ALIBABA CLOUD

阿里云

云数据库HBase版

云数据库HBase产品简介

文档版本：20220225

阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。

2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。

3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。

4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。

5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。

6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	危险 重置操作将丢失用户配置数据。
警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	警告 重启操作将导致业务中断，恢复业务时间约十分钟。
注意	用于警示信息、补充说明等，是用户必须了解的内容。	注意 权重设置为0，该服务器不会再接受新请求。
说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击**设置**> **网络**> **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
`Courier字体`	命令或代码。	执行 `cd /d C:/window` 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	`bae log list --instanceid` *Instance_ID*
[] 或者 [a|b]	表示可选项，至多选择一个。	`ipconfig [-all|-t]`
{} 或者 {a|b}	表示必选项，至多选择一个。	`switch {active|stand}`

目录

1.产品概述 05
2.产品优势 06
3.专家服务 09
4.应用场景 10
5.集群规格选型 12
6.HBase版本选择 15
7.产品形态（单节点/集群版本） 17
8.实例节点规格 18
9.主备方案介绍 20
10.如何设计Rowkey 26

1.产品概述

阿里云HBase是低成本、高扩展、云智能的大数据NoSQL，兼容标准HBase访问协议，提供低成本存储、高扩展吞吐、智能数据处理等核心能力，是为淘宝推荐、花呗风控、广告投放、监控大屏、菜鸟物流轨迹、支付宝账单、手淘消息等众多阿里巴巴核心服务提供支撑的数据库，具备PB规模、千万级并发、秒级伸缩、毫秒响应、跨机房高可用、全托管、全球分布等企业能力。

浅析场景

阿里云HBase全面提供海量半结构/非结构化数据下的实时存储、高并发吞吐、轻SQL分析（参见Lindorm产品）、全文检索（参见Lindorm产品）等能力， 结合完备的工具服务，丰富的生态融合，一站式高效满足企业在大数据量场景下的存储、检索、分析需求，是风控、推荐、广告、物联网、车联网、Feeds流、数据大屏等场景首选数据库。

代表场景	类型	存储对象	引擎接口	优势
社交分发或聊天信息等	KV	key和value。	HBase API	延迟低、并发高。
传统关系型数据库升级	SQL	关系模型。	Phoenix SQL	存储量大、分布式SQL数据库。
风控或画像	表格存储	稀疏Table。	HBase API	动态Schema、存储量大。
新闻	文档存储	JSON、XML或HTML。	HBase API	存储量大、并发高。
图片或视频	对象存储	图片、视频等。	HBase API	存储量大。
传感器或物联网	时序数据	时间维度的连续数据。	OpenTSDB或HBase API	存储量大。
车联网或运动App	时空数据	有空间维度或轨迹。	GeoMesa或HBase API	存储量大，写入性能高。
全文查询	全文	全文。	solr或HBase API	吞吐量大。

典型案例

应用场景，请参考案例集合。

产品优势

请参考产品优势。

2.产品优势

本文介绍HBase的优势。

类别		云数据库HBase增强版（Lindorm）	HBase开源版
核心功能	HBase API	支持	支持
	数据模型	支持宽列（HBase API）、表格（SQL-Like API）、队列等多种，对其他模型感兴趣的话，请联系我们。	仅支持宽列
	全局二级索引	内置，查询透明、高性能、按需冗余非索引列，请参见二级索引。	依赖外部组件，复杂。
	全文检索	智能集成搜索引擎Solr，提供面向海量数据的存储、多维查询、全文索引等统一访问的混合型能力，请参见全文索引服务。	不支持
性能	吞吐性能	单机吞吐是开源HBase的7倍，请参见测试结果。	无优化
	请求毛刺	P99延迟是开源HBase的1/10，请参见测试结果。	无优化
成本	数据压缩	深度优化的ZSTD算法，JNA重写，可杜绝Core dump，并且基于字典采样优化，压缩率比snappy提高50%，数据压缩比可高达10:1。	默认为Snappy，使用ZSTD需依赖Hadoop-3.0，存在Core Dump的概率。
	编码	IndexableDelta算法，相比DIFF算法，保持相同压缩率，访问速度快一倍。	推荐DIFF，随机访问较慢。
	冷热分离	冷热数据自动分层存储，冷数据使用高压缩和廉价存储介质，减少70%成本，热数据可提升访问性能15%，请参见冷热分离。	不支持

类别		云数据库HBase增强版（Lindorm）	HBase开源版
	存储介质	支持高效云盘、SSD云盘、本地HDD、本地SSD。支持冷存储（OSS）、容量型云盘（超性价比云盘，即将发布）。	N/A
可靠性	主备双活	成熟，支持自动容灾切换、双集群请求并发等高级能力，支持与自建HBase构建混合主备，请参见主备双活。	无优化，不支持切换。
	备份恢复	支持100TB+规模的数据备份至OSS，并提供与规模无关的RTO（<30分钟）、按需备份、指定时间点恢复等高级能力，请参见开通备份恢复。	不支持
	MTTR	深度优化，宕机恢复速度是开源HBase的10倍。	不支持
多租户	认证与ACL	支持易用的账号密码认证和ACL管理，请参见用户和ACL管理。	较复杂
	资源隔离	支持RS Group功能，实现租户间的资源物理隔离。	不支持
运维诊断	运维工具	界面化集群管理工具，支持表，Namespace，Group，ACL等管理，请参见集群管理系统。	HBase Shell
	数据查询	HBase Shell+集群管理系统内支持图形化SQL交互查询，请参见数据查询。	HBase Shell
	数据搬迁	支持HBase各个版本之间的在线、跨版本、自动化、高效搬迁，应用零影响、零改造，请参见LTS（原BDS）服务介绍。	只能离线迁移
	MySQL数据同步	请参见LTS（原BDS）服务介绍，支持MySQL数据到HBase的全量同步、在线实时同步。	借助工具、不支持在线增量。

类别 生态		云数据库HBase增强版（Lindorm）	HBase开源版
	Spark分析	产品化深度集成，支持Spark SQL分析HBase，HBase数据到Spark（HDFS/OSS）的增量归档，离线分析结果回流到HBase等。	无优化，数据集成需要较大开发。
	MaxCompute	产品化集成，请参见全量导出MaxCompute。	数据集成需要较大开发。
	日志服务（SLS）	请参见LTS（原BDS）服务介绍，支持SLS数据到HBase的日志服务（SLS）增量导入。	数据集成需要较大开发。
服务能力	可用性SLA	提供SLA保障，单集群99.9%，双集群高可用99.99%。	N/A
	运维成本	全托管，无需复杂的数据库运维投入。	N/A
	技术团队	由多名Apache社区PMC&Committer组成的专家队伍提供技术服务支持。	N/A
	实践经验	支持9年天猫双十一，阿里部署上万台。	N/A

