

# HBase 文档



#### 【版权声明】

版权所有©百度在线网络技术（北京）有限公司、北京百度网讯科技有限公司。未经本公司书面许可，任何单位和个人不得擅自摘抄、复制、传播本文档内容，否则本公司有权依法追究法律责任。

#### 【商标声明】



和其他百度系商标，均为百度在线网络技术（北京）有限公司、北京百度网讯科技有限公司的商标。本文档涉及的第三方商标，依法由相关权利人所有。未经商标权利人书面许可，不得擅自对其商标进行使用、复制、修改、传播等行为。

#### 【免责声明】

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导。如您购买本文档介绍的产品、服务，您的权利与义务将依据百度智能云产品服务合同条款予以具体约定。本文档内容不作任何明示或暗示的保证。

## 目录

目录	2
功能发布记录	6
产品公告	6
云数据库HBase公测说明与免责声明协议	6
云数据库HBase商业化发布通知	7
产品描述	7
产品介绍	7
产品优势	8
应用场景	8
实例节点规格	10
与其他云服务的关系	11
产品定价	11
计费说明	11
余额不足提醒和欠费处理	12
手动续费包年包月实例	12
开通或关闭自动续费包年包月实例	13
快速入门	15
操作指南	17
开通服务	17
创建实例	17
设置白名单	19
访问实例	21
使用HBaseShell连接	21
使用JavaAPI连接	22
连通性检测	25
访问HBase Web UI	26
管理实例	30
查看实例信息	30
重启实例	31
概述	31
操作指南	31
查看实例列表	32
实例扩容	33
数据库连接	36
释放实例	37
多用户访问控制	38
标签管理	41
解绑标签	41
实例列表	42
根据标签筛选实例	43

绑定标签	44
创建标签	47
左侧导航栏	47
企业菜单	48
修改参数	49
备份恢复	51
手动备份	51
恢复数据列表	55
查看详情	56
计费变更	56
开启OpenTSDS	58
查看参数修改历史	59
API参考	62
API简介	62
调用说明	63
实例相关接口	66
查询实例列表	66
改配实例 - 更改TSDB节点规格	68
改配实例 - 添加TSDB节点	69
改配实例 - 更改Core节点磁盘大小	70
改配实例 - 添加Core节点	71
更改实例名称	72
创建实例	73
查询实例详情	77
释放实例	81
重启实例	81
白名单相关接口	83
删除白名单	83
添加白名单	84
获取白名单列表	85
付费相关接口	86
取消预付费转后付费	86
后付费转预付费	87
预付费转后付费	88
附录	89
备份接口	92
删除实例备份	92
查看实例备份详情	93
修改备份描述信息	95
创建备份	96
修改备份策略	97

查询备份策略	99
查询备份列表	100
附录	102
查看实例恢复记录	103
查询恢复记录列表	104
恢复数据库	106
开发指南	107
数据模型	107
命名空间 (Namespace)	108
表 (Table)	108
行 (Row)	108
列 (Column)	108
列族 (Column Family)	108
列标识符 (Column Qualifier)	108
单元格 (Cell)	109
时间戳 (Timestamp)	109
Schema设计原则	111
Rowkey设计原则	112
HBase SQL服务	115
HBase SQL(Phoenix)使用说明	115
HBase SQL(Phoenix)二级索引	116
HBase SQL(Phoenix)快速开始	117
HBase时序引擎	118
OpenTSDB概述	118
连接与使用OpenTSDB	119
连接与使用OpenTSDB	119
开启OpenTSDB	119
连接OpenTSDB	119
修改OpenTSDB节点配置	120
常见问题	120
如何选择规格	120
通用问题	121
一般问题	121
典型实践	122
数据迁移	122
使用Snapshot工具进行数据迁移	122
使用CopyTable工具进行数据迁移	123
不停机迁移	124
服务等级协议SLA	125
云数据库HBase服务等级协议SLA	125



# 功能发布记录

## 新功能发布

发布时间	功能概述
2023-3	<ul style="list-style-type: none"> <li>• HBase支持通过控制台修改实例参数，您可以通过修改参数的方式对集群进行性能调优。功能说明详见<a href="#">参数修改</a></li> <li>• HBase发布OPEN API，您可通过调用API，进行快速高效的数据库管理操作。功能说明详见<a href="#">API简介</a></li> </ul>
2023-1	HBase支持时序引擎OpenTSDB，它是一个具有可伸缩性的，基于HBase的分布式时间序列数据库，其数据存储存储在HBase上。OpenTSDB的主要用途是采集与保存监控数据，并且用户可以十分简便地访问与绘制这些数据。功能说明详见 <a href="#">时序引擎</a> ，您可通过控制台直接购买启用时序引擎实例，操作说明详见 <a href="#">开发者指南-时序引擎开通与管理</a>
2022-12	<ul style="list-style-type: none"> <li>• HBase发布华东-苏州，华南-广州地域</li> <li>• HBase支持标签管理，您可以创建实例时设置标签，也可对存量的实例进行标签管理，标签相关说明详见<a href="#">标签管理</a></li> <li>• 支持多用户访问控制，您可灵活授权进行HBase访问控制，详见<a href="#">多用户访问控制</a></li> </ul>
2022-10	<ul style="list-style-type: none"> <li>• HBase于2022年10月28日正式商业化，并新增预付费计费方式，详见<a href="#">计费说明</a></li> <li>• 华北-保定地域全量开放，价格与原有地域保持一致，详见<a href="#">价格详情</a></li> </ul>
2022-09	<ul style="list-style-type: none"> <li>• 数据的存储类型新增冷存储。创建实例时，存储类型支持选择冷存储，冷数据将被存储在对象存储BOS中，具有低成本、高灵活性等优势，详情请见 <a href="#">创建实例</a></li> <li>• HBase 新增 4 项监控项，涵盖读写请求次数、Cache命中率等</li> </ul>
2022-08	HBase Web UI访问支持通过公网Web UI访问，详情请见 <a href="#">访问HBase Web UI</a>
2022-07	云数据库HBase发布公测，支持轻松创建高可用集群版全托管NoSql数据库。

## 产品公告

### 云数据库HBase公测说明与免责声明协议

公测起止时间/协议生效时间：2022年7月28日-2022年10月27日

请您在申请公测前，务必仔细阅读并充分理解本协议内容。当您勾选同意本协议，即表示您已充分阅读、理解并接受本协议的全部内容。

#### 公测说明

1. 参与公测方式：填写公测申请表，提交测试申请。
2. 公测期间若您提交申请公测并通过审核后，将默认提供一个免费集群版实例进行公测，且申请公测的实例大小默认为最小实例，实例组成为3个规格为1核2GBZK节点，2个规格为4核8GB master 节点，2个规格为4核16GB core节点，单个core节点存储为通用型SSD云磁盘400GB。若以上测试规格不能满足您的业务场景，可在申请公测表单中，详细说明测试规格与使用场景，百度智能云将进行评估，最终试用实例规格以百度智能云评估结果为准。

- 我们将为您的发放对应金额的定向支付云数据库HBase实例的代金券，您获取代金券后可购买实例进行测试，特别说明代金券仅可抵扣云数据库HBase费用，若您测试过程中使用其他产品产生费用将为其付费，例如[云主机BCC](#)。
- 公测周期为45天，起始时间以代金券成功发放到您的账户的后展示的开始时间为准。代金券到期前3天与到期当天将通知您，试用到期后，则将按[产品计费说明](#)收取费用，价格参考[云数据库HBase价格详情](#)。

#### 🔗 特别说明

#### 🔗 责任的限制和免除

- 您理解并同意：在云数据库HBase服务公测期间，百度智能云不对任何服务可用性、可靠性做出承诺，亦不对您使用公测服务的工作或结果承担任何责任；但百度智能云愿意同您一同合作解决问题，努力提升服务质量及服务水平。如果出现任何情况，您可通过官网客服等联系途径反馈给百度智能云以获得进一步技术支持，如果无法解决相关问题，您可选择立即停止使用该公测服务。
- 对于因电信系统故障、互联网网络堵塞、应用系统/或将被创建的服务本身的原因，包括但不限于服务升级、应用程序编程接口关闭、库存不足、启动时间过长以及不可抗力的原因，致使云数据库HBase服务不可用，百度智能云将予以免责。

#### 🔗 公测服务的提前终止或变更

- 经百度智能云提前通知，结束公测期的；
- 百度智能云有权基于公测情况随时调整公测服务，包括但不限于下架、停服、调整产品性能或服务内容等。
- 为技术升级、服务体系升级、或因经营策略调整或配合国家重大技术、法规政策等变化，百度智能云不承诺永久提供某种服务，并有权调整所提供服务的形式、规格或其他方面（如服务的价格和计费模式），在终止该种服务或进行相关调整前，百度智能云将提前以网站公告、站内信、邮件或短信等方式进行通知。

## 云数据库HBase商业化发布通知

#### 🔗 说明

云数据库HBase将在2022年10月27日结束公测，计划于2022年10月28日正式商业化，商业化后您需付费使用，计费策略已在官网公布，详见[计费说明](#)，详细价格说明，您可参见[产品价格](#)。

诚邀您了解并使用我们的云数据库HBase产品，点此[立即购买](#)。

#### 🔗 商业化发布时间

自2022年10月28日起，将正式商用化，具体发布时间，请以通知（站内信、邮件及短信）中的通知时间为准。

## 产品描述

### 产品介绍

云数据库HBase (HBase) 是百度智能云基于HBase深度扩展打造的高性能，高可用，易扩展，低成本的NoSQL 数据库，100%兼容开源HBase，支持PB规模、千万级并发、毫秒响应、低成本存储、弹性扩容、全托管等企业级服务能力，满足用户高吞吐、低成本以及轻运维等业务场景需求。

#### 🔗 优势

#### 🔗 高可用集群，弹性扩展

拥有丰富功能集，双master节点构造HA集群，安全、可靠、可弹性扩展，适用任何规模的企业生产。

#### 🔗 极致性价比

支持存储扩容，用户可按需在线扩容。支持多种存储类型，可为冷热数据提供最适宜的存储介质，高压缩算法。

### 🔗 运维能力专业

提供可视化监控，支持丰富的监控信息和自定义告警规则，极大简化系统运维。同时，内部打磨多年，提供专业的HBase运维服务，动态扩容业务无感知，稳定可靠，使客户免于复杂运维，专注业务发展。

### 🔗 高吞吐，低延时

存储计算分离，支持PB级数据存储与高并发写入，ms级数据查询，优化请求延时，平均读写延迟较开源HBase优越。

### 🔗 简单易用

100%兼容HBase原生接口，可使用丰富的生态工具，业务可以无缝迁移，服务全托管，控制台提供可视化表数据管理与监控告警功能，可随时了解集群动态。

### 🔗 支持地域

支持华北-北京、华北-保定、华东-苏州、华南-广州及更多地域支持请提交工单。

## 产品优势

### 🔗 高可用

- 双master节点构造HA集群，安全、可靠、可弹性扩展，适用任何规模的企业生产。

### 🔗 弹性扩展

- 云原生与计算存储分离架构，计算和存储资源独立，用户无感扩容，可支持PB级存储空间。

### 🔗 极致性价比

- 支持存储扩容，用户可按需在线扩容。
- 支持多种存储类型，可为冷热数据提供最适宜的存储介质。
- 支持更优秀压缩算法。

### 🔗 运维能力专业

- 提供可视化监控，支持丰富的监控信息和自定义告警规则，极大简化系统运维。
- 内部打磨多年，提供专业的HBase运维服务，动态扩容业务无感知，稳定可靠，使客户免于复杂运维，专注业务发展。

### 🔗 高吞吐，低延时

- 存储计算分离架构，支持PB级数据存储与高并发写入。
- ms级数据查询，优化请求延时，平均读写延迟较开源HBase优越。

### 🔗 简单易用

- 100%兼容HBase原生接口，可使用丰富的生态工具，业务可以无缝迁移。
- 服务全托管。
- 控制台提供可视化表数据管理与监控告警功能，可随时了解集群动态。

## 应用场景

### 🔗 画像数据存储与查询

用户画像数据被广泛应用于市场决策、推荐以及广告系统中，画像的集合是不确定的，且数据更新频繁，数据需实时存储在HBase中，根据用户画像可实现实时且精准的推荐业务。

### 优势

- 低时延

深度优化延时，毛刺少，性能稳定。

- 稀疏矩阵，结构灵活

HBase的稀疏矩阵模型，天然适合非结构化数据的存储，数据表无需预先定义schema，行与行之间不需要严格的列定义，适合灵活多变的业务。

- 高扩展性

服务能力可线性可扩展，无需分库分表。

## 🔗 物联网监控数据存储和查询

水务、电力、化工、燃气、互联网等IoT设备广泛接入云端，众多的设备数据和分析结果需实时高效写入到HBase中，通过HBase将时序结果输出到用户的前端监控系统进行展现，实现物联网设备实时监控分析系统，适合物联网、金融K线、监控等多个应用场景。

### 优势

- 高性能读写

存储计算分离，支持PB级数据存储与高并发写入，ms级数据查询。

- 低成本

存储按使用量计费，可弹性扩容，从容应对业务的不确定性，多种存储类型，冷热数据分别存储降低存储成本，支持高压压缩比，成本更低。

- 海量存储

存储容量可无限扩展，存储数据安全可靠，无感弹性扩容。

## 🔗 车联网位置大数据应用

在车联网中，通常会包含几类数据：车辆、驾驶员等基本信息，车况、电池、电机等监控数据车辆路线，可实现车辆，物流轨迹监控与分析。

### 优势

- 高性能读写

支持PB级数据存储与高性能并发写入，分布式架构，存储计算分离。

- 低成本存储

支持多种存储类型，适合冷热数据，有效降低成本。支持高压压缩比算法。

- 海量存储

支持离线、在线海量KeyValue数据存储，存储容量可扩展。

## 🔗 消息日志类数据存储分析

消息数据、报表数据、风控类数据、日志数据、订单数据等结构化、半结构化的KeyValue数据均可以存储和查询。支持企业低成本、大容量存储和查询各类日志、消息、交易、用户行为、画像等结构化/半结构化数据，支持高吞吐量实时入库及数据实时查询，实现数据资源高效智慧化运营。

### 优势

- **海量存储**

支持离线、在线海量KeyValue数据存储，支持GB-PB级存储，存储容量可弹性无限扩展。

- **生态丰富**

Hadoop多项生态组件与百度云智能云产品完成高度整合，轻松构建整体应用。

- **高性能读写**

支持亿级写入吞吐量、ms级查询。

#### 海量数据存储分析

支持海量数据快速批量导入以及实时增量数据快速写入并提供低延时查询，通过计算引擎轻松完成海量数据离线分析。

#### 优势

- **海量存储**

支持离线、在线海量KeyValue数据存储，存储容量可轻松扩展。

- **无缝对接分析能力**

与百度云与大数据计算生态无缝对接实现优秀的分析能力。

- **低成本**

存储按使用量计费、弹性扩容并提供丰富存储介质，选择合适存储类型，轻松实现成本更低。

## 实例节点规格

#### 云数据库HBase各节点规格

HBase高可用集群版由ZK，Master和Core组成，同时提供了对时序引擎的支持。各节点规格具体说明如下：

zk节点规格	master节点规格	core节点规格	时序引擎节点规格
1核2G	4核8G	8核16G	4核16G
-	8核16G	16核32G	8核32G
-	16核32G	32核64G	16核64G
-	32核64G	4核16G	32核128G
-	4核16G	8核32G	-
-	8核32G	16核64G	-
-	16核64G	32核128G	-

#### core节点存储类型

core节点存储容量支持按需扩展，根据您的业务不同，支持3种云磁盘存储类型，具体说明如下：

磁盘类型	细分
云磁盘	通用型SSD
	增强型SSD
	高性能云磁盘
	通用型HDD
	本地磁盘SDD

## 与其他云服务的关系

云服务器BCC：您需要通过云主机连接HBase服务，该云主机需与HBase服务必须在同一地域，并且IP加入到HBase访问白名单中，参见[云服务器](#)。

云监控BCM：云监控平台提供全方位的监控与告警服务，可以向用户展示购买的云数据库HBase集群使用状况及系统健康状况等，便于您随时了解集情况，参见[云监控](#)。

## 产品定价

### 计费说明

#### 概述

HBase以实例为单位进行计价，计费项为包含节点规格与存储容量，计费方式上目前支持预付费和后付费模式，详细价格参考[云数据库HBase价格详情](#)。

#### 计费项

集群	计费项
HA集群版	<ul style="list-style-type: none"> <li>zk节点规格费用</li> <li>master节点规格费用</li> <li>core节点规格费用</li> <li>core节点存储空间费用</li> </ul>

#### 计算公式

总费用 = zk节点规格费用 + master节点规格费用 + core节点规格与存储空间费用。

说明：公网流量优惠活动中，流出流量全部免费，收费时间待定。

#### 计费方式

云数据库HBase 支持预付费（包年包月计费）和后付费（按量计费）两种计费方式，详细的计费规则说明如下：

计费方式	预付费-包年包月计费	后付费-按量计费
说明	<ul style="list-style-type: none"> <li>- 即包年、包月付费，在创建实例时需要支付费用</li> <li>- 适合业务稳定的长期需求，费用比后付费更低，且购买时间越长，折扣越多。</li> </ul>	<ul style="list-style-type: none"> <li>- 按分钟计费，不足1分钟按1分钟计</li> <li>- 根据实例配在北京时间整点扣费并生成账单。出账单时间是当前计费周期结束后1小时内，具体以系统出账时间为准</li> <li>- 适合业务瞬间有较大变化的场景，资源使用完即可立即释放，降低成本</li> </ul>
注意事项	<ul style="list-style-type: none"> <li>- 预付费实例支持配置升级（增加节点/磁盘扩容）</li> </ul>	<ul style="list-style-type: none"> <li>- 后付费实例可随时被释放</li> <li>- 后付费实例支持配置升级（增加节点/磁盘扩容）</li> <li>- 购买前需保证账户无欠款，且账户余额和可用代金券总和≥100元</li> </ul>

预付费和后付费模式的欠费处理方式详见[余额不足提醒和欠费处理](#)。

#### 🔗 计费案例

如：创建集群为，zk节点规格为1c2g，数量3个，master节点规格为4c8g，数量为2，core节点规格为8c16g，且配置100GB通用型SSD云磁盘，core节点数量为2个。采用后付费计费模式。

1.节点用户创建集群时选择集群节点规格及数量， $3 \times 0.003368056\text{元/分钟} + 2 \times 0.0134375\text{元/分钟} + 2 \times 0.024097222\text{元/分钟} = 0.085173611\text{元/分钟}$ 。

2.磁盘用户创建集群时选择集群节点配套数据盘类型及容量大小，如 1000GB通用型SSD 则磁盘费用为： $1000\text{GB} \times 0.0000208\text{元/分钟} = 0.0208\text{元/分钟}$ 。

共计： $0.085173611 + 0.0208 = 0.105973611\text{元/分钟}$ 。

#### 🔗 账单查看

您可以登录百度智能云控制台，进入财务页面查看[消费明细](#)。

## 余额不足提醒和欠费处理

#### 🔗 预付费模式处理方式

- 在实例到期前7天，系统将给您发送“即将到期”通知。
- 实例到期后停服，数据为您保留7天。停服期间不收取费用，系统将给您发送续费通知。
- 若实例到期时间超过7天，实例将被释放，数据无法恢复。
- 在释放实例前1天和释放实例时，系统将给您发送释放实例通知。

#### 🔗 后付费模式处理方式

#### 🔗 余额不足提醒

- 根据用户最近3天的账单金额来判断账户余额（含可用代金券）是否足够支付未来3天的费用，若不足以支付，系统发送提醒。
- 根据最近1天的账单金额来判断您的账户余额（含可用代金券）是否足以支付未来1天的费用，若不足以支付，系统发送续费提醒。

#### 🔗 欠费处理

- 欠费后云数据库HBase立即停服，停服后其中数据将保留7天，期间不收取存储费用。
- 7天内若未充值则数据彻底删除，无法通过任何方式找回。
- 系统会在停服时、数据删除前3天、前1天、资源删除时给您以短信和邮件的方式发送提醒消息。

#### 🔗 操作建议

- 按量付费实例根据实际使用时间计费，无需续费，请保障账户资金充足，若收到欠费提醒请及时充值。

## 手动续费包年包月实例

预付费（包年包月）实例有到期时间，如果到期未续费，会导致业务中断甚至数据丢失，建议您及时手动续费。

#### 🔗 背景信息

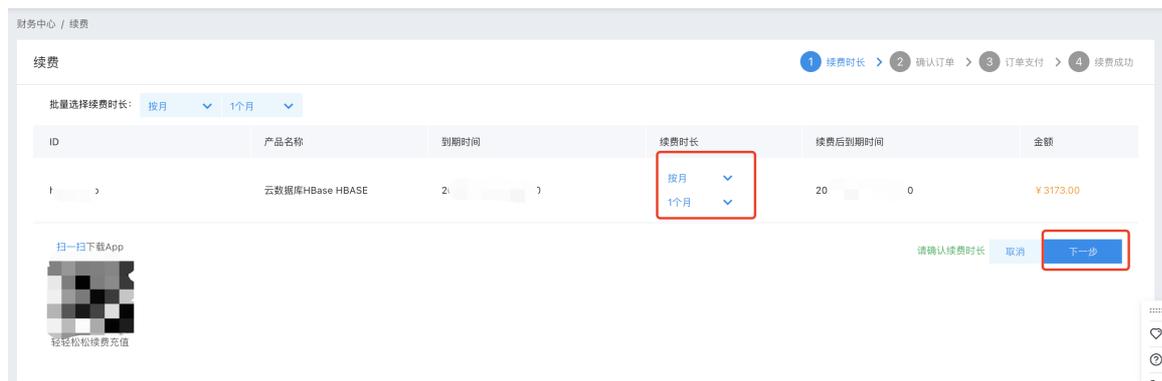
预付费实例在到期后，请在7天内进行续费，否则实例会被释放，其数据也会被删除且不可恢复。

