

# About me

- Hadoop Tech Leader in Baidu
- HADOOP NUMBERS in Baidu
  - Nodes: 1.5w
  - Input data/per day: > 10PB
  - Clusters: 10
  - Big Cluster: 3000 Nodes
  - *The most busy Hadoop Clusters in the world*

# Agenda

- Community - Hadoop 2.0
  - HDFS 2.0
  - MapReduce 2.0
- Baidu - Hadoop 2.0
  - Baidu - HDFS 2.0, HDFS 3.0
  - Baidu - MapReduce 2.0
- TODO
  - CloudTransfer
  - MR-ontime(App Stability)
  - Big, Big, Big Cluster?

# Community-HDFS2.0-Scalability

- Scalability
  - 文件数、块数
  - 负载

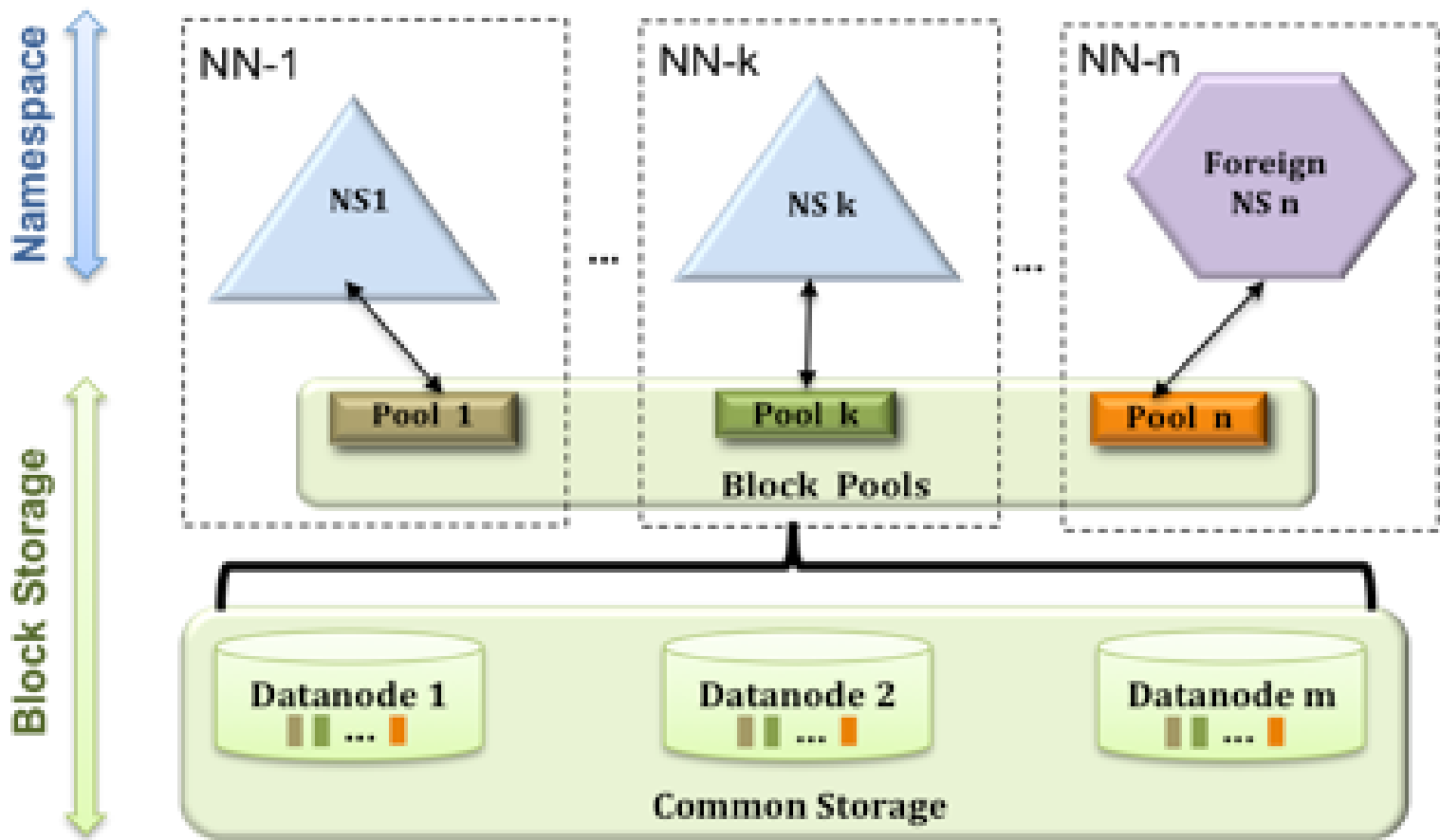
## Cluster Summary

156146781 files and directories, 120438664 blocks = 276585445 total.

Heap Memory used 89.51 GB is 69% of Committed Heap Memory 127.95 GB. Max Heap Memory is 127.95 GB.

Non Heap Memory used 29.99 MB is 66% of Committed Non Heap Memory 45.42 MB. Max Non Heap Memory is 132 MB.

- HDFS Federation
  - 4 months
  - HDFS-1052
  - hadoop-0.23 (coming soon, 2011-11)



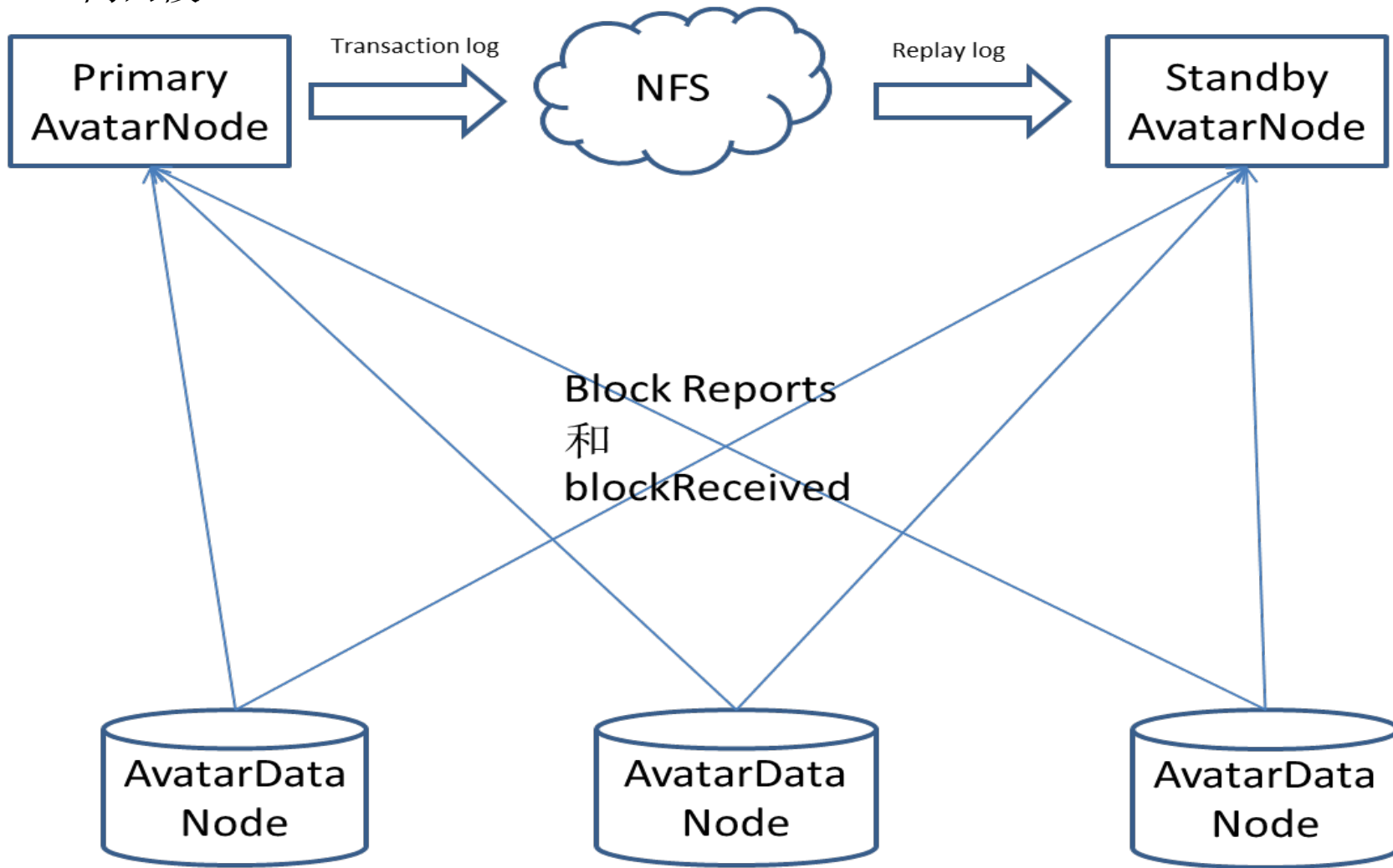
# Community-HDFS2.0-Availability

- Availability
  - NameNode单点
  - 1.5亿文件+1.5亿块+2000节点：重启花费40分钟
- Avatar NameNode
- Backup NameNode

