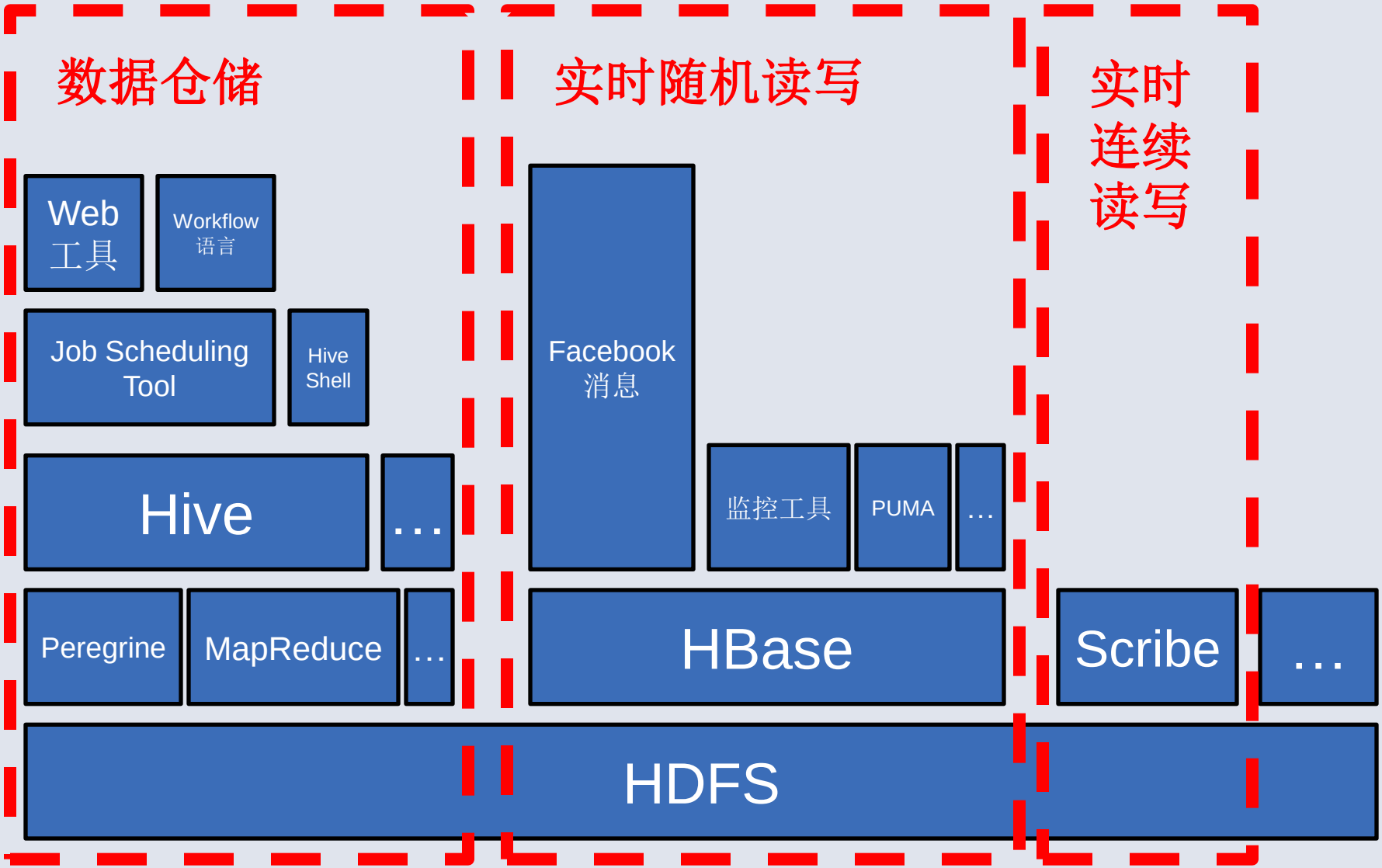
facebook

Facebook开发的HDFS和HBase新特性

董思颖，软件工程师， Facebook数据基础设施组

siying.d@fb.com

Facebook对HDFS及Hbase的使用



代码 GitHub!

