



# **HBASECON** ASIA2019

**THE COMMUNITY EVENT FOR APACHE HBASE™**

**July 20th, 2019 - Beijing, China**





# **HBASECON** ASIA2019

**THE COMMUNITY EVENT FOR APACHE HBASE™**

**July 20th, 2019 - Beijing, China**





# Opening Speech

---

Baoqiu Cui | Xiaomi

小米集团副总裁 集团技术委员会主席

“ 技术事关小米生死存亡，技术立业  
是小米血液里最重要的东西！ ”

— 雷军

2019.2.26 集团技术委员会成立

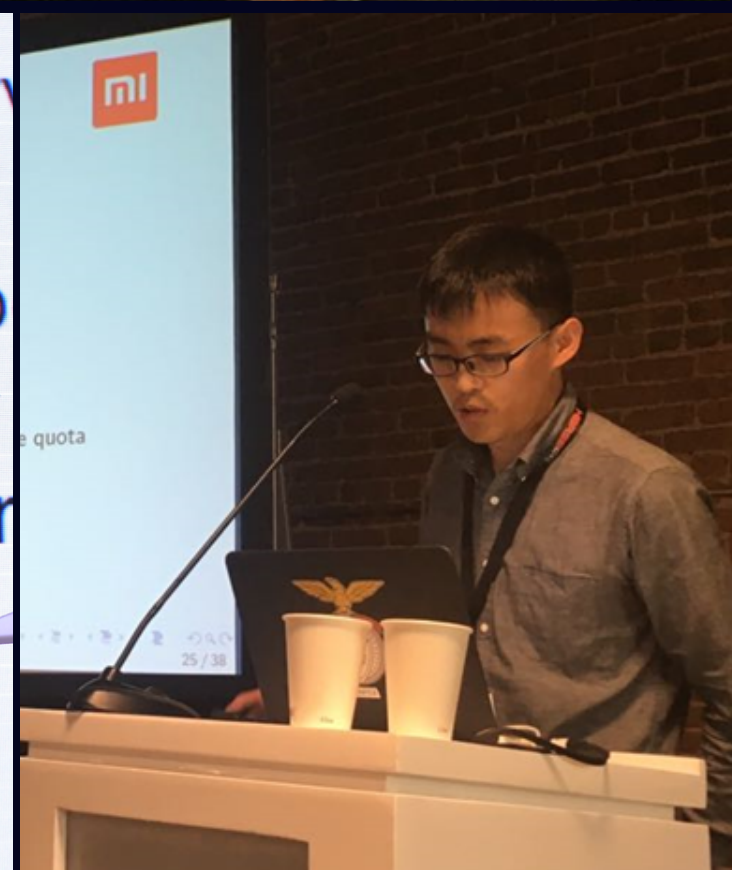


# 拥抱开源是小米的工程文化

开源 · 开放 · 平等 · 全球化



# 小米和 HBase 的渊源





小米将持续贡献，回馈社区！



感谢 HBase 社区！  
感谢来自全球的 HBase 贡献者！





# The current status of HBase

---

Duo Zhang | Xiaomi

HBase PMC主席

Duo Zhang <zhangduo@apache.org>

Michael Stack <stack@apache.org>



# Pervasive...

...distributed, scalable, big data store





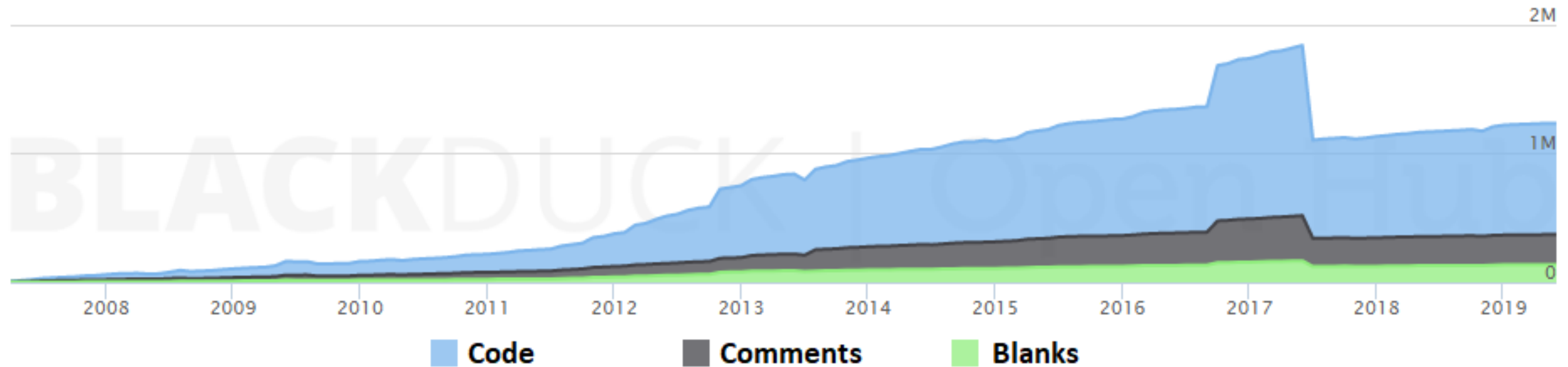
# In a nutshell...

- ...16,504 commits made by 391 contributors
- ...representing 862,469 lines of code
- ...mostly written in Java
- ...has a well established, mature codebase
- ...maintained by a very large development team
- ...with stable Y-O-Y commits
- ...took an estimated 240 years of effort (COCOMO model)
- ...starting with its first commit in April, 2007 (>10 years old!)

Source <https://www.openhub.net/p/hbase>



# LOC

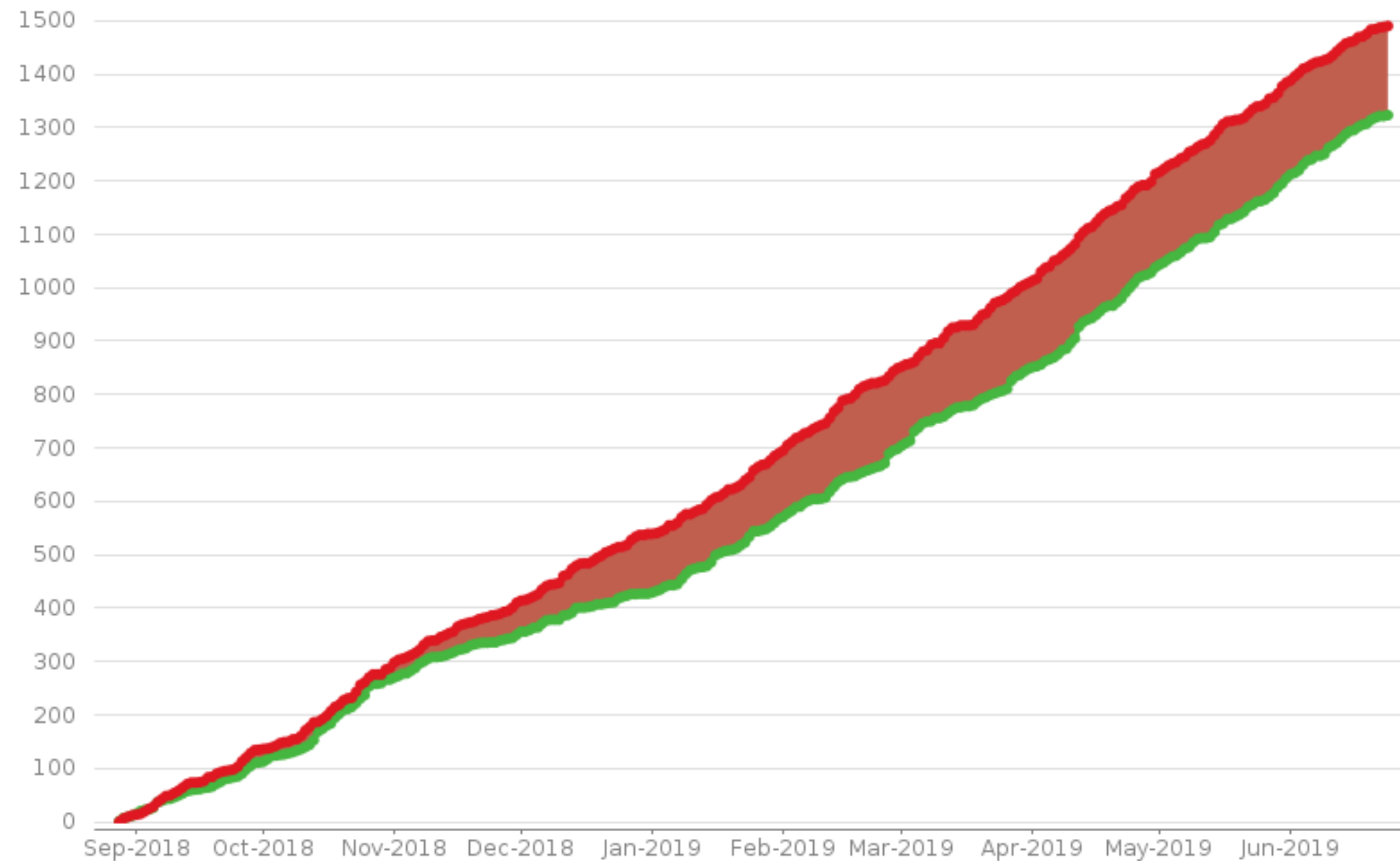


Source <https://www.openhub.net/p/hbase>



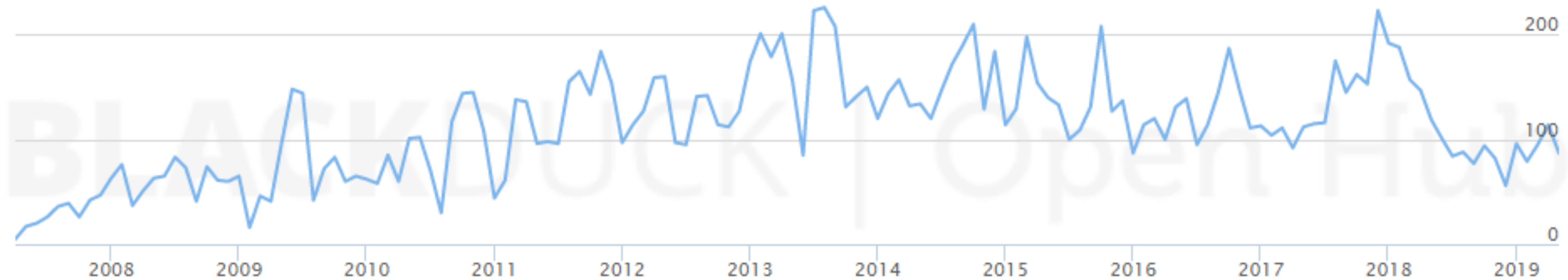
# Issues

This chart shows the number of issues **created** vs. the number of issues **resolved** in the last 300 days.