与传统数据库对

		阿里云HBase	传统关系型数据库(MySQL/PG)
性能规模	存储	平滑扩展至PB级	百GB内较合适，最大扩展至几TB
	请求	~1亿+	~10w合适
	单表行数	~万亿+	~亿内合适
	单表列数	~十万+	~100内合适，最大1000
	行大小	0~10MB	~8KB内合适，最大64KB
可用性	RT	毫秒级	毫秒级
	SLA	99.9%(默认，单机房)，99.99%(可选，双机房)	99.99%(默认，2或3机房)
	MTTR(单节点)	2~5分钟	<3分钟
成本	存储介质	本地HDD，OSS，高效云盘，本地SSD，SSD云盘	本地SSD，SSD云盘
	数据压缩	压缩率达5~10倍	空间碎片，压缩率~3倍(可选，会影响写性能)
	单节点吞吐	单行: ~万/秒，批量: ~几十万/秒	~万/秒，写吞吐会落后于HBase
功能	事务	弱，单行事务	强
	标准SQL	支持	支持，成熟完善
	索引	支持二级索引，并配合solr提供全文检索	支持
	灵活Schema	强，动态列、TTL、多列簇等	弱
生态	数据可视化	弱	丰富
	大数据生态	与Spark、Hadoop生态结合紧密	提供一般支持

3.专家服务

您可通过官网，阿里云开发者社区，或专家服务钉钉群等途径，来进一步了解HBase或解决疑问。

- 官网：您可登录官网，了解HBase的详细信息、快速操作、常见问题处理等信息。
- 开发者社区：关注HBase的开发者社区。
- 云HBase专家服务群：搜索钉钉群号『23135738 』加入技术专家服务群。
- 工单：通过提交工单来提交遇到的困难或问题。

4.应用场景

云数据库HBase是一个键值/宽表型的分布式数据库，适用于任何数据规模，可以提供单个毫秒响应的性能，尤其擅长低成本、高并发的场景，支持水平扩展到PB级存储和千万级QPS，在淘宝、支付宝、菜鸟等众多阿里巴巴核心服务中起到了关键支撑的作用。

背景

HBase的设计之初是为了满足互联网的大数据场景，几乎所有非强事务的结构化、半结构化的存储需求都可以使用HBase来满足。ApsaraDB for HBase是NoSQL（Not only SQL），也支持SQL及二级索引。

大数据场景

云数据库HBase支持海量全量数据的低成本存储、快速批量导入和实时访问，具备高效的增量及全量数据通道，可轻松与Spark、MaxCompute等大数据平台集成，完成数据的大规模离线分析。

优势如下：

- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 数据通道：通过BDS构建云HBase与异构计算系统的高效、易用的数据链路。
- 快速导入：通过BulkLoad将海量数据快速导入HBase，效率比传统方式提升一个数量级。
- 高并发：水平扩展至千万级QPS。
- 弹性：存储计算分离架构，支持独立伸缩，自动化扩容。

广告场景

使用云数据库HBase存储广告营销中的画像特征、用户事件、点击流、广告物料等重要数据，提供高并发、低延迟、灵活可靠的能力，帮助您构建领先的实时竞价、广告定位投放等系统服务。

优势如下：

- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 高并发：水平扩展至千万级QPS。
- 使用灵活：动态列，自由增减特征/标签属性；TTL，数据自动过期。
- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 数据通道：通过BDS构建云HBase与异构计算系统的高效、易用的数据链路。
- 高可用：主备双活容灾，请求自动容错，满足99.99% SLA。

车联网

使用云数据库HBase存储车联网中的行驶轨迹、车辆状况、精准定位等重要数据，提供低成本、弹性、灵活可靠的能力，帮助您构建领先的网约车、物流运输、新能源车检测等场景服务。

优势如下：

- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 弹性：存储计算分离架构，支持独立伸缩，自动化扩容。
- 使用灵活：动态列，自由增减特征/标签属性；TTL，数据自动过期；多版本。
- 数据通道：通过BDS构建云HBase与异构计算系统的高效、易用的数据链路。
- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 高可用：主备双活容灾，请求自动容错，满足99.99% SLA。

互联网

使用云数据库HBase存储社交场景中的聊天、评论、帖子、点赞等重要数据，提供易开发、高可用、延迟的能力，帮助您快速构建稳定可靠的现代社交Feed流系统。

金融和零售

使用云数据库HBase存储金融与零售交易中的海量订单记录，金融风控中的用户事件、画像特征、规则模型、设备指纹等重要数据，提供低成本、高并发、灵活可靠的能力，帮助您构建领先的金融交易与风控服务。

优势如下：

- 低成本：高压缩比，数据冷热分离，支持HDD/OSS存储。
- 高并发：水平扩展至千万级QPS。
- 使用灵活：动态列，自由增减特征/标签属性；TTL，数据自动过期；多版本。
- 低延迟：单个毫秒响应，支持双集群请求并发加速。
- 数据通道：通过BDS构建云HBase与异构计算系统的高效、易用的数据链路。
- 高可用：主备双活容灾，请求自动容错，满足99.99% SLA。

5.集群规格选型

阿里云数据库HBase支持自由组合Master和Core节点规格，数量和盘的种类。您可以根据业务特征（如QPS、存储量、读写请求、相应延迟、稳定性等）自由搭配。

选型主要包括以下几个方面：

- Master机器的规格。
- Core节点的规格和数量。
- 盘的大小和种类。
- HBase版本。参见HBase版本选择。
- ECS版本。
  - 独享规格：完全没有争抢的规格，资源保障稳定，延迟敏感的业务请务必使用独占且SSD云盘。
  - 通用规格：大于等于8CPU，不提供通用。通用会有一定的争抢。

> **说明** 业务生产集群不建议使用通用规格。

Master选择

Master节点不带存储，默认是主备保障单点容灾。Master节点上会部署HBase的Master，HDFS的Namenode和Zookeeper等，是集群中重要节点。如果master的CPU不够或者内存不够，会严重制约HBase的性能。

core的规模	master选择
< 4台	4CPU/8 G
4 <= core数 < 8	8CPU/16 G **（小规模集群推荐Master规格）**
8 <= core数 <16	8CPU/32 G
大于16台	16CPU/64 G或者以上

> **注意** Master的规格不仅受到Core节点数量的影响，还和集群上表的数量和地域的数量有关，如果集群的表数量或者地域数量非常多，Master节点也需要适当选择高规格。

Core选择

Core节点指HBase的RegionServer，您可根据业务请求量来选择合适的Core节点规格。Core节点支持的范围为：4CPU/8 G 至 32CPU/128 G。