Hadoop:

https://github.com/facebook/hadoop-20

Hadoop稳定版:

https://github.com/facebook/hadoop-20/tree/production

HBase: https://github.com/apache/hbase/tree/0.89-fb

HDFS的新特性

HDFS广泛的新需求和新挑战

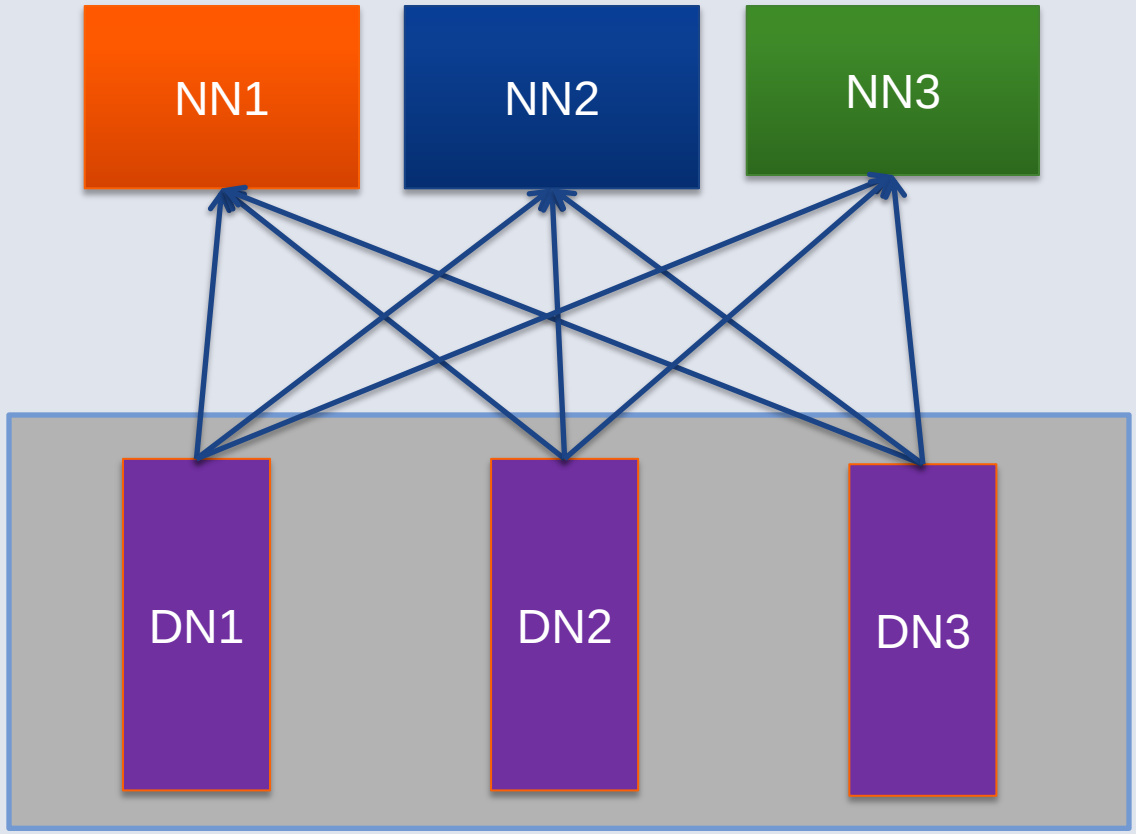
	数据仓储	HBase	Scribe	数据备份和恢复
NameNode 不停机升级	高	高	中	低
NameNode自动故障转移	中	高	中	低
文件数量多	高	低	低	高
数据容量大	高	低	低	低
文件客户端永久存在	低	高	高	低
大量随机读	低	高	低	低
节省存储空间	高	中	低	中
减少高延迟读写	低	高	中	低
DataNode存储大量Block	高	低	低	中
隔离不同应用	中	低	低	低
远程客户端	中	低	中	低