# Facebook-Avatar NameNode 架构

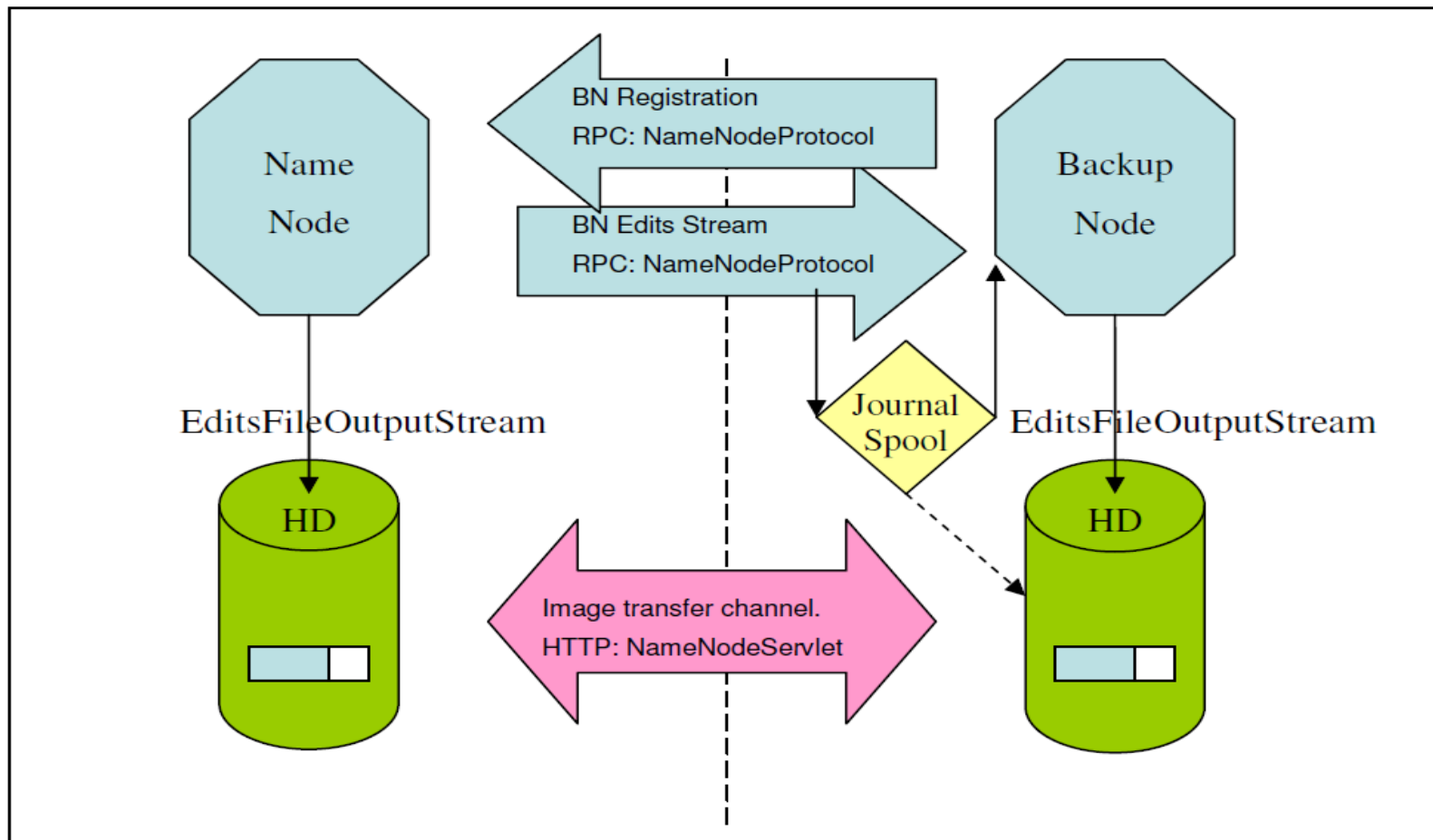


- 1. NetApp Filer: 100w
- 2. VIP – 同网段



# 社区-Backup NameNode 架构

1. 复杂
2. Backup Node 的问题可能造成服务不稳定

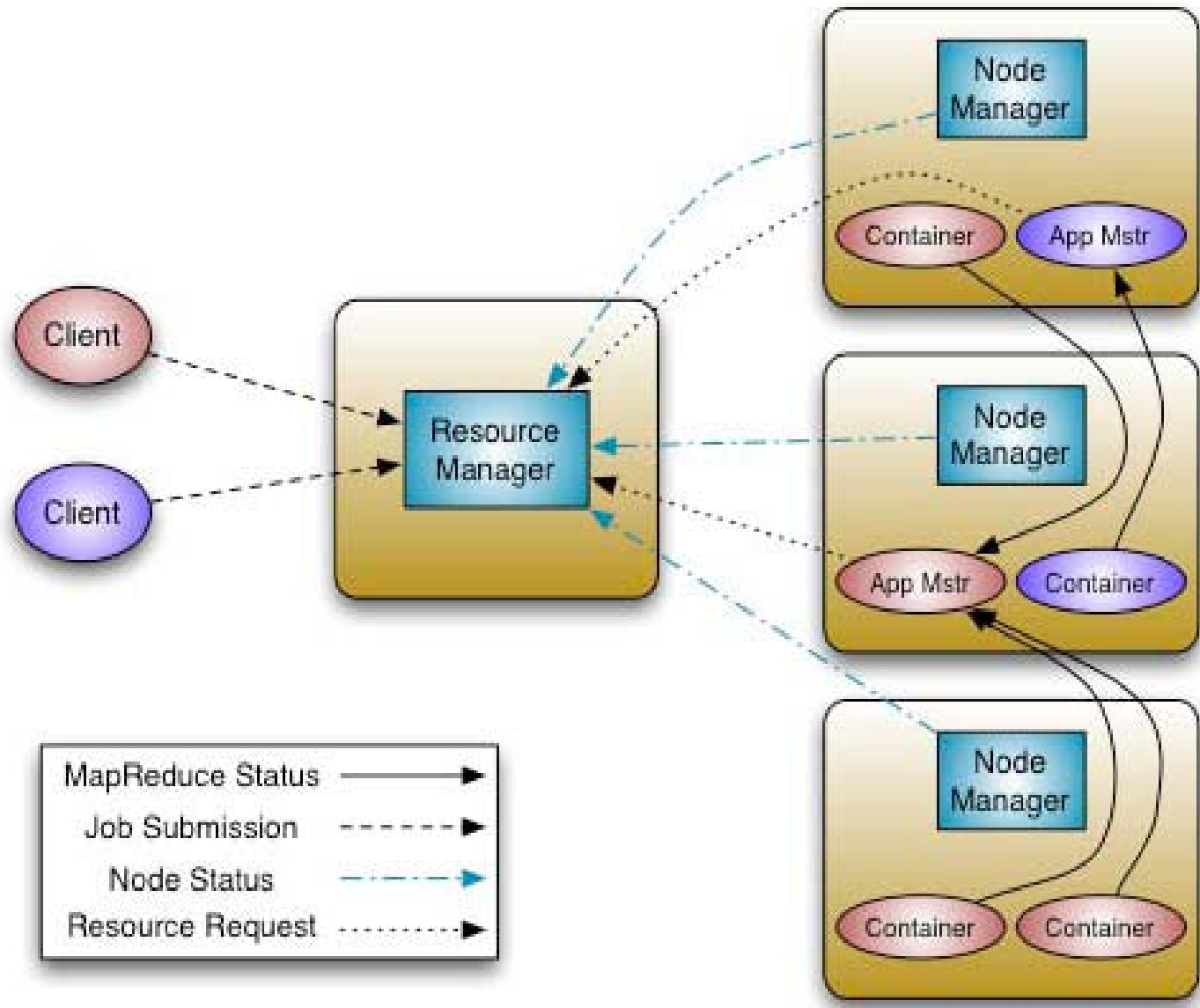


- MapR – HDFS
  - 待补充.....