为了达到最佳性能，通常我们以元信息能够全部放入缓冲作为高性能的重要依据。针对不同的实例，如下给出的节点选择示例供参考：

- 小实例：推荐使用4CPU/16 G或8CPU/32 G。
- 中大实例：根据内存与数据量的比例，来选择Core节点。
  - 存储量大：推荐使用16CPU/64 G或32CPU/128 G。
  - 存储量小：推荐使用16CPU/32 G或32CPU/64 G。

> **说明** 请求量的多少并不是Core节点选择的唯一标准，进行节点选择时，需要综合多方面考虑。例如：某个业务的请求量为每秒数百，根据请求量判断，选择4CPU/8 G的节点即可。但如果该业务中，若假设存在以下几种情况，选择4CPU/8 G的Core节点可能会导致业务不稳定和延迟上涨。
>
> - 用户请求的一行数据非常大，存在几KB，甚至数MB。
> - Scan请求中存在复杂的Filter。
> - 请求的Cache命中率很低，每个请求都会请求磁盘。
> - 集群上表和region非常多。
>
> 若需要辅助容量评估，请在钉钉上联系 `云HBase答疑` 钉钉号或者发起工单咨询。

下表给出一些简单的建议，但是评估Core节点的规格需要综合多方面考虑。

TPS+QPS	core的个数及选择	建议
0 ~ 1000	2台 4CPU/16G	推荐的最低规格，适合低负载访问。每台Core节点上的region不宜超过600个。 **云HBase中能买到的最低Core节点为4CPU/8 G，但不推荐4CPU/8 G规格，8 GB的内存过小，在面对高峰和大KV时容易出现OOM等不稳定现象。**
1000 ~ 2W	2~3台 8CPU/32G	**8CPU/32 G是我们推荐的适合中低负载的规格，价格相比8CPU/16 G贵的不多，但内存翻倍，稳定性更好。**
2W以上	8CPU/32G，16CPU/32G，16CPU/64G，32CPU/64G，32CPU/128G，或者以上	按照实际请求量选择Core节点的数量，如果是在线型业务，适当选择大内存机型来增加cache命中率来提高性能。如果有MR/Spark等离线大负载任务要运行或者TPS/QPS非常高，需要适当选择大CPU机器。

选择高配置机器还是更多Core节点？

HBase支持水平无限扩展，当出现负载过高，延迟变高，不稳定，可以通过水平增加Core节点解决。但是由于业务设计的问题或者使用问题，很可能出现单机热点，因此Core节点的规格**决定了单机抗热点的能力**。因此把Core节点的规格选择的特别低，只通过水平扩展Core节点的方式，在面对风险时的稳定性会没有高配机型来得强。如果有瞬时的大请求打过来，或者偶发的热点流量（在一个地域），低配的机器可能会出现负载过高或者OOM影响整体的稳定性。

因此，我们建议**Core节点的规格要根据业务事先做好评估，选择合适规格的Core节点**。

如果事先选择的Core节点或者Master节点规格无法满足要求，我们还将提供原地升配的功能，将Core/Master的配置升级到合适规格，详情请钉钉上联系 `云HBase答疑` 钉钉号或者发起工单咨询。

存储介质

存储介质分为3个大类，云盘，本地盘和冷存储。

- **云盘**： 云盘的特点是灵活和高可靠。云盘本身有副本冗余，完全屏蔽了硬件细节，不会因为坏盘等原因丢失数据。并且可以自由扩容，是业务的首选存储介质。云盘分为SSD云盘和高效云盘。
- **本地盘**： 本地盘即真实的物理盘，特点是价格比云盘便宜，但是本地盘大小与机型强绑定，不能对单块磁盘扩容，只能通过增加Core节点来增加存储容量，而且**本地盘机型ECS也不支持升配**，灵活性没有云盘强。同时，本地盘如果出现坏盘会对用户产生影响（单块坏盘不会出现丢数据，出现坏盘后，云HBase团队会第一时间处理更换）。本地盘购买的起步较高，适合存储容量大的大客户。
- **冷存储**：阿里云HBase特有存储，基于OSS实现，冷存储可以搭配云盘使用，用户可以将不常用数据存储

到冷存储或者使用冷热分离功能自动实现冷数据归档来降低成本。

在选定了存储介质后，集群的存储介质不可以再更改，其中云盘可以通过扩容或者增加Core节点来增加容量，但本地盘只能通过增加Core节点扩容。例外的是冷存储， 冷存储功能不一定需要在开通HBase集群时购买，可以在之后的使用过程中随时开通和扩容。

冷热程度	介质类型	业务类型
高性能	SSD云盘/本地SSD盘/ESSD云盘	适合对延迟要求高的在线业务，如广告，推荐，Feed流，人物画像等业务，SSD/ESSD盘的延迟低，通常在1~2ms左右，最重要的是毛刺率低。对P99延迟（99%请求的最大延迟）在意的用户首选。
高效能	高效云盘/本地HDD盘	适合对延迟要求一般的在线业务，HDD盘的延迟通常在10ms量级，但毛刺率要比SSD盘要大。
冷数据	OSS（冷存储）	准在线业务，或者归档业务。冷存储配合云盘/本地盘写吞吐基本与云盘/本地盘一致。但读冷存储QPS有限制，读延迟通常在数十毫秒量级。

6.HBase版本选择

本文介绍HBase的各个版本信息以及指导用户如何选择。

云数据库HBase标准版

HBase标准版使用的是社区的开源HBase分支，并在此基础上定制部分能力，同时优化了稳定性和运维能力。这些能力主要来源包括：

- 内部分支（增强版）的优化。
- 开源新版本中的patch。
- 测试及生产中发现的bug修复、体验优化。

这些定制的能力都具有比较灵活的移植能力，所以当社区发布新的大版本时，我们也会快速跟进发布基于此的定制优化版，并结合阿里云内部的完善测试回归体系相应的保证我们的客户既能用到最新最全的功能，同时又享受稳定性保障和一定的能力优化。目前HBase企业标准版有1.1版和2.0版本，分别基于开源HBase1.1.2版本和HBase2.0.0版本打造，100%兼容社区HBase。

云数据库HBase增强版

HBase增强版是基于云原生多模数据库内核研发的、完全兼容HBase的云上托管数据库，已经在阿里集团内部发展10年多，支撑服务了淘宝、支付宝、菜鸟、优酷、高德等业务中的大量核心应用，在成本、性能、稳定性、功能、安全、易用性等方面相比社区版HBase拥有诸多优势和企业级能力。在使用上，用户可以通过HBase官方客户端及标准API访问HBase增强版，HBase增强版同时兼容HBase-1.x和HBase-2.x客户端访问，在整体功能及API上保证近100%的兼容性。HBase增强版提供最高7倍于HBase开源版本的性能和一半的存储成本，详情请参见产品优势。