# Commits per month



Source <https://www.openhub.net/p/hbase>



# Contributors



Source <https://www.openhub.net/p/hbase>



# Our project

- Apache HBase is an Open Source Apache project.
- It's what **we** want to make of it.
- No owners!
- Anyone can help!
- All welcome!
- The more, the merrier!





# From Our PMC Chair

*“The HBase project represents solid, useful computer science that solves problems and runs businesses every day. To keep that going, we need to keep bringing new ideas and approaches into the project...we need to continue to attract people from all backgrounds and parts of the world. I'd love to see more women, more people of color, and even more worldwide diversity. I'd like to see more contributions from people not employed by big data platform companies.”*

*“If we continue to strive for a diversity of ideas and experiences, we'll keep innovating so that HBase remains relevant for years to come.”*

Misty Linville, Vice-President of the Apache HBase Project



# Active branches

<i>Active Branches</i>	<i>Latest Branch Release</i>	<i>Release Manager</i>
branch-1.2	EOL'd 05/2019	Sean Busbey
branch-1.3	1.3.5 (Yahoo)	Francis Liu
branch-1.4	1.4.10 (Current Stable)	Andrew Purtell
branch-1.5	Coming...	Andrew Purtell
branch-2.0	2.0.5 (EOL after 2.0.6)	Michael Stack
branch-2.1	2.1.5	Duo Zhang
branch-2.2	2.2.0	Guanghao Zhang
branch-3	Coming...	Volunteer wanted!



# HBase 2.2.0



# In general

- Redesign of the assignment related procedures
  - A TransitRegionStateProcedure rules all
  - No AssignProcedure, UnassignProcedure, MoveRegionProcedure
- More proc-v2 based operations
  - Split WAL (experimental)
- Upgrading
  - No effect on 1.x release
  - 2.1.x & 2.0.x
- The candidate of next stable pointer



# TransitRegionStateProcedure

## ➤ 4 Types

- Assign, Unassign, Move, Reopen

## ➤ Sub procedures

- OpenRegionProcedure

- CloseRegionProcedure

- One time, no retry

## ➤ Why a new one?

- MoveRegionProcedure does not work well with ServerCrashProcedure

- Lots of conditions to deal with tons of corner cases, but still buggy



# More improvements

- More fencings
  - Master side on reportRegionStateTransition
  - RegionServer side on executeProcedures
- WALProcedureStore
  - Fix bugs
  - Make it more robust

# More proc-v2 based operations

- Performance
  - Send request to RegionServer directly
- Staged execution
  - Supported by the proc-v2 framework
- ZooKeeper less
  - Ideally, only use ZooKeeper as an external storage
  - Support etcd, or other HA systems
  - Cloud Native



# Upgrading to 2.2.0

## ➤ From 1.x

- Upgrade to 1.4.x first for safety
- Turn on zk-less assignment
- Rolling upgrade region servers first
- Upgrade masters

## ➤ From 2.1.x/2.0.x

- Make sure there are no RITs, then upgrade master first, and then rolling upgrade region servers
- If there are old RIT procedures the new master will quit to make sure there are no damages to the cluster
- There is an option for helping upgrading to 2.2.0, please see the upgrading section in our ref guide

# The stable pointer

- Currently still on 1.4.x
- AM-v2 is still not stable enough
  - AM-v1 is not stable either!
- HCK2
  - Can fix basic region assignment problems
  - A bit different from HCK1
  - Still not powerful enough compare to HCK1
  - Cluster status report



# What's Next

# New projects

## ➤ HBase Connectors

- Kafka Proxy

- Spark

- 1.0.0

## ➤ HBase Filesystem

- HBOSS, HBase OSS(S3) Adaptor

## ➤ HBase Operator Tools

- HBCK2

## ➤ HBase Native Client

- HBASE-14850



# HBase 3.0.0

- Plan to cut branch-3 by the last quarter
- Need to have a 'stable' 2.x release line first
- 'New' features
  - Fold namespace table into meta table(HBASE-21154)
  - Synchronous replication(HBASE-19064)
  - Off-Heap read starting from DFSCClient(HBASE-21879)
  - Proc-v2 based ACL(HBASE-21602)
  - Reimplement sync client on top of async client(HBASE-21512)
  - ...

# Delayed...

- SQL Engine
  - Like CQL?
  - Phoenix?
  - Upstream of in-house solution?
- Splitable meta
  - Scalability
  - Optimistic to have this in 4.0.0...
- WAL abstraction
  - Remove the last unavoidable HDFS dependency
  - Cloud Native
  - HBASE-20952





# GitHub PR is now available!

- Still need a jira account to file an issue...
- The commit message for the PR should start with the issue number, for example, “HBASE-12345 Test Github PR”
- All other things can be done on GitHub!

# HBASE-22638 : Zookeeper Utility enhancements #345

Edit

 Open virajjasani wants to merge 6 commits into `apache:master` from `virajjasani:HBASE-22638-master` 

 Conversation 33

 Commits 6

 Checks 0

 Files changed 6



+89 -74 



virajjasani commented 6 days ago • edited ▾

Contributor +  ⋮

- final arguments for Constructor with args
- try with resources
- removal of redundant null check
- avoid possible NPE due to KeeperException


  HBASE-22638 : Checkstyle changes for Zookeeper Utility classes

Unverified 2ae4e1e



Apache-HBase commented 6 days ago

+  ⋮

 +1 overall

Vote	Subsystem	Runtime	Comment
0	rexxec	153	Docker mode activated

Reviewers




 maoling



 jatsakthi



 HorizonNet



Assignees



No one—assign yourself

Labels



None yet

Projects



None yet







HBase / HBASE-22638

# Checkstyle changes for hbase-zookeeper util classes

- Edit
- Comment
- Assign
- More ▾
- Submit Patch
- Resolve Issue

-   Export ▾

## Details

Type:	 Improvement	Status:	<b>OPEN</b>
Priority:	 Minor	Resolution:	Unresolved
Affects Version/s:	3.0.0	Fix Version/s:	None
Component/s:	<a href="#">Zookeeper</a>		
Labels:	None		

## Description

Checkstyle and cosmetic changes for Zookeeper Util classes - ZKUtil, MiniZookeeperCluster etc



## Attachments

 Drop files to attach, or [browse](#).

## Issue Links

links to  [GitHub Pull Request #345](#)

## People

- Assignee:  Viraj Jasani  
[Assign to me](#)
- Reporter:  Viraj Jasani
- Votes:  0 [Vote for this issue](#)
- Watchers:  1 [Start watching this issue](#)

## Dates

- Created: 6 days ago
- Updated: 13 minutes ago

## Agile

[View on Board](#)

