hadoop=framework

->storage+caculation (distributed)

# distributed storage(PB level)

just data,not os,stored in multiple computers

e.g. BaiduCloud 360Cloud

in Hadoop,HDFS(Hadoop Distributed File System)

# distributed caculation

a task -> subtasks one task by many computers

target:reduce compute time

tranditional compute:drive -> memory IO cost

~:parallel IO

in Hadoop:MR(Map Reduce)

# version

Apache (official)

Cloudera(commercial,introduced some patch)

HortonWork(integrated apache)

# module

common:serial,rpg communication,

HDFS:storage

yarn:resource management platform

mapreduce:caculation

# derived framework

Ambari : Management Monitor

Avro : Serial

Cassandra : no single fault

Chukwa : Data collection (from all nodes to different destination-db,console)

HBase : Big Table,Structed data warehouse

Hive : Data aggregate & query instantly

Mahout : Machine learning & data mining library

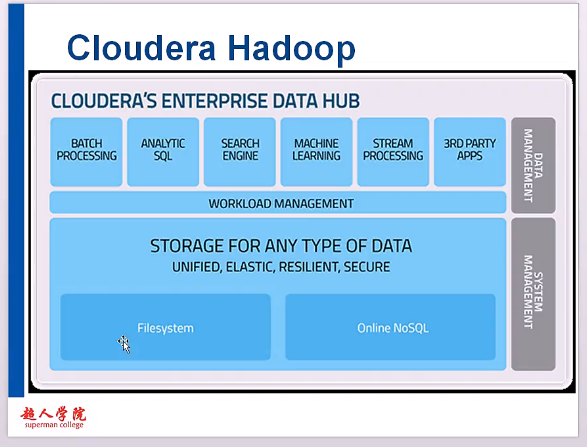
Pig : superior data stream language

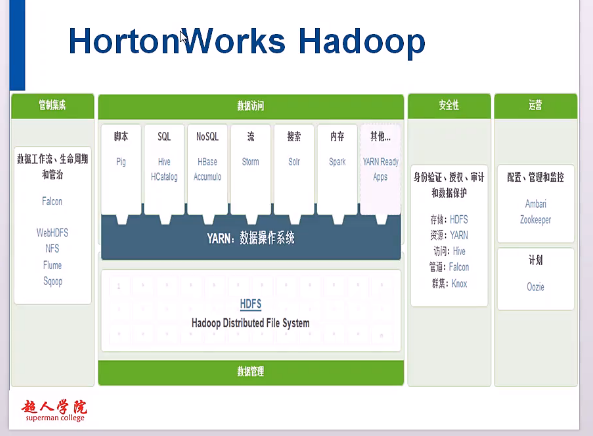
Spark : (In Memory) Common caculation engine

Tez : Common caculation stream framework

Zookeeper : a framework for coporation

# 3rd party-Hadoop







# Hadoop Components



compare:

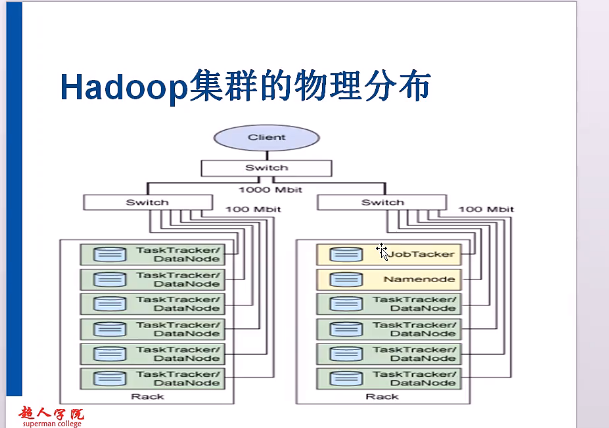
Map-Reduce:Batch processing

Storm:Streaming data process

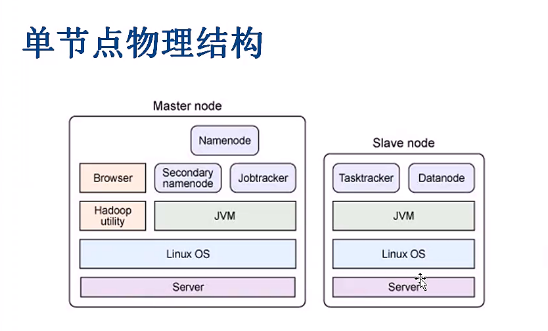
Spark:Caculate in memory

Distrubuted App:(Storm,Spark,HDFS...)

1. Master-Slave Structure
   1. HDFS(NameNode(M),Datanode(S))
   2. YARN(ResourceManager(M),Nodemanager(S))
   3. MR(JobTracker(M),TaskTracker(S))
   4. Storm(Nimbus(M),Supervisor(S))

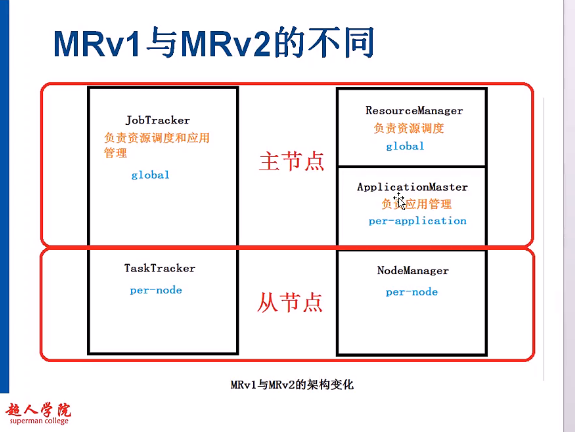


代码一样 进程不一样



Hadoop1 jdk1.6

Hadoop2 jdk1.7



左边列是v1 右边v2

ApplicationMaster 是YARN对外接口