HDFS Scalability

如何使用密度更高的机器，存储更多的数据和更多的文件？

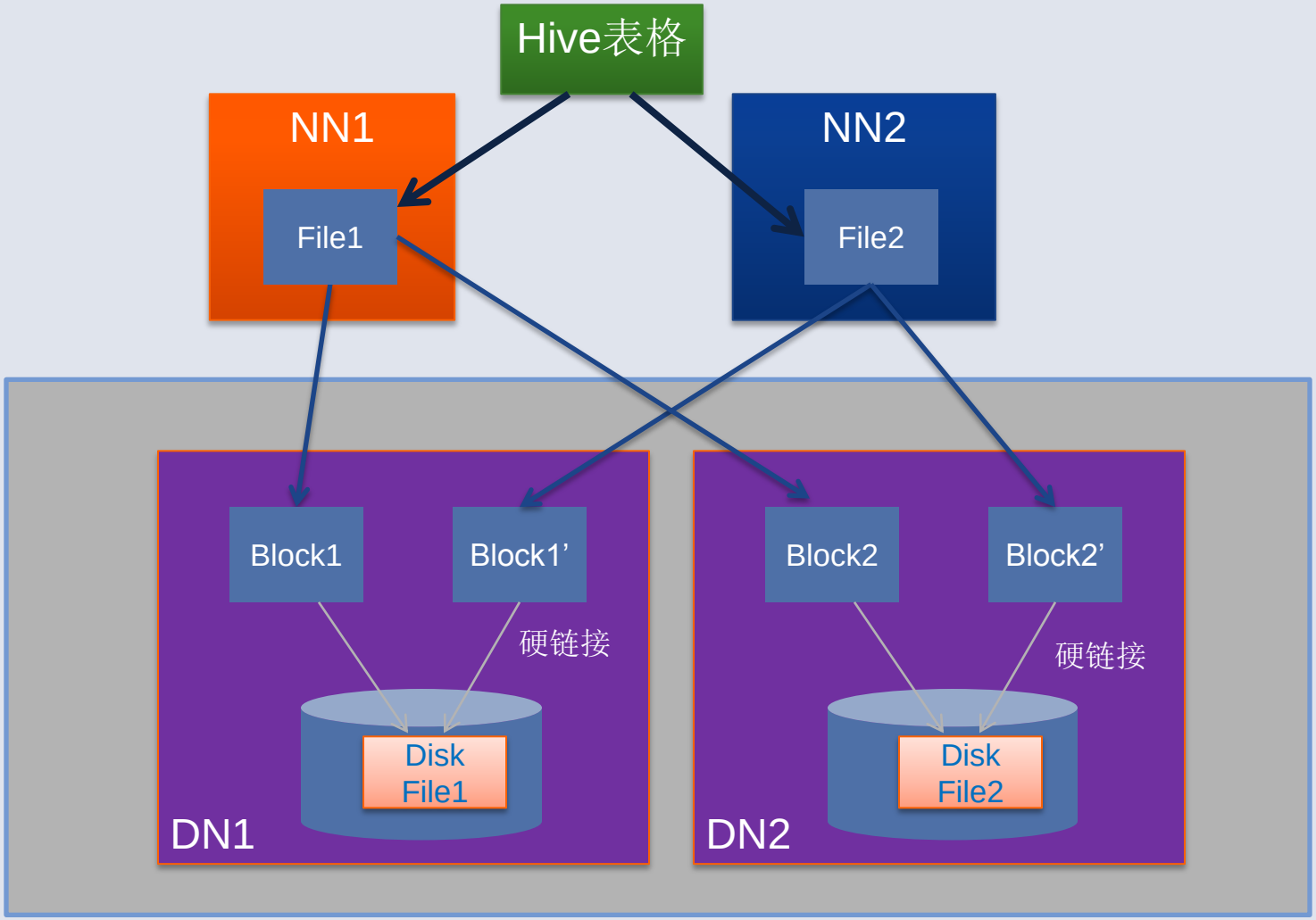
NameNode Scalability – Federation (1)

什么是Federation



NameNode Scalability – Federation (2)