# The Apache Way

## ➤ Independence

- Only PMC can control the direction of the project
- Diversity

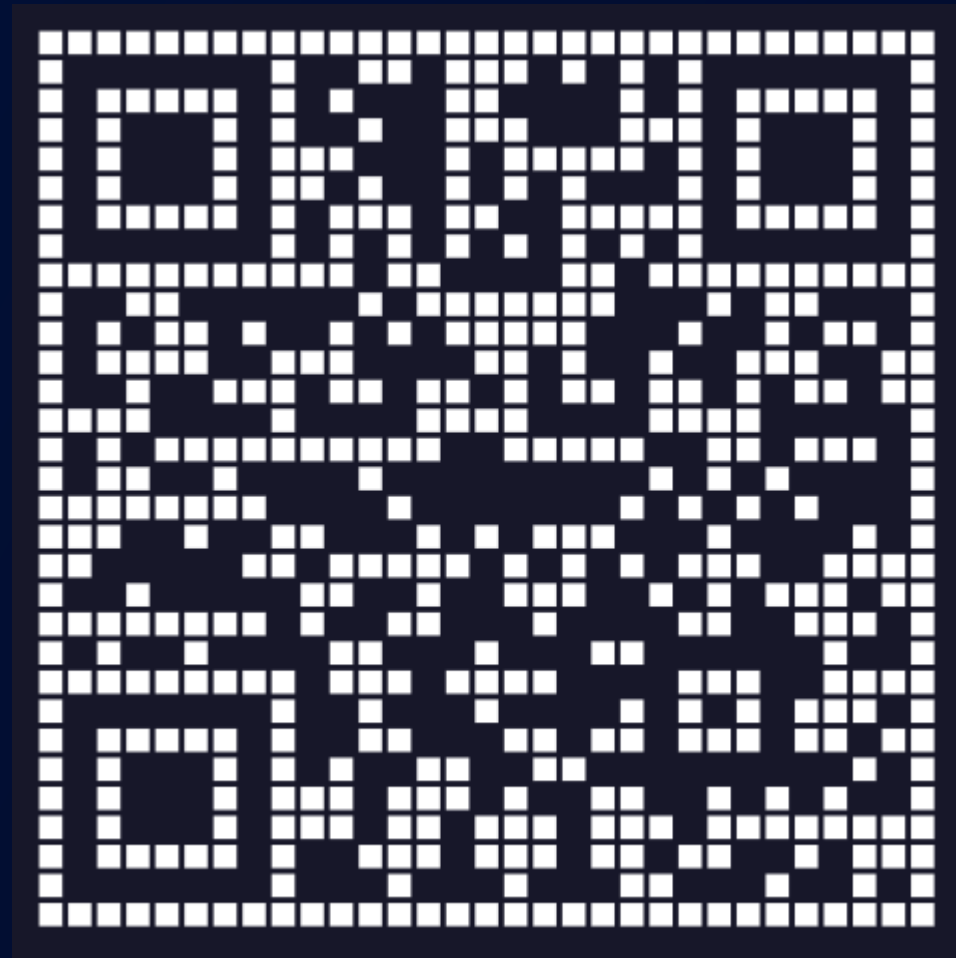
## ➤ Community Over Code

- A healthy community is a higher priority than good code

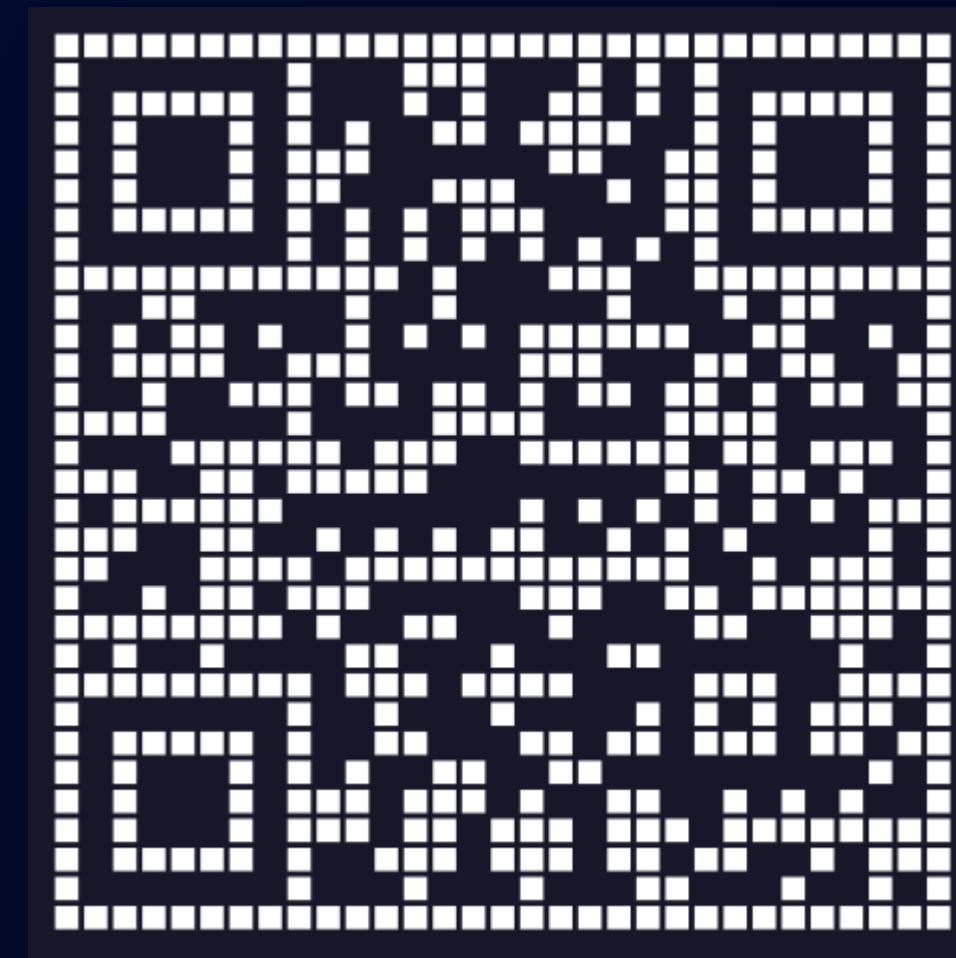


# Thank you!

小米云技术



中国HBase技术社区





# The advantages and technology trend of HBase on the cloud

沈春辉 Chunhui Shen | Alibaba

Apache HBase PMC 阿里巴巴 资深专家 阿里云 HBase负责人



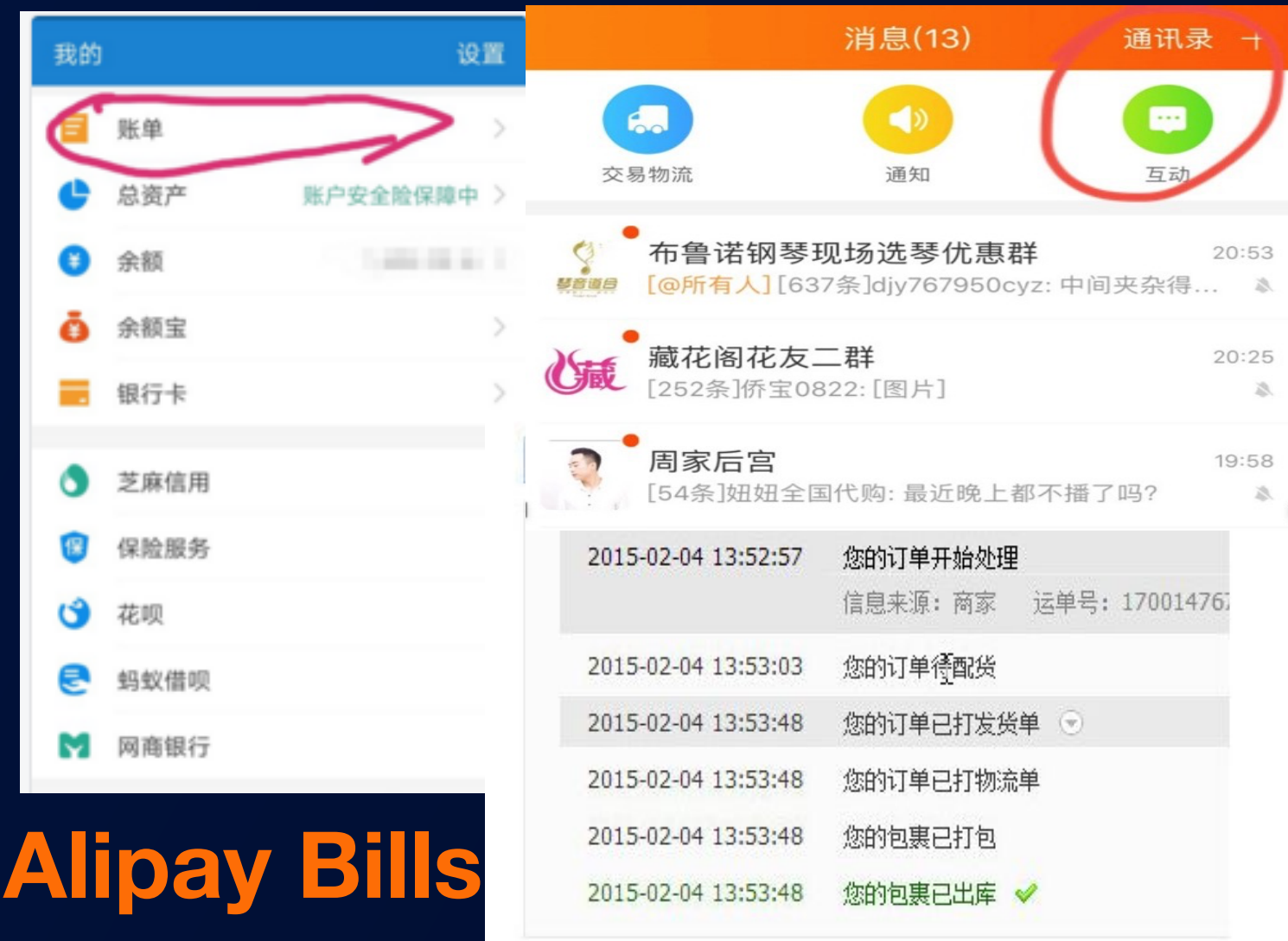
# HBase在阿里的使用情况

## The current status of HBase at Alibaba

# 使用场景

## Core Scenarios

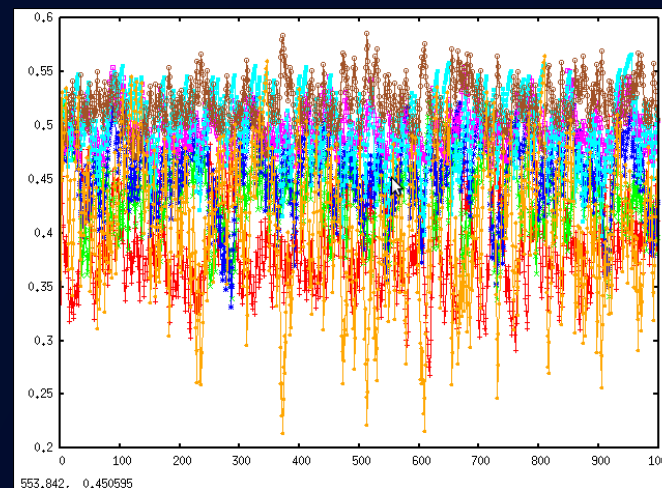
Message, Orders, Feeds ...



Alipay Bills

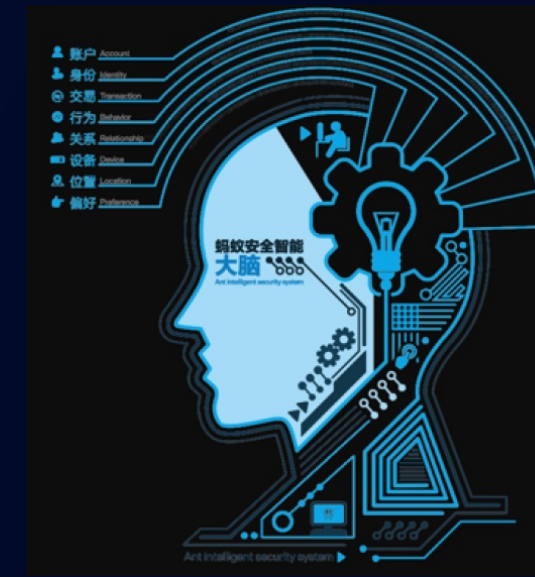
Cainiao Logistics

Monitor, Log,  
Tracking, IoT Data...



Log

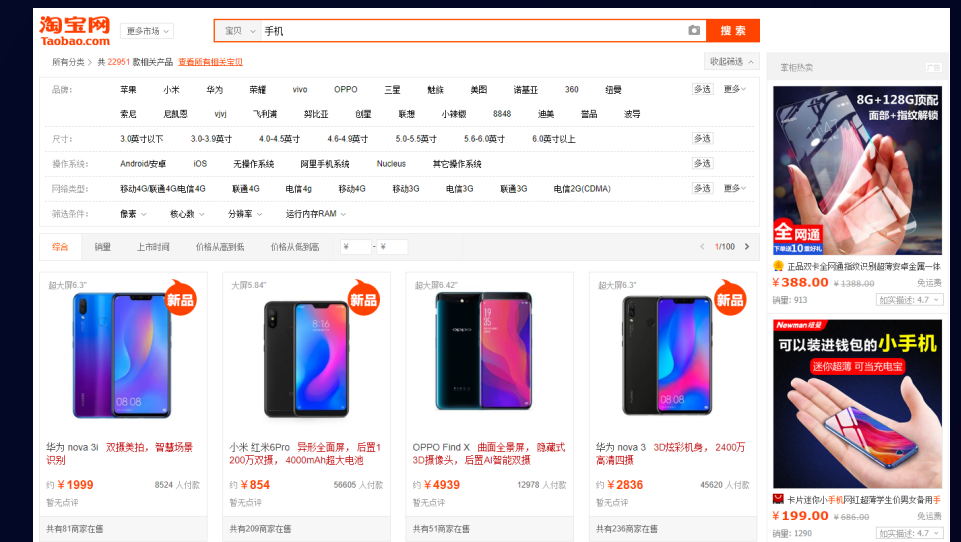
AI Storage  
Ant Intelligent Security



Intelligent Customer Service



Recommendation  
Search, BI Report...