# Community-MR2.0

- Next MapReduce (MapReduce 2.0)
  - Scalability
    - Cluster resource management
    - Application lifecycle management
  - Utilization
    - cpu, memory, io, network
    - remove fixed partition of map and reduce slots
  - Support for programming paradigms other than MR
    - MPI
    - Multi-version Hadoop
    - Spark
    - .....
- 100人月， hadoop-0.23

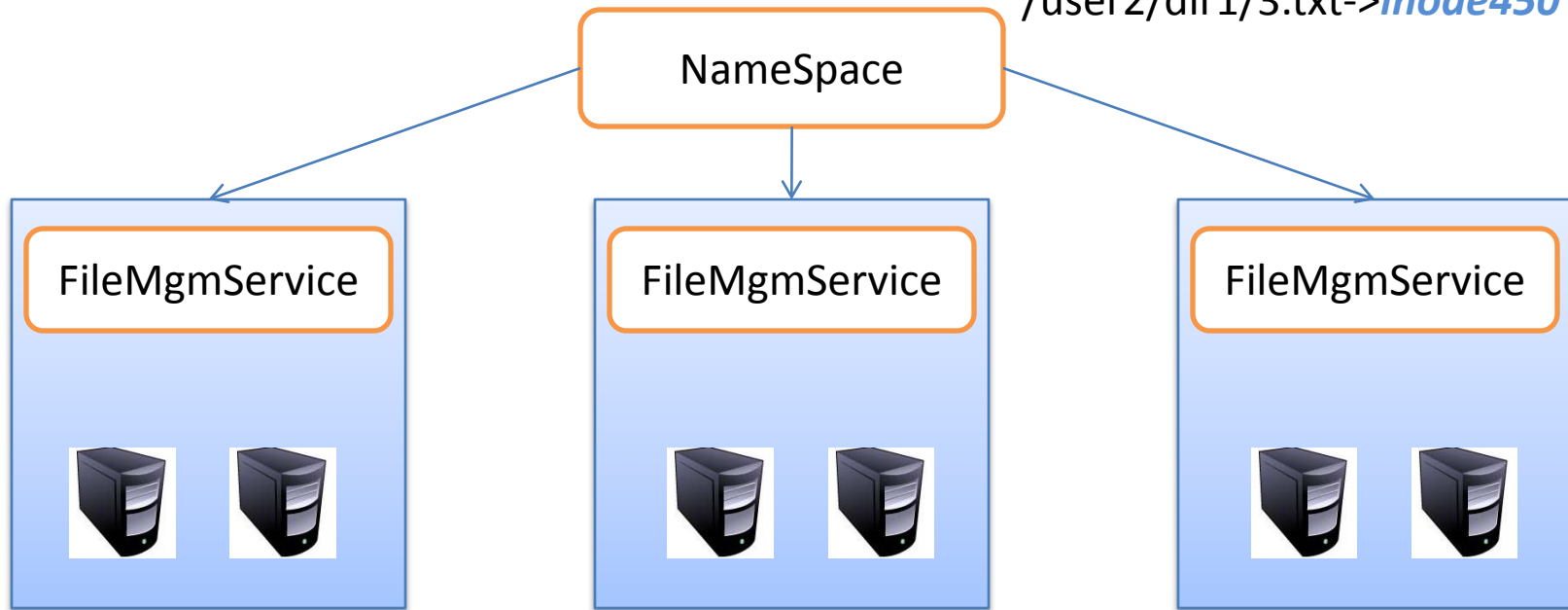


# Baidu-HDFS2-Scalability

- 3kw个文件，3kw个块
  - 块管理 $\approx 7.8\text{G}$ ，包括全部块副本信息
  - 目录树 $\approx 4.3\text{G}$ ，目录层次结构，包含文件块列表信息
- 到10亿文件、10亿块的数据规模
  - 块管理 $\approx 240\text{GB}$
  - 目录树 $\approx 140\text{GB}$
- 负载
  - 集群规模扩大后，单点的NameNode请求压力也会同时增大

## Baidu-hdfs2-architecture

/user1/1.txt -> **inode120**  
/user2/2.txt -> **inode231**  
/user2/dir1/3.txt->**inode450**



**inode120**->

user,group, rwxr-x---, size, repl  
blkX, blkY, blkZ

**inode231** -> xxxxxx

inodexxx -> xxxxxx

**inode450**->

user,group, rwxr-x---, size, repl  
blkX, blkY, blkZ

## 支持各类名字空间

Tree-NameSpace

Flat-NameSpace

MySQL/File/Memo  
ry/Your Brain

FileMgmService



FileMgmService



FileMgmService



- HDFS2

- 内存: 10亿文件, 10亿块

- Namespace: 66GB文件数据 + 1GB目录
    - 单节点就可以管理

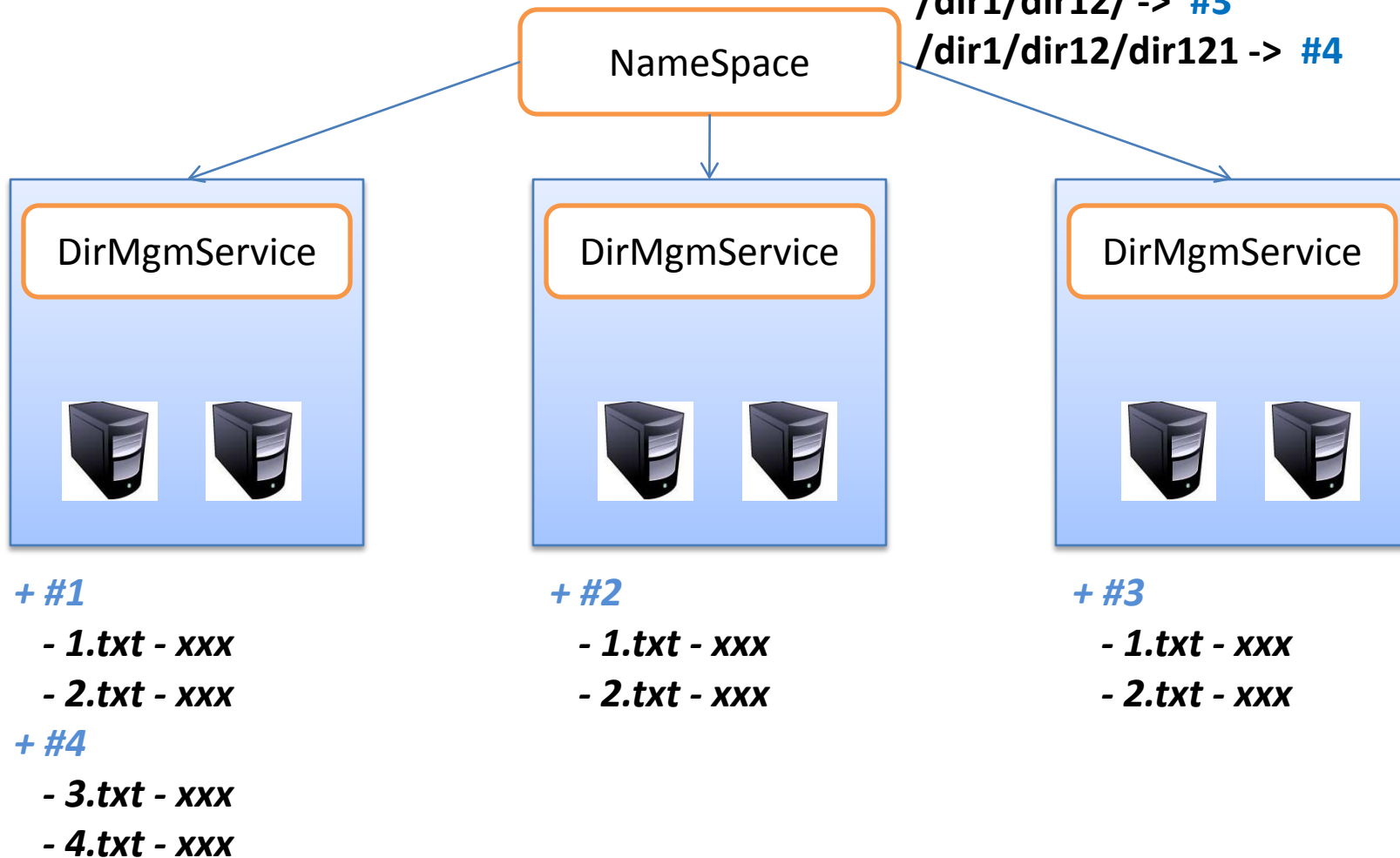
- 请求负载:

- 大部分耗时操作都属于文件对象管理层, 不用经过Namespace<sup>[2]</sup>
    - 最耗CPU资源的若干操作中, 仍需经过Namespace的只占13.7%<sup>[2]</sup>
    - 命名空间管理不再维护块信息, 大部分操作都不需要加全局锁, 可以更充分利用CPU资源

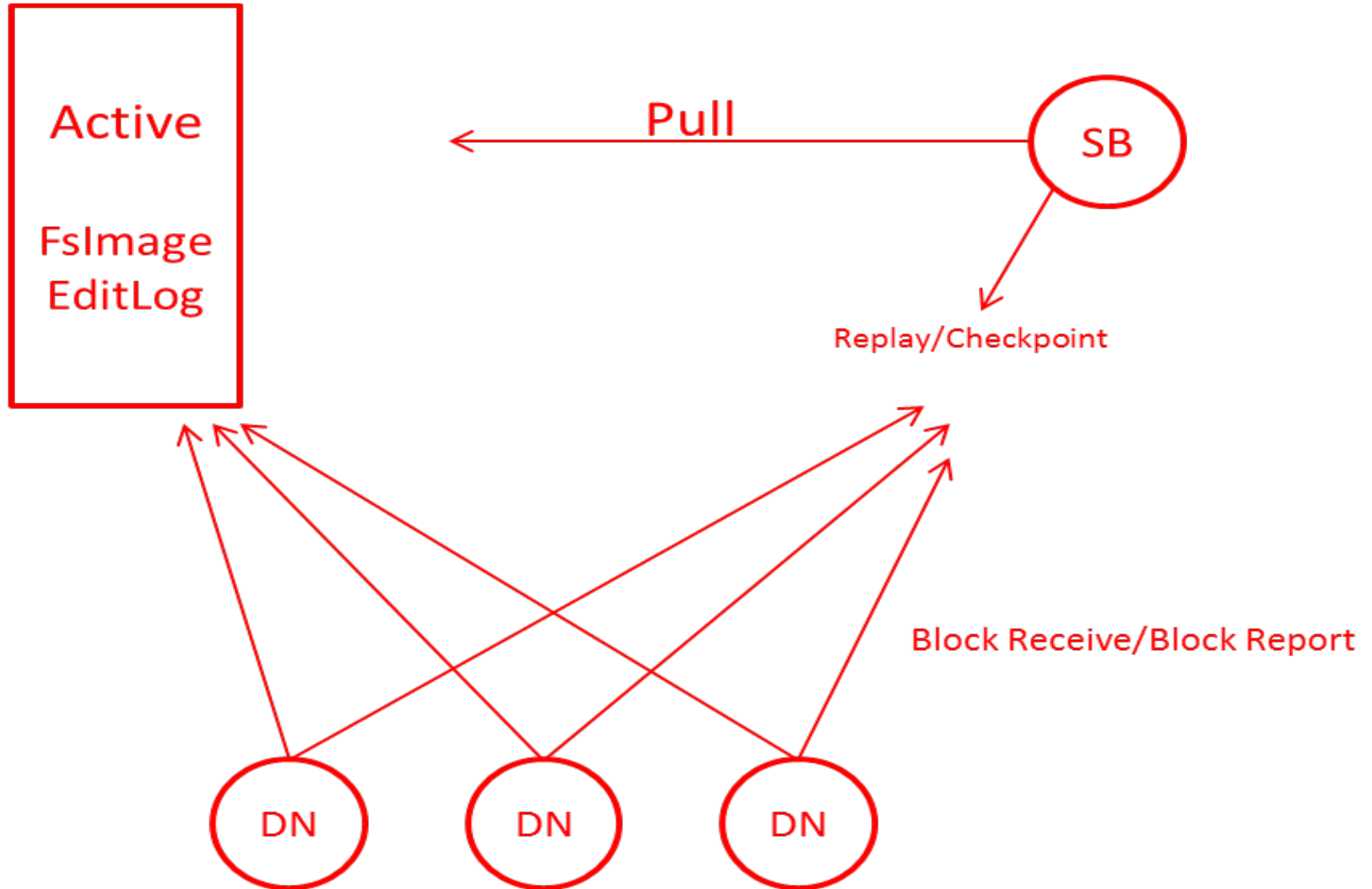
## Baidu-hdfs3-architecture (?, 如有必要)