#### 🔗 操作步骤

1. 登录HBase管理控制台。
2. 在页面右上角单击**财务**，然后单击**续费管理**，进入续费管理界面，选择目标实例进行**续费**，如下图所示。



3. 进入续费页面，如下图所示，选择按月或按年，选择时长，单击**下一步**按钮，进入支付页面。



4. 单击去支付按钮，完成手动续费。

## 开通或关闭自动续费包年包月实例

开通自动续费预付费（包年包月）实例可以免去您定期手动续费实例的麻烦，且不会因忘记续费而导致业务中断。

### 背景信息

目前百度智能云仅支持在购买预付费实例时选择自动续费，同时也支持对已有预付费实例开通自动续费。设置成功后，系统将自动以选择的续费时长进行续费。例如，您选择了3个月的续费时长，则每次自动续费时会缴纳3个月的费用。

#### 说明：

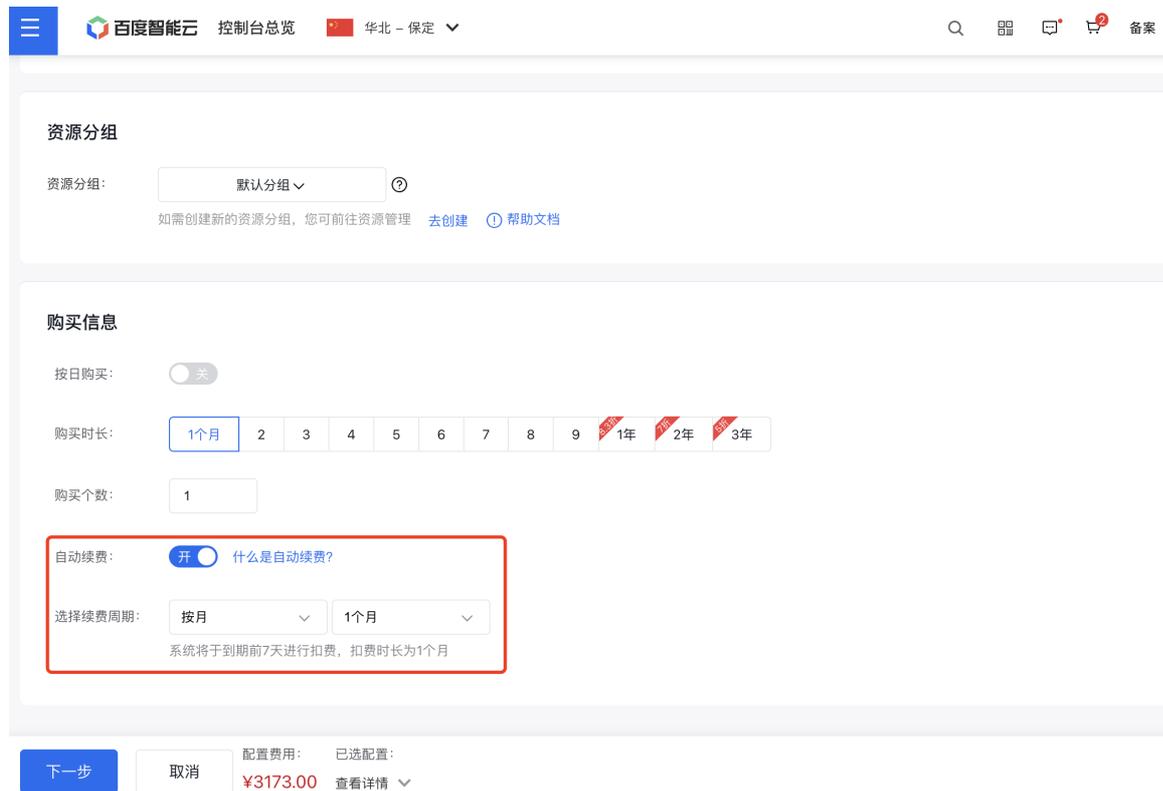
- 在购买实例选择购买时长时，您可以勾选自动续费。
- 仅支持预付费进行续费，后付费自动按量扣费。
- 包月实例：自动续费周期支持1~9个月。
- 包年实例：自动续费周期为1~3年。

### 操作步骤

#### 新购实例开通自动续费操作

1. 登录HBase管理控制台，在页面左上角，选择实例所在的地域。
2. 左侧菜单栏选择**实例管理**，在实例列表上方点击**创建实例**。
3. 付费方式选择**预付费**。

4. 在**购买信息**中的自动续费，单击**开关按钮**，选择续费周期（按月或按年）以及时长。



5. 单击**下一步按钮**，购买实例。

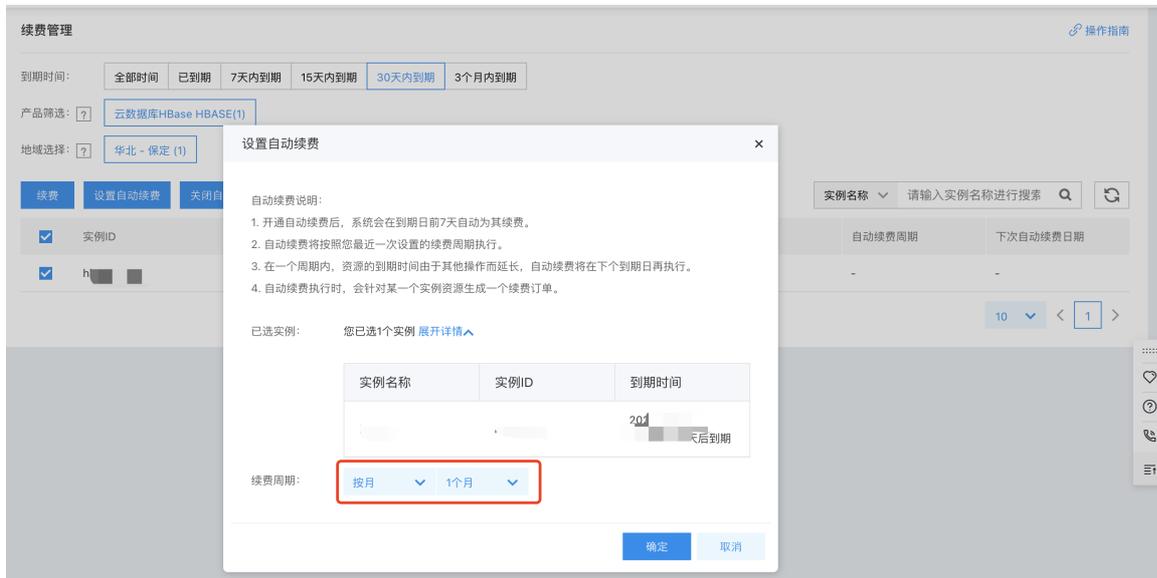
## 存量实例开通自动续费操作

1. 登录HBase管理控制台。

2. 在页面右上角单击**财务**，然后单击**续费管理**，进入续费管理界面，然后选择一个或多个目标实例进行，点击**设置自动续费按钮**，如下图所示。



3. 单击**设置自动续费**，在弹出的设置自动续费对话框里，选择**续费周期**，单击**确定按钮**，就完成了设置自动续费，如下图所示。



## 🔗 关闭自动续费操作

1. 登录HBase管理控制台。
2. 在页面右上角单击**财务**，然后单击**续费管理**，进入续费管理界面，然后选择一个或多个目标实例，单击**关闭自动续费**按钮，如下图所示。



3. 单击关闭自动续费，在弹出的取消自动续费对话框里，单击**确定**按钮，就完成了关闭自动续费，如下图所示。



# 快速入门

## 🔗 概览

本文介绍如何使用控制台快速创建一个HBase实例，学习掌握访问与管理实例。

基本操作流程如下图所示：



1. **注册及实名认证**：注册百度智能云账号，并完成实名认证。
2. **开通云数据库HBase**：控制台开通云数据库HBase服务。
3. **购买实例**：购买和配置实例。
4. **设置白名单**：提升数据库访问的安全性，将允许访问云数据库的IP地址设置为白名单，防止其他外部地址的访问。
5. **访问实例**：用户通过同地域云主机，并使用社区开源版客户端访问。
6. **释放实例**：当您不再需要HBase实例，可以将实例释放，释放后实例资源和数据将被立即清除，并停止计费。

## 🔗 注册及实名认证

使用百度智能云云数据库HBase前，用户需要拥有一个百度智能云账号并完成实名认证，具体操作如下：

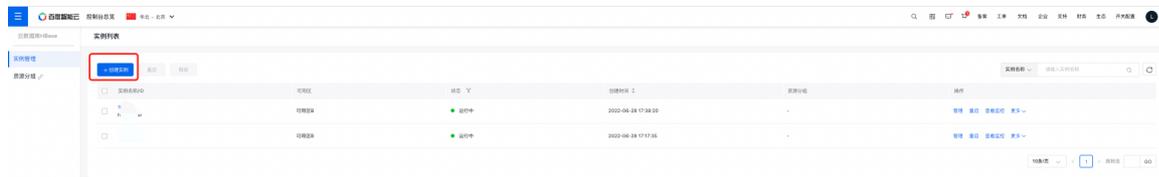
1. 注册并登录百度智能云平台，请参考 [注册](#) 和 [登录](#)。
2. 完成实名认证，操作细节请参考 [实名认证](#)，实名认证后方可购买云数据库HBase。

## 🔗 开通云数据库HBase服务

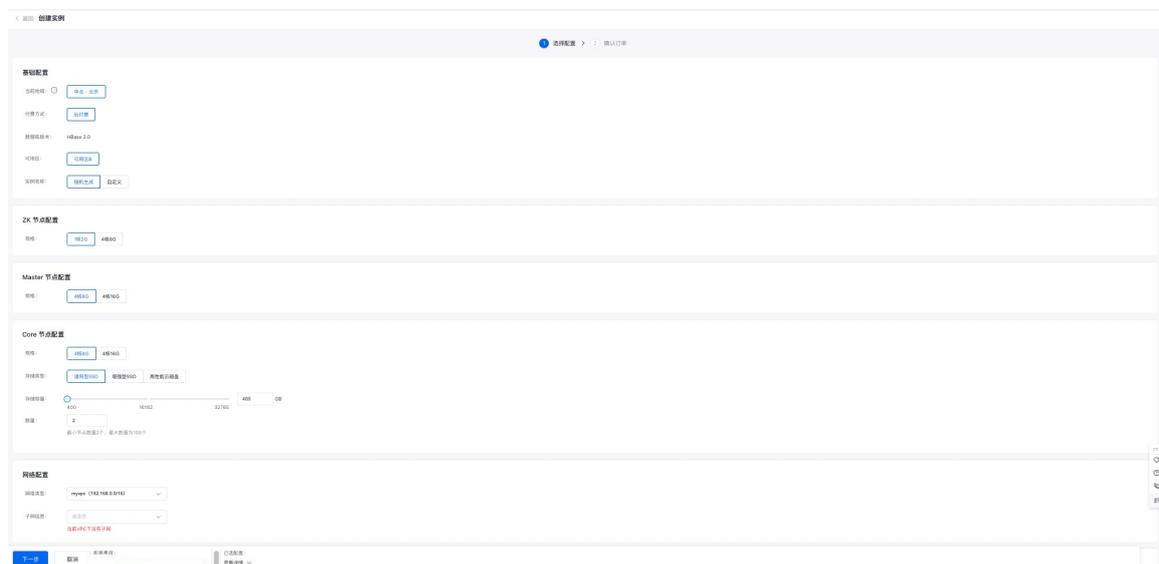
登录 [云数据库HBase管理控制台](#) 成功后，新用户需开通服务后即可开始使用，若您已经开通则可直接购买HBase实例。

## 🔗 购买实例

1. 您可以登录 [云数据库HBase控制台](#) 进行购买。
2. 在云数据库HBase控制台 [集群列表](#) 页面，单击【创建实例】按钮。



3. 按照界面提示设置基础配置数据、Zk节点配置、Master节点配置、Core节点配置、网络配置等参数。



4. 参数设置完成后，单击【下一步】按钮，根据提示完成支付流程。

- 购买参数说明请参考 [操作指南-创建实例](#)

- 规格选型建议，请参考[如何选择规格](#)。

## 🔗 设置访问白名单

具体操作请参考[设置白名单](#)。

## 🔗 访问实例

## 🔗 访问准备

由于云数据库HBase目前不支持公网访问，建议您通过购买百度智能云云主机（BCC）实现HBase实例的访问。

1. 购买与HBase集群同一地域下的云主机，详细步骤请参考[创建云主机](#)。
2. 完成白名单配置，请参考[设置白名单](#)。
3. 准备运行HBase客户端环境，您需要先安装Java编译运行环境，云数据库HBase依赖JDK 1.8及以上版本。

## 🔗 访问实例

- 下载和安装HBase客户端，并修改客户端目录下conf/hbase-site配置文件，即可实现访问HBase。
- 详细步骤请参考 [使用HBaseShell连接](#) 或[使用Java API连接](#)。

## 🔗 释放实例

当您不再需要HBase实例，可以在购买后将实例删除，删除操作后，实例资源和数据将被立即清除，并停止计费，资源与数据不可找回，请谨慎操作，详细步骤可以参考[删除实例](#)。

# 操作指南

## 开通服务

用户第一次使用百度智能云数据库HBase产品时，首先需要开通服务。

1. 首次开通，在实例管理页面会出现开通页面。
2. 在开通页面，勾选【已阅读并同意接受《百度智能云服务协议》】后，点击【立即开通】按钮，即可使用云数据库HBase。

The screenshot shows the Baidu Smart Cloud console interface. At the top, there's a navigation bar with '百度智能云' (Baidu Smart Cloud) logo, '控制台总览' (Control Panel Overview), and location '华北 - 北京'. Below the navigation bar, the '实例管理' (Instance Management) section is active. A notification banner at the top of the instance management page states: '您的账号未通过实名认证，将无法完成开通，建议尽快进行实名认证。' (Your account has not passed real-name authentication, you cannot complete the opening, it is recommended to complete real-name authentication as soon as possible.) Below this, there's a large blue banner for '云数据库 HBase' (Cloud Database HBase) with the tagline '高性能，高可用，易扩展，低成本的 NoSQL 数据库' (High performance, high availability, easy to expand, low cost NoSQL database). The banner includes a description of HBase and a '立即开通' (Open Now) button. Below the banner, there's a '流程引导' (Process Guide) section with four steps: 1. 创建云数据库 HBase 实例 (Create Cloud Database HBase Instance), 2. 数据向云上迁移 (Migrate data to the cloud), 3. 向实例写入新数据 (Write new data to the instance), and 4. 管理监控 (Management and monitoring).

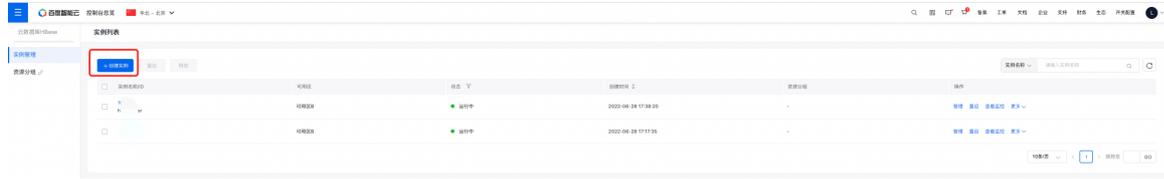
## 创建实例

本文档主要介绍如何购买HBase实例。

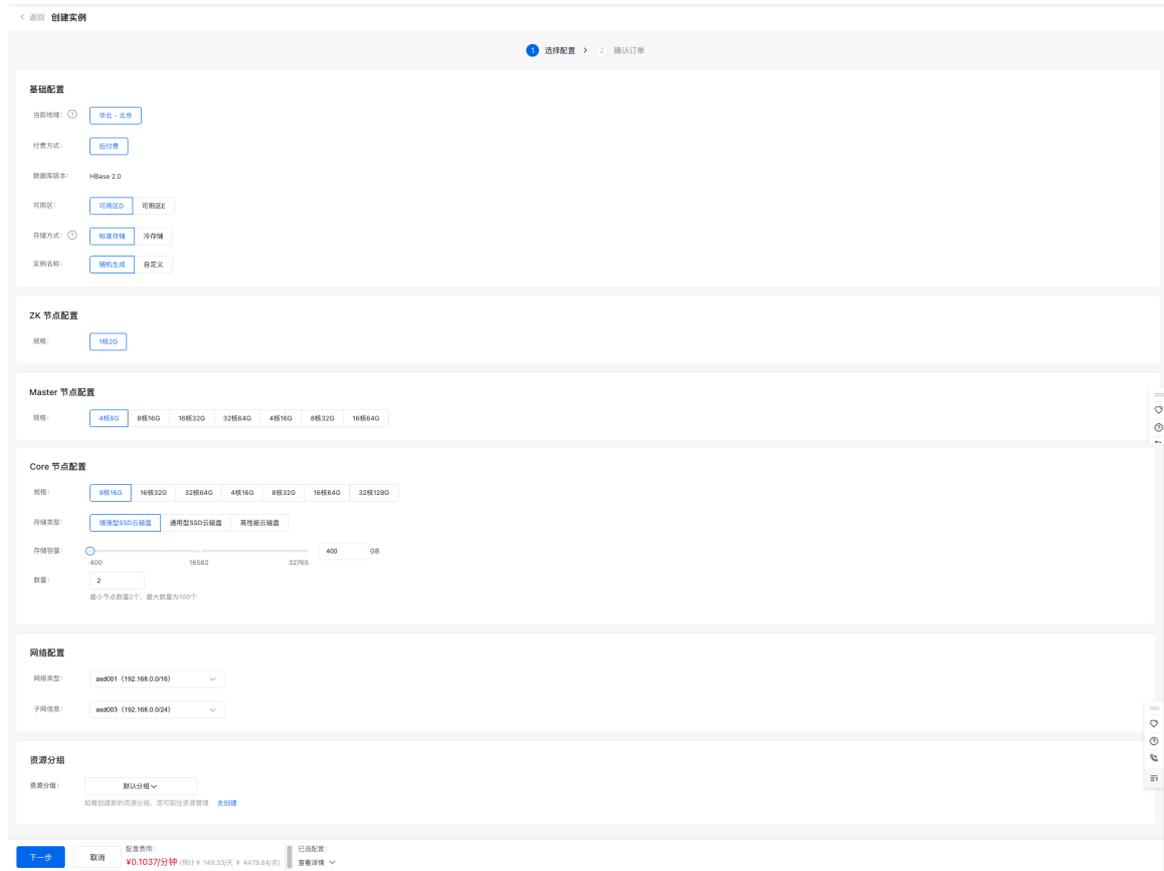
前提：您已经完成[使用前准备](#)。

## 购买集群

- 1.您可以登录[云数据库HBase控制台](#)进行购买。
- 2.在云数据库HBase控制台[集群列表](#)页面，单击【创建实例】按钮。



3.按照界面提示设置基础配置、Zk节点配置、Master节点配置、Core节点配置等购买参数。



类别	参数	说明
基础配置	地域	集群所在的地理位置。购买后无法更换地域，请确保两个问题： 1.根据目标用户所在的地理位置就近选择地域，提升用户访问速度 2.HBase集群与需要连接的BCC创建于同一个地域，否则它们无法通过内网互通，无法发挥最佳性能
	付费方式	后付费：按量付费，根据用户选购的配置，系统按用户的实际使用计费，即开即停
	可用区	可用区是指在同一地域内，电力和网络互相独立的物理区域。在同一专有网络内可用区与可用区之间私网互通，可用区之间能做到故障隔离

	存储方式	提供2种存储类型供用户选择：  - 标准存储：数据将被存储至云磁盘 HDFS  - 冷存储：数据将被存储至对象存储 BOS ，可在降低存储成本的同时获取更高的存储灵活性与数据可靠性。BOS 中默认的数据存储类型为标准存储，BOS 存储类型请参见 <a href="#">BOS分级存储</a> ，BOS 产品介绍请参见 <a href="#">BOS产品介绍</a>
	存储路径	存储方式为冷存储时，需指定同账号下同地域的BOS Bucket名称和目录。由于归档数据不便于取回，目前仅支持指定存储类型为非归档存储的Bucket
ZK节点配置	ZK节点规格	ZK规格选择参考 <a href="#">实例节点规格</a>
Master节点配置	Master节点规格	Master默认配置为高可用，数量为2个，Master规格选择请参考 <a href="#">实例节点规格</a>
Core节点配置	Core节点规格	Core规格选择参考 <a href="#">实例节点规格</a>
	存储（磁盘）类型	提供3种磁盘类型供用户选择： - 增强型SSD云磁盘：数据经常被访问或者高I/O负载的应用场景，适合中大型核心业务数据库，NoSQL数据库 - 通用型SSD云磁盘：I/O密集型应用，中小型关系数据库，如大型 OLTP 业务 - 高性能云磁盘：IO较高，且具有较高性价比适合开发与测试业务
	单节点存储容量	Core单节点容量选择范围为400GB~32765GB，请按需选择
	数量	默认为2节点，开通后可再次添加节点，数量范围2~100
网络配置	网络类型	实例所属的VPC
	子网信息	实例所属的子网信息
资源分组	资源分组	通过设置资源分组，可以对单个云账号下多个地域、多种云资源进行集中的分组管理。用户也可以选择加入已创建的资源组或者创建一个资源组加入