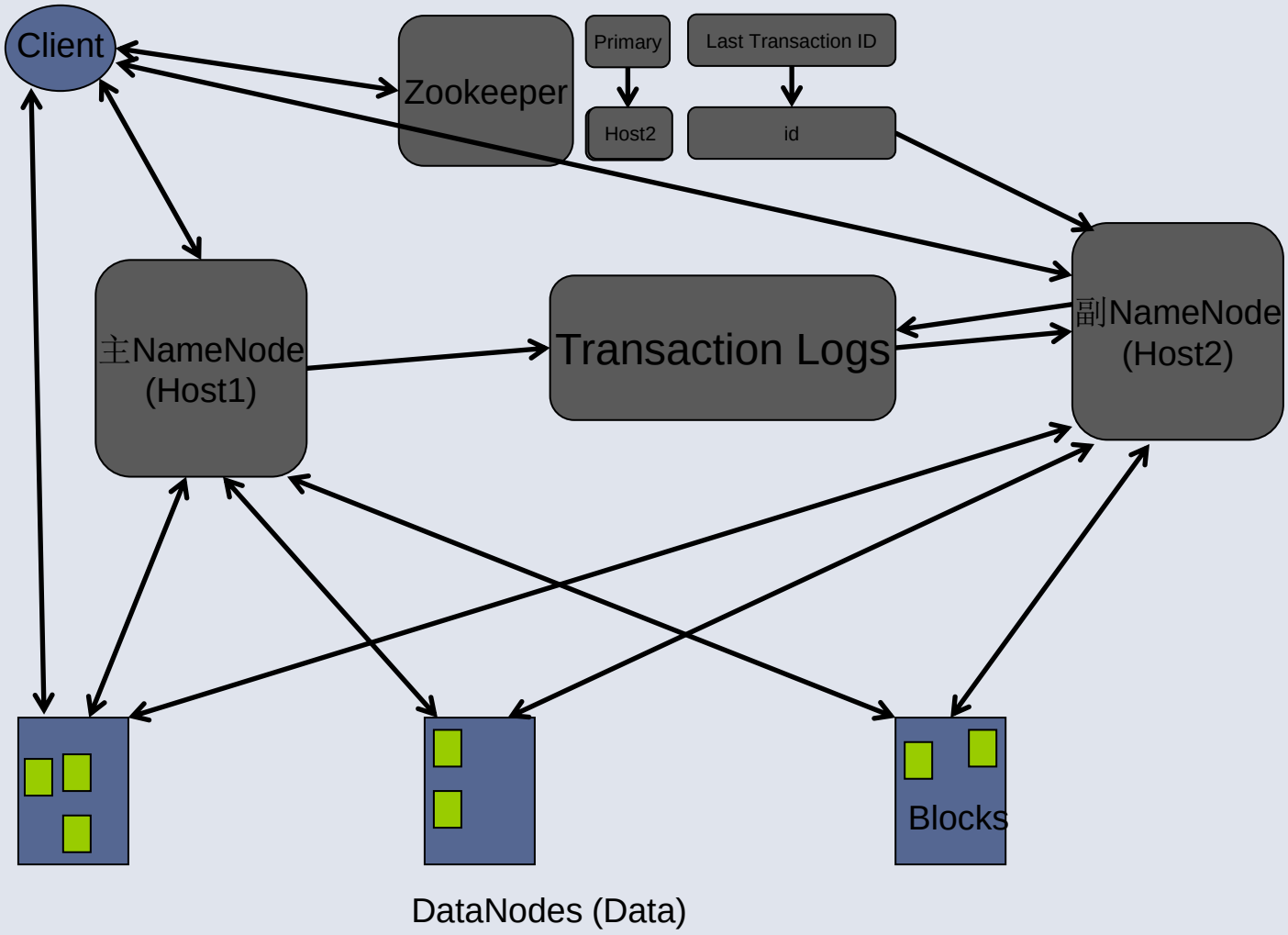
在不同NameNode之间移动文件？FastCopy！



HDFS不停机升级

NameNode升级是造成服务不可用的首要原因，它一定需要停机吗？

NameNode不停机升级



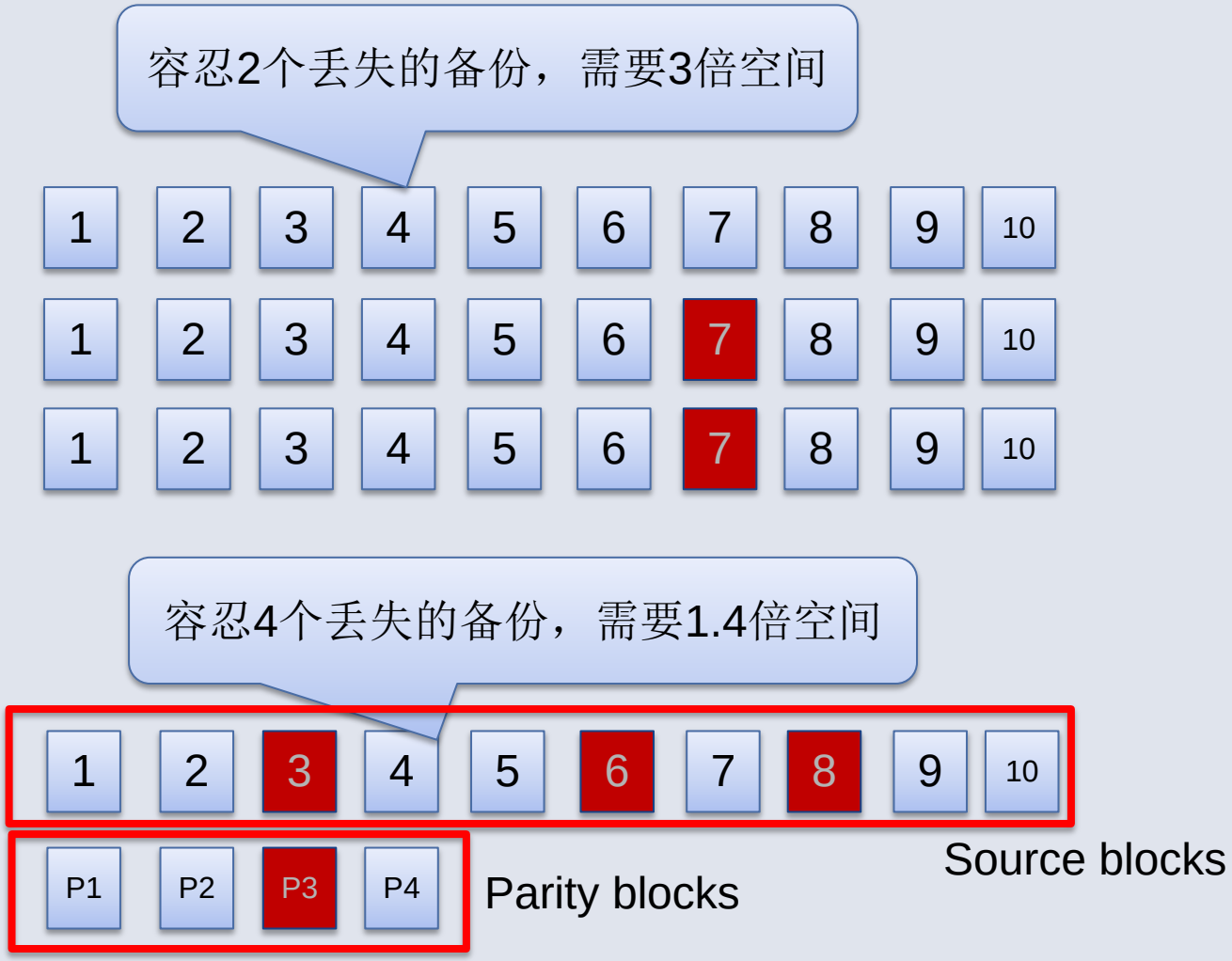
NameNode不停机升级——遇到的问题

- 如何验证正确性？
  - 确认Transaction ID
  - 确认Block数量
- 暂停时间用在哪里？
  - 等待主NameNode退出
  - 副NameNode读取剩余记录
  - 副NameNode等待Block报告

HDFS节省存储空间

节省存储空间——“RAID”

Reed Solomon校正码



节省存储空间——“RAID”