- 1) File可以rename, 但是不能在目录间移动
- 2) Dir可以mv
- 3) 文件数能力可以支撑到100亿到200亿, 负载继续下移

**/dir1/ -> #1**  
**/dir1/dir11/ -> #2**  
**/dir1/dir12/ -> #3**  
**/dir1/dir12/dir121 -> #4**



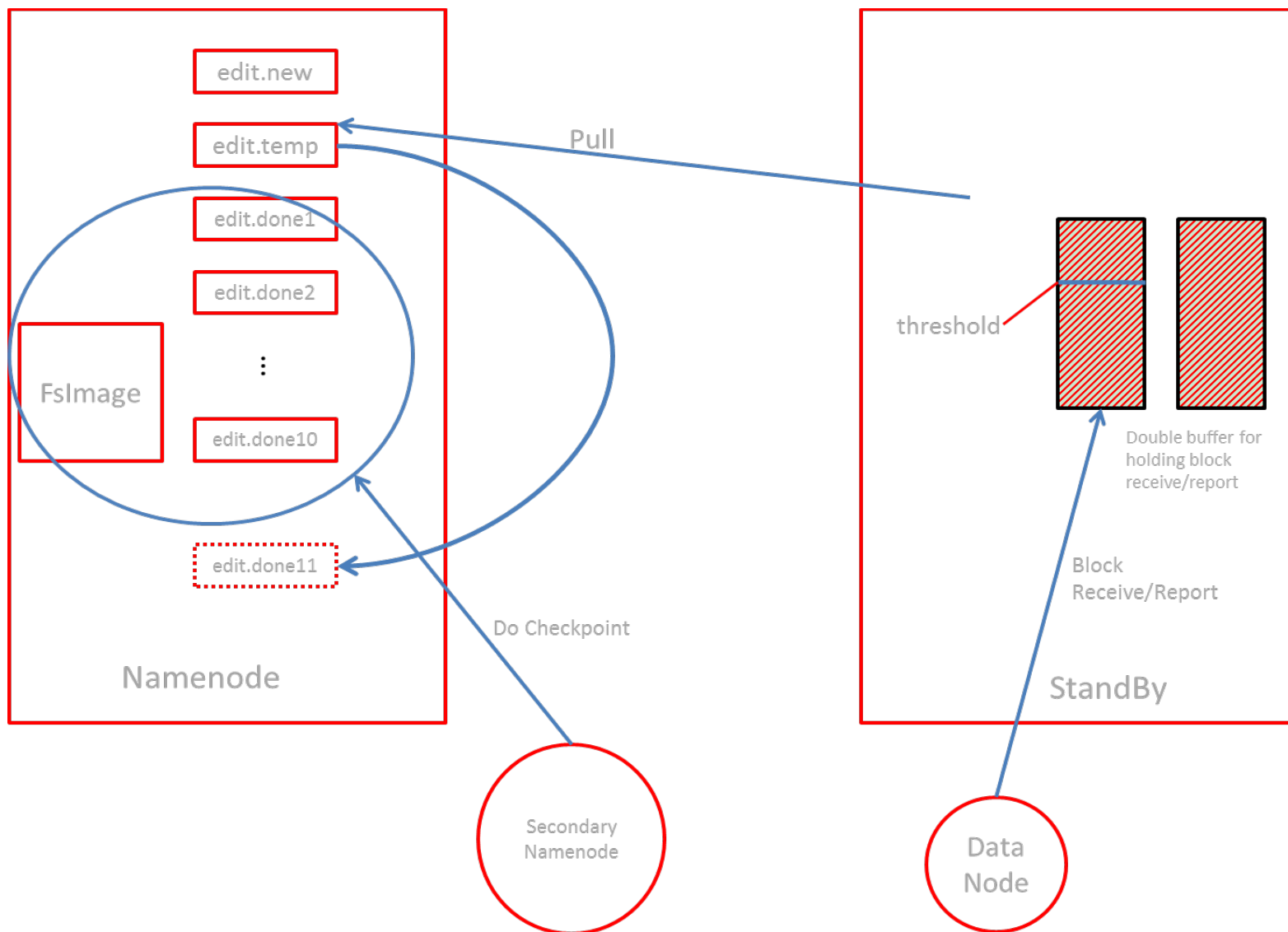
# Baidu-HDFS2-Availability



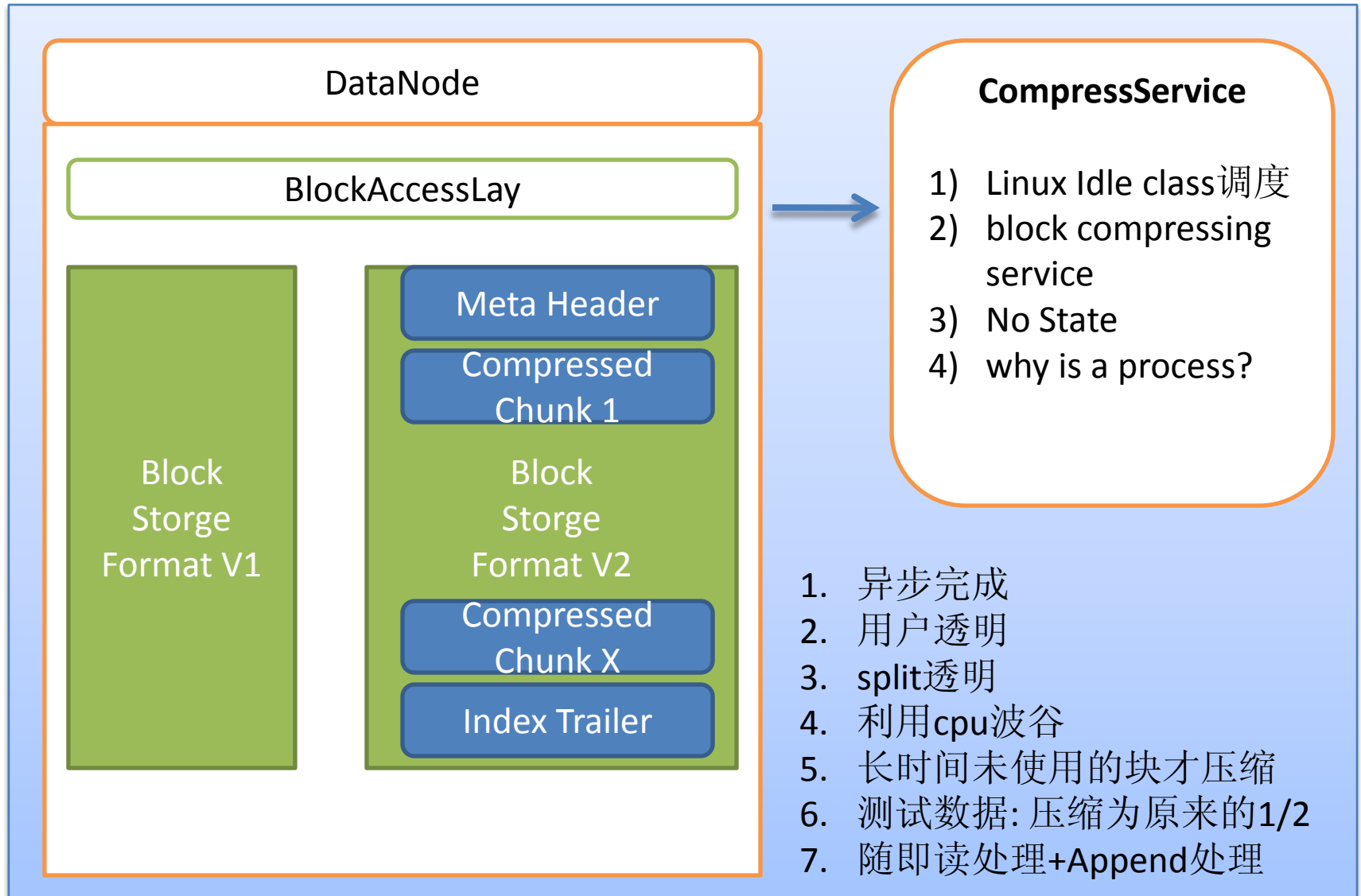


## Baidu-hdfs2-availability

- 1) 允许丢失1分钟的数据
- 2) 改动简单，对namenode服务影响最小

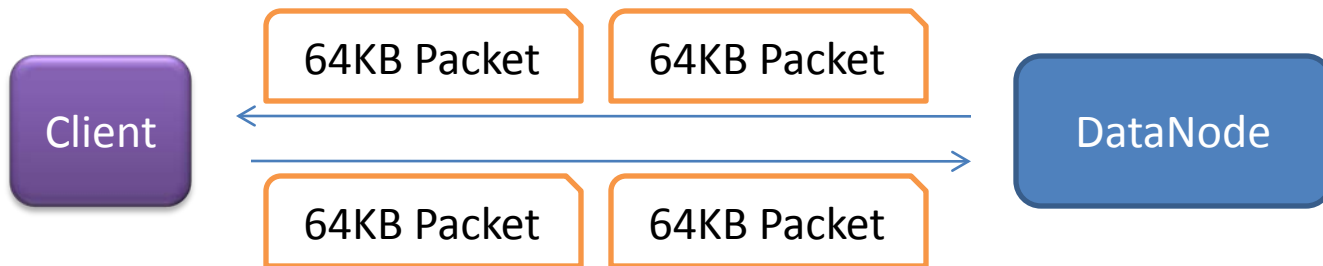


# Baidu-HDFS2-透明压缩

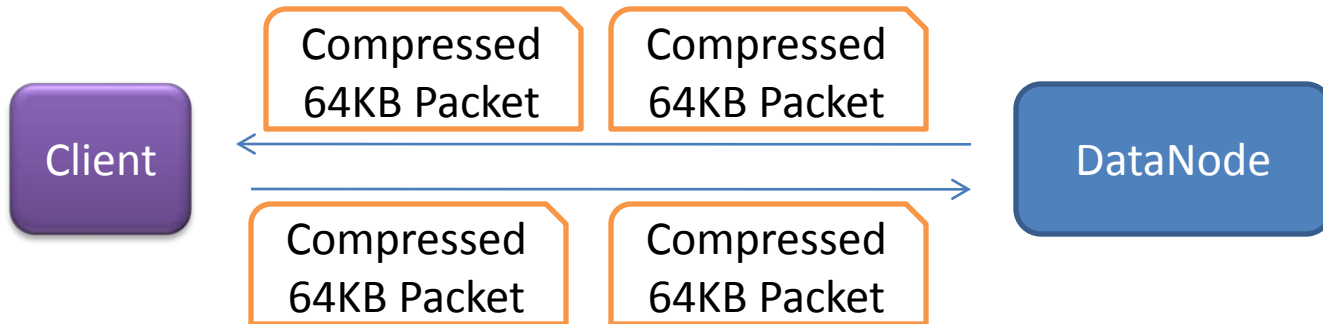


# Baidu-HDFS2-透明传输

改进前



改进后

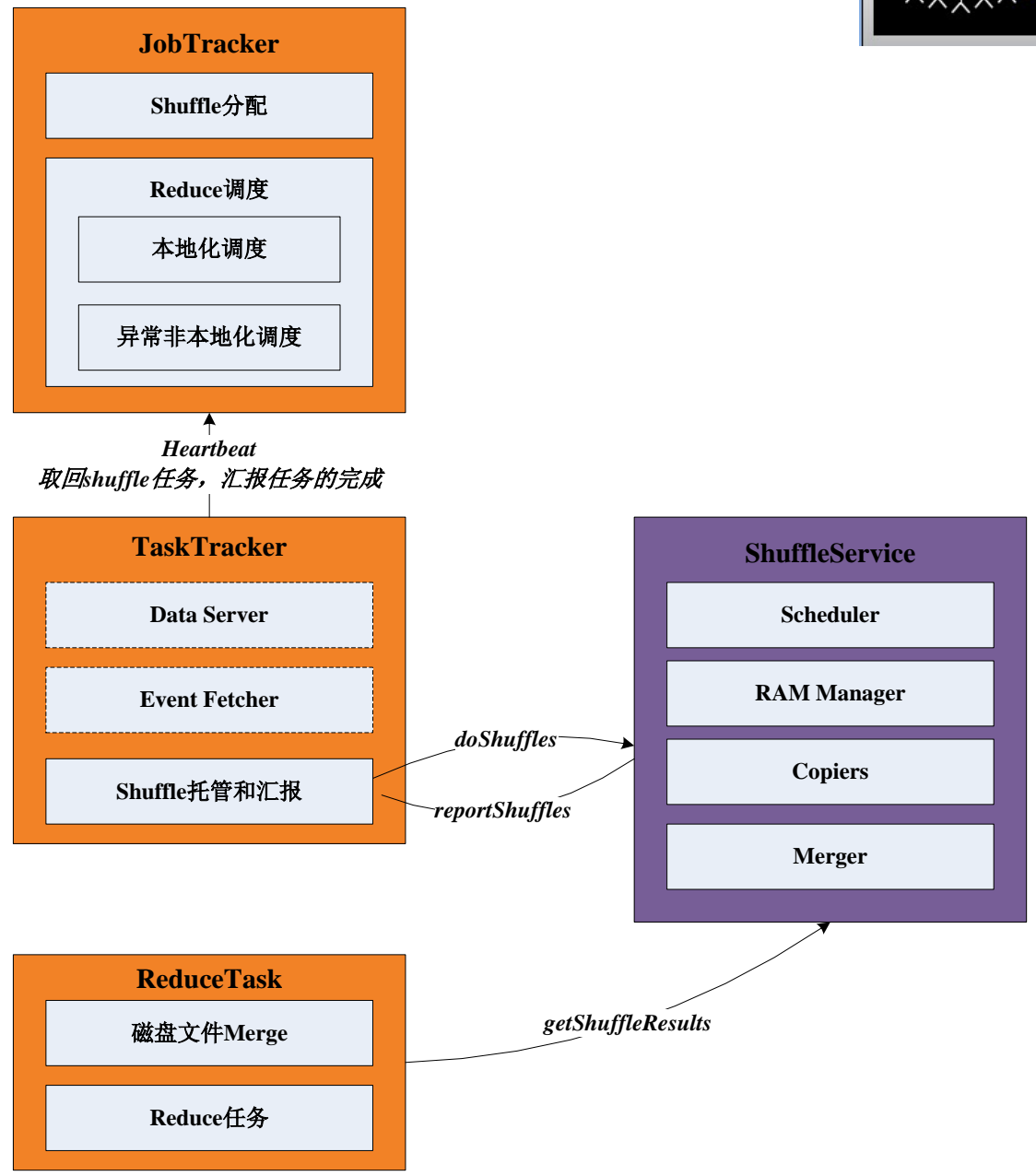


测试数据: 100MB/s -> 175MB/s

# Baidu-MR2.0

- Shuffle独立 \*
- Map和Reduce Task同治调度
- 资源调度
  - 去除slots, 按cpu、mem调度
  - 资源隔离: cgroup、Linux container
- Scalability - Distributed JobTracker \*
- AppTracker + JobTracker
- Availability – JobTracker Failover
  - job recover
  - task recover

# Shuffle独立架构



# Hadoop job\_201108291638\_0002 on [jx-hadoop-r](#)



User: root

Job Name: streamjob34428.jar

Job Priority: NORMAL

Job Map Capacity: 100

Job Map Over Capacity Allowed: true

Job Reduce Capacity: 100

Job Reduce Over Capacity Allowed: true

Job Queue: defaultqueue

Job Groups: default

Job File: [hdfs://jx-hadoop-rmaster.jx:34002/system/mapred/job\\_201108291638\\_0002/job.xml](hdfs://jx-hadoop-rmaster.jx:34002/system/mapred/job_201108291638_0002/job.xml)

Job Setup: Successful

Status: Running

Started at: Mon Aug 29 16:49:15 CST 2011

Running for: 1mins, 30sec

Job Cleanup: Pending

Job Error: No error

Shuffle and Reduce Cache Info: [Shuffle and Reduce-Cache Info](#)