云数据库HBase Serverless版

HBase Serverless版是基于增强版内核构建的Serverless模式服务，提供6元/月的超低起步价格，是一种极致弹性、按需计费的产品形态，适合于培训学习、测试、间歇性负载、小规模生产等场景，更多内容请参见基本介绍。

如何选择

- HBase增强版在成熟度、经济性、规模性等方面具有非常大的领先，具备大量专业的企业级功能、管理工具，并经过阿里核心业务的大规模验证，适合于通用的企业级客户场景。
- HBase标准版是经过优化的开源定制版，适合于增强版暂时无法满足的用户场景。
- HBase Serverless版是一种Serverless模式的服务，适合于低成本的学习、测试的用户场景。

> **说明** HBase增强版目前支持HDFS、Spark、Hive、OpenTSDB等常见大数据生态（暂不支持Phoenix），但还不支持部分组件（Kylin、Geomesa），我们正在加紧适配中。对于HBase原生的功能、客户端API，增强版与开源标准保持一致。 HBase增强版内核使用的是Lindorm宽表引擎，更多强大的企业级功能如SQL支持和高性能原生二级索引，请参见Lindorm产品。

版本功能对比

> **说明** 下表中✔代表支持，✘代表不支持。

功能详情	HBase版本		
	云数据库HBase增强版	云数据库HBase标准版	HBase开源版
性能优化	✓	✓	✗
全局二级索引	✓	✗	✗
全文检索（兼容Solr）	✓	✗	✗
冷热分离	✓	✗	✗
企业级备份	✓	✓	✗
主备容灾	✓	✓	✗
主备双活	✓	✓	✗
跨机房强一致	✓	✗	✗
智能诊断与管理	✓	✓	✗
大数据生态	✓	✓	✓

7.产品形态（单节点/集群版本）

本文主要介绍云数据库HBase支持产品形态。

-	集群版	单节点版	Serverless版	跨可用区双集群版（欢迎咨询）
费用（以实际售卖为准）	>2500元/月	100-1000元/月	>6元/月	>5000元/月
适用场景	任何规模的企业生产	开发测试生产预发小规模生产	培训学习、测试间歇性负载小规模生产	任何规模的企业生产
服务等级协议	标准SLA	无SLA	无SLA	更高SLA
功能	全	全	基础数据访问（不支持高级功能，具体见下）	全
扩展性	强（接近无限规模）	中（可在线迁移至集群版）	中（一定规模下弹性）	强（接近无限规模）
计费模式	资源价格x节点数量支持包年包月支持小时计费	资源价格x节点数量支持包年包月支持小时计费	请求/存储计费支持包年包月支持小时计费	资源价格x节点数量支持包年包月支持小时计费
技术架构	分布式部署多副本、自动容灾	单机部署无容灾能力	多租户共享分布式部署多副本、自动容灾	双集群部署机房级容灾

- 支持原生HBase API。
- 不支持HDFS的开放访问。

更多详情，请参见Serverless文档。

8.实例节点规格

根据所选引擎的不同，选择如下所属的Master和Core规格信息。

引擎类别	Master规格	Core规格

HBase标准版/Hbase增强版	• 4 核 8 GB（独享规格）hbase.sn1.large
	• 8 核 16 GB（独享规格）hbase.sn1.2xlarge
	• 8 核 32 GB（独享规格）hbase.sn2.2xlarge
	• 16 核 32 GB（独享规格）hbase.sn1.4xlarge
	• 16核 64 GB（独享规格）hbase.sn2.4xlarge
		• 云盘2倍mem/cpu比（磁盘类型=cloud_ssd,cloud_essd_pl1,cloud_efficiency）
		  ○ 4 核 8 GB（独享规格）hbase.sn1.large
		  ○ 8 核 16 GB（独享规格）hbase.sn1.2xlarge
		  ○ 16 核 32 GB（独享规格）hbase.sn1.4xlarge
		  ○ 32 核 64 GB（独享规格）hbase.sn1.8xlarge
		• 云盘4倍mem/cpu比（磁盘类型=cloud_ssd,cloud_essd_pl1,cloud_efficiency）
		  ○ 4 核 16 GB（独享规格）hbase.sn2.large
		  ○ 8 核 32 GB（独享规格）hbase.sn2.2xlarge
		  ○ 16 核 64 GB（独享规格）hbase.sn2.4xlarge
		  ○ 32 核 128 GB（独享规格）hbase.sn2.8xlarge
		• 本地盘HDD（磁盘类型=local_hdd_pro）
		  ○ 8 核 32 GB（本地盘-独享规格）hbase.d1.2xlarge
		  ○ 16 核 64 GB（本地盘-独享规格）hbase.d1.4xlarge
		  ○ 24 核 96 GB（本地盘-独享规格）hbase.d1.6xlarge
		  ○ 32 核 128 GB（本地盘-独享规格）hbase.d1.8xlarge

引擎类别	Master规格	Core规格
		○ 32 核 128 GB（本地盘-独享规格） hbase.d1-c8d3.8xlarge ● 本地盘SSD（磁盘类型 local_ssd_pro） ○ 4 核 32 GB（本地-ssd） hbase.i2.xlarge ○ 8 核 64 GB（本地-ssd） hbase.i2.2xlarge ○ 16 核 128 GB（本地-ssd） hbase.i2.4xlarge ○ 32 核 256 GB（本地-ssd） hbase.i2.8xlarge

引擎为Bds

引擎类型	Master规格	Core规格
Bds	可为空字符串	● 4 核 8 GB（独享规格）hbase.sn1.large ● 8 核 16 GB（独享规格）hbase.sn1.2xlarge ● 16 核 32 GB（独享规格）hbase.sn1.4xlarge ● 32 核 64 GB（独享规格）hbase.sn1.8xlarge

引擎为HBase单机版

createCluster openapi nodeCount = 1

terraform core_instance_quantity = 1

引擎类型	Master类型	Core类型
HBase单机版	可为空字符串	● 2 核 4 GB hbase.sn1ne.large ● 4 核 8 GB hbase.sn1ne.xlarge

9.主备方案介绍

云数据库HBase存储海量大数据，在业务场景中往往承载着重要数据，为保障数据的高可用性和安全性，云数据库HBase提供了主备双活和主备容灾特性。本文介绍云数据库HBase的主备双活和主备容灾特性功能。