点击【下一步】按钮，根据提示完成支付流程。

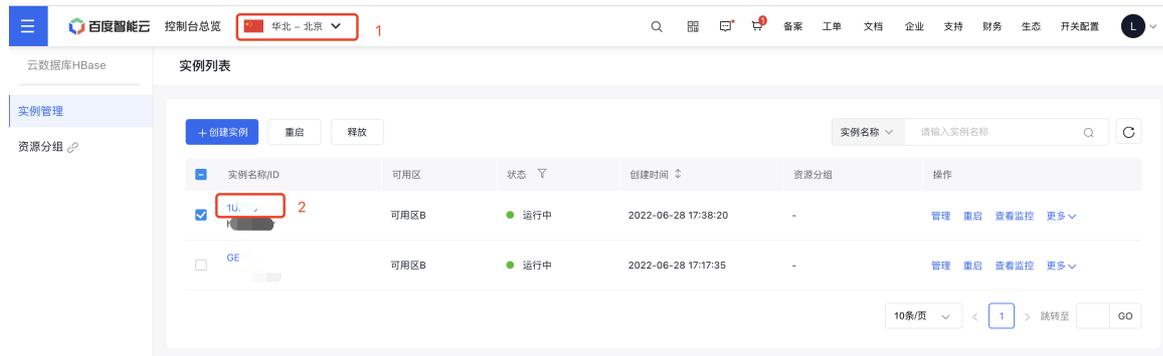
- 购买参数说明请参考[操作指南-创建实例](#)。
- 规格选型建议，请参考[如何选择规格](#)。

## 设置白名单

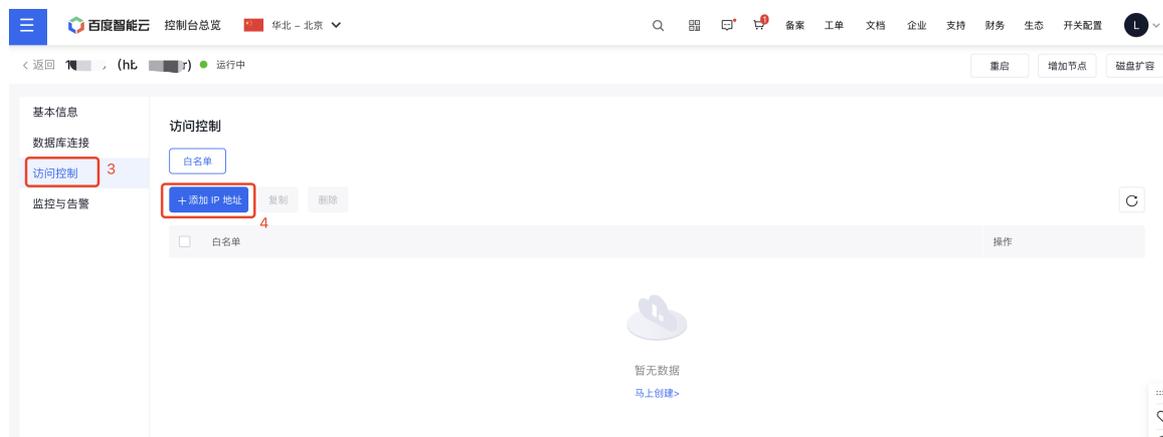
创建HBase集群后，您需要设置HBase集群的白名单分组或者添加ECS安全组，以允许外部设备访问该集群。

特别提示：为了数据库的安全，新创建的集群被默认设置为无法访问。

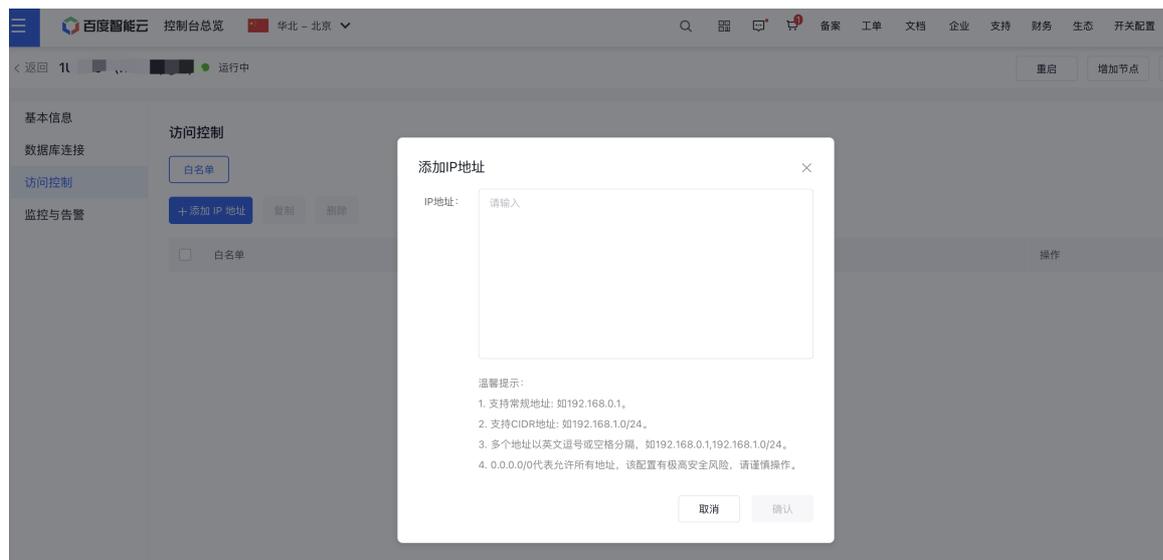
1. 登录 [云数据库HBase控制台](#)。
2. 在左上方选择集群所在地域。
3. 在实例列表中找到目标实例，单击【实例名称】。



4. 单击左侧导航栏中的访问控制。
5. 在白名单设置页签下单击【添加IP地址】按钮。



6. 在修改分组白名单对话框中，填写需要访问该集群的IP地址或IP段，然后单击【确定】按钮。



## 说明

- 支持常规地址：如192.168.0.1。
- 支持CIDR地址：如192.168.1.0/24。
- 多个地址以英文逗号或空格分隔，如192.168.0.1,192.168.1.0/24。
- 0.0.0.0/0代表允许所有地址，该配置有极高安全风险，请谨慎操作。

# 访问实例

## 使用HBaseShell连接

本文介绍如何通过HBase Shell工具访问云数据库HBase。

### 前提条件

通过内网连接时，请确认已购买与HBase实例在相同专有网络VPC的BCC服务器，并将BCC服务器的IP地址在HBase中添加为白名单。

- 若未购买BCC服务器，具体操作请参见[创建BCC实例](#)。
- 若未配置白名单，具体操作请参见[设置白名单](#)。

### 操作步骤

1. 下载[HBase Shell工具包](#)。
2. 添加集群ZK链接地址，具体操作请参考【[数据库连接](#)】。

**注意** 若您未添加有效访问IP，地址默认隐藏，您需要先配置访问白名单。



添加白名单后获取ZK链接地址：在数据库连接信息中，复制该实例的ZK链接地址。



3. 解压缩下载的HBase Shell工具包，打开conf/hbase-site.xml文件并修改配置如下参数，其中\$ZK\_IP1,\$ZK\_IP2,\$ZK\_IP3，需使用实际实例的zk地址替换。

```
<configuration>
  <property>
    <name>zookeeper.znode.parent</name>
    <value>/hbase</value>
  </property>
  <property>
    <name>hbase.zookeeper.quorum</name>
    <value>$ZK_IP1,$ZK_IP2,$ZK_IP3</value>
  </property>
</configuration>
```

4.访问集群：通过命令bin/hbase shell，可访问集群。

**注意** 云数据库HBase的默认超级账户为root与hbase，您可以使用任意一个账户连接云数据库HBase并获得超级用户权限。

## 使用JavaAPI连接

HBase Java API是HBase最常规和最高效的访问方式，Java客户端其实是shell客户端的一种实现，操作命令基本上也是shell客户端命令的一个映射，适合Hadoop MapReduce Job并行批处理 HBase表数据等场景。

建立完连接后，即可使用Java API访问HBase集群，下面提供一些简单的Java 示例。

### 🔗 表操作常见方法

#### 🔗 获取Table

```
import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.Connection;
import org.apache.hadoop.hbase.client.Table;
public Table getTable(String myTableName) {
    TableName tableName = TableName.valueOf(myTableName);
    Table table = conn.getTable(tableName);
    return table;
}
```

#### 🔗 列举Table

```
import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.Connection;
import org.apache.hadoop.hbase.client.ConnectionFactory;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.security.UserGroupInformation;
public class Test {
    public static void main(String[] args) {
        try {
            TableName[] names = conn.getAdmin().listTableNames();
            for (TableName name : names) {
                System.out.println(name.getNameAsString());
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

#### 🔗 Scan表数据表数据

```

import org.apache.hadoop.hbase.client.Result;
import org.apache.hadoop.hbase.client.ResultScanner;
import org.apache.hadoop.hbase.client.Scan;
import org.apache.hadoop.hbase.client.Table;
public void scanTable(Table table, byte[] startRow, byte[] stopRow) throws IOException {
    Scan scan = new Scan();
    scan.withStartRow(startRow);
    scan.withStartRow(stopRow);
    ResultScanner rs = table.getScanner(scan);
    for (Result r = rs.next(); r != null; r = rs.next()) {
        for (Cell kv : r.rawCells()) {
            //scan cell
        }
    }
    rs.close();
}
//Get表数据
public void get(Table table,byte[] rowkey) throws IOException {
    final byte[] CF = Bytes.toBytes("f");
    final byte[] QUALIFIER = Bytes.toBytes("c1");
    Get get = new Get(rowkey);
    Result result = table.get(get);
    byte[] cValue = result.getValue(CF,QUALIFIER);
}

```

#### 🔗 Get Table

```

public void get(Table table,byte[] rowkey) throws IOException {
    final byte[] CF = Bytes.toBytes("f");
    final byte[] QUALIFIER = Bytes.toBytes("c1");
    Get get = new Get(rowkey);
    Result result = table.get(get);
    byte[] cValue = result.getValue(CF,QUALIFIER);
}

```

#### 🔗 创建Table (基于HBase2.2版本)

```

import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.Admin;
import org.apache.hadoop.hbase.client.ColumnFamilyDescriptor;
import org.apache.hadoop.hbase.client.ColumnFamilyDescriptorBuilder;
import org.apache.hadoop.hbase.client.Connection;
import org.apache.hadoop.hbase.client.TableDescriptorBuilder;
import org.apache.hadoop.hbase.util.Bytes;
public void createTable(String myTableName, String myColumnFamily) throws IOException {
    Admin admin = getConnection().getAdmin();
    TableName tableName = TableName.valueOf(myTableName);
    TableDescriptorBuilder htd = TableDescriptorBuilder.newBuilder(tableName);
    byte[] cf = Bytes.toBytes(myColumnFamily);
    ColumnFamilyDescriptorBuilder hcd = ColumnFamilyDescriptorBuilder.newBuilder(cf);
    ColumnFamilyDescriptor columnFamilyDescriptor = hcd.build();
    htd.setColumnFamily(columnFamilyDescriptor);
    admin.createTable(htd.build());
}

```

#### 🔗 删除Table

```
import org.apache.hadoop.hbase.TableName;
import org.apache.hadoop.hbase.client.Admin;
public void deleteTable(String myTableName) {
    Admin admin = getAdmin();
    TableName tableName = TableName.valueOf(myTableName);
    if (!admin.isTableDisabled(tableName)) {
        admin.disableTable(tableName);
    }
    admin.deleteTable(tableName);
    admin.close();
}
```

## 🔗 行操作常用方法

### 🔗 获取Row

```
import org.apache.hadoop.hbase.client.Get;
import org.apache.hadoop.hbase.client.Result;
import org.apache.hadoop.hbase.client.Table;
public void getRow(Table table, byte[] rowKey) {
    Get get = new Get(rowKey);
    Result result = table.get(get);
    // do something
}
```

### 🔗 插入Row

```
import org.apache.hadoop.hbase.client.Put;
import org.apache.hadoop.hbase.client.Table;
public void putRow(Table table, byte[] rowKey, byte[] family, byte[] qualifier, byte[] value) {
    Put put = new Put(rowKey);
    put.addColumn(family, qualifier, value);
    table.put(put);
}
//插入多条
public void put(Table table, List<Put> puts) {
    table.put(puts);
}
```

### 🔗 删除Row

```
import org.apache.hadoop.hbase.client.Delete;
import org.apache.hadoop.hbase.client.Table;
public void deleteRow(Table table, byte[] rowKey) {
    Delete delete = new Delete(rowKey);
    table.delete(delete);
}

Admin admin = conn.getAdmin();
TableName tableName = TableName.valueOf(tableNameStr);
HTableDescriptor htd = new HTableDescriptor(tableName);
byte[] cf = Bytes.toBytes(columnFamily);
HColumnDescriptor hcd = new HColumnDescriptor(cf);
htd.addFamily(hcd);
RegionSplitter.HexStringSplit hexStringSplit = new RegionSplitter.HexStringSplit();
hexStringSplit.setFirstRow(Bytes.toBytes("00000000"));
hexStringSplit.setLastRow(Bytes.toBytes("ffffff"));
byte[][] splitKeys = hexStringSplit.split(100);
admin.createTable(htd,splitKeys);
admin.close();
```

## 连通性检测

### 🔗 连通性检测

本文介绍如何对云数据库HBase产品进行连通性检测。

### 🔗 背景信息

为了保障安全性，云数据库HBase不提供Ping的ICMP协议，请不要使用Ping命令测试实例连通性。Telnet命令是一个替代选项，您可以使用Telnet命令判断实例的连通性。请注意，如果您需要通过公网访问您的实例，请确保实例的公网访问选项已开通。

### 🔗 检测步骤

1. 随意选择一个zookeeper的域名地址。
2. 使用telnet命令，访问该地址的2181端口。
3. 如果您与您的实例网络互通，您会看到如下结果：

```
[user_name@your_computer ~]# telnet zk-xxxxx-1.hbase.baidubce.com 2181
Trying 10.10.10.10...
Connected to zk-o9jwa9-1.hbase.baidubce.com.
Escape character is '^'.
```

### 🔗 与云数据库HBase网络不通时，我应该怎么做？

如果您与云数据库HBase产品网络不通，您可能会看到如下结果：

```
[user_name@your_computer ~]# telnet zk-xxxxx-1.hbase.baidubce.com 2181
Trying 10.10.10.10...
telnet: connect to address 10.10.10.10: Operation timed out
telnet: Unable to connect to remote host
```

此时，您可以：

- 查看您的云数据库HBase实例的白名单是否配置正确。具体操作请参考[设置白名单](#)。
- 查看您的本地网络是否开通了出口白名单。

- 如果您需要使用百度智能云的VPC服务，请确认您的本地环境是否能够访问VPC环境。
- 如果您与您的实例在不同的VPC环境下，请确认您的VPC服务配置正确。具体操作请参考[对等连接](#)。

如果您经过自我排查后仍然无法连通您的实例，您可以尝试提交工单，[云数据库HBase运维团队](#)会帮助您解决问题。

## 访问HBase Web UI

Web UI展示了HBase集群的状态，其中包括整个集群概况信息、RegionServer和Master的信息、快照、运行进程等信息。通过Web UI提供的信息可以对整个HBase集群的状况有一定的了解。本文主要介绍内网环境下如何访问HBase Web UI。

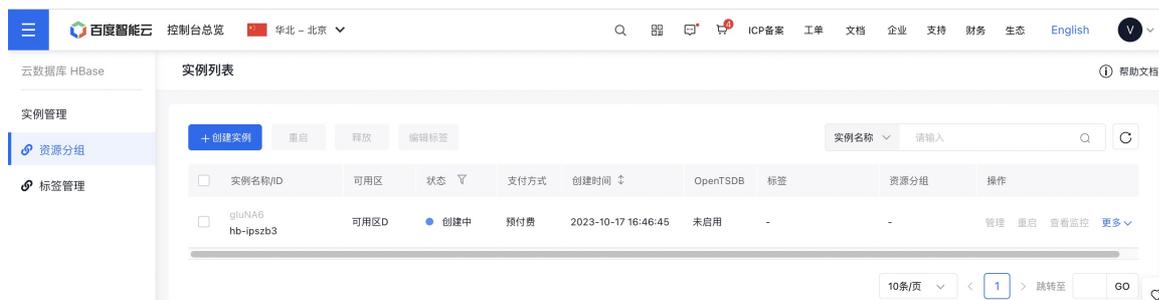
### 使用内网访问Web UI

#### 前提条件

- 准备一台和HBase实例在同一个vpc内的Windows云服务器。具体操作可参考[快速配置Windows云服务器](#)。
- 已将Windows云服务器的内网IP地址加入到HBase实例的白名单中。

#### 操作步骤

##### 1. 登录HBase管理控制台。



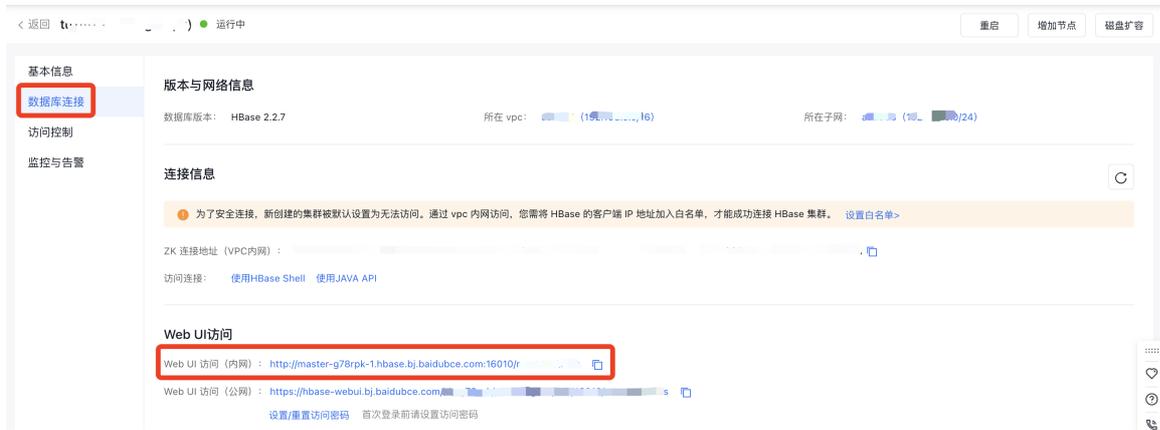
##### 2. 在页面左上角，选择实例所在的地域。



##### 3. 找到目标实例，单击实例名称或具体实例-操作栏的【管理】按钮进入【数据库连接】页签。



##### 5. 在实例数据库连接页中，获取实例内网访问Web UI连接地址。



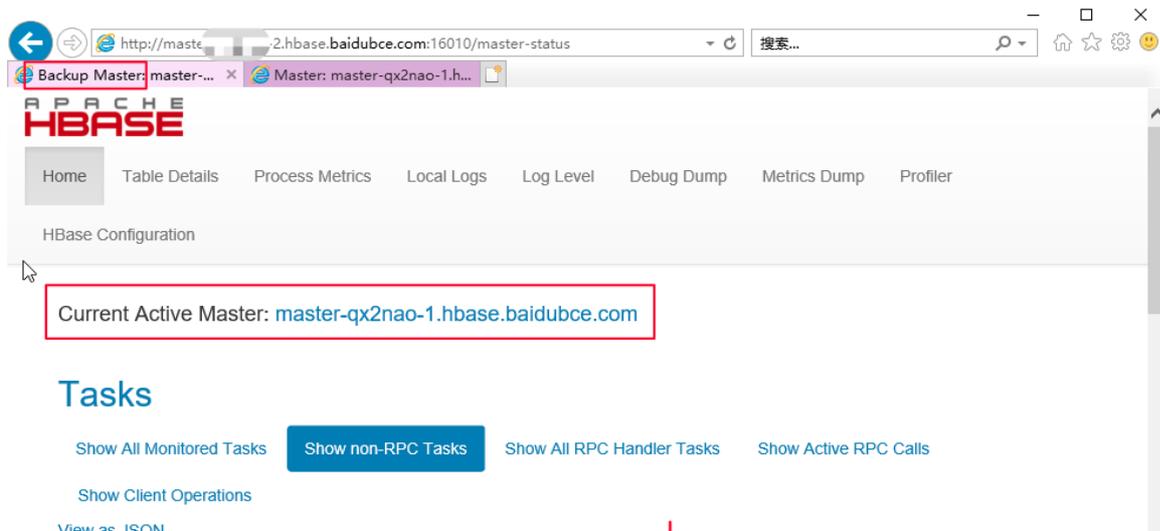
5. 登录Windows云服务器。您可以[使用VNC远程登录Windows云服务器](#)。

6. 在Windows云服务器的浏览器中输入如下地址：