| Kind                   | % Complete | Num Tasks | Pending             | Running            | Complete            | Killed | <a href="#">Failed/Killed Task Attempts</a> |
|------------------------|------------|-----------|---------------------|--------------------|---------------------|--------|---|
| <a href="#">map</a>    | 63.28%<br> | 420       | <a href="#">122</a> | <a href="#">70</a> | <a href="#">228</a> | 0      | 0 / 0                                       |
| <a href="#">reduce</a> | 0.00%<br>  | 200       | <a href="#">200</a> | 0                  | 0                   | 0      | 0 / 0                                       |

|              | Counter              | Map            | Reduce | Total          |
|--------------|----------------------|----------------|--------|----------------|
| File Systems | HDFS bytes read      | 34,356,154,491 | 0      | 34,356,154,491 |
|              | Local bytes read     | 7,852,015,752  | 0      | 7,852,015,752  |
|              | Local bytes written  | 15,722,691,206 | 0      | 15,722,691,206 |
| Job Counters | Rack-local map tasks | 0              | 0      | 82             |
|              | Launched map tasks   | 0              | 0      | 298            |

# Hadoop job\_201108291638\_0001 on [jx-hadoop-rm](#)



User: root

Job Name: streamjob26250.jar

Job Priority: NORMAL

Job Map Capacity: 100

Job Map Over Capacity Allowed: true

Job Reduce Capacity: 100

Job Reduce Over Capacity Allowed: true

Job Queue: defaultqueue

Job Groups: default

Job File: [hdfs://jx-hadoop-rmaster.jx:34002/system/mapred/job\\_201108291638\\_0001/job.xml](hdfs://jx-hadoop-rmaster.jx:34002/system/mapred/job_201108291638_0001/job.xml)