使用场景

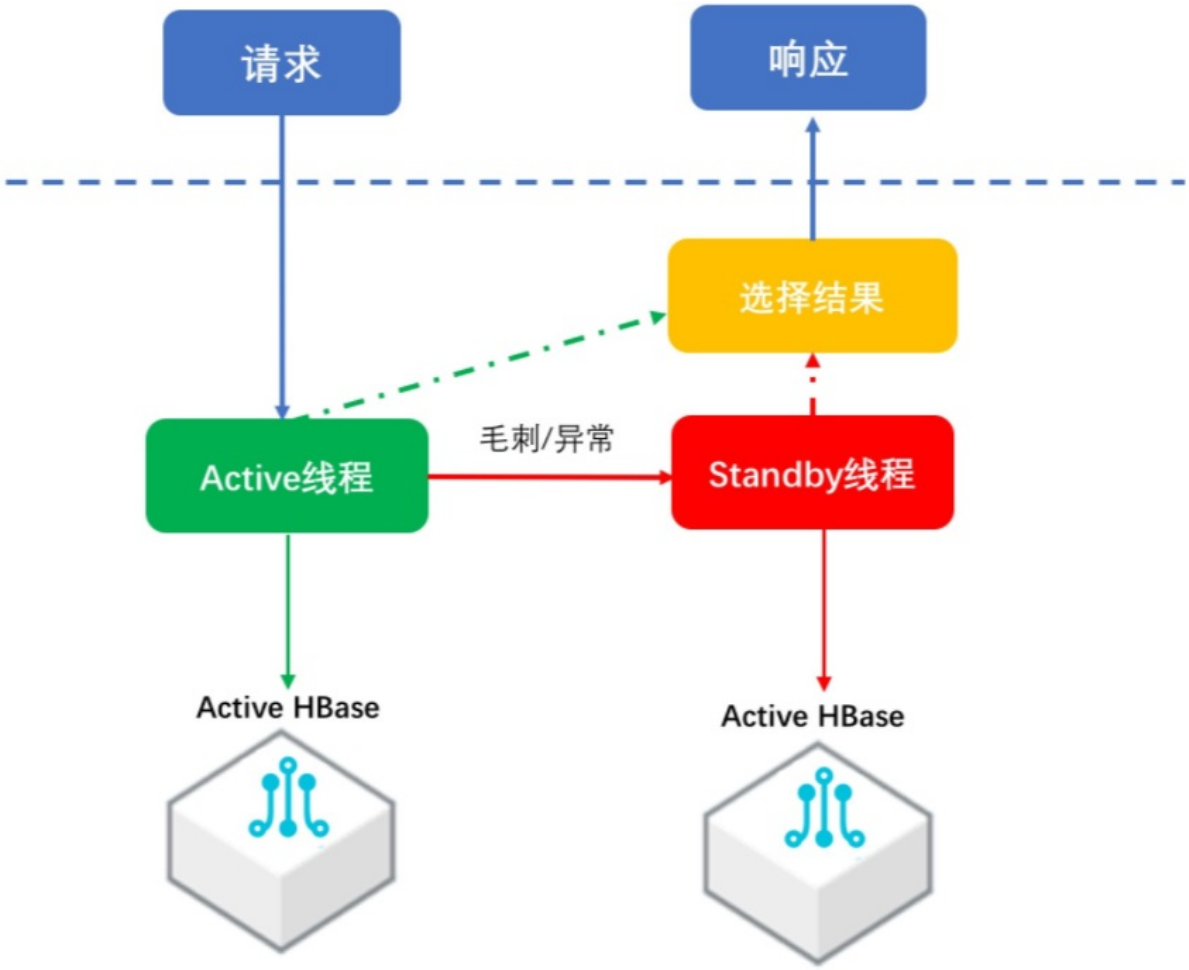
- 主备双活：大数据量随机读响应有高要求（P999小于50ms）的实时在线业务，例如用户推荐，安全风控等。
- 主备容灾：当云数据库HBase实例因不可预料的原因（例如设备故障、机房断电断网等）发生故障，容灾机制可用于保障数据的一致性和业务可用性。

功能优势

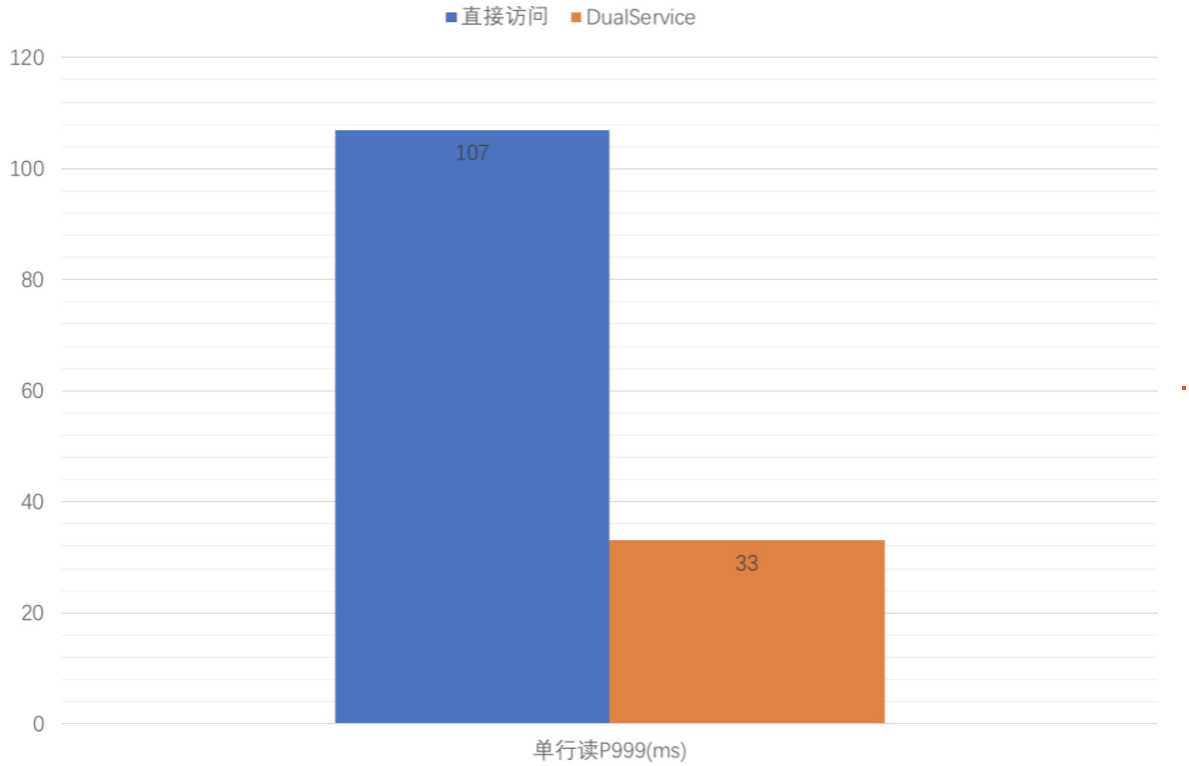
- 主备双活：
  - 请求低毛刺。
  - 故障自动容错。
  - 资源利用率高。
- 主备容灾：
  - 支持云数据库HBase、EMR HBase、自建HBase混合主备。
  - 支持1.x、2.x多个版本。
  - 自动数据同步管理，高效双向同步，延迟在百毫秒级。
  - 故障自动容错。
  - 在主备控制台可查看同步延迟、客户端链接数、主备状态等详细信息。