```
http://master-xxxxx-1.hbase.baidubce.com:16010/master-status
```

其中，“master-xxxxx-1.hbase.baidubce.com”是Master节点的地址，“16010”是Master节点Web UI的端口。

7. 如果打开的是Backup Master的页面，您可点击其中的Active Master节点域名地址跳转到Active Master的页面。

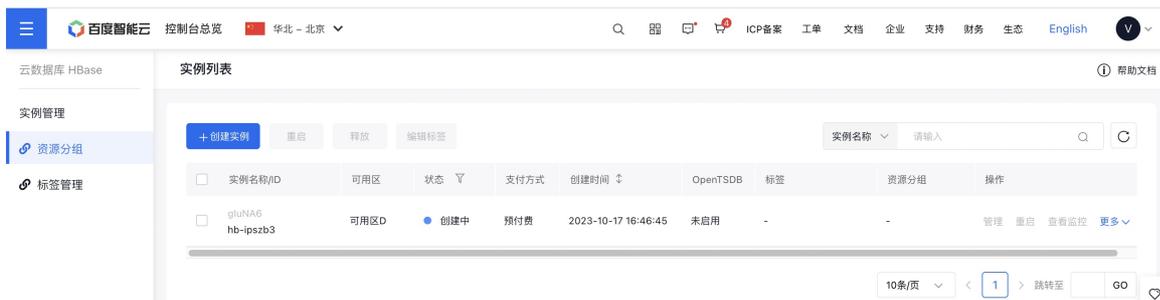


🔗 使用公网访问Web UI

**前提条件**

- 已将客户端公网IP地址加入到HBase实例的白名单中。 [操作步骤](#)

1. 登录HBase管理控制台。



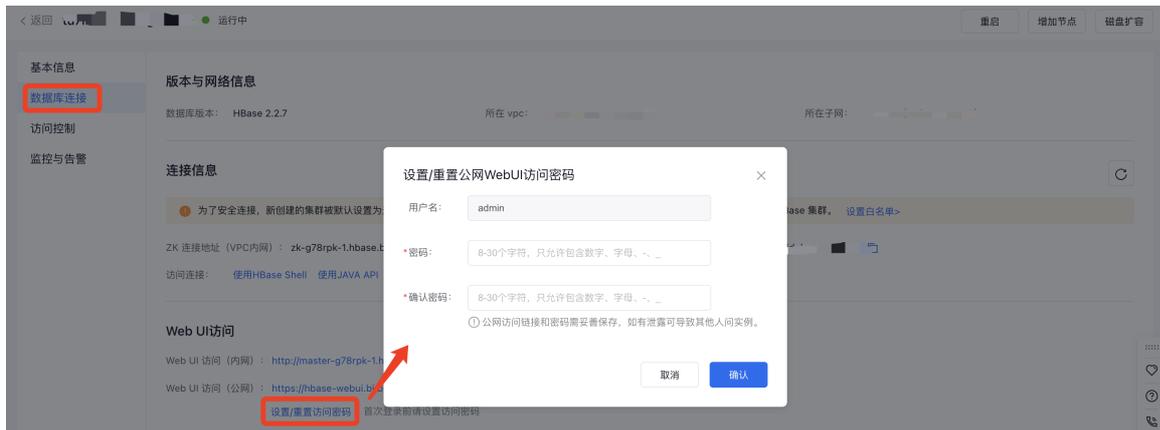
2. 在页面左上角，选择实例所在的地域。



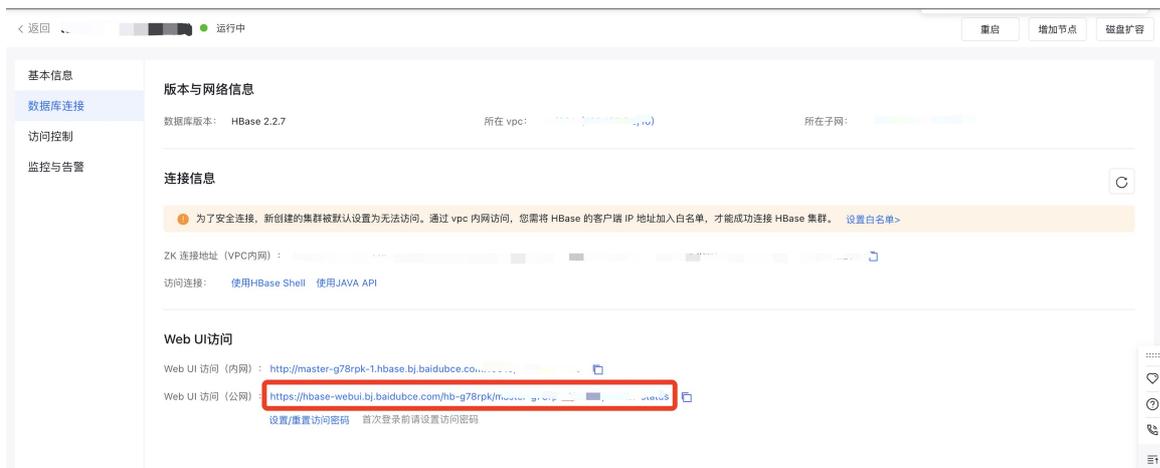
3. 找到目标实例，单击实例名称或具体实例-操作栏的【管理】按钮进入【数据库连接】页签。

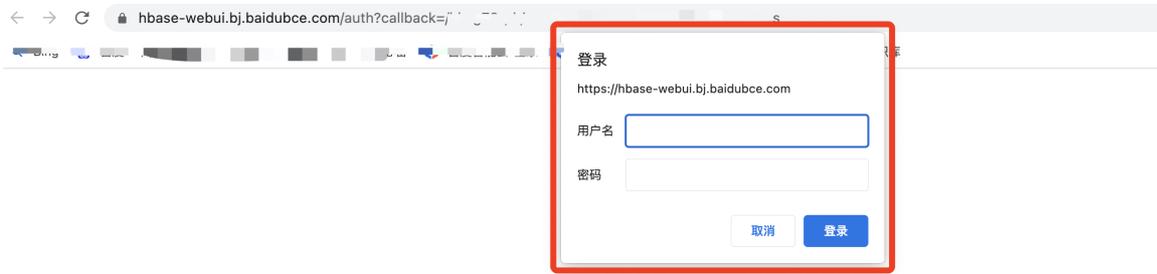


4. 首次使用公网访问Web UI时，您需设置访问登录密码。请妥善保管密码，防止泄露，建议您定期更换密码，您可通过【重置访问密码】重新设置。



5. 点击公网Web UI连接地址，输入用户名和密码，点击登录。





6. 如果打开的是Backup Master的页面，您可点击其中的Active Master节点域名地址跳转到Active Master的页面（同内网访问步骤7）。

## Web UI信息简介

1. 在Active Master的Web UI页面中，**Home**页面展示的是HBase实例的概况信息，主要包括以下内容：

- Region Servers页面展示了RegionServer的基本信息

### Region Servers

ServerName	Start time	Last contact	Version	Requests Per Second	Num. Regions
core-1-hbase.baidubce.com,16020,1658480419339	Fri Jul 22 17:00:19 CST 2022	0 s	2.2.7	0	5
core-2-hbase.baidubce.com,16020,1658480430114	Fri Jul 22 17:00:30 CST 2022	1 s	2.2.7	0	4
Total:2				0	9

- Backup Master页面展示了Backup Master的基本信息

### Backup Masters

ServerName	Port	Start Time
master-2-hbase.baidubce.com	16000	Fri Jul 22 17:00:09 CST 2022
Total:1		

- Tables页面显示了实例中表的信息，包括User Tables、System Tables、Snapshots

### Tables

Namespace	Name	State	Regions										Description
			OPEN	OPENING	CLOSED	CLOSING	OFFLINE	FAILED	SPLIT	Other			
default		ENABLED	1	0	0	0	0	0	0	0	0		
default		ENABLED	4	0	0	0	0	0	0	0	0		

2. 在Active Master的Web UI页面中，**HBase Configuration**页面展示的是HBase实例的配置信息

```

<?xml version="1.0" encoding="UTF-8"?>
- <configuration>
  - <property>
    <name>dfs.journalnode.rpc-address</name>
    <value>...</value>
    <final>false</final>
    <source>hdfs-default.xml</source>
  </property>
  - <property>
    <name>io.storefile.bloom.block.size</name>
    <value>...</value>
    <final>false</final>
    <source>hbase-default.xml</source>
  </property>
+ <property>
- <property>
    <name>hbase.regionserver.hostname</name>
    <value>...</value>
    <final>false</final>
    <source>hbase-site.xml</source>
  </property>
- <property>
    <name>mapreduce.job.maxtaskfailures.per.tracker</name>
    <value>...</value>
    <final>false</final>
    <source>mapred-default.xml</source>
  </property>
- <property>
    <name>yarn.client.max-cached-nodemangers-proxies</name>
    <value>...</value>
    <final>false</final>
    <source>yarn-default.xml</source>
  </property>
- <property>
    <name>mapreduce.job.speculative.retry-after-speculate</name>
    <value>...</value>
    <final>false</final>
    <source>mapred-default.xml</source>
  </property>
- <property>
    <name>hbase.rest.threads.min</name>
    <value>...</value>
    <final>false</final>
    <source>hbase-default.xml</source>
  </property>
- <property>
    <name>hbase.rs.cacheblocksonwrite</name>
    <value>...</value>
    <final>false</final>
    <source>hbase-default.xml</source>
  </property>
</configuration>