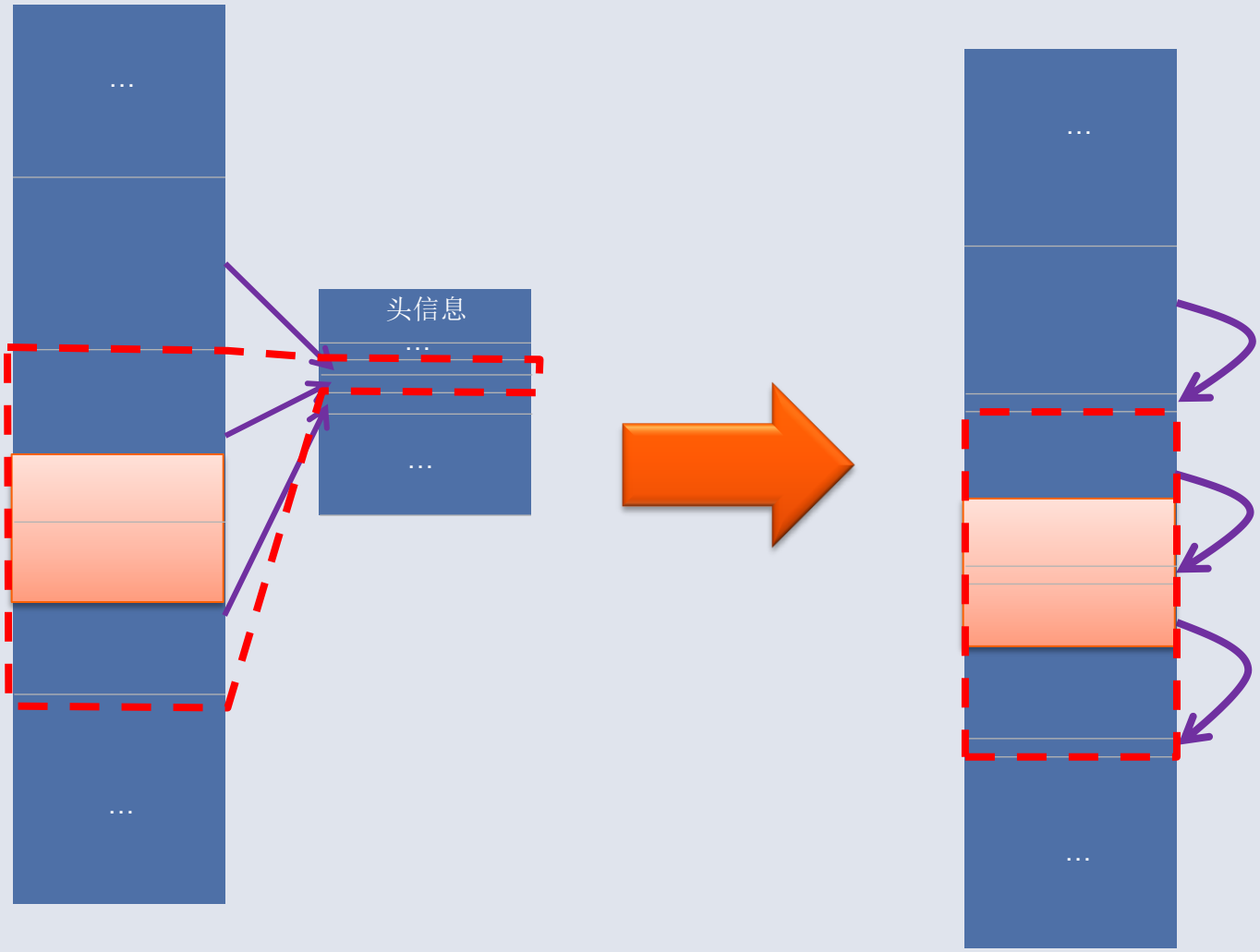
运维中遇到的问题

- **“Decommission”**:一台机器存储TB级别的数据，需要多长时间全部复制到别的机器？
- 数据重建：如果一台机器完全失效，需要多少系统资源重建TB级别的数据？
  需要在不更换机器的情况下更换硬盘！
- 文件太小？定期将旧数据合并成大文件
- 无法直接生成RAID文件，需再次扫描数据生成校验数据
- 随机读数据重建：线程模型影响性能

HDFS性能和可靠性

提高随机读数据的吞吐量

“Inline Checksum”