Ali-HBase



# 使用规模

Use Scale

10000+  
Nodes

100+  
Clusters

300+  
Million OPS

200+  
PB Data

9000+  
Users

# 部署演进

## Evolution of deployment

Begin:

1. Physical machine
2. One HBase per App
3. Exclusive ZK&HDFS per HBase

**Pain:** Maintenance Cost 、 Resource Fragmentation 、 Waste of MetaServer

Phase 2  
2012~2014

Change:

1. Exclusive ZK&HDFS per HBase

**Pain:** move machines  
**Weak:** Storage Fragmentation

Phase 4  
2018~Now

Phase 1  
2010~2012

Change:

1. Multi-App on Shared HBase
2. Use RSGroup for isolation
3. HBase on Shared ZK&HDFS

**Pain:** stability

Phase 3  
2014~Now

Change:

1. Run on Cloud Infrastructure

**Challenge:** Be Cloud Native



# 关键考量

## Key points in deployment

稳定性  
Stability

弹性  
Elasticity

成本  
Cost

效率  
Efficiency

# 当HBase运行在云上时

When HBase runs on the cloud

“云”的能力如何发挥

How to unleash the energy of Cloud

HBase的技术如何演进

The technology challenge and trend of HBase



# 当HBase在云上运行时

When HBase runs on the Cloud

## 优势一：超弹性

Advantage 1: Hyper Elasticity

# 需要弹性 Need elasticity



大促扩容  
Expansion for  
shopping day



突发流量  
Burst traffic

Hot Key

异常隔离  
Anomaly isolation

# 弹性的衡量#1

## Measurement of elasticity #1

扩容时间 = 资源准备 + 环境准备 + 节点加入 + 节点服务

Time of capacity  
expansion

resource preparation

environment  
preparation

node join in

node in full service

Traditional Datacenter:  
Hours ~ Days

HBase: 1 ~ 5 Mins

Non-matched



# 云上的资源创建

## Create instance on the cloud



Compute  
virtualization

Storage  
virtualization

Network  
virtualization

Resource Scheduling Management

Physical Resource Pool

虚拟化 + 池化  
Virtualization Resource pool

快速创建实例  
Quickly create an instance

# 弹性的衡量#2

## Measurement of elasticity #1

扩容时间 = 资源准备 + 环境准备 + 节点加入 + 节点服务



Matched

# 弹性的价值-成本

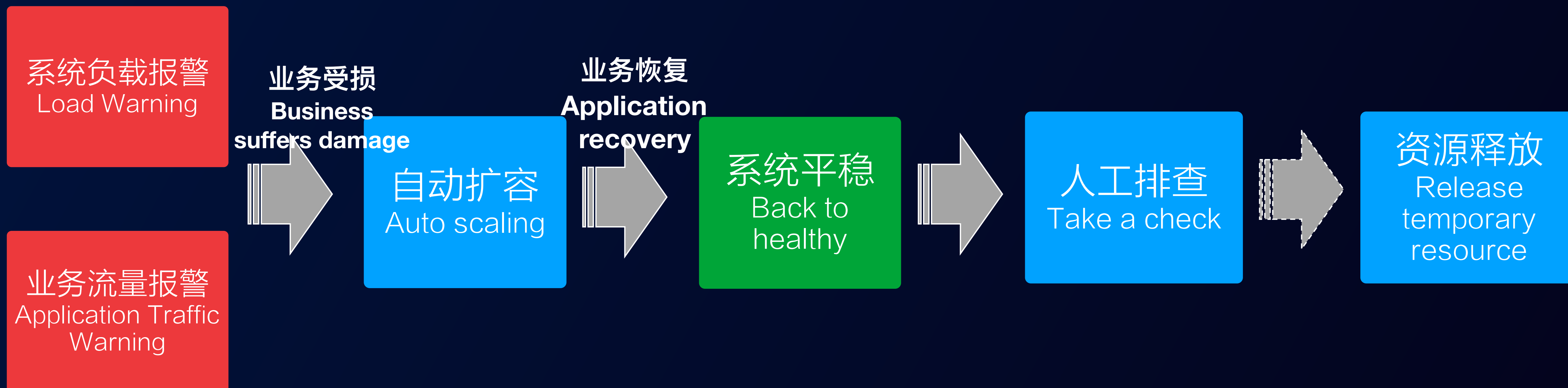
The value of elasticity - cost





# 弹性的价值-稳定#1

The value of elasticity - stability #1



# 弹性的价值-稳定#2

## The value of elasticity - stability #2

### Normal Group

t1-region1

t2-region1

RS1

t3-region1

t1-region2

RS2

t2-region2

t3-region2

RS3

t4-region1

### Abnormal Group

t4-region1

t4-region2

RS4

#### 1 T4表请求出现异常

Abnormal request to table t4

#### 2 扩容资源

Add new node RS4

#### 3 隔离T4表

Move Table T4 to the new group

# 当HBase在云上运行时

When HBase runs on the Cloud

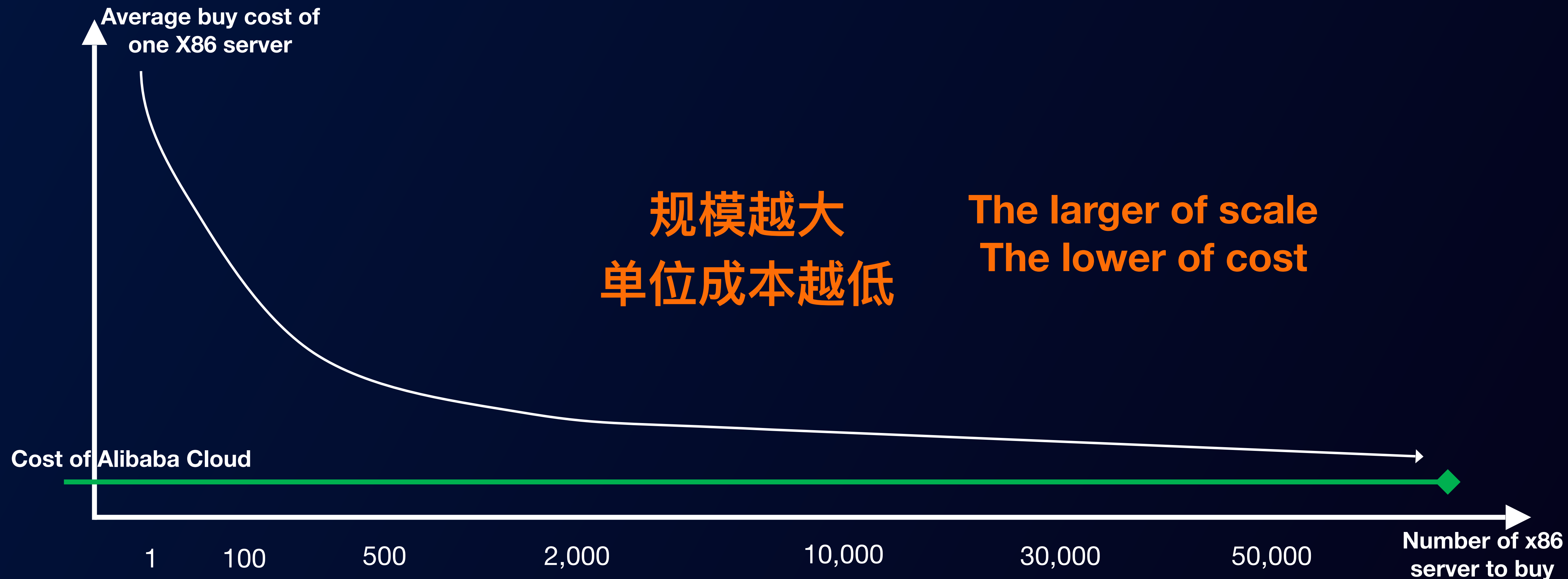
## 优势二：低成本

Advantage 2: Low Cost



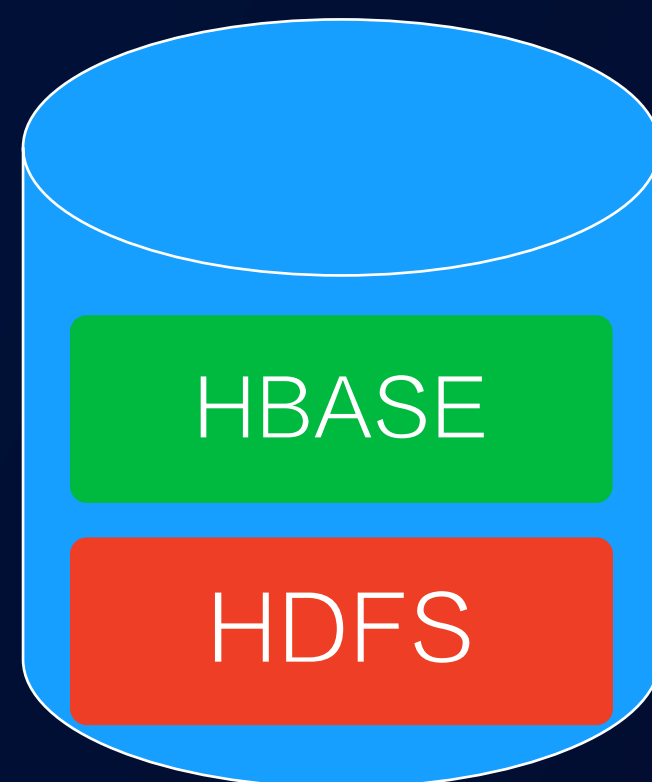
# 服务器采购成本

The cost of buying machines



# 物理部署下的常见情况

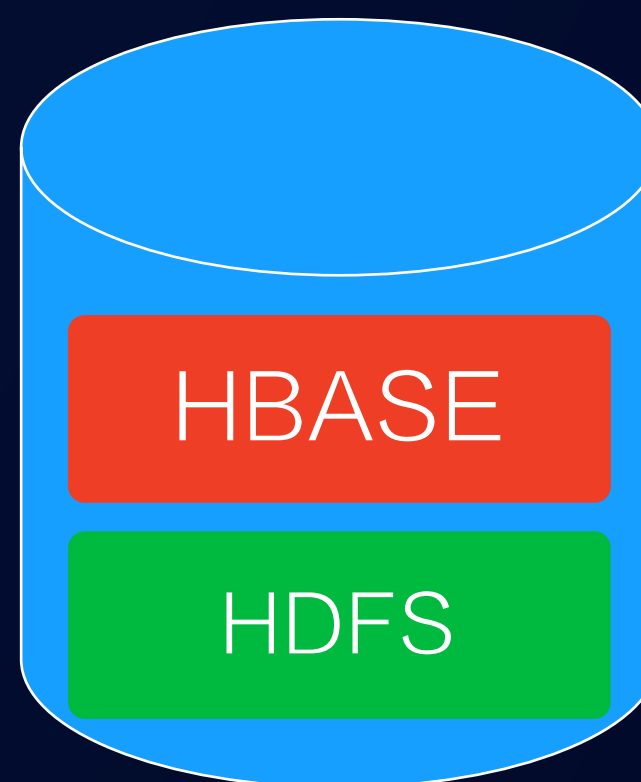
Common situation when deploy on physical machines



