**When should we use HBASE, list some of the scenarios for the same in real time**

**We should HBase in the following cases**

**1.Unstructured data**

**2.Large volume**

**3.Column-Oriented data**

**4.Versioned Data**

**5.High scalability**

**Unstructured Data:**

Since the data generated these days is highly unstructured and it not only contains text but also other formats such as audio and video file which cannot be stored in conventional RDBMS

So in this case we can use HBASE

**Large volume:**

Data generated these days is very large and data is also generated with high velocity.In order to store this large data conventional database requires high memory and It adds upto cost.So in that case we use HBase

**Column-Oriented data**

In conventional RDBMS we use a specific schema and data will be allocated regardless of any value is there or not further more say if we change the existing column of a table in new version in older DBMS this columns cannot be added but in this hbase since Data are stored in columnar manner this additional column can be added in fly

**Versioned Data**

In RDBMS if we update the record the old data will be deleted but in case Hbase we can maintain last 3 version of updated data

**High scalability**

Scalabilty can be achieved in this database

**Some of the important real time use cases are:**

**Apache HBase:**[**Powered By HBase Wiki**](http://wiki.apache.org/hadoop/Hbase/PoweredBy)

**Mozilla:**[**Moving Socorro to HBase**](http://blog.mozilla.com/webdev/2010/07/26/moving-socorro-to-hbase/)

**Facebook:**[**Facebook’s New Real-Time Messaging System: HBase**](http://highscalability.com/blog/2010/11/16/facebooks-new-real-time-messaging-system-hbase-to-store-135.html)

**StumbleUpon: [HBase at StumbleUpon](http://www.stumbleupon.com/devblog/hbase_at_stumbleupon/" \t "_about)**

**What are the different modes in which Hbase can be run?**

**Hbase runs in two modes**

**1.Standalone Mode**

**2.Distributed Mode**

**Standalone Mode**

1. In standalone mode, HBase does not use HDFS -- it uses the local filesystem instead

**2.** it runs all HBase daemons and a local ZooKeeper all up in the same JVM

3. Zookeeper binds to a well known port so clients may talk to HBase.

4.It is the default mode

**Distributed Mode** 1. Distributed mode can be subdivided into distributed but all daemons run on a single node -- a.k.a pseudo-distributed-- and fully-distributed where the daemons are spread across all nodes in the cluster

2. Distributed modes require an instance of the Hadoop Distributed File System (HDFS).

Pseudo Distributed Mode

1. Configuration is required in given 3 files for this mode
2. Replication factory is one for HDFS.
3. Here one node will be used as Master Node / Data Node / Job Tracker / Task Tracker
4. Used for Real Code to test in HDFS.
5. Pseudo distributed cluster is a cluster where all daemons are  
   running on one node itself.

 Full Distributed mode

1. This is a Production Phase
2. Data are used and distributed across many nodes.
3. Different Nodes will be used as Master Node / Data Node / Job Tracker / Task Tracker

**Need and working of zookeeper in Hbase?**

**zookeeper**

1.Zookeeper is an open-source project that provides services like maintaining configuration information, naming, providing distributed synchronization, etc.

2.Zookeeper has ephemeral nodes representing different region servers. Master servers use these nodes to discover available servers.

3.In addition to availability, the nodes are also used to track server failures or network partitions.

4.Clients communicate with region servers via zookeeper.

5.In pseudo and standalone modes, HBase itself will take care of zookeeper.