主备双活原理介绍

- 低毛刺：两个独立节点同时产生毛刺概率比单节点产生毛刺概率低，采用双节点服务相同数据在最终一致性条件中能够提高数据的稳定性。当用户请求发起时先向主库发送请求，一段时间后如果主库没有返回结果，则向备库发送请求，最终取最快返回的值作为结果。发送请求和响应的流程图如下。

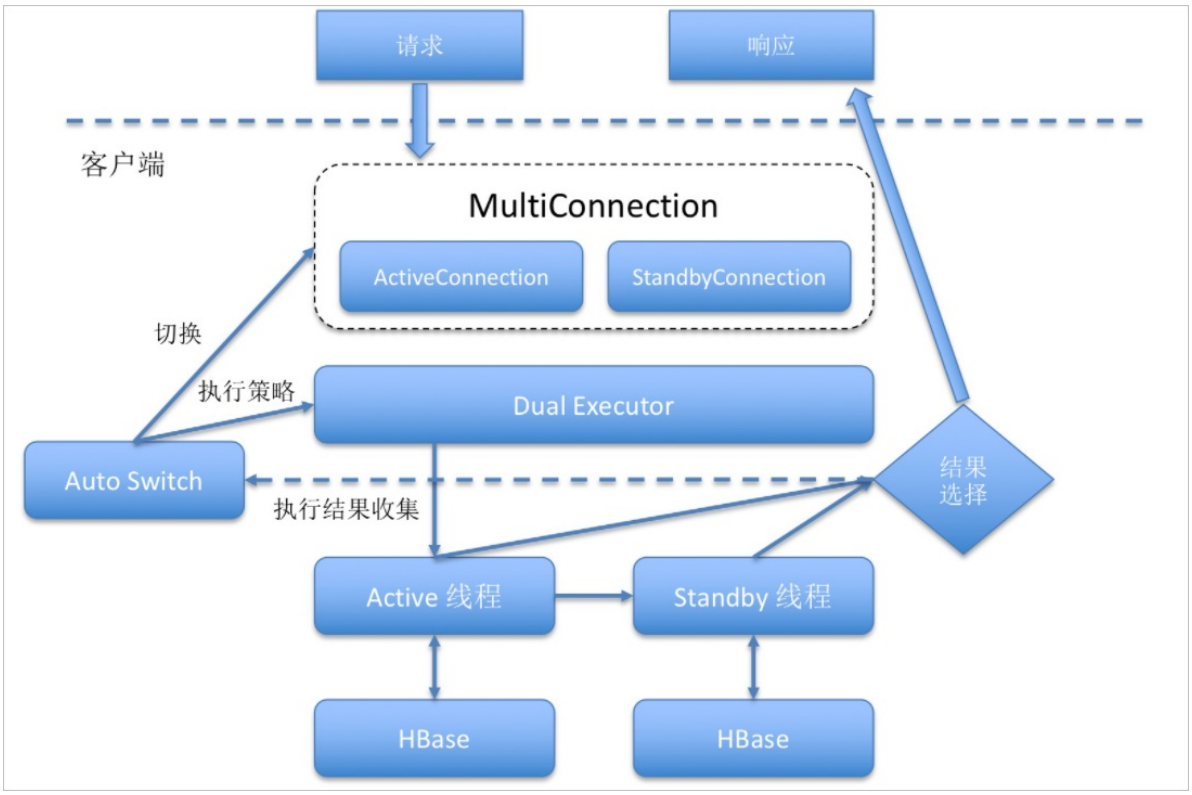


以下是直接访问和使用主备双活功能访问的时间对比。



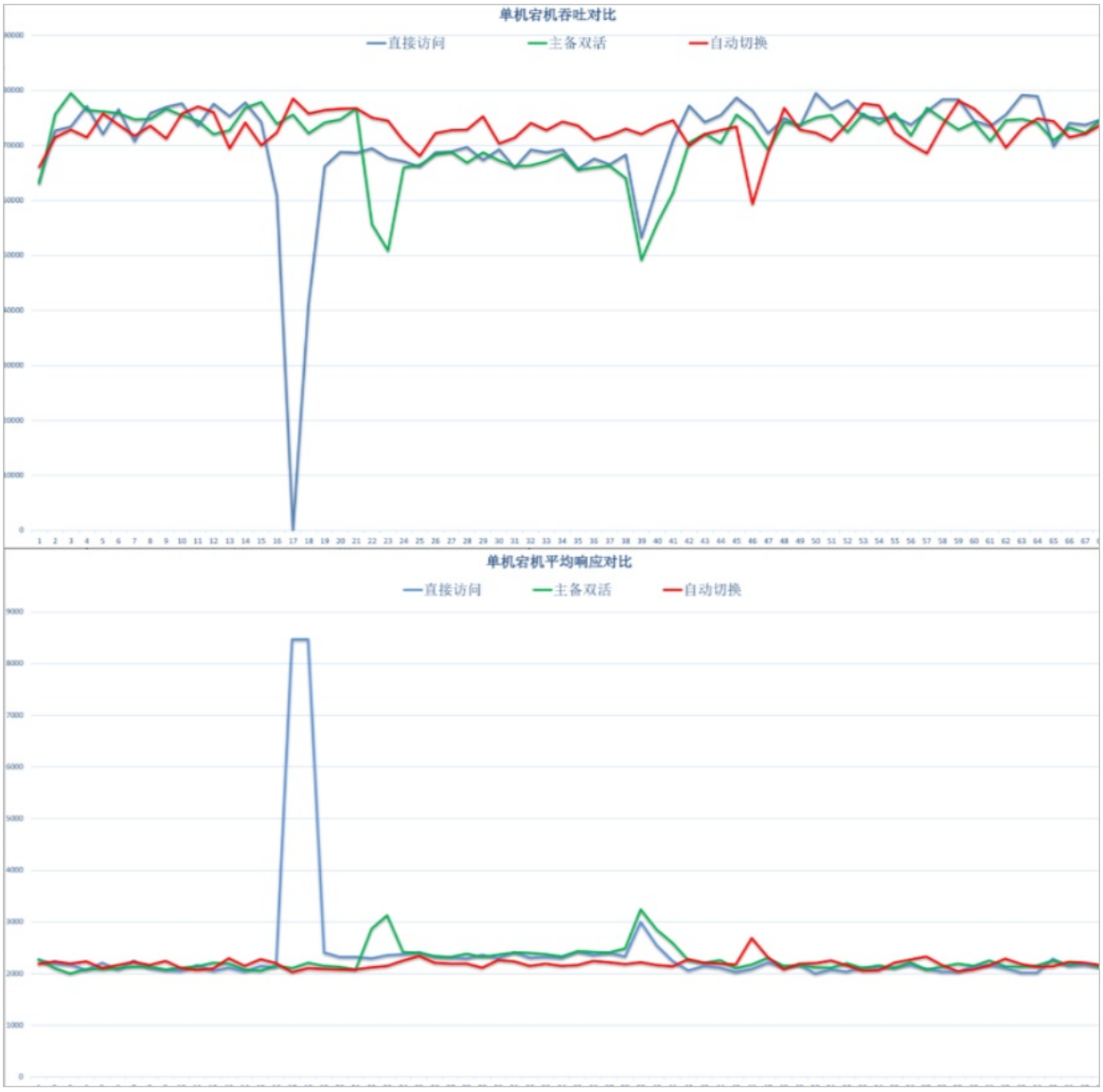
- 故障自动容错：主备双活功能提供故障自动容错能力，在故障场景下可以自动进行主备节点切换。通常需要进行主备集群切换的场景如下：
  - 机房出现断网断电等故障。无法正常连接主集群，请求全部报错。
  - 由于软件Bug导致全集群宕机。
  - 由于慢盘或者坏盘造成的集群访问超时。