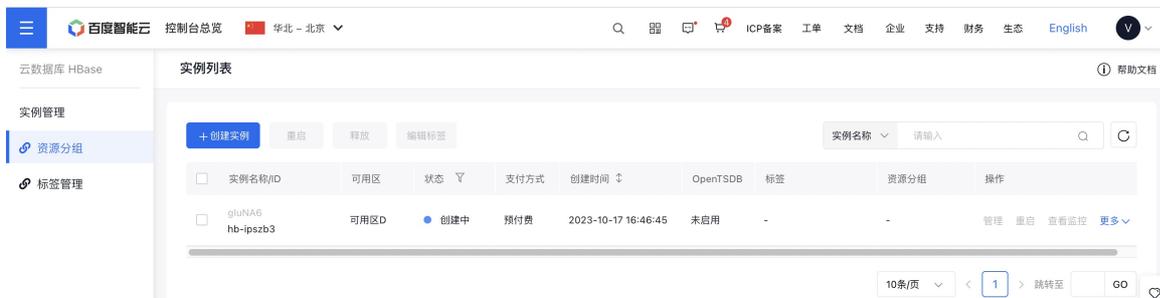
```

注意：Windows云服务器自带的IE浏览器默认开启了IE增强模块，为方便资源访问，可将该功能关闭，操作步骤可参考文档[windowsServer关闭IE增强机制教程](#)

## 管理实例

### 查看实例信息

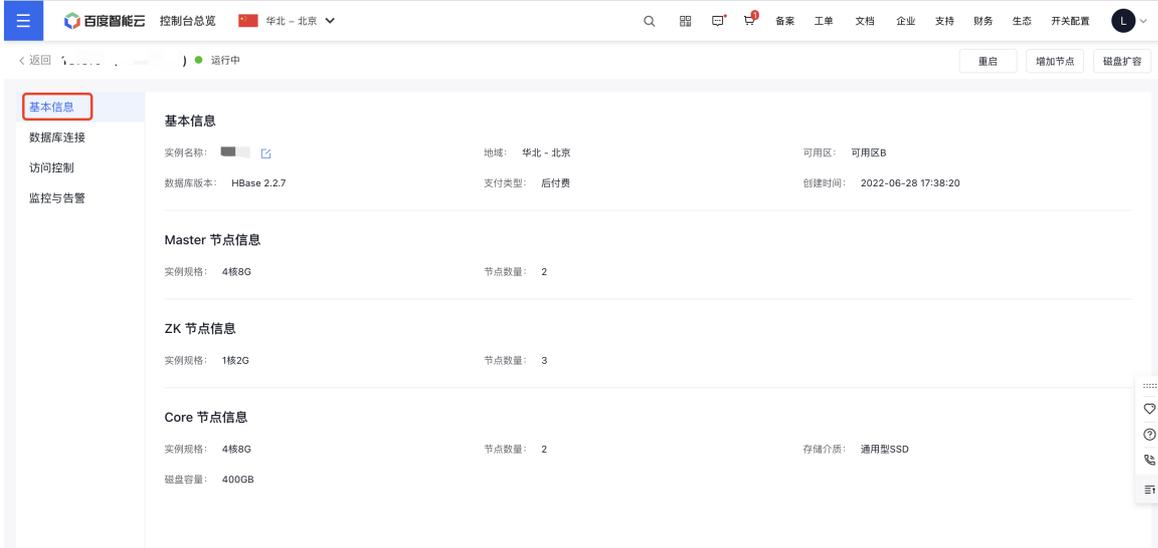
1. 登录HBase管理控制台。



2. 在实例列表中找到需要查看的实例，单击实例名称，或【操作】栏中的【管理】按钮。



3. 进入实例【基本信息】页面，可查看基本信息、节点信息及存储配置及使用详情等，如下图所示。



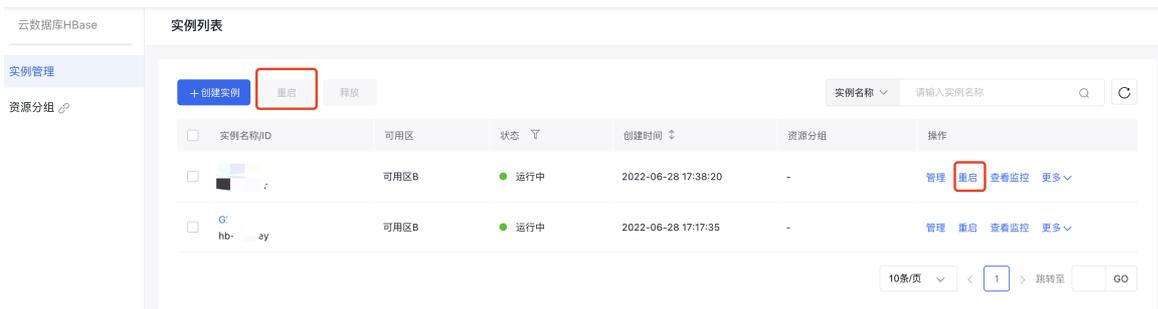
## 重启实例

### 概述

如果您的HBase实例异常或者有参数调整后重启生效，可以手动重启该实例。

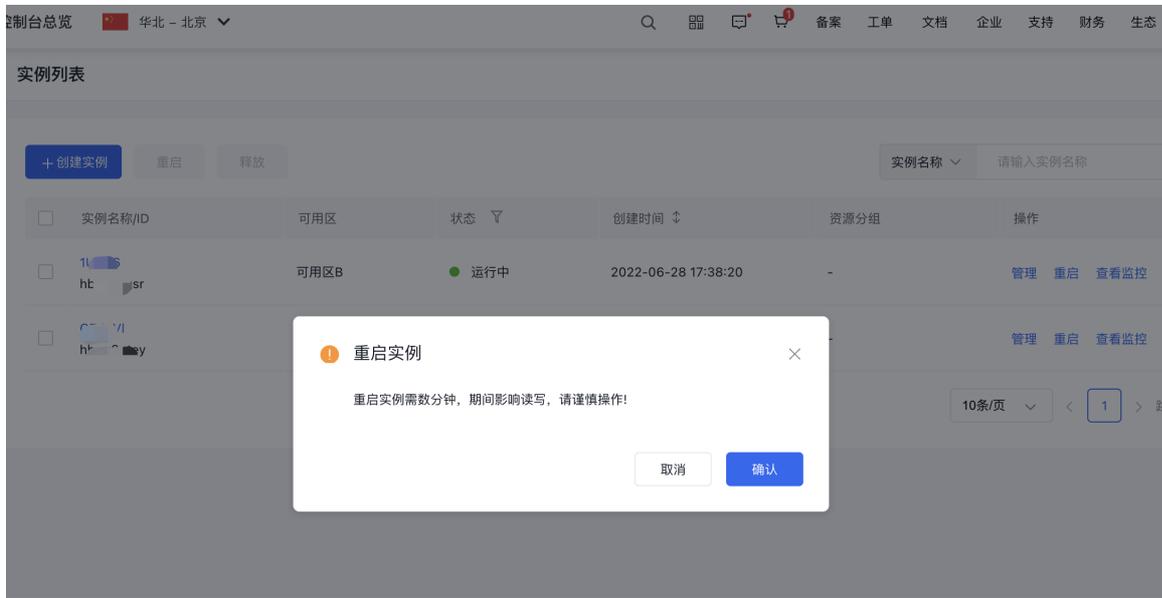
### 操作指南

1. 登录云数据库HBase控制台。
2. 选择单个实例，单击【操作】栏中【重启】按钮，也可以选择多个实例进行批量重启。



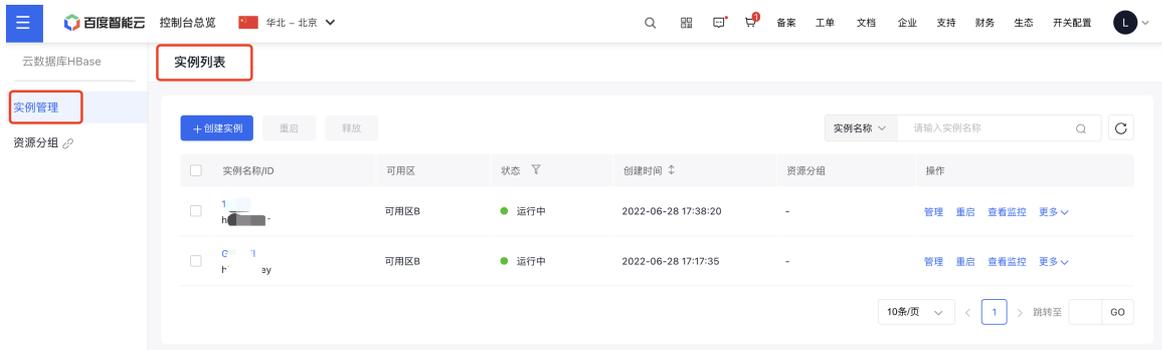
3. 在重启确认弹窗中，单击【确认】按钮，将立即重启该实例。

确认重启后，该实例的所有进程将会关闭并重新启动，会导致您的业务中断，影响读写需要谨慎操作。



### 查看实例列表

1.进入云数据库HBase控制台，您可以查看已购买的所有HBase实例，如下图所示。



2.管理列表参数说明如下表所示。

信息	说明
实例名称/ID	实例的名称，创建集群时设置, ID系统自动创建
可用区	实例所在的可用区名称
状态	实例运行状态信息。集群状态包括创建中、创建失败、运行中、异常、重启中和冻结，实例状态说明 参考本文3. 状态参数说明
创建时间	实例创建的时间
资源分组	实例设置的资源组
操作	实例可执行的操作，包括管理、重启、查看监控、增加节点、扩容磁盘、释放

3.状态参数说明 如下表所示

状态	状态说明	可执行的操作
创建中	实例正在创建，创建成功后将进入【运行中】状态	-
创建失败	实例创建失败，不会计费，无需操作	-
运行中	实例正常运行，服务可用	管理、重启、查看监控、扩容、释放
改配中	实例进行扩容磁盘或增加节点个数变配操作且成功后，实例状态变为【运行中】	可以配置白名单，但不可以查看监控、扩容、释放
改配失败	实例改配失败，原实例状态依然为运行中	管理、重启、查看监控、扩容、释放
重启中	实例正在重启，重启成功，集群状态变为【运行中】	可以配置白名单，但不可以查看监控，不可进行扩容、释放、增加节点
实例欠费停服	欠费状态停止服务	-
实例续费启动中	后付费欠费实例释放前完成充值续费操作，实例启动	-

## 实例扩容

云数据库HBase可通过控制台可以对集群进行在线节点扩容及磁盘扩容的功能。

### 增加节点

实例创建后可以根据业务需要增加core(regionserver)节点，加的节点与之前core节点规格一致，

包括:节点的CPU、内存及磁盘，目前单个数据库实例支持最大规模100台core节点，单次最多扩容20台，如果您的业务超过限制，可提交工单联系提高配额。

注意：扩容节点，会添加磁盘，相当于同时扩容了磁盘。

对实例的影响：

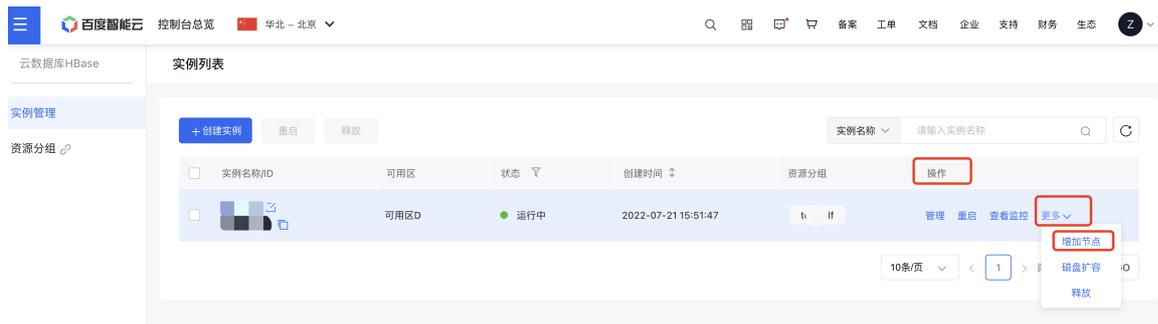
正常情况下影响可控，但建议您业务低峰期执行。扩容后分区会执行负载均衡，部分分区会转移到新节点，单个分区转移过程中会有短暂不可用（正常200ms内），如果客户端配置了重试（默认开启）一般都会重试成功。但如果客户端的超时时间设置的比较短则可能出现请求报错现象。

操作说明：

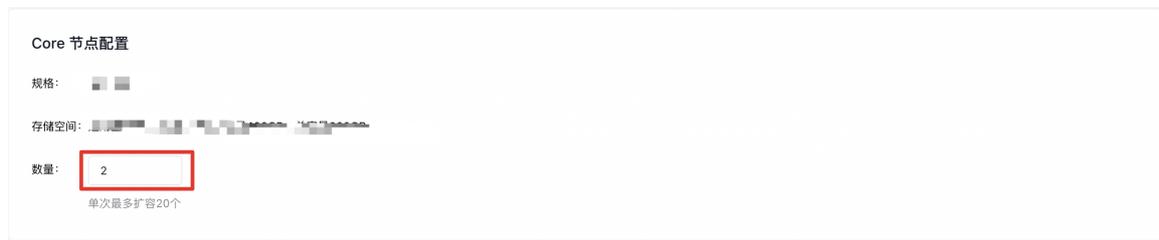
### 方式一：实例列表

1.您可以登录[云数据库HBase控制台](#)，对特定实例进行操作。

2.点击【操作】-【更多】，选择【增加节点】，可进行扩容。



3.在CORE节点配置的数量列中，输入节点扩容数量。最大值20个



4. 点击【下一步】按钮，根据后续提示完成支付操作。

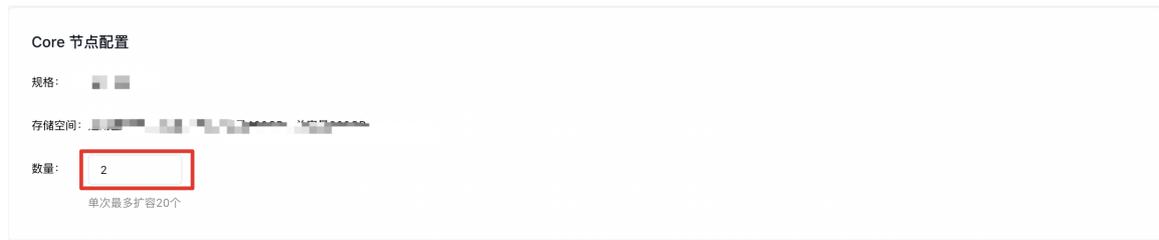


方式二：实例详情页 1.您可以登录[云数据库HBase控制台](#)，对特定实例进行操作。

2.在实例列表页，点击【管理】,进入实例详情页面,可在页面上方操作【增加节点】按钮，进行扩容。



3.在CORE节点配置的【数量】列中，输入节点扩容数量。最大值20个



4. 点击【下一步】按钮，根据后续提示完成支付操作。



## 🔗 磁盘扩容

可以对core节点的容量直接添加，仅可直接增加容量，但是不可变更存储介质类型，每个磁盘单次增加至少50GB的大小。

对集群的影响：

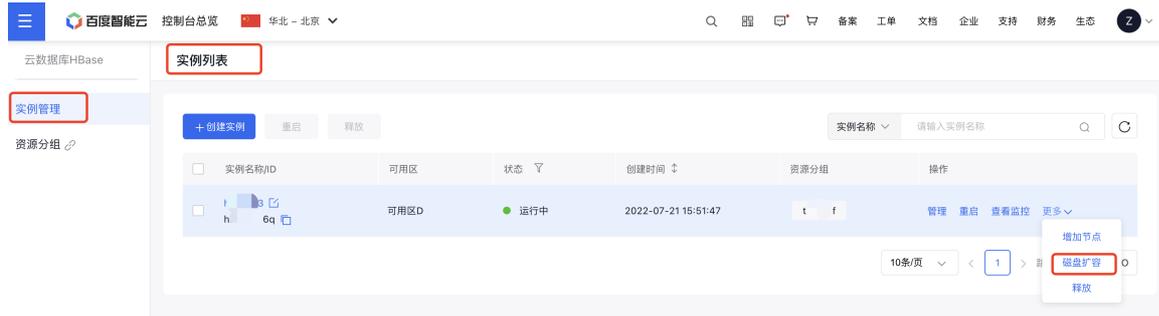
本地盘会滚动重启regionserver及datanode，对集群影响较小，云磁盘不需要重启进程，对集群基本无影响。

操作说明：

### 🔗 方式一：实例列表

1.您可以登录[云数据库HBase控制台](#)，对特定实例进行操作。

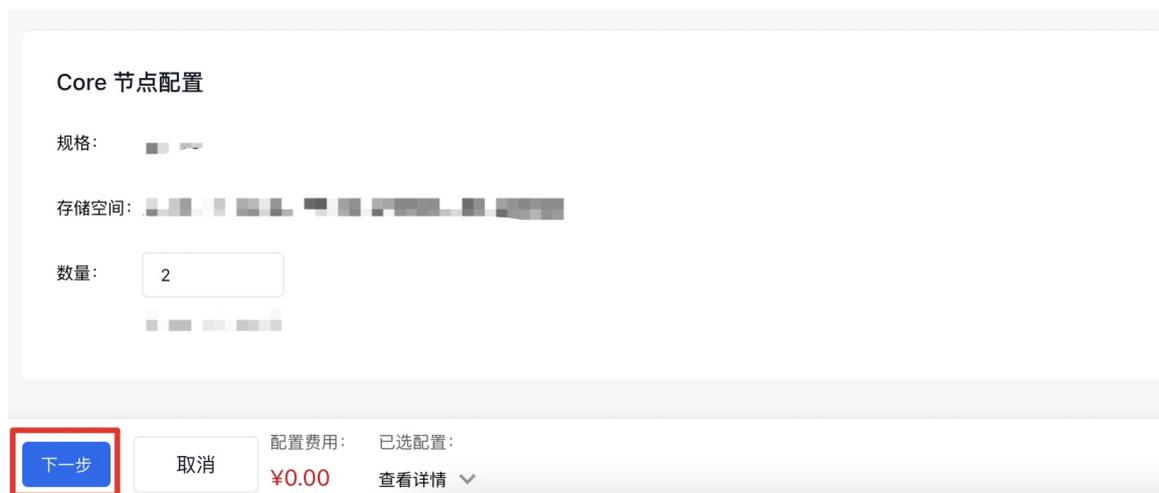
2.选择需要增加CORE节点的实例，在【操作】栏中单击【增加磁盘】，可进行扩容。



3.在实例列表详情页，Core节点配置的【数据盘】，可选择拖动【滑块】或输入扩容数字，来确定扩容的磁盘大小。



4. 点击【下一步】按钮，根据后续提示完成支付操作。



方式二：实例详情页 1.您可以登录[云数据库HBase控制台](#)，对特定实例进行操作。

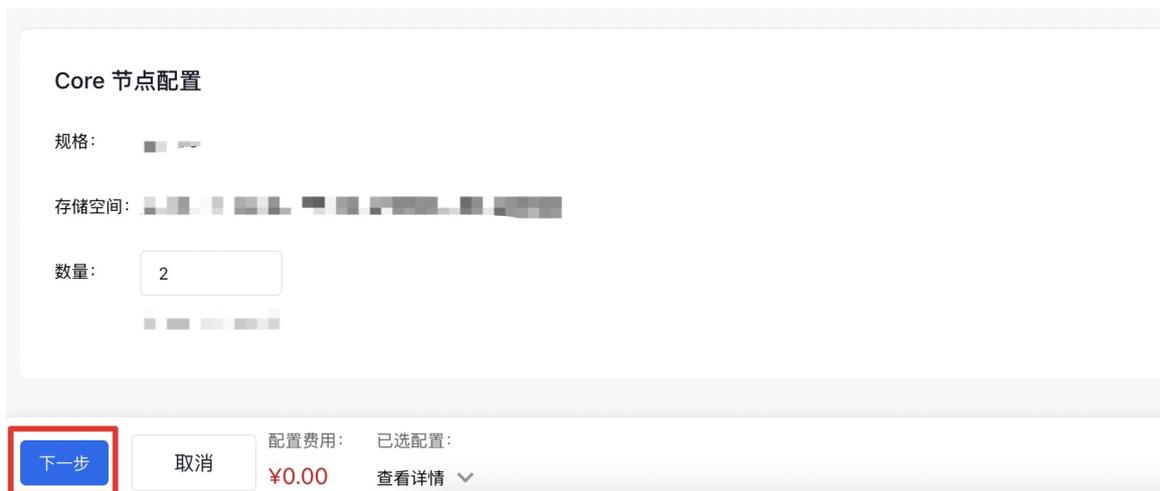
2.或者您可通过点击【管理】，进入实例详情页，可在页面上方操作【磁盘扩容】按钮。



3. 在实例列表详情页面，Core节点配置的【数据盘】，可选择拖动【滑块】或输入扩容数字，来确定扩容的磁盘大小。

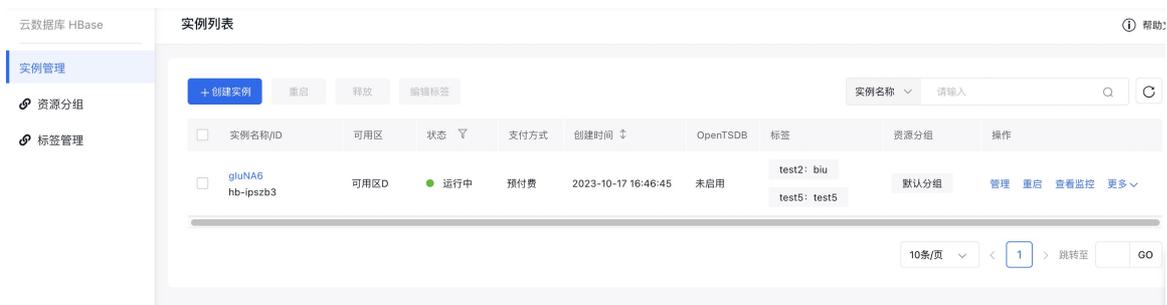


4. 点击【下一步】按钮，根据后续提示完成支付操作。

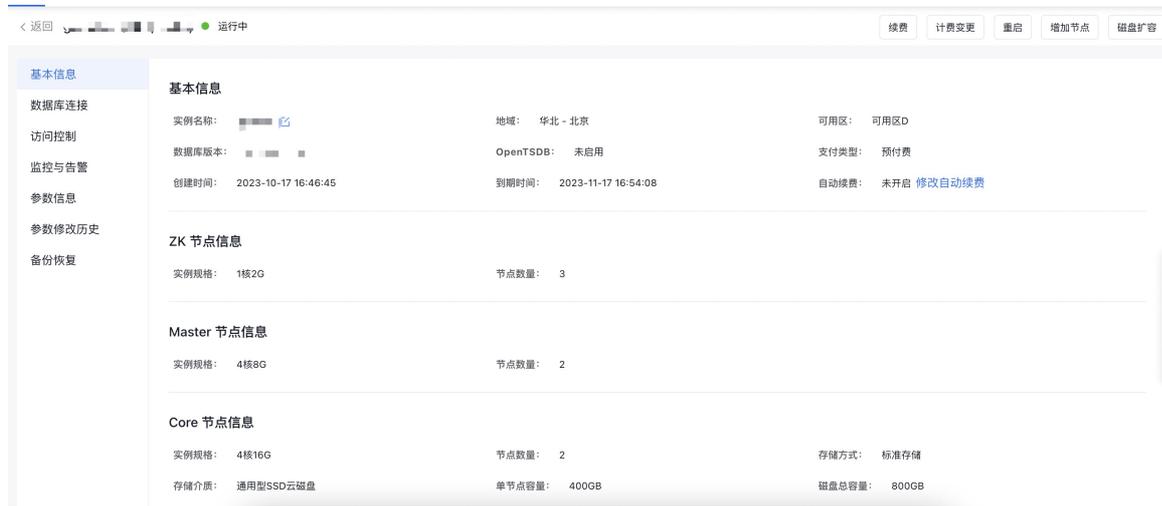


## 数据库连接

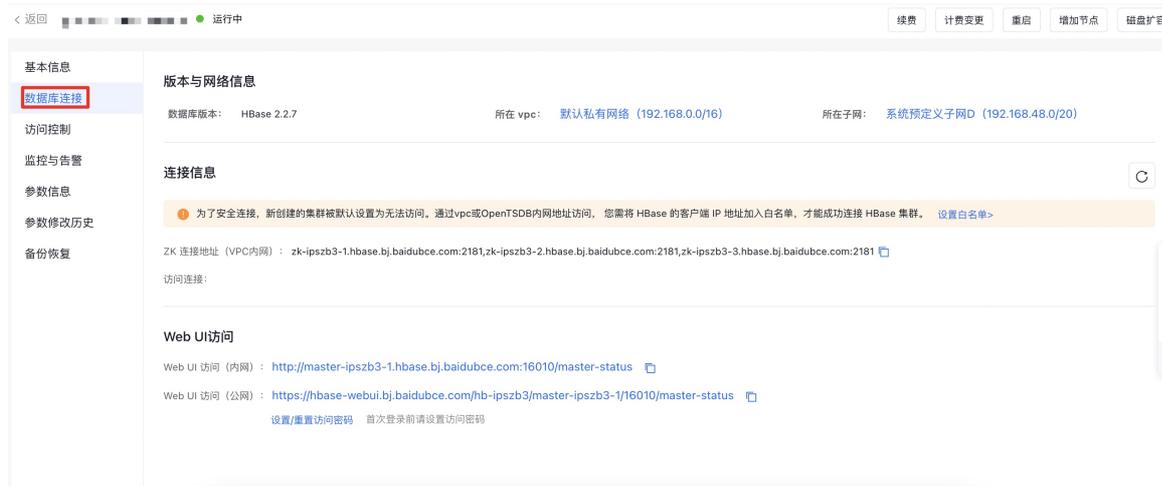
1. 登录HBase管理控制台。



2. 在实例列表中找到需要查看的集群，单击集群名称，进入基本信息页面。



3. 点击数据库连接，进入数据库连接信息页面。页面包括版本与网络信息、连接信息、WebUI访问。



4. 可以点击链接信息后面的复制按钮。

## 注意

连接信息需要讲IP地址添加到白名单方可访问集群。

## 释放实例

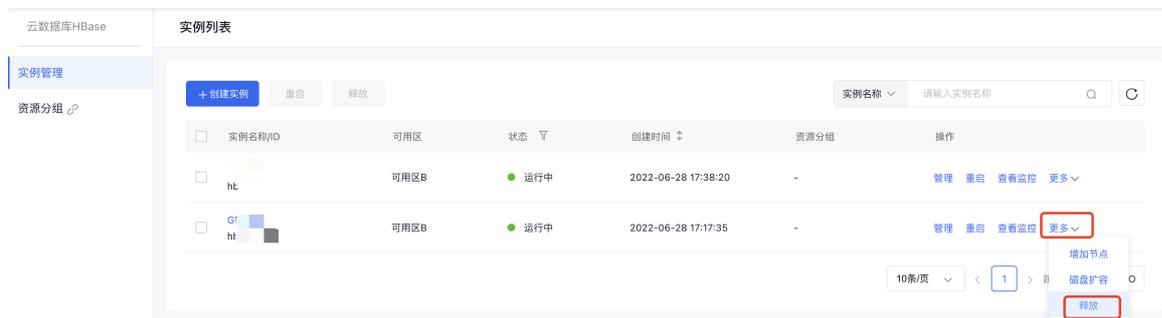
### 概述

当您不再需要HBase实例，可以在将实例释放，释放后资源和数据将被立即清除，并停止计费，资源与数据无法恢复请您谨慎操作。

### 单个释放实例

1. 需要登录 [云数据库HBase控制台](#)。

2. 在实例列表选择您需要删除的实例，在操作列选择【更多】菜单下的释放，点击【释放】文字，



- 弹出释放对话框，并阅读释放前提示，若确定删除，则点击【确定】按钮。

## ！ 释放实例 ×

释放后，将会清空该实例所有数据和日志，且不可恢复。



### 🔗 批量释放实例

- 登录[云数据库HBase控制台](#)。
- 在实例列表中可通过勾选多个需要释放的实例。



- 点击列表上方的【释放】按钮。



- 弹出**释放实例**对话框，并阅读释放前提示，若确定删除，则点击【确定】按钮，可实现批量释放实例。

## ！ 释放实例 ×

释放后，将会清空该实例所有数据和日志，且不可恢复。



## 多用户访问控制

### 🔗 介绍

多用户访问控制，主要用于帮助用户管理云账户下资源的访问权限，适用于企业内的不同角色，可以对不同的工作人员赋予使

用产品的不同权限，当您的企业存在多用户协同操作资源时，推荐您使用多用户访问控制。

适用于下列使用场景：

- 中大型企业客户：对公司内多个员工授权管理。
- 偏技术型vendor或SAAS的平台商：对代理客户进行资源和权限管理。
- 中小开发者或小企业：添加项目成员或协作者，进行资源管理。

## 创建用户

1.主账号用户登录后在控制台选择“多用户访问控制”进入用户管理页面。



2.在左侧导航栏点击“用户管理”，在“子用户管理列表”页，点击“新建用户”。

3.在弹出的“新建用户”对话框中，完成填写“用户名”和确认，返回“子用户管理列表”区可以查看到刚刚创建的子用户。

## 配置策略

云数据库HBase支持系统策略和用户自定义策略两种，分别实现产品级权限和实例级权限控制。

- 系统策略：百度智能云系统为管理资源而预定义的权限集，这类策略可直接为子用户授权，用户只能使用而不能修改。
- 自定义策略：由用户自己创建，更细化的管理资源的权限集，可以针对单个实例配置权限，更加灵活的满足账户对不同用户的差异化权限管理。

## 系统策略

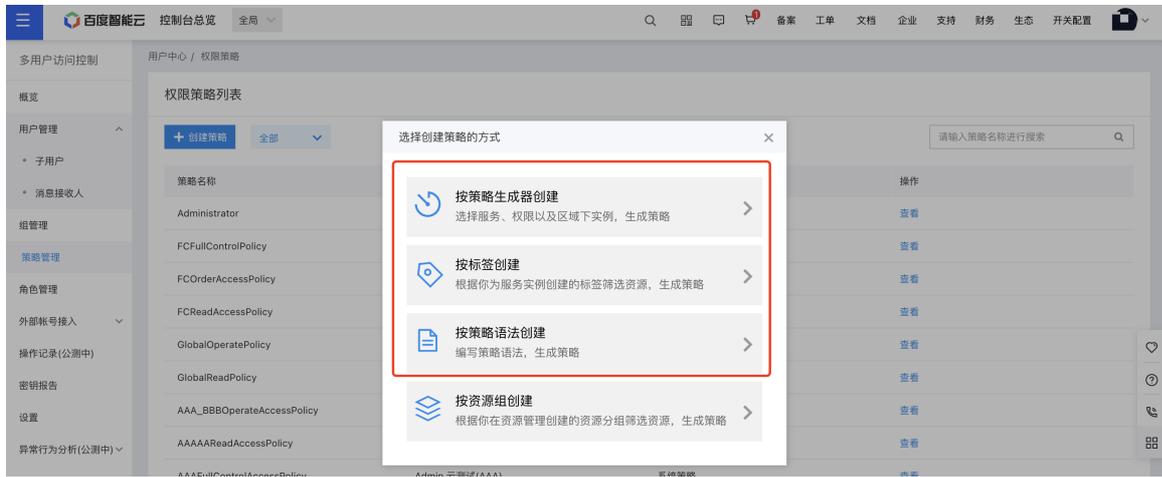
系统策略包含管理权限、运维权限和只读权限3种策略，权限范围详细如下：

策略名称	权限说明	权限范围
HBASEFullControlAccessPolicy	完全控制管理云数据库HBASE的权限	创建实例、释放实例、实例变配、预付费实例续费、修改实例名称、重启实例、重设实例WebUI密码、设置白名单列表、删除白名单列表、更新标签绑定、查询实例列表、获取可用区列表、获取实例详情、获取白名单列表
HBASEOperateAccessPolicy	运维操作云数据库HBASE的权限	查询实例列表、修改实例名称、重启实例、重设实例WebUI密码、设置白名单列表、删除白名单列表、更新标签绑定、获取可用区列表、获取实例详情、获取白名单列表
HBASEReadAccessPolicy	只读访问云数据库HBASE的权限	查询实例列表、获取可用区列表、获取实例详情、获取白名单列表

## 自定义策略

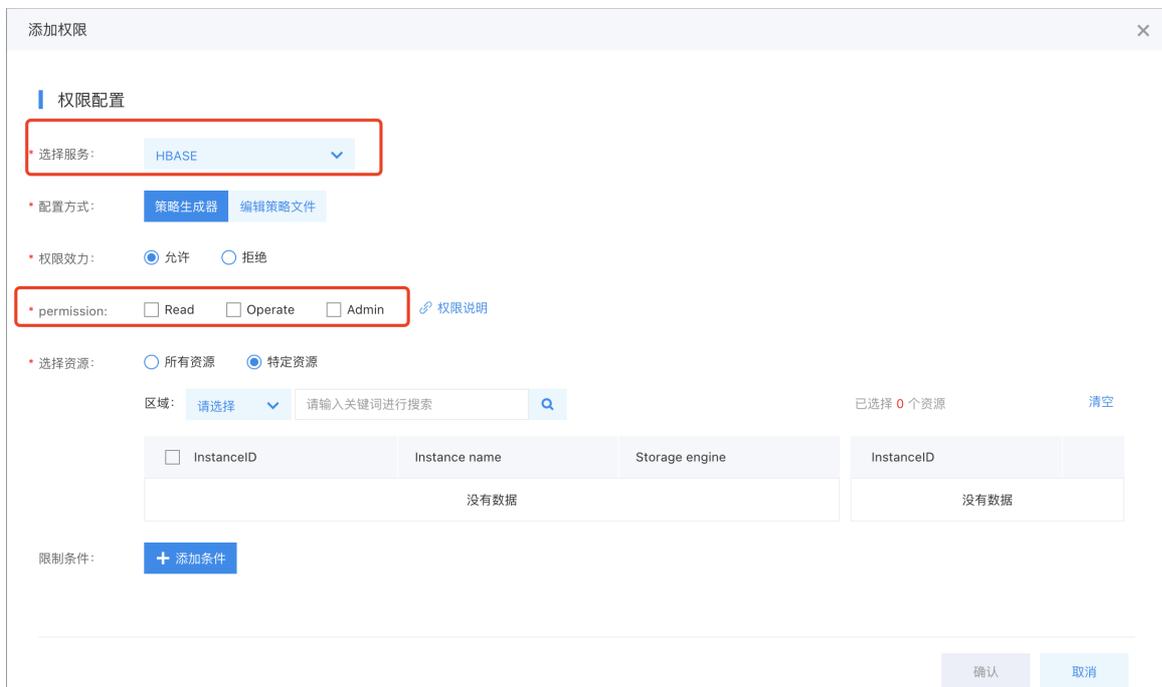
自定义策略可以实现精细的实例维度授权，与系统策略不同，可灵活配置允许或拒绝某些实例的操作，可细粒度的进行资源范围的限定。

1.子用户先通过左侧导航栏进入【策略管理】，然后点击“创建策略”，云数据库HBase按照策略生成器、标签、策略语法3种创建方式。



以按照策略生成器创建为例

用户填写合法策略名称后，点击创建“添加权限”选择服务类型为“HBase”，其中策略生成方式默认为策略生成器，不需要修改。



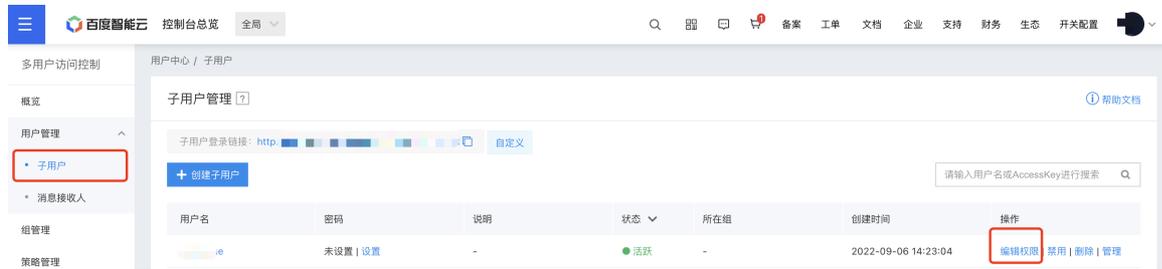
自定义

权限范围详细如下：

分类	操作
只读 (Read)	查询实例列表、获取可用区列表、获取实例详情、获取白名单列表
运维 (Operate)	查询实例列表、修改实例名称、重启实例、重设实例WebUI密码、设置白名单列表、删除白名单列表、更新标签绑定、获取可用区列表、获取实例详情、获取白名单列表
管理 (Admin)	创建实例、释放实例、实例变配、预付费实例续费、查询实例列表、修改实例名称、重启实例、重设实例WebUI密码、设置白名单列表、删除白名单列表、更新标签绑定、获取可用区列表、获取实例详情、获取白名单列表

🔗 用户授权

在“用户管理子菜单下的子用户管理列表页”的对应子用户的“操作”列选择“添加权限”，并为用户选择系统权限或已经创建好的自定义策略进行授权。



## 子用户登录

主账号完成对子用户的授权后，可以将链接发送给子用户；子用户可以通过IAM用户登录链接登录主账号的管理控制台，根据被授权的策略对主账号资源进行操作和查看。

其他详细操作参考：[多用户访问控制](#)。

## 标签管理

### 解绑标签

#### 概述

当HBase实例不再需要某个标签时，您可以解除实例与该标签的绑定关系。

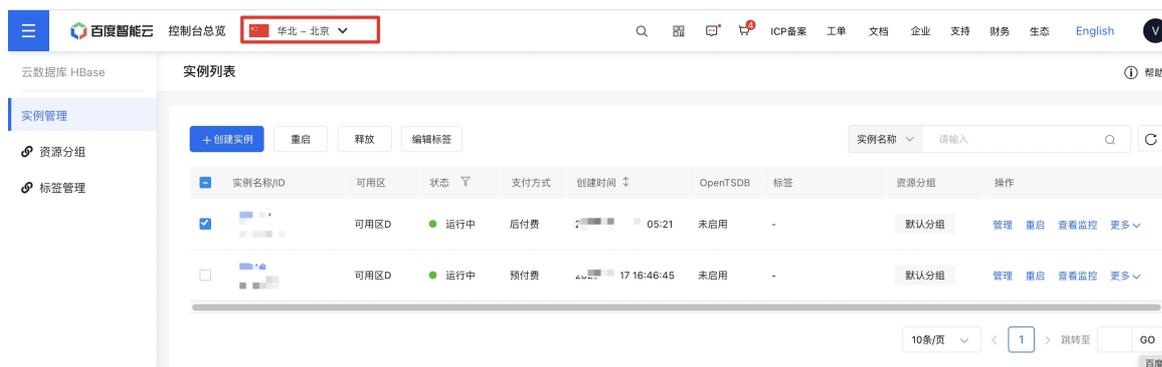
#### 注意事项

- 只能为一个实例解绑标签，不能同时为多个实例解绑标签。
- 单次解绑的标签数量不能超过20个。
- 解绑标签的操作对实例运行无影响。当实例的所有标签被解绑后，该实例不能通过标签被筛选到。

#### 操作步骤

##### 编辑标签

- 登录HBase管理控制台。
- 在页面左上角，选择实例所在的地域。



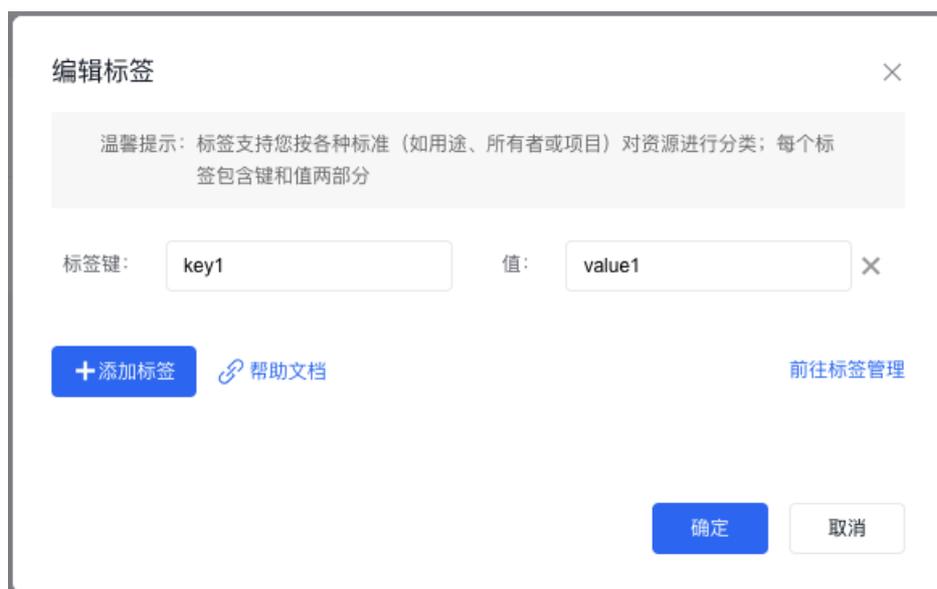
- 在左侧导航栏，单击实例管理。



- 选择目标实例单击实例列表上方的**编辑标签**。

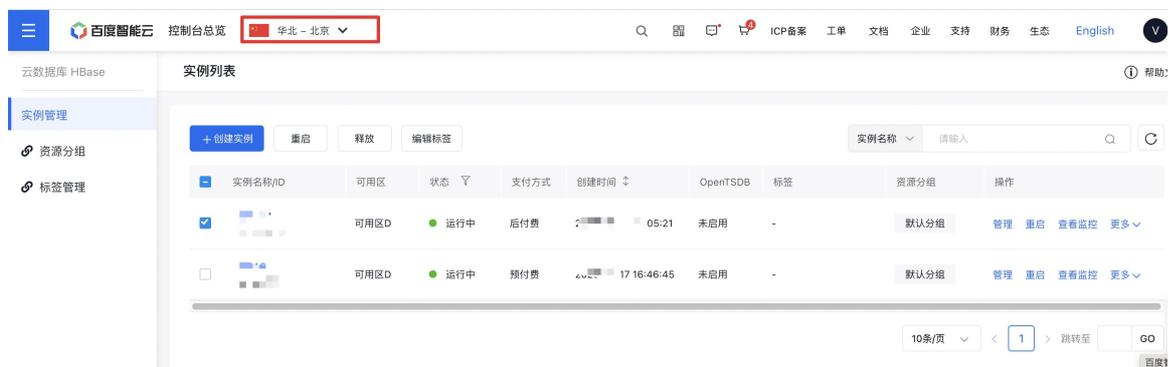


- 弹出的标签编辑窗口会包含实例已经绑定的标签。您可以在弹出窗口对实例绑定标签的状态进行编辑，点击需要解绑的标签右方的“x”，点击确认后，实例绑定的标签的将被更新。



## 实例列表

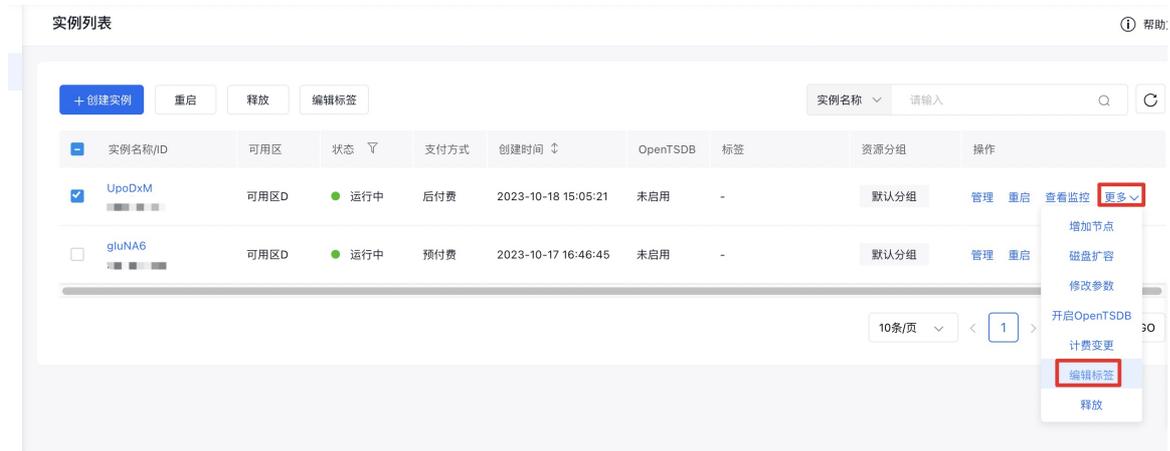
- 登录HBase管理控制台。
- 在页面左上角，选择实例所在的地域。



- 在左侧导航栏，单击**实例管理**。



4. 选择目标实例，单击实例列表操作栏的**更多-编辑标签**。



5. 弹出的标签编辑窗口会包含实例已经绑定的标签。您可以在弹出窗口对实例绑定标签的状态进行编辑，点击需要解绑的标签右方的“x”，点击确认后，实例绑定的标签的将被更新。

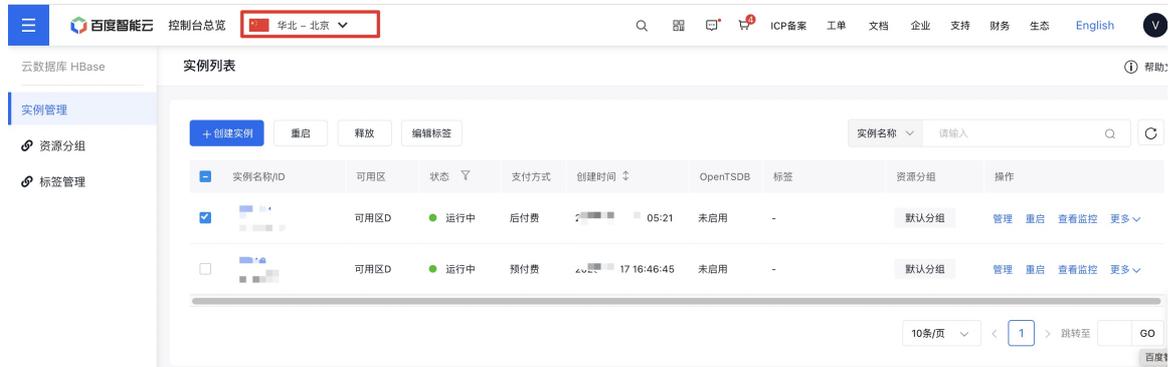


## 根据标签筛选实例

为HBase实例绑定标签后，您可以在实例列表中通过标签来筛选实例，快速查找指定分类的实例以进行管理。

### 🔗 操作步骤

1. 登录HBase管理控制台。
2. 在页面左上角，选择实例所在的地域。



3. 在左侧导航栏，单击实例管理。



4. 单击标签搜索框左侧，选择**标签**，表示以标签来进行筛选实例，然后选择待搜索标签键，最后选择值，单击 ，则绑定了指定标签的实例将被筛选出来，如下图所示：



说明：

- 如需清除标签筛选条件，直接删除选择的值即可。

## 绑定标签

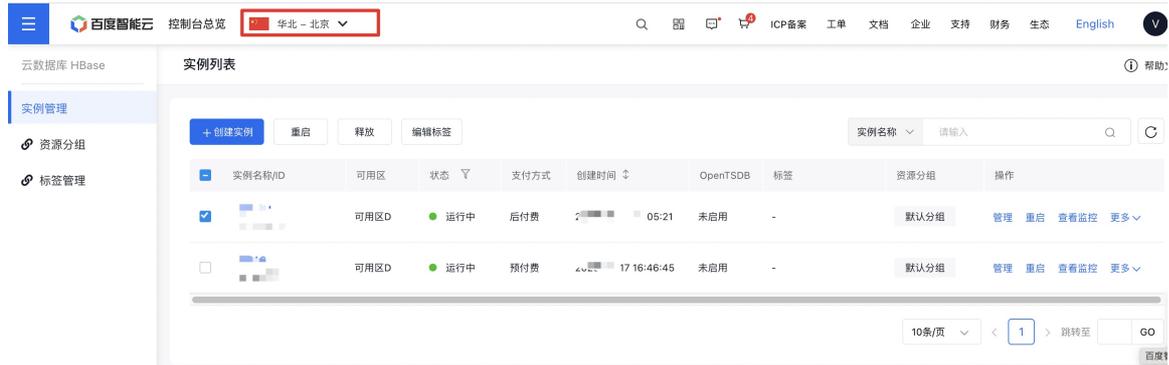
当您完成了新建标签的操作后，您可以为实例绑定若干个的已有的标签，也可以在绑定的同时新建标签，后续可通过标签筛选实例。

### 🔗 注意事项

- 编辑一个实例的标签是批量更新标签的操作。
- 编辑多个实例的标签是追加标签的操作。
- 最多同时选中50个实例进行批量标签绑定。
- 每个实例最多可以绑定20个标签。
- 单次绑定或者解绑的标签数量不能超过20个。

### 🔗 操作步骤

1. 登录HBase管理控制台。
2. 在页面左上角，选择实例所在的地域。



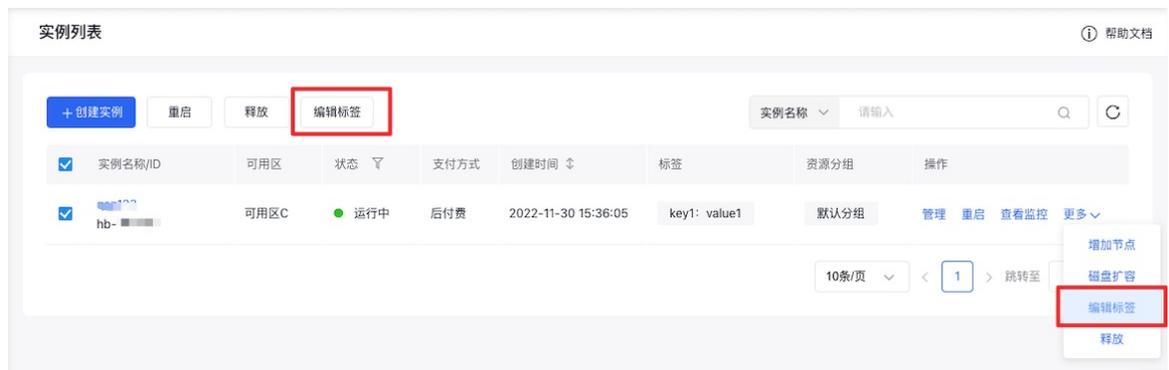
3. 在左侧导航栏，单击实例管理。



4. 根据绑定标签的方式，选择下述操作步骤。

- 为单个实例绑定标签：

选择目标实例单击实例列表上方的**编辑标签**，或点击**更多-编辑标签**，如下图所示：



弹出的标签编辑窗口会包含实例已经绑定的标签。您可以在弹出窗口对实例绑定标签的状态进行编辑，点击确认后，实例绑定的标签的将被更新。如果添加的标签原本不存在，则系统将新建该标签并绑定到实例：



- 为多个实例批量绑定标签：

勾选两个或以上目标实例，单击实例列表上方的**编辑标签**。



不同于单个实例绑定标签，实例批量绑定标签时，弹出的标签编辑窗口不会显示任何标签。

## 编辑标签

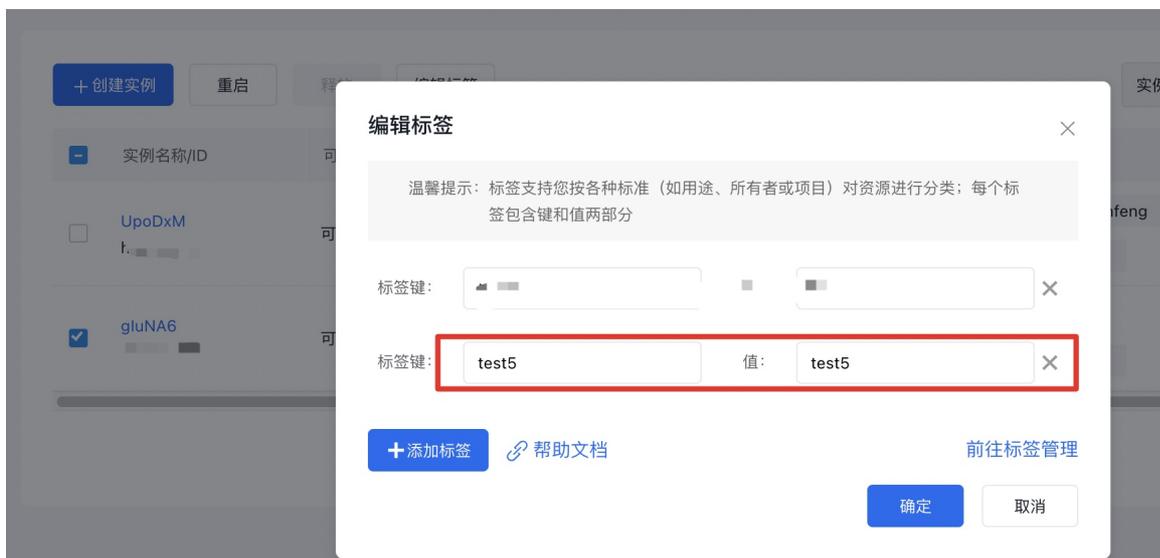


您可以在弹出窗口添加想要绑定的标签，点击确认后，这些标签会自动绑定到选中的实例。如果添加的标签原本不存在，则系统将新建该标签并绑定到实例，下图批量绑定标签后的效果

UpoDxM 实例：



gluNA6 实例：



## 创建标签

### 概述

在实例数量较多的情况下，您可以创建多个标签，为实例绑定不同的标签对其进行分类，后续可通过标签筛选实例。

### 注意事项

- 标签由一对键值组成，键在同账号同地域下唯一，值无此限制。

**说明：**标签键和标签值是1对N的关系，N可以是0。

- 最多同时选中50个实例进行批量标签编辑。
- 每个实例最多可以绑定20个标签。
- 单次绑定或者解绑的标签数量不能超过20个。

### 操作步骤

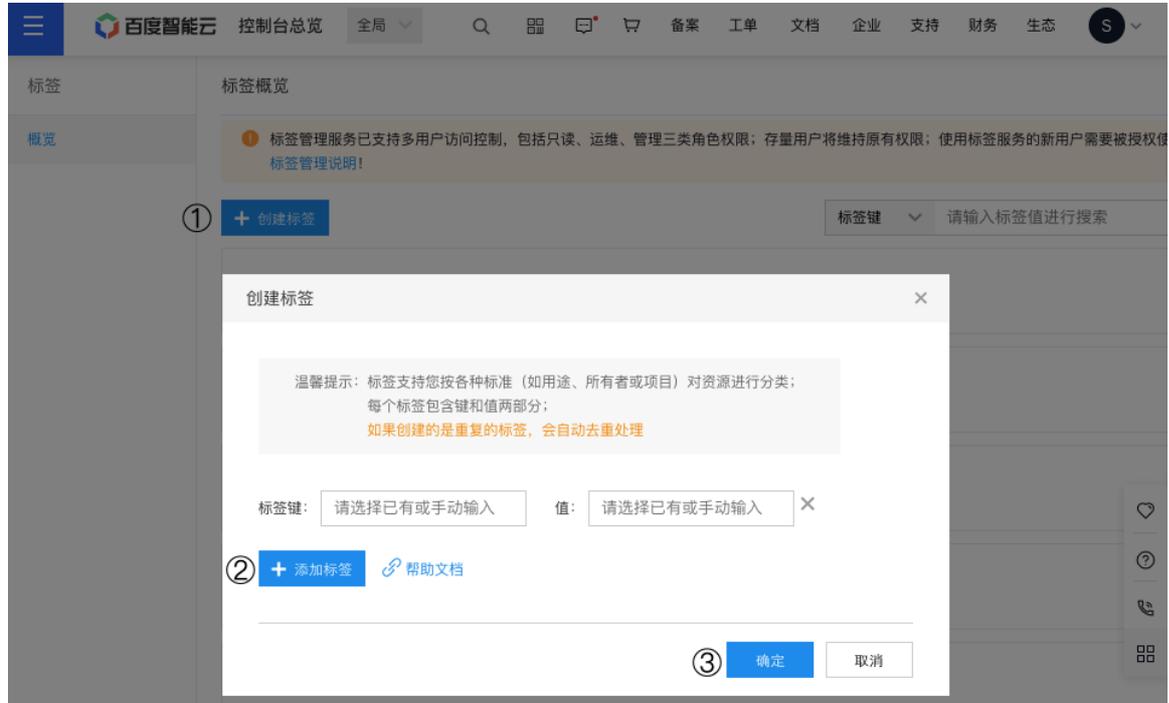
#### 左侧导航栏

1. 登录HBase管理控制台。

2. 在左侧导航栏，单击**标签管理**。



3. 在标签管理页，单击**创建标签**，输入需要创建的标签；如果需要添加多个，则单击**添加标签**按钮；编辑完成后单击**确认**。



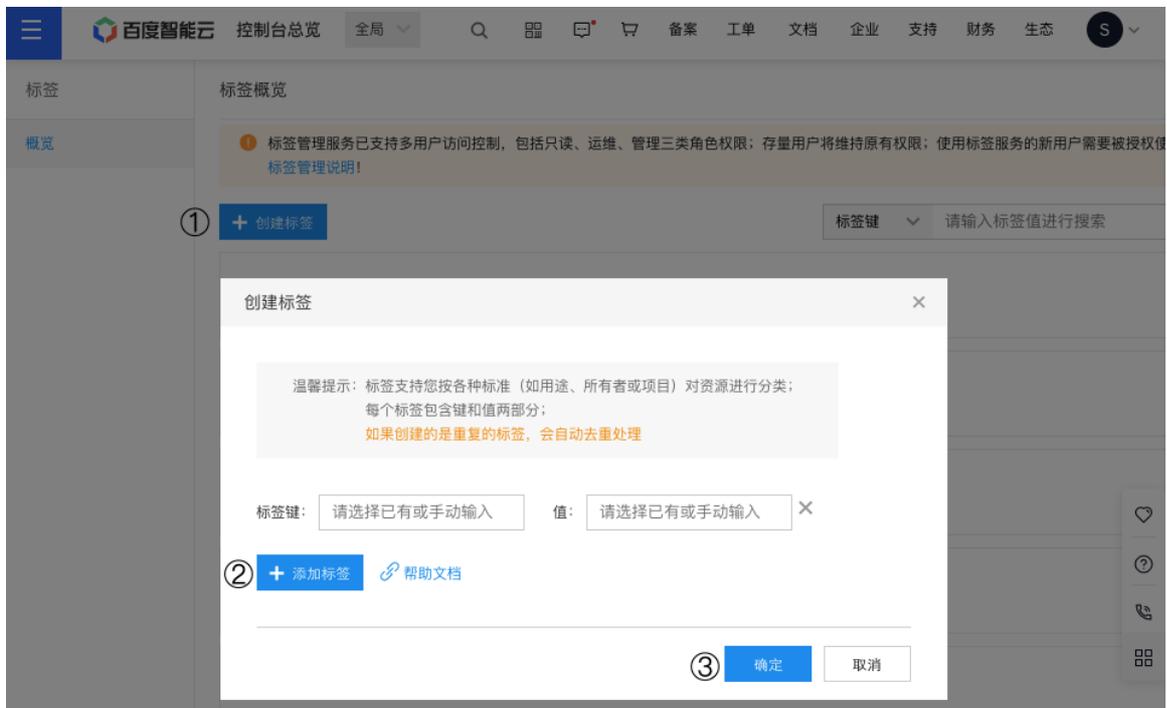
说明：完成标签的新建后，您可以在其它产品中绑定已创建的标签。

## 企业菜单

1. 登录HBase管理控制台。
2. 在左侧导航栏，点击右上方的**企业-标签管理**。



3. 在标签管理页，单击**创建标签**，输入需要创建的标签；如果需要添加多个，则单击**添加标签**按钮；编辑完成后单击**确认**。



## 修改参数

创建HBase集群后，您可以通过修改参数的方式对集群进行性能调优。

### 单个修改参数

1. 登陆[云数据库HBase控制台](#)
2. 点击侧边栏中的[参数信息](#)
3. 选择对应的参数，点击操作列的修改。

参数名称	所属服务	参数默认值	运行参数值	单位	是否需要重启	可修改参数范围	参数描述	操作
<input checked="" type="checkbox"/> hbase.wal.regiongrouping.strategy	HBase	bounded	bounded	ENUM	是	[identity,bounded,na...	region分租策略，每个region group...	修改
<input type="checkbox"/> hbase.wal.regiongrouping.numgroups	HBase	2	3	INT32	是	[1-100]	bounded分租策略下，region group...	修改
<input type="checkbox"/> hbase.wal.provider	HBase	asyncfs	multitw	ENUM	是	[asyncfs,filesystem,...	使用配置设置“hbase.wal.provider”...	修改
<input type="checkbox"/> hbase.regionserver.thread.compaction.small	HBase	1	1	INT32	是	[1-10]	small compaction线程数量	修改
<input type="checkbox"/> hbase.regionserver.thread.compaction.large	HBase	1	1	INT32	是	[1-10]	large compaction线程数量	修改

4. 弹出修改参数弹窗，输入新参数值和修改目的。

## 修改参数



参数修改成功后需要重启才能生效，所有服务进程将会关闭并重新启动，整个过程需要几分钟时间（集群规模越大消耗时间越多），重启中集群仍然可以正常服务，您的服务不会受到影响。

参数名称	可修改参数范围	原参数值	新参数值
hbase.wal.regiongrouping.strategy	[identity,bounded,namespace]	bounded	bounded

修改目的:

请输入内容

0/100

取消

确定

5. 输入完成后点击**确定**按钮。

## 批量修改参数

1. 登陆[云数据库HBase控制台](#)
2. 点击左侧边栏中的“参数信息”一栏
3. 选择对应的参数可以勾选多个，点击列表上方**修改参数**按钮。

参数名称	所属服务	参数默认值	运行参数值	单位	是否需重启	可修改参数范围	参数描述	操作
<input checked="" type="checkbox"/> hbase.wal.regiongrouping.strategy	HBase	bounded	bounded	ENUM	是	[identity,bounded,na...	region分租策略，每个region group...	
<input type="checkbox"/> hbase.wal.regiongrouping.numgroups	HBase	2	3	INT32	是	[1-100]	bounded分租策略下，region group...	
<input type="checkbox"/> hbase.wal.provider	HBase	asyncts	multiwal	ENUM	是	[asyncts,fileystem,...	使用配置设置“hbase.wal.provider”...	
<input type="checkbox"/> hbase.regionserver.thread.compaction.small	HBase	1	1	INT32	是	[1-10]	small compaction线程数量	
<input type="checkbox"/> hbase.regionserver.thread.compaction.large	HBase	1	1	INT32	是	[1-10]	large compaction线程数量	

4. 弹出修改参数弹窗，输入对应修改的新参数值和修改目的。

## 修改参数



参数修改成功后需要重启才能生效，所有服务进程将会关闭并重新启动，整个过程需要几分钟时间（集群规模越大消耗时间越多），重启中集群仍然可以正常服务，您的服务不会受到影响。

参数名称	可修改参数范围	原参数值	新参数值
hbase.wal.regiongrouping.strategy	[identity,bounded,namespace]	bounded	bounded
hbase.wal.storage.policy	[LAZY_PERSIST,ALL_SSD,ONE_SSD,HOT,WARM,COLD]	HOT	HOT

修改目的:

请输入内容

0/100

取消

确定

5. 输入完成后点击**确定**按钮。

需要注意的是，一些参数需要**重启集群**才能生效，这可能会对集群造成一定影响。我们采用了滚动重启的方式，保证集群中的大部分节点可以正常工作。

## 备份恢复

### 备份列表

#### 手动备份

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入**恢复备份**，进入**恢复备份**页面，默认选中**备份列表**页签。

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
■■■■■■■■	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	-	查看详情 删除 恢复

3. 点击备份列表上方的**手动备份**按钮，弹出**手动备份**窗口，输入备份表和备注。

## 手动备份



\* 备份表:

请输入备份表

1. 如果需要备份多个表，则用英文逗号 (,) 隔开，default:table1,namesapce1:\*。\*表示所有表
2. 单独传入 \* 表示进行全量备份

备注:

请输入备注信息，最多256个字符

0/256

取消

确认

4. 点击**确认**按钮。

### 🔗 备份策略设置

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入**恢复备份**，进入**恢复备份**页面，默认选中**备份列表**页签。

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
■■■■■■■	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	- [🔗]	查看详情 删除 恢复

3. 点击备份列表上方的**备份策略设置**按钮，弹出**备份策略设置**窗口，设置**自动备份**是否开启、备份表、自动备份周期、自动备份时间及备份保留个数。

## 备份策略设置



\* 自动备份:  开启  关闭

\* 备份表:

\*

\* 自动备份周期:

周一



\* 自动备份时间:

09时



00分



\* 备份保留个数:

3

含自动备份和用户主动备份数量。如果有效备份数大于阈值，系统将删除最早的有效备份。

取消

确认

4. 点击**确认**按钮。

## 数据恢复

1. 登陆[云数据库HBase控制台](#)。

2. 点击左侧边栏中进入**恢复备份**，进入**恢复备份**页面，默认选中**备份列表**页签。

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
■■■■■	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	-	<a href="#">查看详情</a> <a href="#">删除</a> <a href="#">恢复</a>