**Cluster A**

**CPU: 5%**

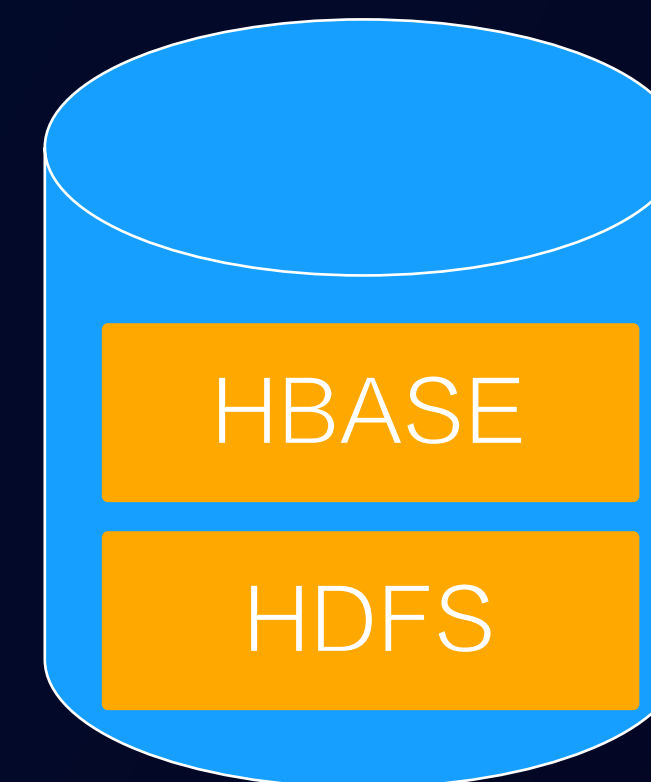
**Storage: 80%**



**Cluster B**

**CPU: 40%**

**Storage : 30%**



**Cluster C**

**CPU: 20%**

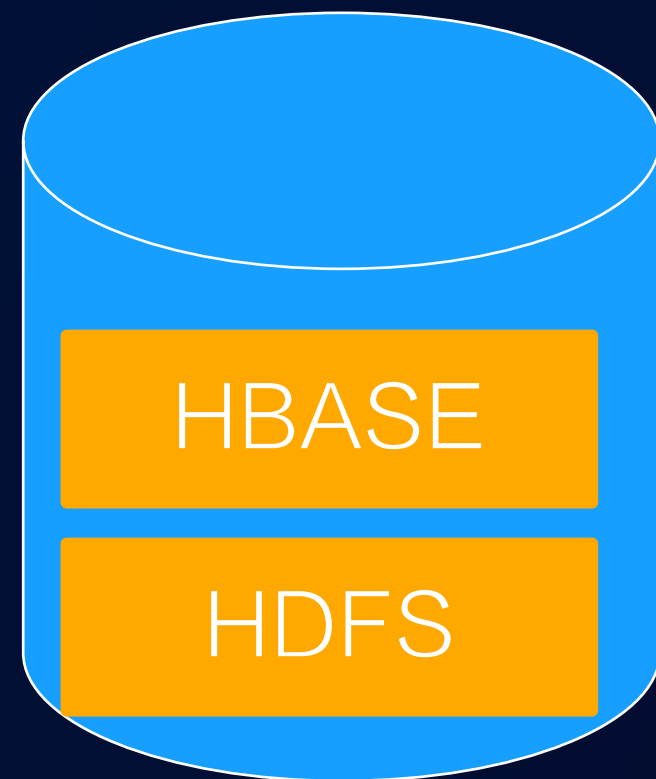
**Storage : 70%**

不同集群(workload)使用同配置的服务器

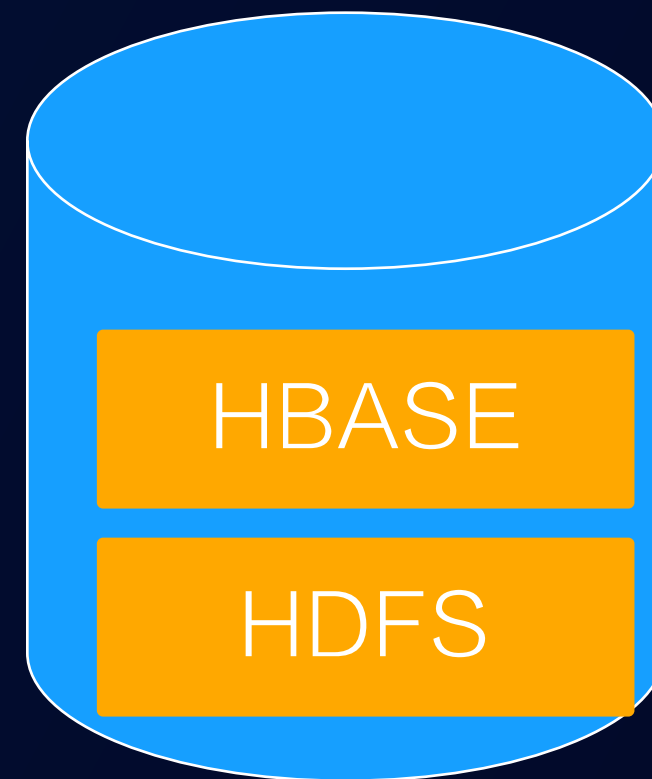
Different clusters(workload) have servers with the same hardware configuration