Job Setup: Successful

Status: Running

Started at: Mon Aug 29 16:42:46 CST 2011

Running for: 3mins, 38sec

Job Cleanup: Pending

Job Error: No error

Shuffle and Reduce Cache Info: [Shuffle and Reduce-Cache Info](#)

| Kind                   | % Complete   | Num Tasks | Pending            | Running            | Complete            | Killed | <a href="#">Failed/Killed Task Attempts</a> |
|------------------------|--|-----------|--------------------|--------------------|---------------------|--------|---|
| <a href="#">map</a>    | <div style="width: 100.00%;"><div style="width: 100.00%;"></div></div> 100.00% | 420       | 0                  | 0                  | <a href="#">420</a> | 0      | 0 / 0                                       |
| <a href="#">reduce</a> | <div style="width: 44.25%;"><div style="width: 44.25%;"></div></div> 44.25%    | 200       | <a href="#">87</a> | <a href="#">70</a> | <a href="#">43</a>  | 0      | 0 / 0                                       |

|              | Counter               | Map            | Reduce        | Total          |
|--------------|-----------------------|----------------|---------------|----------------|
| File Systems | HDFS bytes read       | 60,026,968,560 | 0             | 60,026,968,560 |
|              | HDFS bytes written    | 0              | 7,499,480,000 | 7,499,480,000  |
|              | Local bytes read      | 13,429,715,177 | 1,712,680,519 | 15,142,395,696 |
|              | Local bytes written   | 27,233,186,398 | 0             | 27,233,186,398 |
| Job Counters | Launched reduce tasks | 0              | 0             | 113            |
|              | Rack-local map tasks  | 0              | 0             | 124            |
|              | Launched map tasks    | 0              | 0             | 420            |



# Shuffle Service Administration



## Ram Manager

| shuffleBufferMegabytes | maxSingleShuffleLimit | startMergePercent(%) | PercentUsed(%) | waitForMemoryThreads | numStarted | numClosed | maxInMemOutputs | m  |
|------------------------|-----------------------|----------------------|----------------|----------------------|------------|-----------|-----------------|----|
| 1024                   | 33554432              | 80.0                 | 0.0            | 0                    | 0          | 0         | 20000           | 0. |

## Shuffle Result

| ShuffleID                 | JobPriority | ResultStatus | ResultFileNum | ResultFileList  |
|---------------------------|-------------|--------------|---------------|---|
| job_201108291638_0001-195 | NORMAL      | DONE         | 2             | &file:/home/disk4/dcmmapred/shuffleService/job_201108291638_0001/195/output/map_283.out&/shuffleService/job_201108291638_0001/195/output/map_46.out |
| job_201108291638_0001-198 | NORMAL      | DONE         | 2             | &file:/home/disk6/dcmmapred/shuffleService/job_201108291638_0001/198/output/map_351.out&/shuffleService/job_201108291638_0001/198/output/map_2.out  |
| job_201108291638_0001-197 | NORMAL      | DONE         | 2             | &file:/home/disk7/dcmmapred/shuffleService/job_201108291638_0001/197/output/map_176.out&/shuffleService/job_201108291638_0001/197/output/map_46.out |
| job_201108291638_0001-199 | NORMAL      | DONE         | 2             | &file:/home/disk8/dcmmapred/shuffleService/job_201108291638_0001/199/output/map_352.out&/shuffleService/job_201108291638_0001/199/output/map_46.out |

## ShuffleWorks

| shuffleWork | totalMaps | remainingMaps | usedMem | inMemorySegments | onDiskSegments | totalFailures | pendingHosts | fetchFailedMaps | penalties |
|-------------|-----------|---------------|---------|------------------|----------------|---------------|--------------|-----------------|-----------|
|-------------|-----------|---------------|---------|------------------|----------------|---------------|--------------|-----------------|-----------|

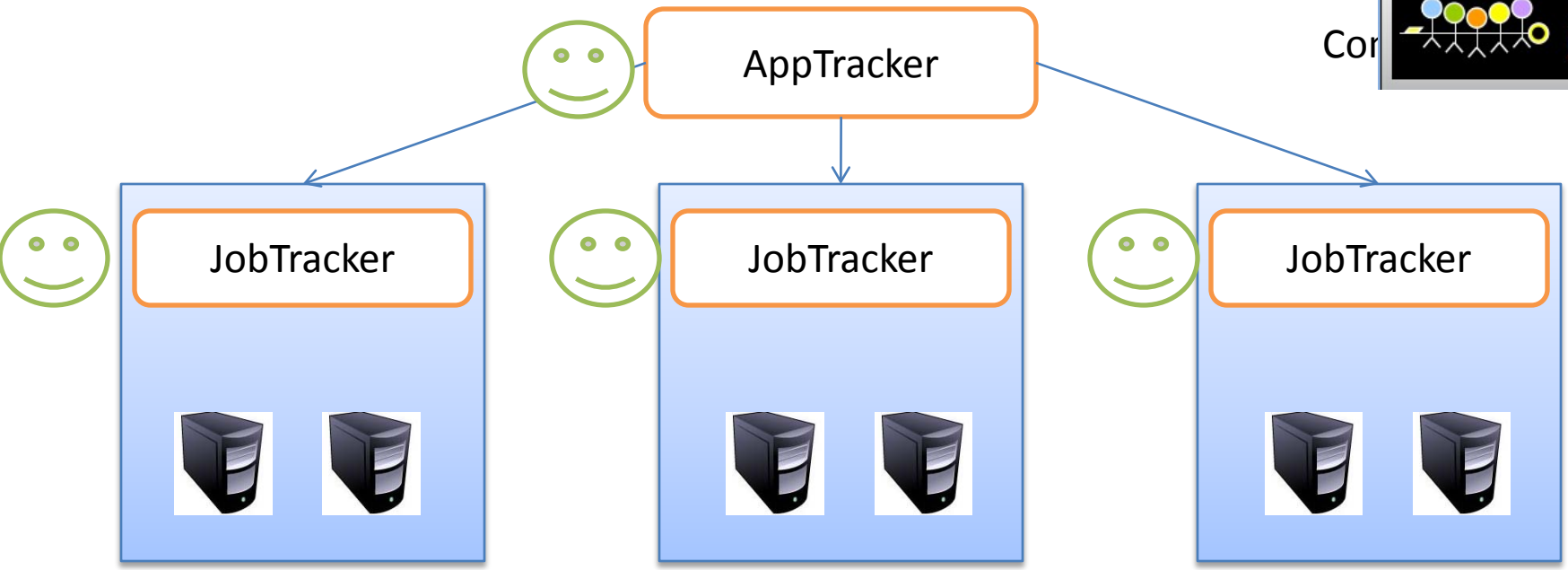
## ShuffleCopiers

| Copier-Id | Running ShuffleWork |
|-----------|---------------------|
| 1         | null                |
| 2         | null                |
| 3         | null                |
| 4         | null                |
| 5         | null                |