3. 点击备份列表上方的**数据恢复**按钮，切换**数据恢复**页面。

### 恢复方式

1. 库表恢复，每次最多可选择 5 万个库表。  
 2. 目标库如果存在一样的表，则恢复过程中，该表不可访问，数据并被覆盖，有丢数据风险。  
 3. 如果您的目标恢复库（或者表）不在当前实例的备份集，无法恢复成功。

恢复位置： 恢复到原实例  恢复到其它实例

选择实例：

恢复方式：

备份集：

---

### 恢复详情

恢复所有库/表： 是  否

已选择的库表总大小：

4. 在数据恢复页面中，恢复方式中可以选择恢复位置、恢复方式（默认按备份集）、备份集；恢复详情：恢复所欲哦库/表（是/否）。

5. 完成后，点击**确定**按钮。

## 查看详情

1. 登陆[云数据库HBase控制台](#)。

2. 点击左侧边栏中进入**恢复备份**，进入恢复备份页面，默认选中**备份列表**页签。

备份列表 恢复列表

备份空间总量：800 GB，已使用空间大小：0 bytes，有效备份：1/3. C

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
■■■■■■■■	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	- <a href="#">🔗</a>	<a href="#">查看详情</a> <a href="#">删除</a> <a href="#">恢复</a>

3. 点击**查看详情**。

.6 (hb-ipszb3) ● 运行中

备份列表 恢复列表

备份空间总量：800 GB，已使用空间大小：0 bytes，有效备份：1/3. C

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
backup-LQQnVC	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	- <a href="#">🔗</a>	<input checked="" type="button" value="查看详情"/> <a href="#">删除</a> <a href="#">恢复</a>

10条/页 <  > 跳转至

4. 弹出**备份详情页**。

### 备份详情 ×

备份ID: backup-LQQnvC      备份状态: 已完成      备份开始时间: 2023-10-19 18:18:30

备份大小: 0 bytes      备份结果: 成功      备份结束时间: 2023-10-19 18:19:04

**备份信息:**

是否备份所有表: 是

备份表信息: 无

备份描述: 无

备份失败原因: 无

### 🔗 删除备份

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入恢复备份，进入恢复备份页面，默认选中备份列表页签。

备份列表    恢复列表

手动备份    数据恢复    备份策略设置    备份空间总量: 800 GB, 已使用空间大小: 0 bytes, 有效备份: 1/3. 🔄

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
██████████	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	- 🔗	查看详情   删除   恢复

3. 在数据列表页，点击删除。

备份列表    恢复列表

手动备份    数据恢复    备份策略设置    备份空间总量: 800 GB, 已使用空间大小: 0 bytes, 有效备份: 1/3. 🔄

备份ID	备份模式	备份状态	备份结果	备份开始时间	备份结束时间	备份大小	备份描述	操作
██████████	手动备份	● 已完成	● 成功	2023-10-19 18:18:30	2023-10-19 18:19:04	0 bytes	- 🔗	查看详情 <span style="border: 2px solid red; padding: 2px;">删除</span> 恢复

4. 弹出删除备份对话框。

!

## 删除备份

×

确定删除备份: ██████████ ██████████ ， 删除的备份无法恢复。

取消
确认

### 🔗 恢复列表

#### 恢复数据列表

1. 登陆[云数据库HBase控制台](#)

2. 点击左侧边栏中进入**恢复备份**，进入恢复备份页面，默认选中**备份列表**页签，切换到**恢复列表**页签，可查看恢复历史记录。

恢复记录ID	备份所属实例ID	备份ID	恢复状态	恢复结果	恢复开始时间	恢复结束时间	操作
██████████	██████████	██████████	● 已完成	● 成功	2023-10-19 18:40:58	2023-10-19 18:41:29	<a href="#">查看详情</a>

## 查看详情

1. 登陆[云数据库HBase控制台](#)
2. 点击左侧边栏中进入**恢复备份**，进入恢复备份页面，默认选中**备份列表**页签，切换到**恢复列表**页签。

恢复记录ID	备份所属实例ID	备份ID	恢复状态	恢复结果	恢复开始时间	恢复结束时间	操作
██████████	██████████	██████████	● 已完成	● 成功	2023-10-19 18:40:58	2023-10-19 18:41:29	<a href="#">查看详情</a>

3. 点击**查看详情**，弹出恢复详情对话框。

## 恢复详情

恢复记录ID: restore-QlpLIH      备份ID: backup-LQQnvc      备份开始时间: 2023-10-19 18:40:58

源端实例ID: hb-ipszb3      恢复状态: 已完成      备份结束时间: 2023-10-19 18:41:29

目标端实例ID: hb-ipszb3      恢复结果: 成功

### 恢复信息:

是否恢复所有表: 是

恢复表信息: 无

恢复失败原因: 无

## 计费变更

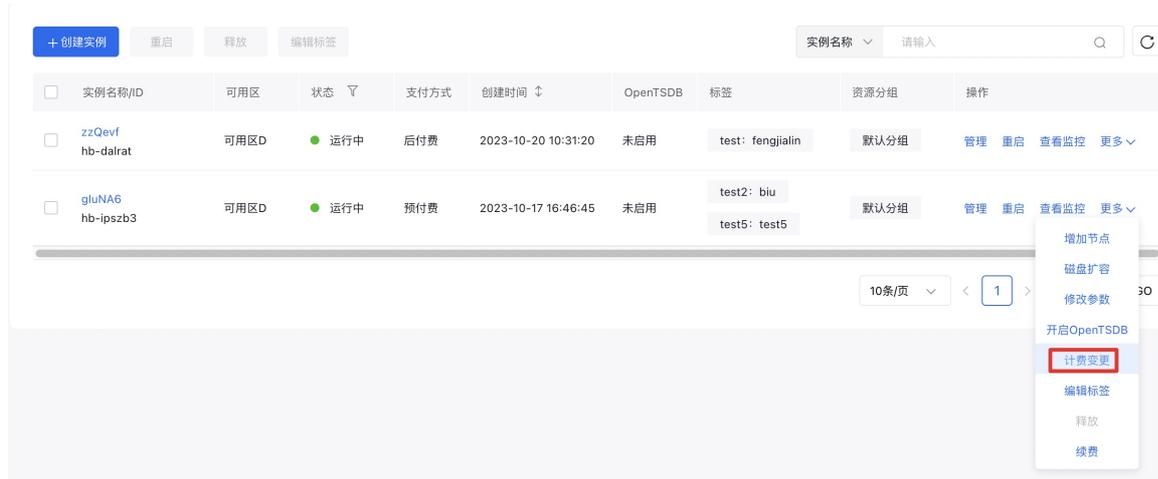
### ☞ 预付费转后付费

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入**实例管理**，进入实例管理页面。

实例名称/ID	可用区	状态	支付方式	创建时间	OpenTSDB	标签	资源分组	操作
<input type="checkbox"/> hb-dalrat	可用区D	● 运行中	后付费	2023-10-20 10:31:20	未启用	test: fengjialin	默认分组	<a href="#">管理</a> <a href="#">重启</a> <a href="#">查看监控</a> <a href="#">更多</a>
<input type="checkbox"/> hb-ipszb3	可用区D	● 运行中	预付费	2023-10-17 16:46:45	未启用	test2: biu test5: test5	默认分组	<a href="#">管理</a> <a href="#">重启</a> <a href="#">查看监控</a> <a href="#">更多</a>

10条/页 < 1 > 跳转至  GO

## 3. 点击实例列表操作列，更多-计费变更。



## 4. 弹出计费变更-预付费转后付费页面，选择购买预付费时长。



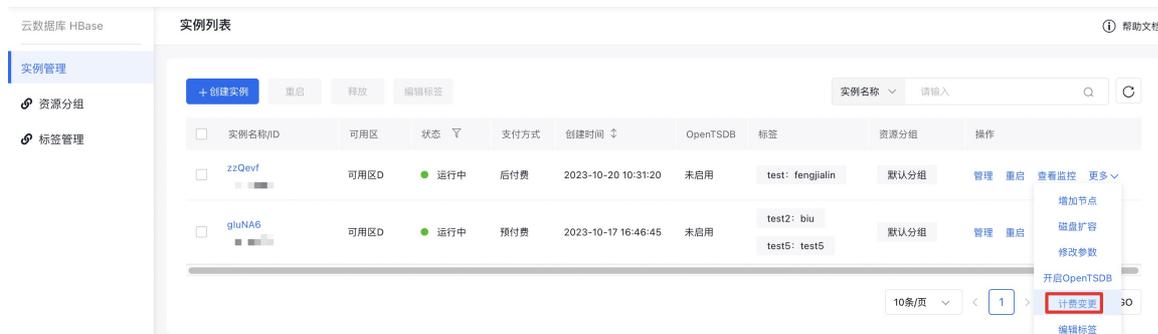
## 5. 点击下一步按钮，完成后续付费操作。

## 后付费转预付费

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入实例管理，进入实例管理页面。



## 3. 点击实例列表操作列，更多-计费变更。



## 4. 弹出计费变更-后付费转预付费页面，选择购买预付费时长。

计费变更-后付费转预付费 1 计费变更 > 2 确认订单 > 3 订单支付 > 4 变更成功

预付费购买时长: 按月 1个月

ID	产品名称	变更前单价	变更前计费方式	时长	变更后单价	变更后计费方式	变更后到期时间	金额
hb-dalrat	云数据库HBase HBASE	¥ 0.108968/分钟/台	按时长	按月 1个月	¥ 3379.00/台	包年包月	2023-11-20 11:18:50	¥ 3379.00

请确认购买时长 取消 下一步

5. 点击下一步按钮，完成后续付费操作。

## 开启OpenTSDS

### 开启OpenTSDS

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入实例管理，进入实例管理页面。

实例名称: 请输入

实例名称/ID	可用区	状态	支付方式	创建时间	OpenTSDS	标签	资源分组	操作
hb-dalrat	可用区D	运行中	后付费	2023-10-20 10:31:20	未启用	test: fengjialin	默认分组	管理 重启 查看监控 更多
hb-ipszb3	可用区D	运行中	预付费	2023-10-17 16:46:45	未启用	test2: bliu test5: test5	默认分组	管理 重启 查看监控 更多

10条/页 < 1 > 跳转至  GO

3. 点击实例列表操作列，更多-OpenTSDS。

实例列表

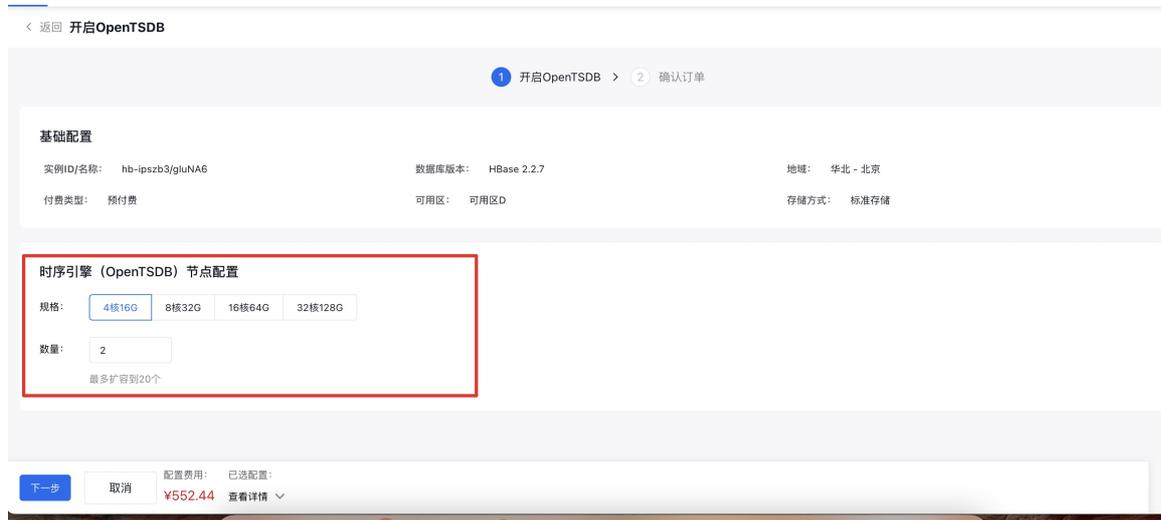
实例名称: 请输入

实例名称/ID	可用区	状态	支付方式	创建时间	OpenTSDS	标签	资源分组	操作
zzQevf	可用区D	运行中	后付费	2023-10-20 10:31:20	未启用	test: fengjialin	默认分组	管理 重启 查看监控 更多
gluNA6	可用区D	运行中	预付费	2023-10-17 16:46:45	未启用	test2: bliu test5: test5	默认分组	管理 重启 查看监控 更多

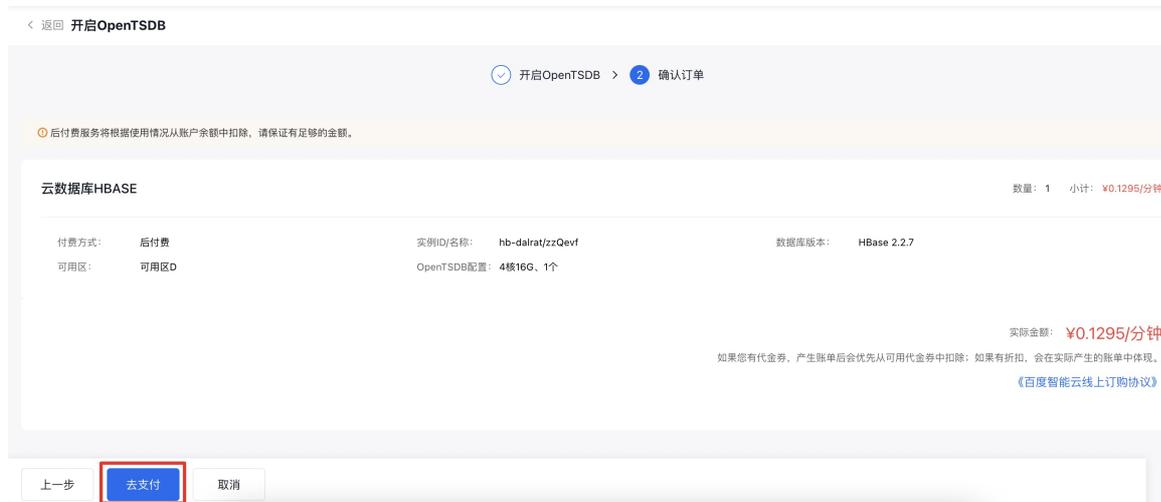
10条/页 < 1 > 跳转至  GO

- 增加节点
- 磁盘扩容
- 修改参数
- 开启OpenTSDS**
- 计费变更
- 编辑标签
- 释放
- 续费

4. 切换到开启OpenTSDS页面，选择规格、数量。



5. 点击下一步按钮，切换到支付页面，点击去支付按钮。



## 查看参数修改历史

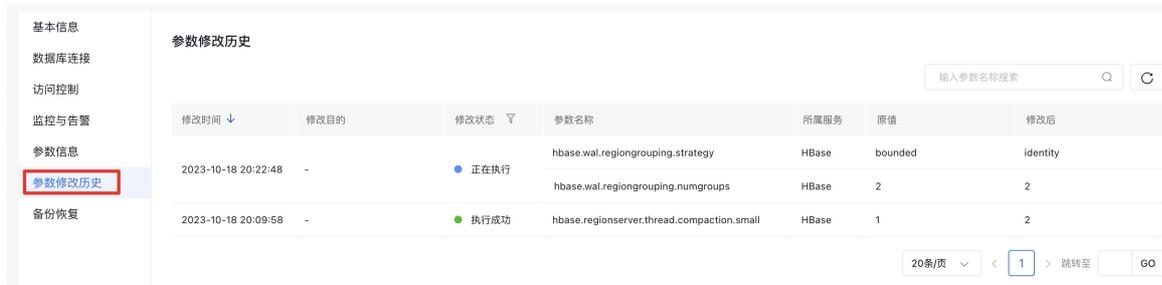
🔗 查看参数修改历史

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入“参数修改历史”，进入[参数修改历史](#)页面。



🔗 搜索参数修改历史

1. 登陆[云数据库HBase控制台](#)。
2. 点击左侧边栏中进入“参数修改历史”，进入[参数修改历史](#)页面。



3.在列表有右上方文本框中输入需要的参数值。



4.点击**搜索**按钮，搜索结果展示在列表页。



### 参数列表

目前开放配置的参数列表如下。需要注意的是，可配置参数范围可能在后续更新中进行修改，具体以控制台为准。

参数名	参数描述
dfs.datanode.max.transfer.threads	指定用于将数据传入和传出DN的最大线程数。
hbase.dfs.client.read.shortcircuit.buffer.size	如果DFSCli配置dfs.client.read.shortcircuit.buffer.size未设置，我们将使用此处配置的内容作为短路读取默认直接字节缓冲区大小。DFSCli本地默认值为1MB；HBase保持其HDFS文件的打开状态，因此由于直接内存不足，文件块的数量 * 1MB很快开始增加并可能OOM。因此，我们将其设置为默认值。使其>HColumnScript中设置的默认hbase block大小，通常为64k
hbase.quota.enabled	启用限流
hbase.mob.cache.evict.period	MOB文件在缓存中的存活时间，默认值为3600秒
hbase.mob.file.cache.size	最大缓存打开文件句柄数。设置一个更大的值使得每个mob缓存能够存储更多的文件句柄，并将减少频繁的文件打开和关闭，从而提升读性能。但是，如果设置得太高，可能会导致“打开的文件处理程序太多”。默认值为1000
hbase.mob.cache.evict.remaining.ratio	当缓存的mob文件数超过hbase.mob.file.cache.size时，将触发逐出后仍缓存的文件的比率（介于0.0和1.0之间）。默认值为0.5f
hbase.hstore.compaction.throughput.higher.bound	聚合压缩吞吐量的目标上限（字节/秒）。允许您在PressureAwareCompactionThroughputController吞吐量控制器处于激活状态时（默认为激活状态）控制聚合压缩吞吐量需求。当Compaction压力在[0.0, 1.0]范围内时，最大吞吐量将在下限和上限之间进行调整。如果压实压力为1.0或更大，则将忽略上限，直到压力恢复到正常范围

hbase.hstore.compaction.throughput.lower.bound	聚合压缩吞吐量的目标下限（字节/秒）。允许您在PressureAwareCompactionThroughputController吞吐量控制器处于激活状态时（默认为激活状态）控制聚合压缩吞吐量需求
hbase.hstore.compaction.throughput.of.fpeak	在非高峰时段，使用固定的吞吐量限制
hbase.offpeak.start.hour	非高峰时段的开始时间，表示为0到23之间的整数（包括0和23）。设置为-1可禁用非峰值
hbase.offpeak.end.hour	非高峰时段的结束时间，表示为0到23之间的整数（包括0和23）。设置为-1可禁用非峰值
hbase.hstore.compaction.throughput.tune.period	限流调整周期,单位毫秒
hbase.regionserver.thread.compaction.large	large compaction线程数量
hbase.regionserver.thread.compaction.small	small compaction线程数量
hbase.coprocessor.enabled	启用或禁用协处理器加载。如果“false”（禁用），将忽略任何其他与协处理器相关的配置
hbase.coprocessor.user.enabled	启用或禁用用户（又名table）协处理器加载。如果“false”（禁用），则将忽略表描述符中的任何表协处理器属性。如果“hbase.coprocessor.enabled”为“false”，则此设置无效
hbase.coprocessor.region.classes	默认情况下在所有表上加载的区域观察器或端点协处理器的逗号分隔列表。对于任何重写协处理器方法，这些类将被顺序调用。实现自己的协处理器后，将其添加到HBase的类路径中，并在此处添加完全限定的类名。还可以通过设置HTableDescriptor或HBase shell按需加载协处理器
hbase.coprocessor.master.classes	一个逗号分隔的org.apache.hadoop.hbase.coprocessor.MasterObserver协处理器列表，默认加载到active HMaster进程上。对于任何重写协处理器方法，这些类将被顺序调用。在实现自己的MasterObserver之后，只需将其放入HBase的类路径中，并在此处添加完全限定的类名
hbase.coprocessor.regionserver.classes	由region server加载的协处理器的逗号分隔列表
hbase.master.loadbalancer.class	用于执行负载均衡的类
hbase.hstore.flusher.count	flush线程数。如果线程不足，Memstore flush操作将被加入等待队列。如果线程足够，memstore flush操作可以并行执行，增加HDFS读取速率，同时可能会增加compaction次数

hbase.region.replication.enabled	是否启用到Secondary region replica。我们有一个单独的实现在不使用通用的集群间复制框架的WAL复制，因此现在我们不会添加任何replication peer
hbase.region.server.storefile.refresh.period	刷新secondary region的存储文件的时间段（以毫秒为单位）。0表示此功能已禁用。一旦secondary region刷新region中的文件列表，secondary region就会看到来自primary region的新文件（来自flush和compaction）。但由于频繁的flush可能会导致额外的Namenode压力。如果文件刷新时间超过HFile TTL（hbase.master.hfilecleaner.TTL），则拒绝请求。建议使用此设置将HFile TTL配置为更大的值
hbase.wal.provider	使用配置设置“hbase.wal.provider”配置使用哪个wal提供程序
hbase.wal.regiongrouping.strategy	region分组策略，每个region group将会拥有一个WAL
hbase.wal.regiongrouping.boundednumgroups	bounded分组策略下，region group的最大数量
hbase.master.logcleaner.ttl	WAL被主线程清理前在archive目录中能够保留的时间长度。该值以毫秒为单位
hbase.master.hfilecleaner.ttl	HFile被主线程清理前在archive目录中能够保留的时间长度。该值以毫秒为单位

## API参考

### API简介

#### 概述

欢迎使用百度智能云数据库HBase。您可以使用本文档介绍的API对云数据库HBase服务进行灵活的操作。

如果您是初次调用百度智能云产品的API，可以观看[API入门视频指南](#)，快速掌握调用API的方法。

云数据库HBase API 提供下列接口类型：

接口类型	描述
实例管理相关接口	通过实例相关接口对实例进行日常管理，包括：创建实例，改配实例，获取实例列表，实例详情，重启实例，释放实例，修改实例名称等
白名单相关接口	通过白名单相关接口，可以完成对实例的访问控制操作

#### API服务域名

云数据库HBase API的服务域名包括：

Region	EndPoint	Protocol
北京	hbase.bj.baidubce.com	HTTP and HTTPS
广州	hbase.gz.baidubce.com	HTTP and HTTPS
苏州	hbase.su.baidubce.com	HTTP and HTTPS
保定	hbase.bd.baidubce.com	HTTP and HTTPS

## 调用说明

### API认证机制

用户可以通过两种方式与百度智能云进行交互，包括认证方式和匿名方式。对于认证方式，需要通过使用Access Key Id / Secret Access Key加密的方法来验证某个请求的发送者身份。Access Key Id (AK) 用于标示用户，Secret Access Key (SK) 是用户用于加密认证字符串和百度智能云用来验证认证字符串的密钥，其中SK必须保密，只有用户和百度智能云知道。

当百度智能云接收到用户的请求后，系统将使用相同的SK和同样的认证机制生成认证字符串，并与用户请求中包含的认证字符串进行比对。如果认证字符串相同，系统认为用户拥有指定的操作权限，并执行相关操作；如果认证字符串不同，系统将忽略该操作并返回错误码。鉴权认证机制的详细内容请参见[鉴权认证机制](#)。

### 排版约定

排版格式	含义
< >	变量
[ ]	可选项
{ }	必选项
	互斥关系

### 时间与日期

日期与时间的表示有多种方式。为统一起见，除非是约定俗成或者有相应规范的，凡需要日期时间表示的地方一律采用UTC时间，遵循ISO 8601，并做以下约束：

1. 表示日期一律采用YYYY-MM-DD方式，例如2016-06-01表示2016年6月1日。
2. 表示时间一律采用hh:mm:ss方式，并在最后加一个大写字母Z表示UTC时间。例如23:00:10Z表示UTC时间23点0分10秒。
3. 凡涉及日期和时间合并表示时，在两者中间加大写字母T，例如2016-06-01T23:00:10Z表示UTC时间2016年6月1日23点0分10秒。

### 通用说明

API调用遵循HTTP协议，各Region采用不同的域名，具体域名为hbase.{region}.baidubce.com。数据交换格式为JSON，所有request/response body内容均采用UTF-8编码。

### 请求参数

请求参数包括如下4种：

参数类型	说明
URI	通常用于指明操作实体,如:POST /v{version}/instance/{instanceId}
Query参数	URL中携带的请求参数
HEADER	通过HTTP头域传入,如:x-bce-date
RequestBody	通过JSON格式组织的请求数据体

### 返回值

返回值分为两种形式：

返回内容	说明
HTTP STATUS CODE	如200,400,403,404等
ResponseBody	JSON格式组织的响应数据体

## API版本号

参数	类型	参数位置	描述	是否必须
version	string	URL参数	API版本号, 当前API版本为v1	是

## 认证机制

所有API的安全认证一律采用Access Key与请求签名机制。Access Key由Access Key ID和Secret Access Key组成, 均为字符串。对于每个HTTP请求, 使用下面所描述的算法生成一个认证字符串。提交认证字符串放在Authorization头域里。服务端根据生成算法验证认证字符串的正确性。认证字符串的格式为**bce-auth-v{version}/{accessKeyId}/{timestamp}/{expirationPeriodInSeconds}/{signedHeaders}/{signature}**。

- version是正整数。
- timestamp是生成签名时的UTC时间。
- expirationPeriodInSeconds表示签名有效期限。
- signedHeaders是签名算法中涉及到的头域列表。头域名之间用分号 (;) 分隔, 如host;x-bce-date。列表按照字典序排列。(本API签名仅使用host和x-bce-date两个header)。
- signature是256位签名的十六进制表示, 由64个小写字母组成。

## 密码加密传输规范定义

所有涉及密码的接口参数都需要加密, 禁止明文传输。密码一律采用AES 128位加密算法进行加密, 用SK的前16位作为密钥, 加密后生成的二进制字节流需要转成十六进制, 并以字符串的形式传到服务端。具体步骤如下:

- byte[] bCiphertext= AES(明文,SK)。
- String strHex = HexStr(bCiphertext)。

## 幂等性

当调用创建接口时如果遇到了请求超时或服务器内部错误, 用户可能会尝试重发请求, 这时用户通过clientToken参数避免创建出比预期要多的资源, 即保证请求的幂等性。

幂等性基于clientToken, clientToken是一个长度不超过64位的ASCII字符串, 通常放在query string里, 如<http://hbase.bj.baidubce.com/v1/instance?clientToken=be31b98c-5e41-4838-9830-9be700de5a20>。

如果用户使用同一个clientToken值调用创建接口, 则服务端会返回相同的请求结果。因此用户在遇到错误进行重试的时候, 可以通过提供相同的clientToken值, 来确保只创建一个资源; 如果用户提供了一个已经使用过的clientToken, 但其他请求参数(包括queryString和requestBody) 不同甚至url Path不同, 则会返回IdempotentParameterMismatch的错误代码。

clientToken的有效期为24小时, 以服务端最后一次收到该clientToken为准。也就是说, 如果客户端不断发送同一个clientToken, 那么该clientToken将长期有效。

## 错误返回

### 错误返回格式

请求发生错误时通过response body返回详细错误信息, 遵循如下格式:

参数名	类型	说明
code	string	错误码
message	string	错误描述
requestId	string	本次请求的requestId

示例：