# 上云之后#1

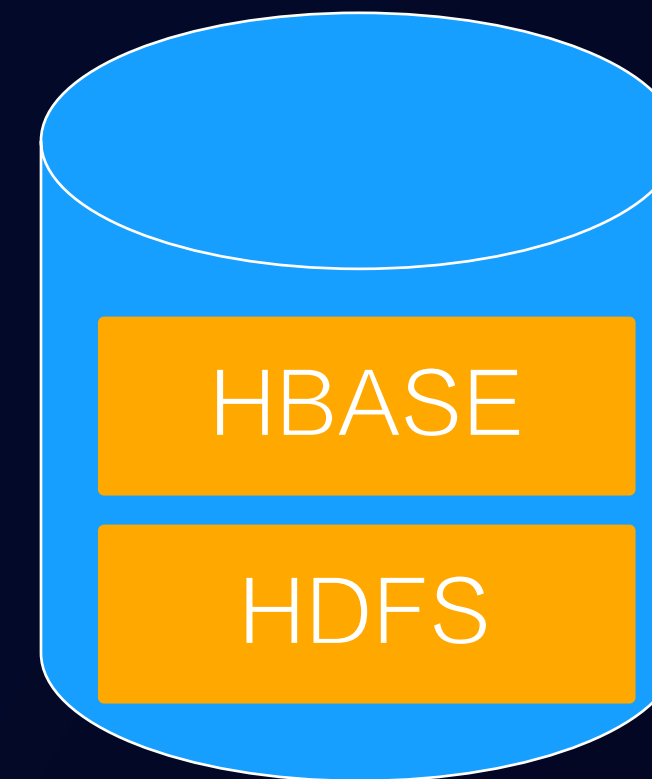
After deploy on the cloud #1



**Cluster A**  
**CPU: 20%**  
**Storage : 70%**



**Cluster B**  
**CPU: 20%**  
**Storage : 70%**



**Cluster C**  
**CPU: 20%**  
**Storage : 70%**



# 上云之后#2

After deploy on the cloud #2

## 灵活的存储计算比

Flexible ratio between storage and computation

不同集群(workload)使用不同配置的虚拟机

Different clusters could have servers with the different hardware configuration

## 存储计算分离

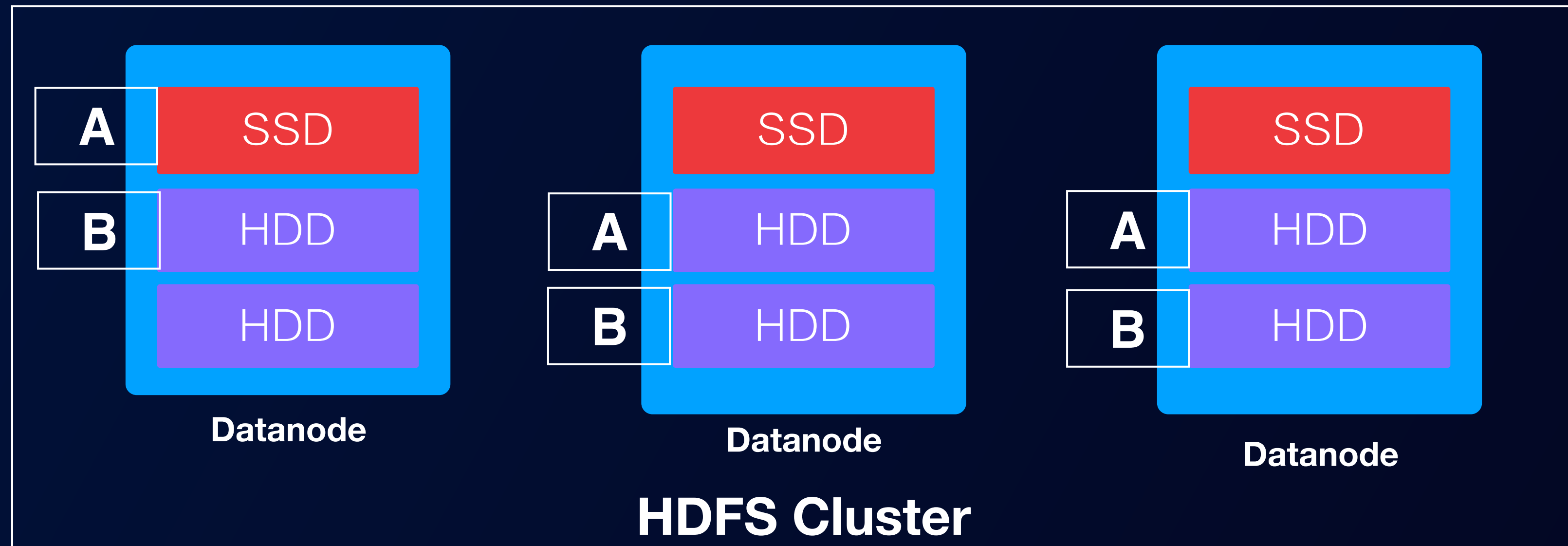
Separation of storage and computation

同集群使用相同配置，让运维保持简单

The same hardware configuration in one cluster, keep the maintenance be easy

# HDFS异构存储#1

## HDFS heterogeneous storage #1



### Advantage:

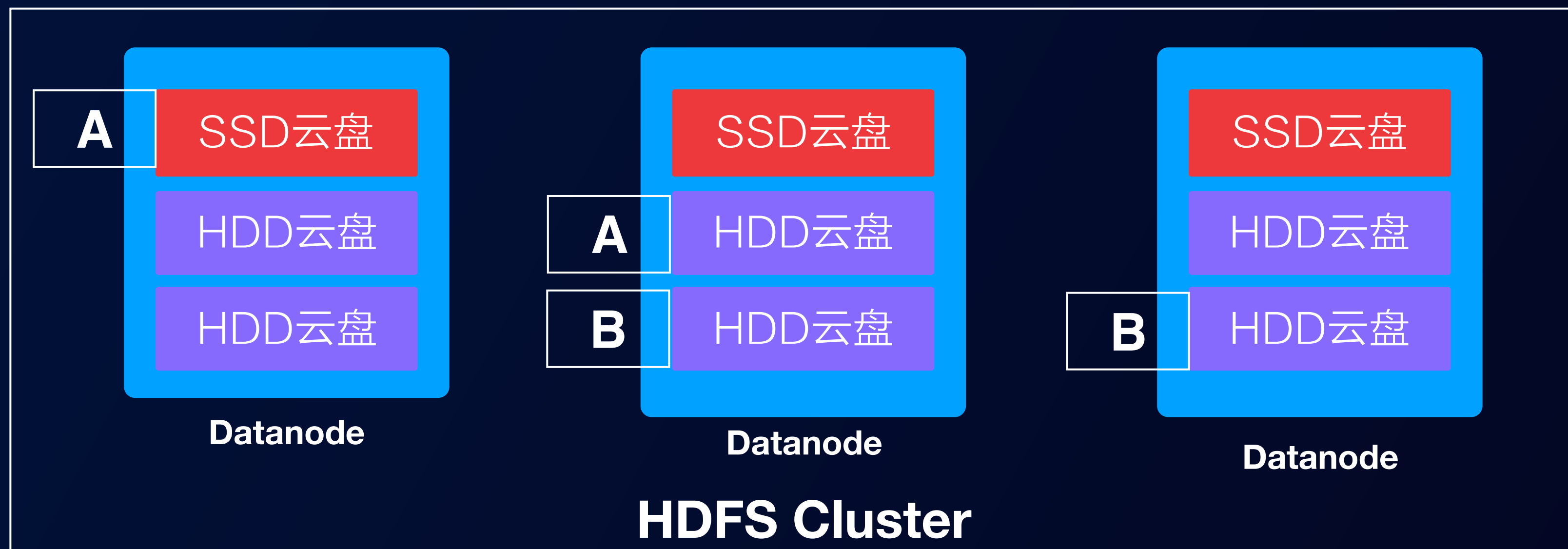
1. Cost-effective
2. Allocate storage according to scenario