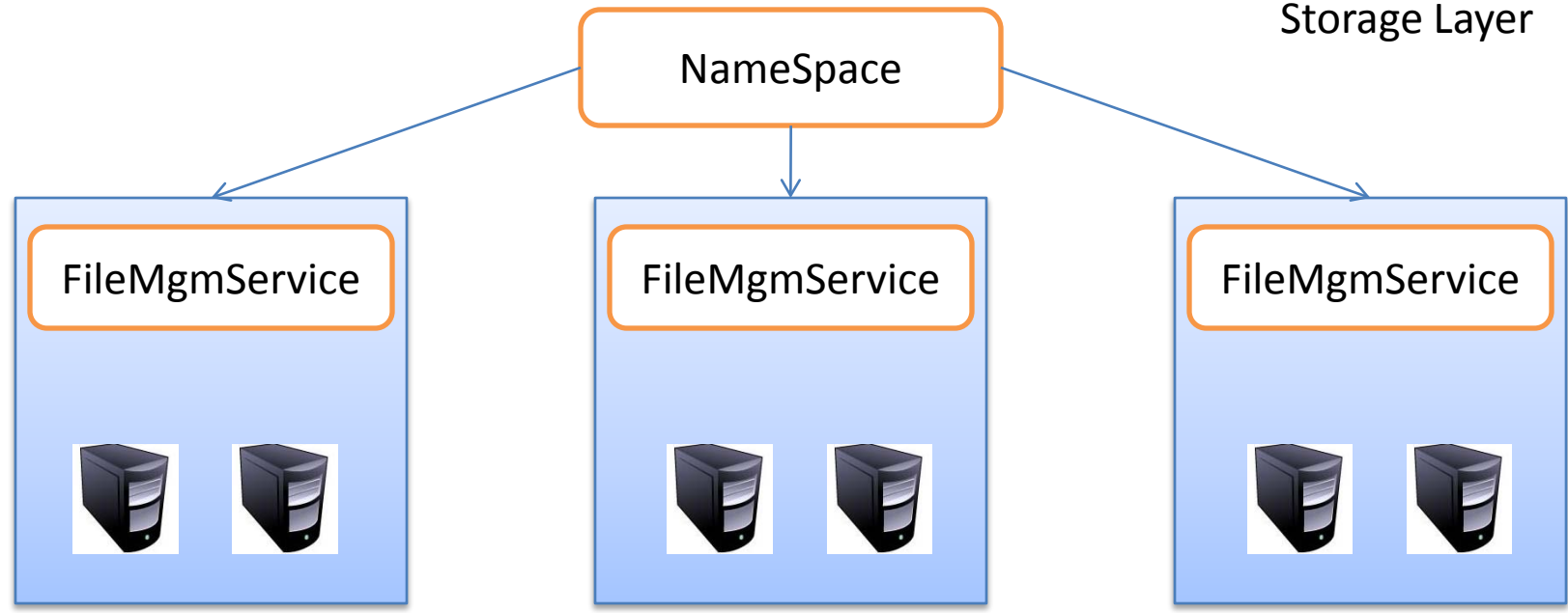


- Scalability - Distributed JobTracker
  - 一个app有多个job组成
  - jobtracker可以独立运行， apptracker只是更高层的一个应用
  - 实现非常简单
  - 规模可以做到很大
  - 需要底层存储支持
  - 适合未来跨集群完成计算
  - 可以融合进Community-MapReduce2.0

Cor



Storage Layer



# TODO- CloudTransfer

- 流式传输
  - flume, scribe
- 批量传输
  - 节点间数据移动
  - 一到一
  - 一到多（树状分发，解决类似jar包分发，cachearchive问题）
  - 节点到hdfs, hdfs到节点
  - hdfs到hdfs
  - 第3方控制

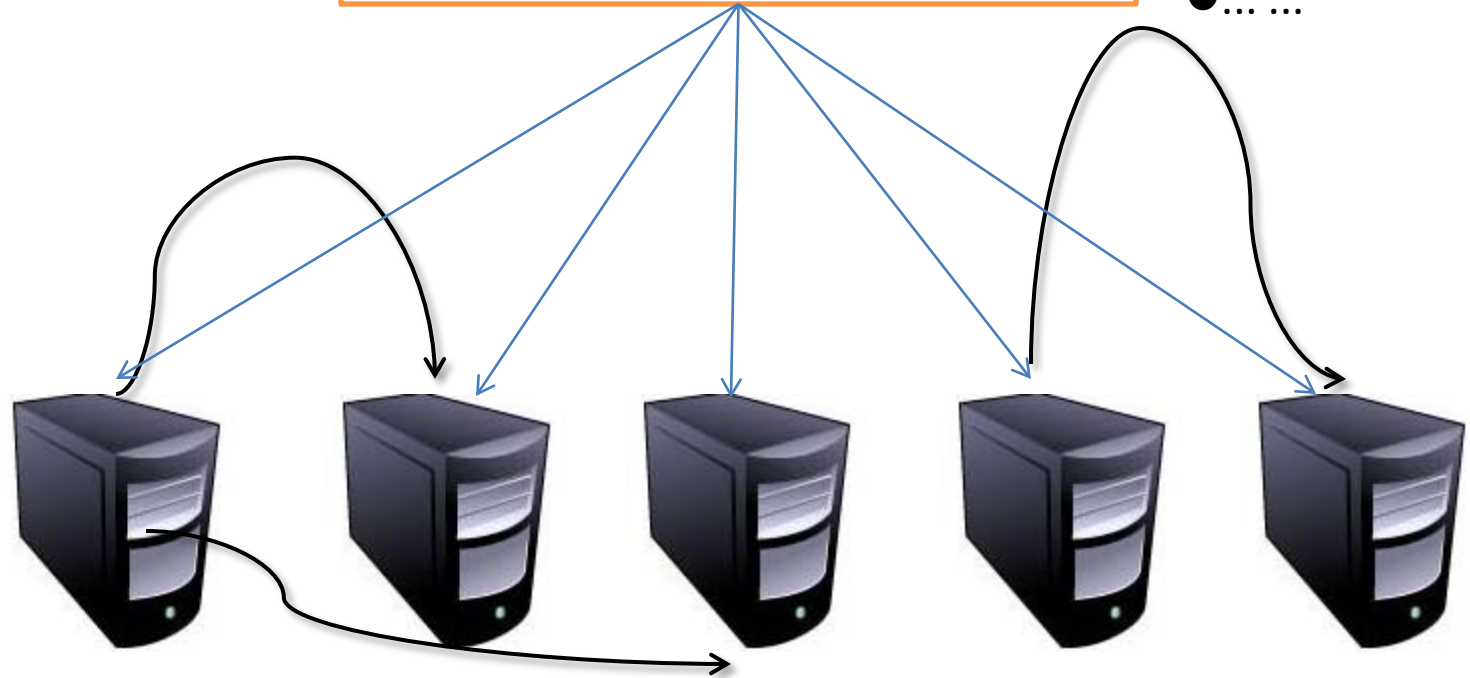
### Master

- task-1: A:/1 -> B:/1
- task-2: C:/2 -> B:/2, D:/3, E:/4
- task-3: F:/5 -> indexserver-\*/5



- 优先级控制
- 带宽控制
- 动态调整
- 出错控制
- stop/resume
- 广域网优化
- ... ..

- compress
- small file 优化
- 带宽控制



# TODO-MR-Ontime

- throughput
  - 时效性不保证(> 2小时)
  - 追求: 吞吐最高(资源利用率)
- realtime
  - 时效性最高(< 10分钟时效性要求)
  - 追求: 完成latency
  - Twitter-Storm/Linkedin-Kafka
- ontime
  - 时效性中等(10分钟 – 2小时)
  - 追求: 就如同地铁, 也许速度不快, 但是预期稳定
- 追求不同, 侧重点就要不同

computing-realtime

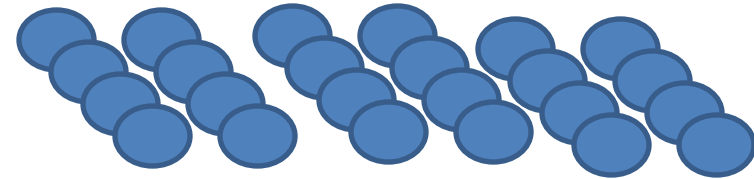
computing-ontime

computing-throughput

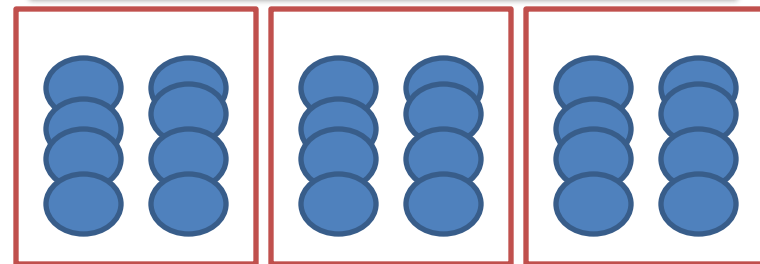
# TODO-Big,Big,Big Cluster?

- Datacenter
  - 1w
- 系统复杂度提高
- 问题影响面太大
- 系统升级也是个大麻烦
- 解决思路:
  - “中央集权”走向“地方分权”
  - storage.com, computing.com

中央集权



元调度



Q & A

THANKS