  故障自动容错原理：

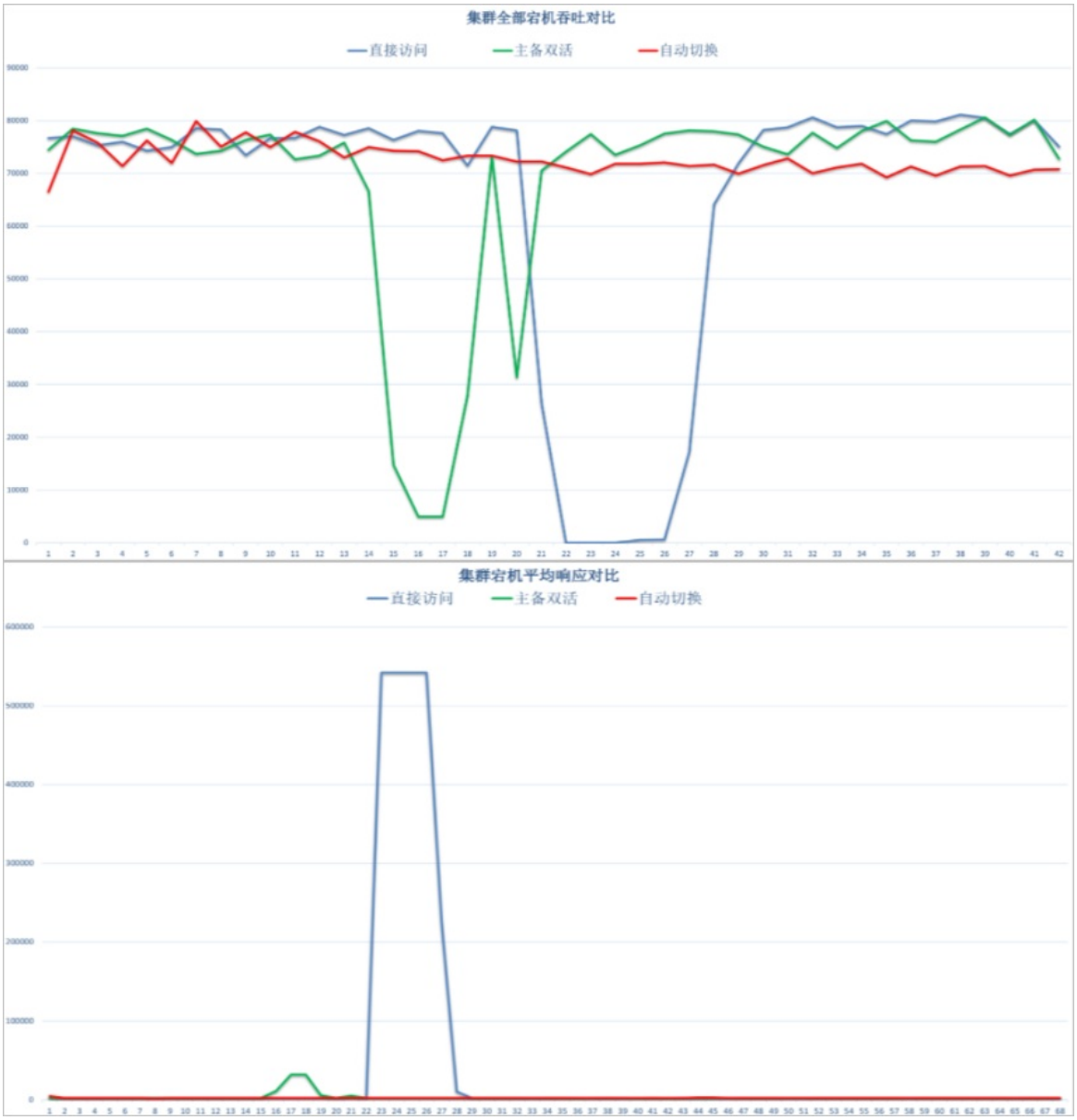


  单机宕机和集群宕机测试结果如下。

  - 以下是单机宕机吞吐对比图和单机宕机平均响应对比图。



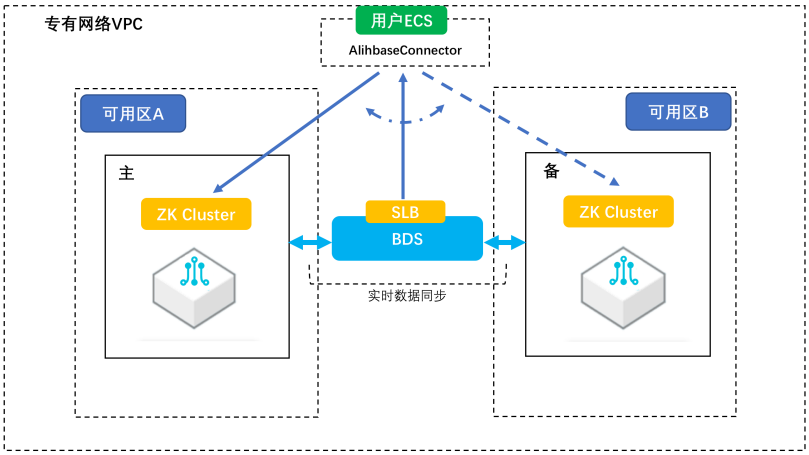
- 以下是集群宕机吞吐对比图和集群宕机平均响应对比图。



主备容灾原理介绍

当云数据库HBase实例因不可预料的原因（例如设备故障、机房断电断网等）发生故障，容灾机制可用于保障数据的一致性和业务可用性。云数据库HBase提供两种灾备方案供您选择，可满足不同的业务场景。