```
{
  "requestId": "ae2225f7-1c2e-427a-a1ad-5413b762957d",
  "code": "AccessDenied",
  "message": "Access denied."
}
```

## 错误返回码

错误码	错误描述	HTTP状态码	中文解释
AccessDenied	Access denied.	403	禁止操作
InvalidAction	Invalid Action.	400	非法操作
InternalFailure	We encountered an internal error. Please try again.	400	内部错误
ValidationError	Validation Error.	400	校验失败
InvalidVersion	The API version specified was invalid.	400	非法版本号
MissingDateHeader	Request must have a "date" or "x-bce-date" header.	400	Header中缺少"date"或"x-bce-date"
MissingAuthToken	Request must have a "authorization" header.	400	Header中缺少"authorization"
RequestExpired	Request has expired.	400	请求中authorization过期
ServiceInternalError	Internal service occurs error.	500	内部（未知）服务器错误
PaymentFailed	payment failed, please check remaining balance	403	支付失败
InsufficientBalance	Insufficient balance, please check remaining balance	403	支付失败，余额不足
NoSuchObject	The specified object is not found or resource do not exist.	404	资源不存在
OperationDenied	No permission to accessor or permission error.	403	您没有权限做此操作
OperationDenied	Operation not available.	409	资源当前状态不允许您做此操作
BadRequest	Bad request parameters or illegal request.	400	参数输入不合法

## 公共头

### 公共请求头

公共头部	描述
Authorization	包含Access Key与请求签名
Content-Type	application/json; charset=utf-8
x-bce-date	表示日期的字符串，符合API规范
Host	表示请求API的域名
x-bce-content-sha256	表示内容部分的SHA256签名的十六进制字符串。这里内容指HTTP Request Payload Body。即Content部分在被HTTP encode之前的原始数据

HTTP协议的标准头域不再这里列出。公共头域将在每个云数据库 HBase API中出现，是必需的头域，其中x-bce-content-sha256头域只出现在POST和PUT请求中。POST、PUT、DELETE等请求数据放在request body中。

### 公共响应头

公共头部	描述
Content-Type	application/json; charset=utf-8
x-bce-request-id	云数据库 HBase 后端生成，并自动设置到响应头域中

## 实例相关接口

### 查询实例列表

#### 说明

- 查询所属于本账号的实例列表。

#### 请求结构