### Disadvantage:

1. Personalized machine type
2. The capacity ratio of SSD to HDD

# 异构存储介质#2

## HDFS heterogeneous storage #2



### Advantage:

1. Cost-effective
2. Allocate storage according to scenario

### Disadvantage:

1. Personalized machine type
2. The capacity ratio of SSD to HDD



# 当HBase在云上运行时

When HBase runs on the Cloud

## 优势三：高稳定性

Advantage 1: High Stability

# 阿里云ECS的稳定性特征

## Stability characteristics of Alibaba Cloud's ECS

99.95%  
Availability  
SLA

机器过保无感知

Not aware of machine replacement

Never  
Retired

机房迁移无感知

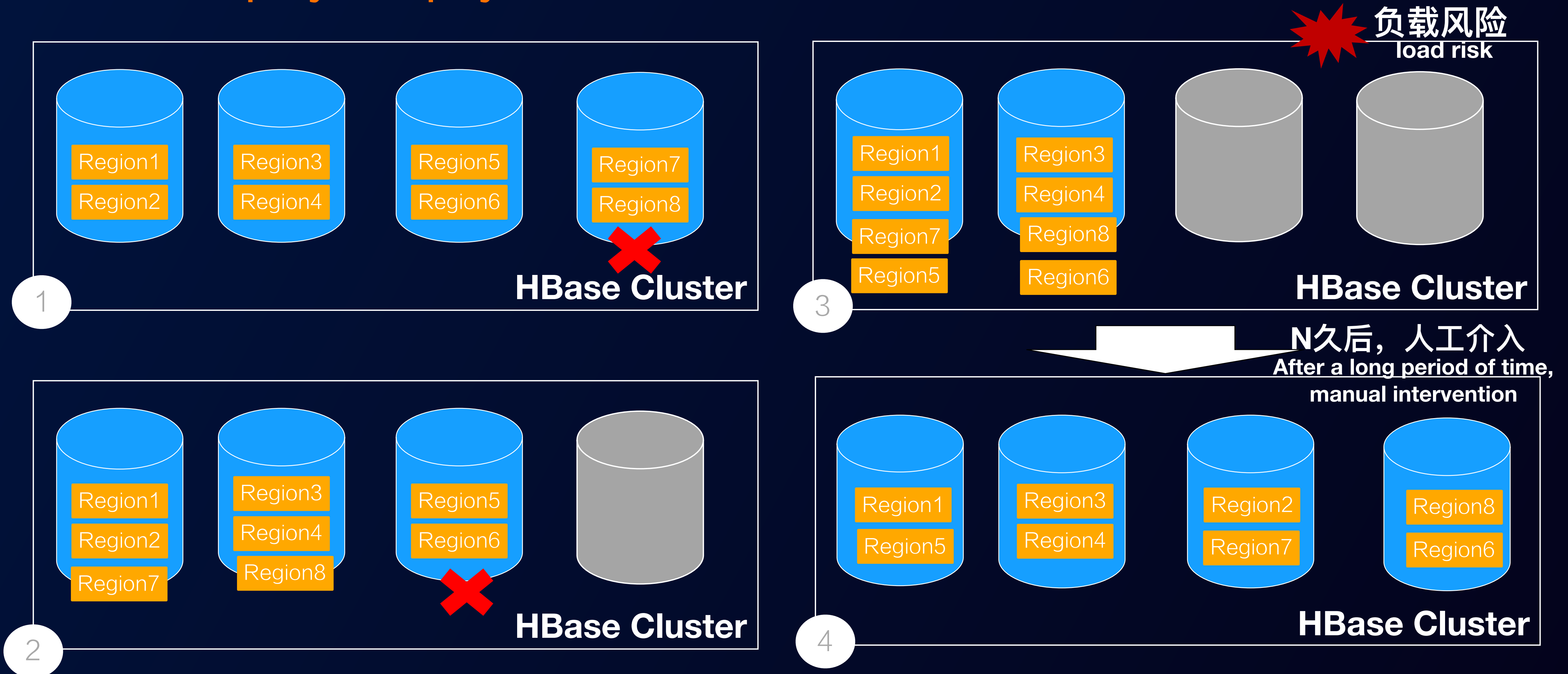
Not aware of datacenter migration

极低物理宕机率

The very low probability of machine downtime

# 物理机房—Auto Recovery, But not sustainable

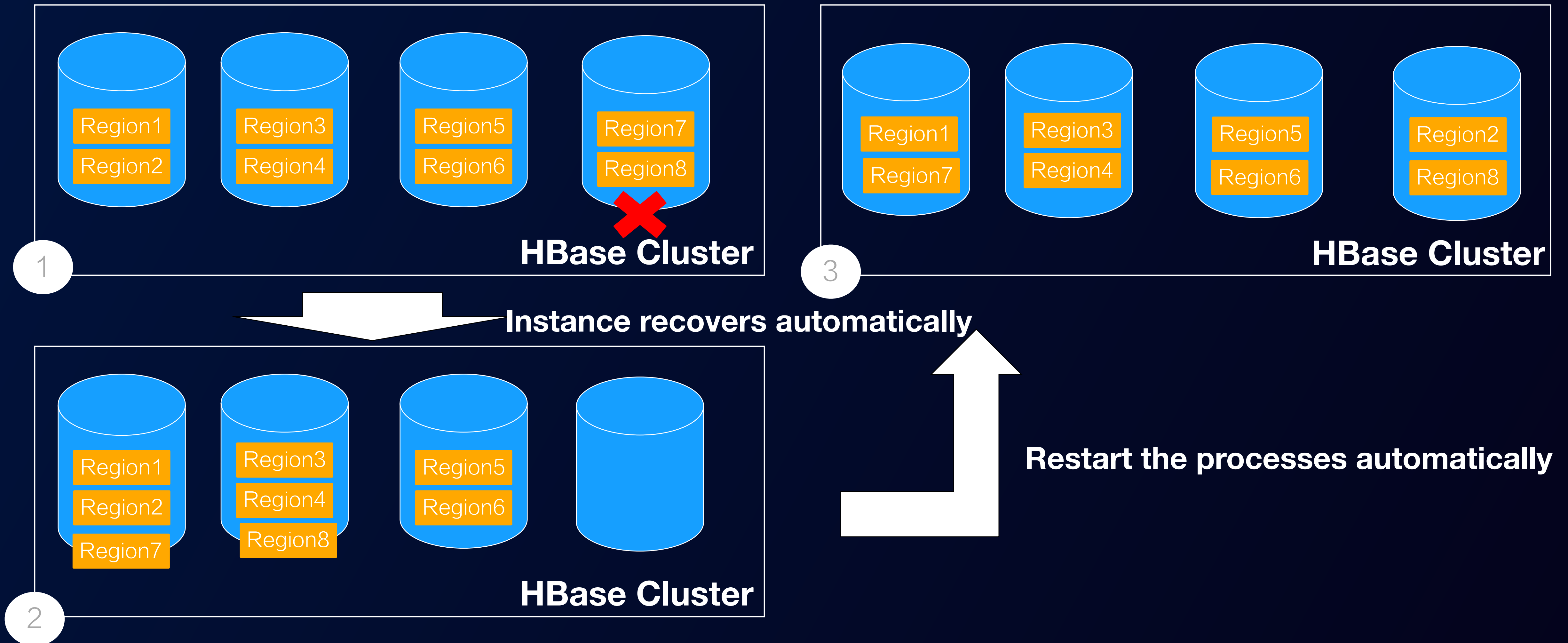
## When deploy on physical datacenter





# 上云之后-Auto recovery forever

After deploy on the cloud



# 其他常见问题

Other common problems

Long Compaction Queue

Replication Delay Too Much

Failover Too Slow

# 没有什么问题是扩容和重启不能解决的

No problem that can't be solved by increasing capacity and rebooting

在云上，问题处理变得更简单

On the cloud, problem handling becomes easier



# 当HBase在云上运行时

When HBase runs on the Cloud

## 优势四：全球部署

Advantage 4: Global Deployment

# 全球多活

Multi-master at global scale

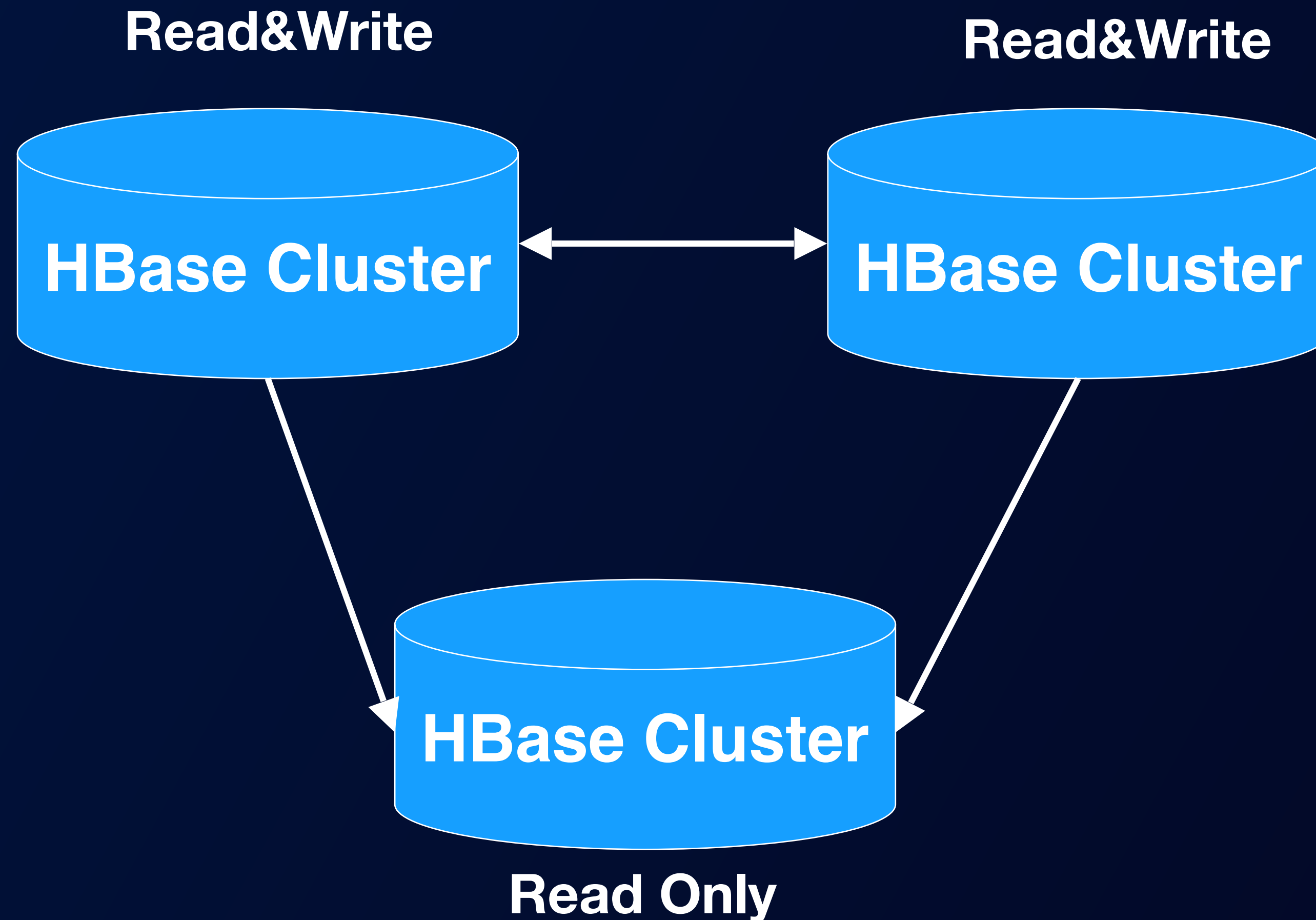


全球服务  
Global service

异地容灾  
Resilience of  
regional failure

就近访问  
Localized access

# HBase Replication



## Replication 特点:

1. 支持多活  
Support for multi-master
2. 自由拓扑  
Free topology
3. 支持循环链路  
Support for circular data links
4. 在线添加/删除链路  
Add and remove data links online