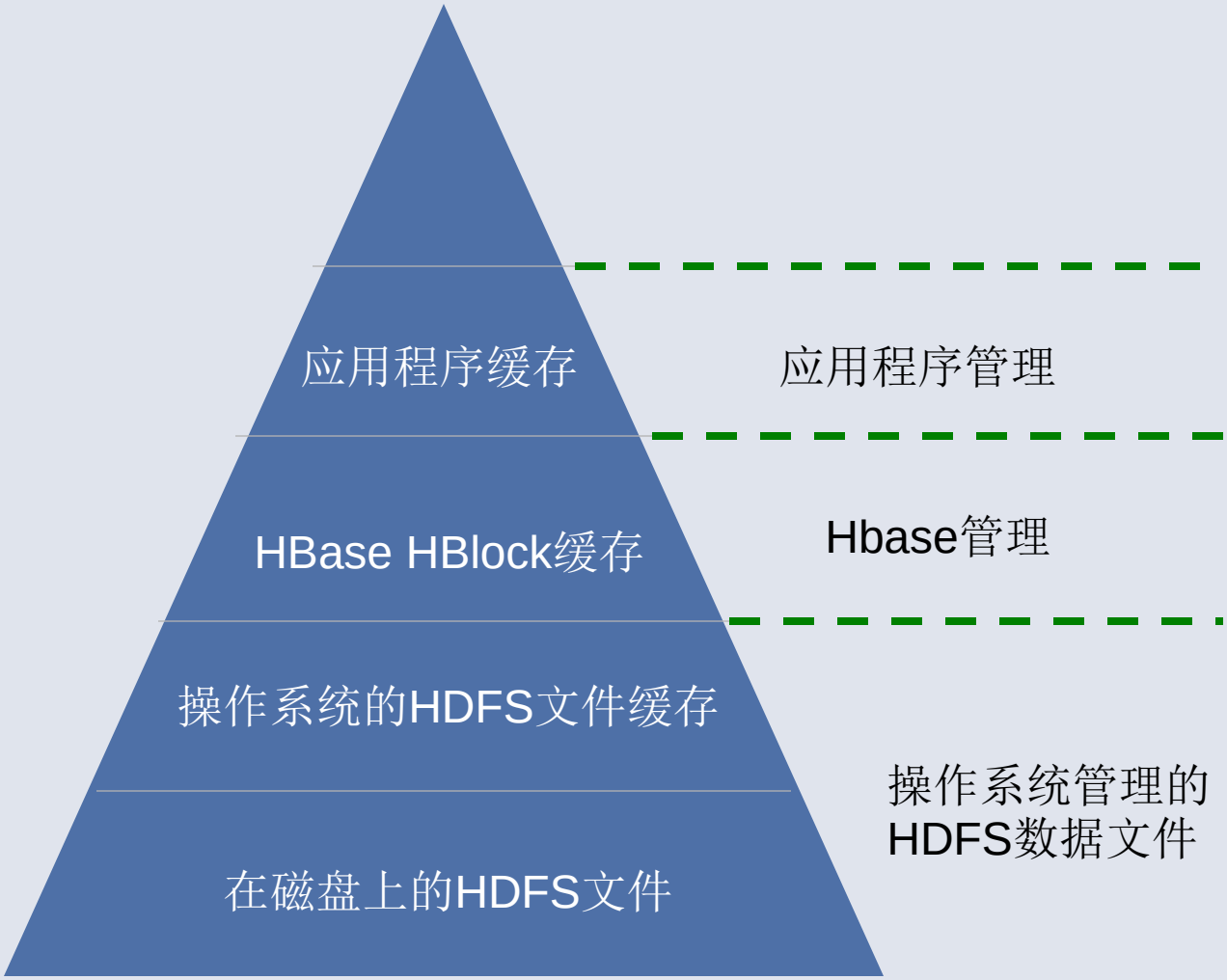
数据读写的稳定性

读写稳定性

- 调整DataNode的锁（FSDataset.lock）：
  - 不必要的操作不加锁
  - 尽量将I/O操作移到锁外
- 修补写操作容错的bug
- 改进写操作的超时检测

