, HBase is also suitable when you need fault tolerant, fast and usable data management in a non-relational environment.

|  |  |  |
| --- | --- | --- |
| **Sl No** | **Hadoop1** | **Hadoop2** |
| **1** | Supports MapReduce (MR) processing model only. Does not support non-MR tools | Allows to work in MR as well as other distributed computing models like Spark, Hama, Giraph, Message Passing Interface) MPI & HBase coprocessors. |
| **2** | MR does both processing and cluster-resource management. | YARN (Yet Another Resource Negotiator) does cluster resource management and processing is done using different processing models. |
| **3** | Has limited scaling of nodes. Limited to 4000 nodes per cluster | Has better scalability. Scalable up to 10000 nodes per cluster |
| **4** | Works on concepts of slots – slots can run either a Map task or a Reduce task only. | Works on concepts of containers. Using containers can run generic tasks. |
| **5** | A single Namenode to manage the entire namespace. | Multiple Namenode servers manage multiple namespaces. |
| **6** | Has Single-Point-of-Failure (SPOF) – because of single Namenode- and in the case of Namenode failure, needs manual intervention to overcome. | Has to feature to overcome SPOF with a standby Namenode and in the case of Namenode failure, it is configured for automatic recovery. |
| **7** | MR API is compatible with Hadoop1x. A program written in Hadoop1 executes in Hadoop1x without any additional files. | MR API requires additional files for a program written in Hadoop1x to execute in Hadoop2x. |
| **8** | Has a limitation to serve as a platform for event processing, streaming and real-time operations. | Can serve as a platform for a wide variety of data analytics-possible to run event processing, streaming and real-time operations. |
| **9** | A Namenode failure affects the stack. | The Hadoop stack – Hive, Pig, HBase etc. are all equipped to handle Namenode failure. |
| **10** | Does not support Microsoft Windows | Added support for Microsoft windows |