上云之后

Build replication on the cloud

HBase Replication

+

Globally Distributed Datacenters

Multiple active masters at global scale with HBase

# 当HBase在云上运行时

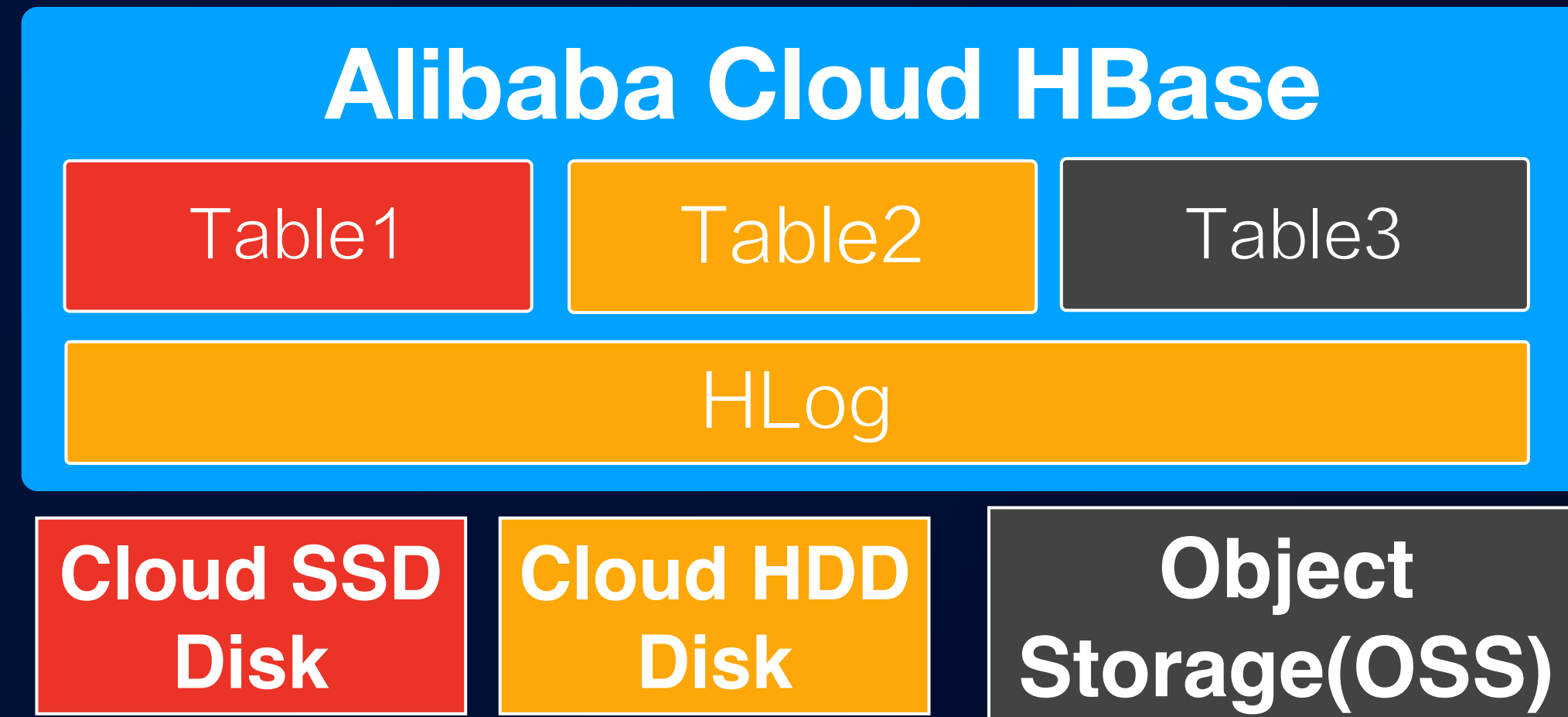
When HBase runs on the Cloud

## 技术的挑战及趋势

Challenge and Trend of Technology

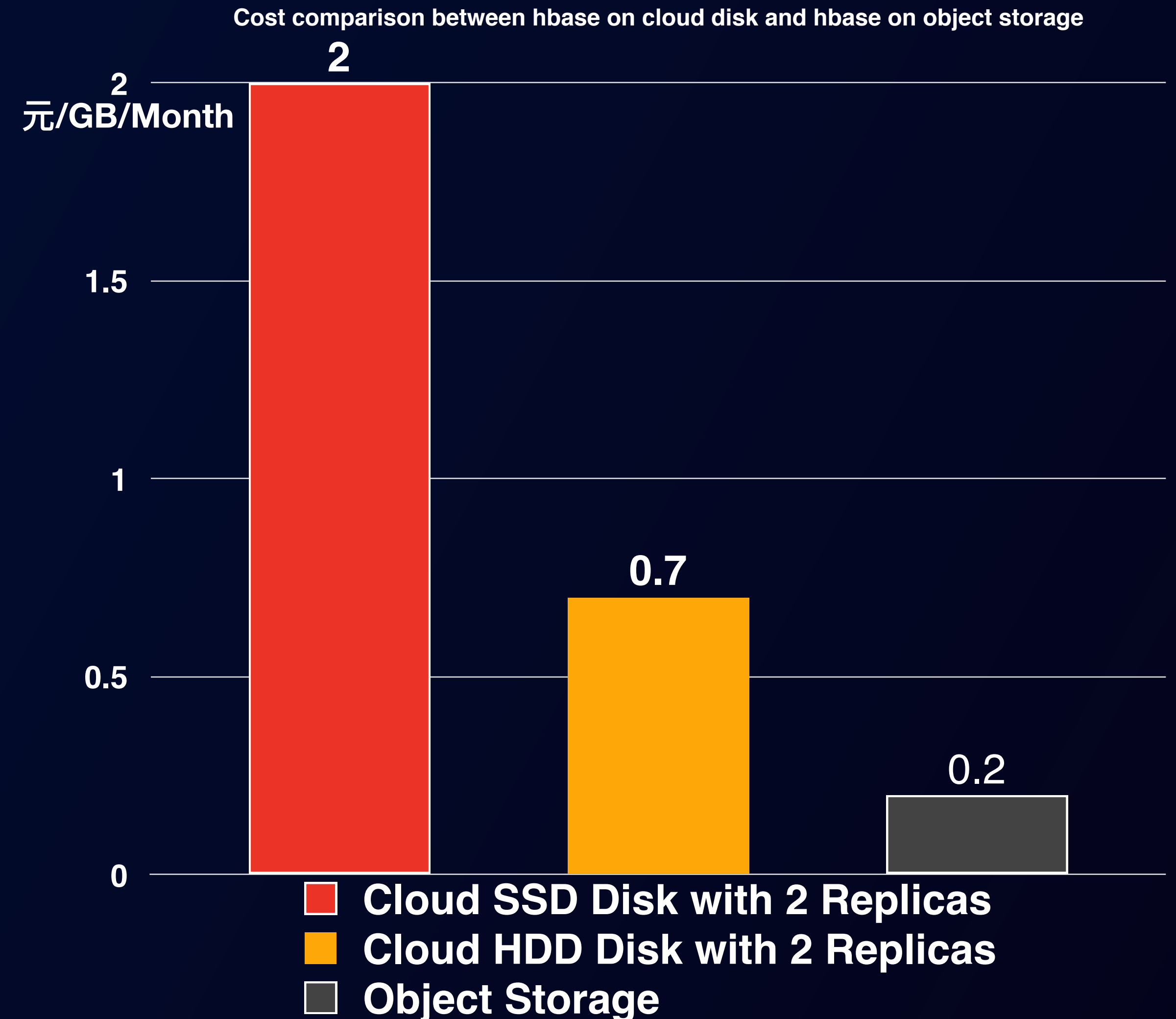
# Log与File分离存储

## Separate Storage of Log and File



Write Performance **1** : **1** : **1**

Read Performance **7** : **3.5** : **1**

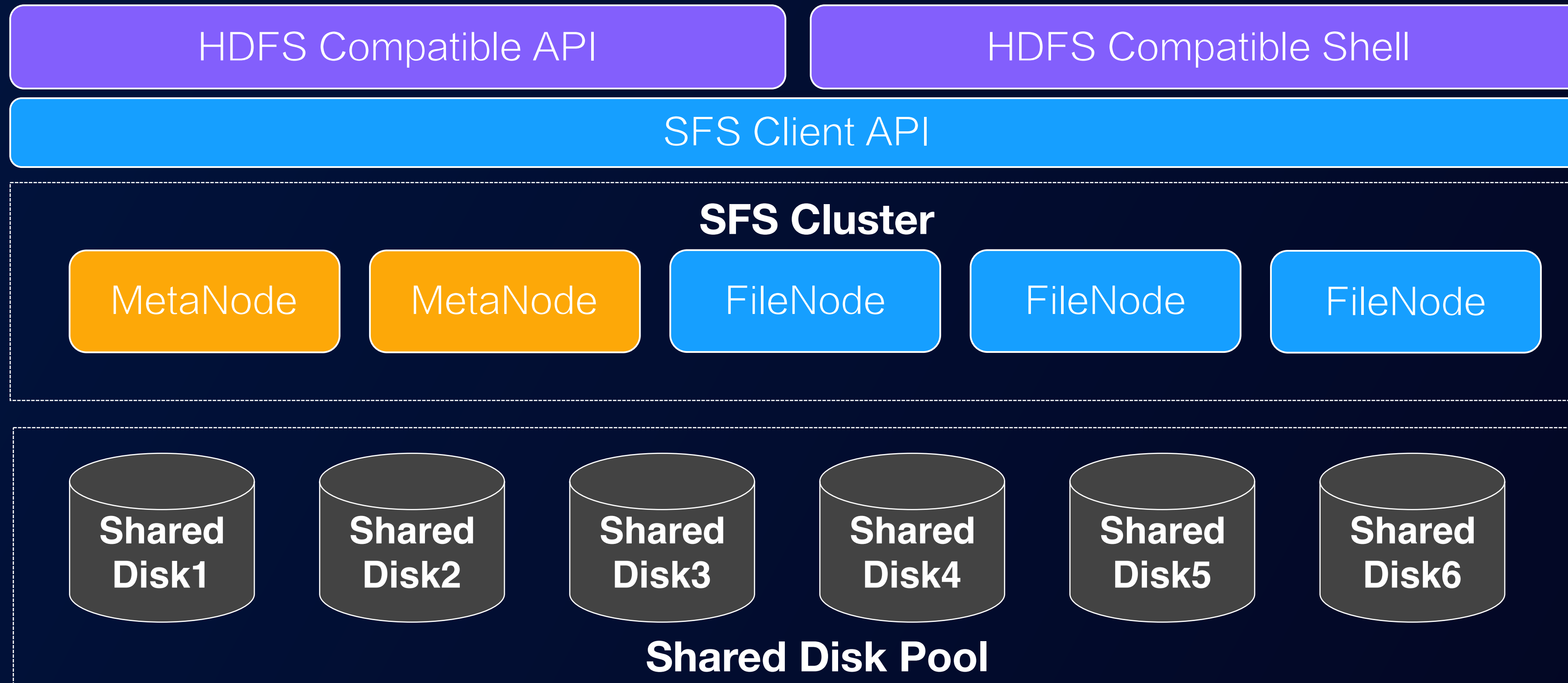


The above data comes from the HBase test on Alibaba Cloud, only for reference



# 使用共享块存储

## Make use of shared block storage



**HDFS on Normal Cloud Disk**  
**At least 2 replicas**

**SFS on Shared Block Storage**  
**Only 1 replica**

**Advantage of SFS:**

- 1. 200% 存储容量提升**  
More available storage than HDFS
- 2. 130% 吞吐性能提升**  
Better throughput
- 3. 兼容HDFS客户端**  
HDFS compatible
- 4. 水平扩展至PB级存储**  
Scale out to PB level
- 5. 单节点故障持续可用**  
Automatic fault tolerance

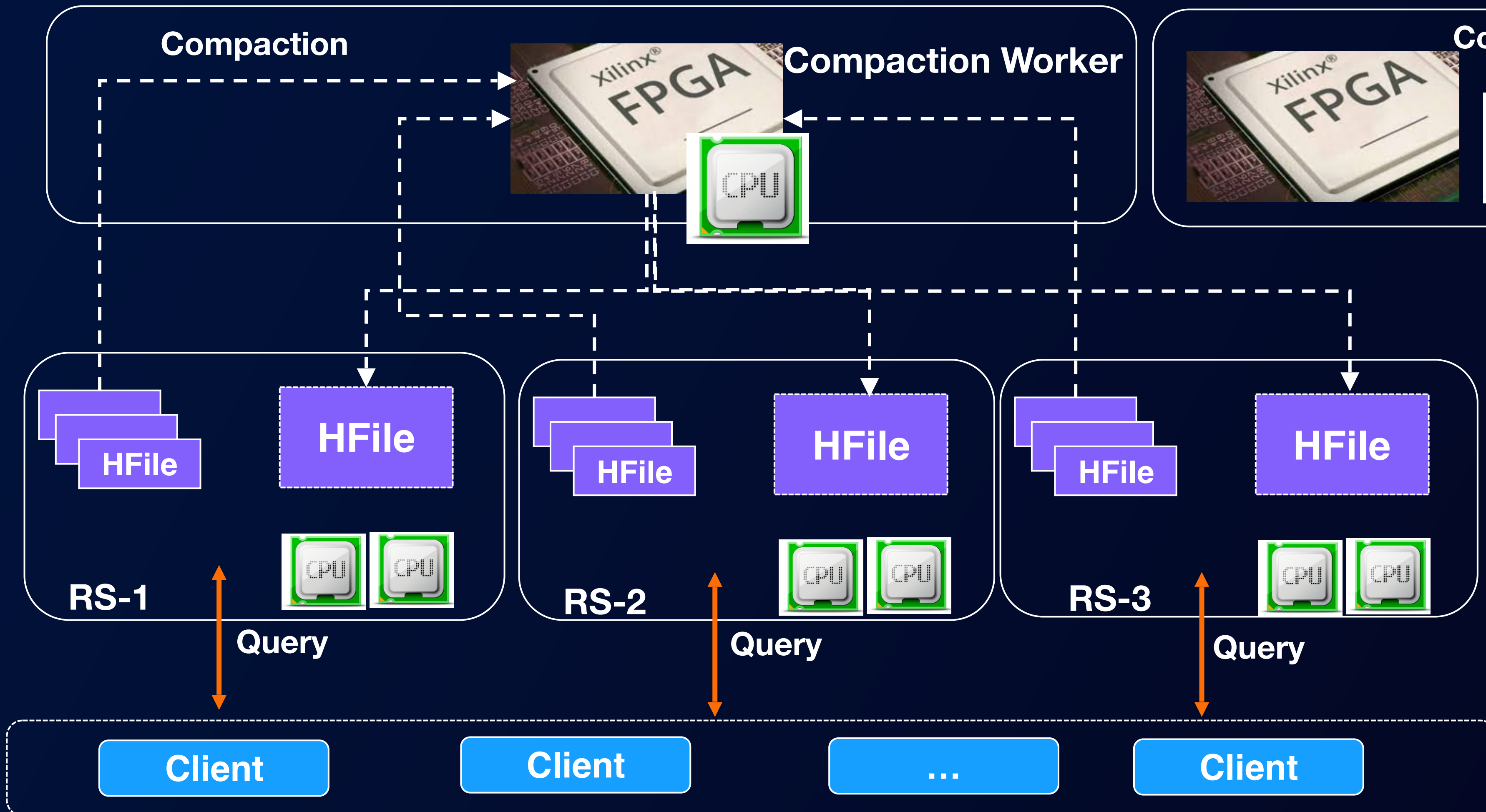
**共享块存储: 一种支持多台ECS实例并发读写访问的数据块级存储设备**

Shared block storage supports concurrent access from multiple ECS instances

**SFS: 基于共享云盘的云原生分布式文件系统**

SFS-A Cloud Native Distributed File System based on cloud shared block storage

# Compaction on FPGA



Cloud accelerates landing  
the new hardware

At the same cost , FPGA shows **3X higher** compaction performance than CPU

**Eliminate the impact** of compaction on online requests

# 使用阿里云HBase，获得最佳云上体验

		阿里云数据库HBase <a href="https://cn.aliyun.com/product/hbase">https://cn.aliyun.com/product/hbase</a>	自建HBase
核心指标	吞吐性能	是社区版的3-7倍	N/A
	毛刺	P99延迟是社区版的1/10	N/A
	压缩率	是社区版的1.5-2倍	N/A
	可用性	SLA保障，单集群99.9%，双集群99.99%	N/A
	主备容灾	成熟、支持跨版本、主无影响、且支持透明切换	无优化，不支持切换
企业特性	数据搬迁	支持在线、跨版本、自动化、高效搬迁，应用0影响、0改造	自己用工具、效率&正确性无保障
	备份恢复	支持数据备份至OSS及恢复	不支持
	冷热分离	支持，冷数据存储在OSS，成本减少70%	不支持
	多租户	支持全局Quota限流	只支持单Server
生态体验	MySQL数据同步	提供产品化功能、支持在线增量同步	自己用工具、不支持在线增量
	Spark分析	产品化深度集成，支持高性能分析、增量归档、结果回流等	无优化，数据集成需要较大开发
功能	二级索引	支持强一致、最终一致、本地索引等多形态，性能大幅优化	依赖外部组件
	全文索引	产品化关联外部搜索引擎Solr、Elasticsearch，自动数据同步	不支持
	专业运维	支持HBase-Manager (表管理、数据查询、热点识别、自动合并、大scan识别等)	无
技术团队	专家	拥有HBase领域专家数十人 Apache社区4 PMC、6 Committer	-
	社区	中国HBase技术社区发起者 定期主办HBase Con、Meetup等中大型会议	-
	实践经验	内部打磨9年，支持天猫双十一，阿里部署10000+台，公有云600+集群	-



# Thanks!

欢迎咨询  
求贤若渴





# **HBASECON** ASIA2019

**THE COMMUNITY EVENT FOR APACHE HBASE™**

**July 20th, 2019 - Beijing, China**