HBase的新特性

Hbase的层次存储


应用程序缓存
HBase HBlock缓存
操作系统的HDFS文件缓存
在磁盘上的HDFS文件
应用程序管理
Hbase管理
操作系统管理的
HDFS数据文件

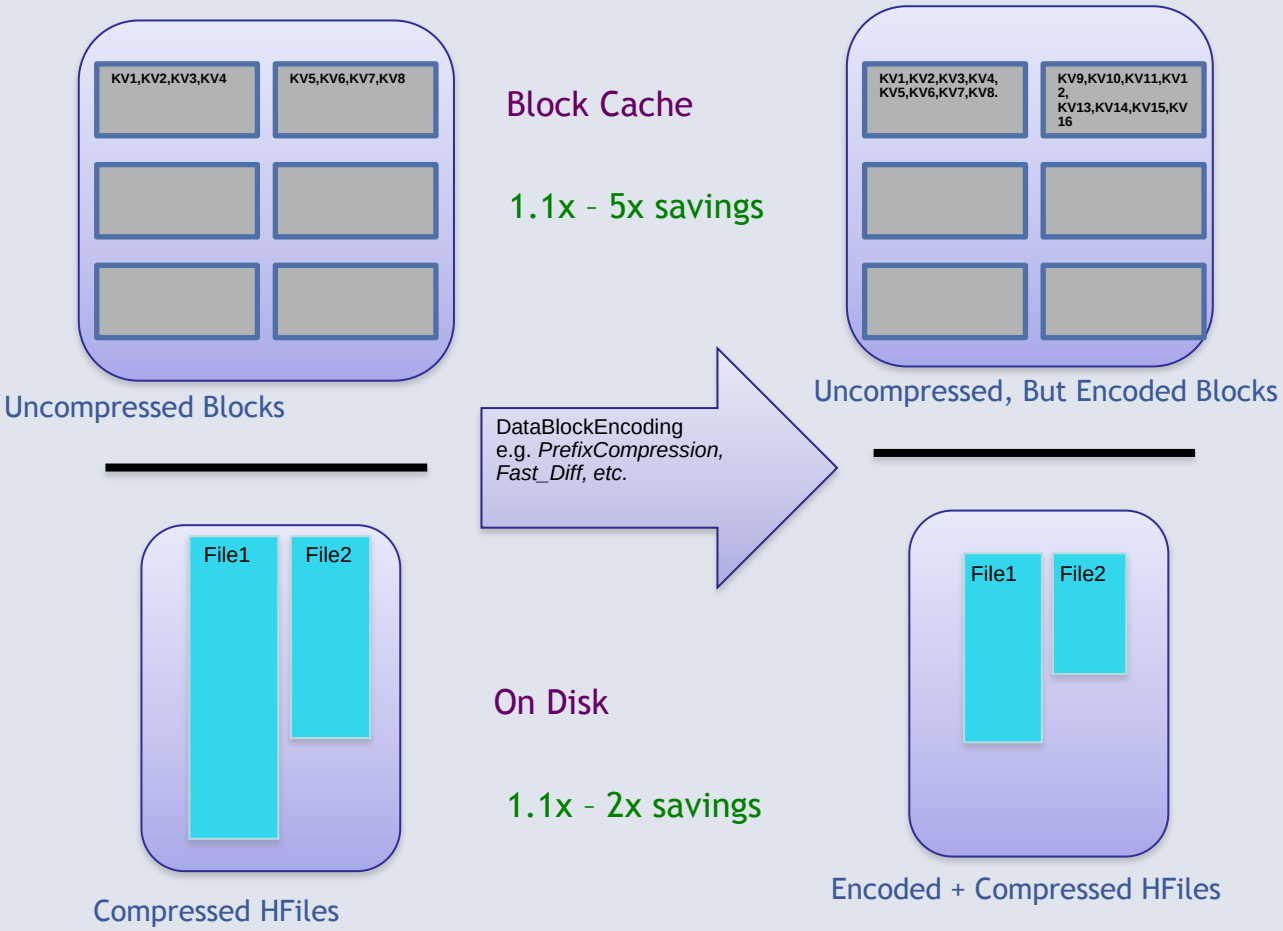
改进容错

交换机重启

- 交换机重启只需几分钟，我们不希望region server等待
- 修复使region server过快自动退出的一些路径
- Master智能检测是某一region server失效还是交换机重启，以此决定超时的时间
- 保证交换机重启后立刻恢复运行

数据编码

- more KVs per block in cache
- on-disk savings too
- seeking done on encoded format
- pluggable framework



针对Hbase的数据块放置算法

Pros:

- locality-aware “region” load-balancing/failover
- avoids network spikes on server failures
- facilitates “smooth” cluster expansion

其他改进简述

- 可靠性
  - 重写Master故障转移代码
  - 加速region重新分配
- RPC优化
- 批量删除优化
- Per-request profiling
- 客户端优化

facebook