灾备方案	说明
单可用区高可用方案	主备节点部署在同一可用区中的不同机器上，当任一节点发生故障时，由高可用HA（High Availability）系统监控主备节点的健康状态并自动执行故障切换，避免单点故障引起的服务中断。

灾备方案	说明
同城容灾方案	主备节点支持云数据库HBase间、云数据库HBase和自建HBase、自建HBase和自建HBase。主备节点分别部署在同一地域下两个不同的可用区，当任一可用区因电力、网络等不可抗因素失去通信时，高可用HA系统将执行故障切换，确保整个节点的持续可用。您可以使用LTS数据通道服务完成主备节点之间实时增量数据双向同步，同时在LTS上进行主备切换操作，您可以通过alihbase-connector插件访问HBase节点，alihbase-connector监听主备节点zk集群的切换事件完成节点切换操作。 

常见问题

- Q：主备数据双向同步中是否会循环发送？

  A：不会，双向同步中会根据集群ID区分是数据同步写入的数据还是客户端写入的数据，保证数据不会循环发送。

- Q：目标实例出现异常无法写入时，LTS是否会缓存数据？

  A：LTS不会缓存数据，数据同步中当目标实例无法写入时，会记录读取WAL的点位，保证数据在目标实例恢复后继续同步数据，数据存储在源集群的Hlog中。

- Q：数据D1写入主实例，但数据D1尚未到达备实例时发生切换，此时备实例中写入一条数据D2后数据D1才到达，备实例中存储的是数据D1还是数据D2？

  A：存储的是数据D1和数据D2中时间戳更大的数据，数据在LTS同步过程中不会改变数据原有的时间戳，一般情况下存储的是数据D2，但是由于主备实例不同可能存在时间戳毫秒级的时间差异造成数据D2的时间戳比数据D1的时间戳小，在这种情况下存储的是数据D1（云原生多模数据库Lindorm实例多可用区版本支持强一致功能，保证存储的是数据D2，如果需要了解详情请提交工单）。

10.如何设计Rowkey

云数据库HBase的Rowkey设计在数据分区和数据查询中很重要，本节介绍设计Rowkey前需要考虑的一些问题以及设计示例。

问题考虑

- 问题一：Rowkey是唯一的吗？

  相同的Rowkey在HBase中认为是同一条数据的多个版本，查询时默认返回最新版本的数据，所以通常Rowkey都需要保证唯一，除非用到多版本特性。

  最佳设计示例：Rowkey相当于数据库的主键。Rowkey表示一条记录。Rowkey可以是一个字段也可以是多个字段接起来。Rowkey为[userid]表示每个用户只有一条记录， Rowkey为[userid][orderid]表示每个用户有多条记录。

- 问题二：满足哪种查询场景？

  Rowkey的设计限制了数据的查询方式，HBase有两种查询方式。

  - 根据完整的Rowkey查询（get方式），例如 `SELECT * FROM table WHERE Rowkey = 'abcde'` 。

    > **说明** get方式需要知道完整的Rowkey，即组成Rowkey所有字段的值都是确定的。

  - 根据Rowkey的范围查询（scan方式），例如 `SELECT * FROM table WHERE 'abc' < Rowkey <'abcx'` 。

    > **说明** scan方式需要知道Rowkey左边的值，例如您使用英文字典查询pre开头的所有单词，也可以查询prefi开头的所有单词，不能查询中间或结尾为prefi的单词。

  最佳设计示例：在有限的查询方式下如何实现复杂查询？以下方法可以帮您实现。

  - 再新建一张表作为索引表。
  - 使用Filter在服务端过滤不需要的数据。
  - 使用二级索引。
  - 使用反向scan方法实现倒序（将新数据排在前面），`scan.setReverse(true)` 。

    > **说明** 反向scan的性能比正常scan性能差，如果大部分是倒序场景可以体现在Rowkey设计上，例如 `[hostname][log-event][timestamp] => [hostname][log-event][Long.MAX_VALUE - timestamp]` 。

- 问题三：数据足够分散，会存在堆积的热点现象吗？

  散列的目的是将数据分散到不同的分区，不至于产生热点使某一台服务器终止，其他服务器空闲，充分发挥分布式和并发的优势。

  最佳设计示例：

  - 设计md5散列算法：`[userId][orderid] => [md5(userid).subStr(0,4)][userId][orderid]` 。
  - 设计反转：`[userId][orderid] => [reverse(userid)][orderid]` 。
  - 设计取模：`[timestamp][hostname][log-event] => [bucket][timestamp][hostname][log-event]; long bucket = timestamp % numBuckets` 。

- 增加随机数：`[userId][orderid] => [userId][orderid][random(100)]`。

- 问题四：Rowkey可以再短点吗？

  短的Rowkey可以减少数据量，提高数据查询和数据写入效率。

  最佳设计示例：

  - 使用Long或Int代替String，例如`'2015122410' => Long(2015122410)`。
  - 使用编码代替名称，例如`'淘宝' => tb`。

- 问题五：使用scan方式会查询出不需要的数据吗？

  会的。场景举例：table1的Rowkey为colume1+ colume2+ colume3，如果您需要查询colume1= host1的所有数据，使用`scan 'table1',{startkey=> 'host1',endkey=> 'host2'}`语句。如果有一条记录为colume1=host12，那么此记录也会查询出来。

  最佳设计示例：

  - 设计字段定长，`[colume1][colume2] => [rpad(colume1,'x',20)][colume2]`。
  - 添加分隔符，`[colume1][colume2] => [colume1][_][colume2]`。

常见设计实例

- 日志类、时间序列数据。列举出三个场景设计Rowkey。
  - 查询某台机器某个指标某段时间内的数据，Rowkey设计为`[hostname][log-event][timestamp]`。
  - 查询某台机器某个指标最新的几条数据，Rowkey设计为`timestamp = Long.MAX_VALUE – timestamp; [hostname][log-event][timestamp]`。
  - 查询的数据存在只有时间一个维度或某一个维度数据量巨大的情况，Rowkey设计为`long bucket = timestamp % numBuckets; [bucket][timestamp][hostname][log-event]`。

- 交易类数据。列举出四个场景设计Rowkey。
  - 查询某个卖家某段时间内的交易记录，Rowkey设计为`[seller id][timestmap][order number]`。
  - 查询某个买家某段时间内的交易记录，Rowkey设计为`[buyer id][timestmap][order number]`。
  - 根据订单号查询，Rowkey设计为`[order number]`。
  - 查询中同时满足三张表，一张买家维度表Rowkey设计为`[buyer id][timestmap][order number]`。一张卖家维度表Rowkey设计为`[seller id][timestmap][order number]`。一张订单索引表Rowkey设计为`[order number]`。