```
GET /v{version}/instance?marker={marker}&maxKeys={maxKeys}&name={name}&status={status} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

#### 请求头域

除公共头域外，无其它特殊头域。

#### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
marker	String	否	Query参数	批量获取列表的查询的起始位置。
maxKeys	Integer	否	Query参数	每页包含的最大数量，最大数量通常不超过1000，缺省值为10。
name	String	否	Query参数	实例名称
status	String	否	Query参数	实例状态

#### 返回头域

除公共头域，无其它特殊头域。

#### 返回参数

参数名称	类型	描述
marker	String	批量获取列表的查询的起始位置。
maxKeys	Integer	每页包含的最大数量。
isTruncated	Boolean	true表示后面还有数据，false表示已经是最后一页
nextMarker	String	获取下一页所需要传递的marker值。当isTruncated为false时，该域不出现
result	List<InstanceModel>	实例列表。

#### 请求示例

```
GET /v1/instance?marker=-1&maxKeys=1&name=P9cZKQ&status=Running HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

#### 返回示例

HTTP/1.1 200 OK

x-bce-request-id: 7869616F-7A68-6977-656E-406261696475

Content-Type: application/json

```
{
  "result": [
    {
      "instanceId": "hb-ABC",
      "uuid": "0946b67d-ABC",
      "name": "P9cZKQ",
      "status": "Running",
      "zkNodeSpec": "c1m2",
      "masterNodeSpec": "c4m8",
      "coreNodeCount": 2,
      "coreNodeSpec": "c4m8",
      "coreNodeDiskType": "cloud_hp1",
      "coreNodeDiskSizeInGB": 400,
      "tsdbProxyCount": 1,
      "tsdbProxySpec": "c4m16",
      "coldStorageDir": null,
      "vpclId": "vpc-bnABC",
      "orderStatus": null,
      "tags": null,
      "resourceGroups": [
        {
          "name": "默认分组",
          "extra": "",
          "parentUuid": "",
          "groupId": "RESG-wHABC",
          "accountId": "ABC",
          "userId": "ABC",
          "createTime": 1614157938000,
          "updateTime": 1614157938000,
          "deleteTime": null,
          "bindTime": "2022-12-19T06:12:36Z"
        }
      ],
      "vpc": {
        "vpclId": "vpc-bnABC",
        "name": "hbase-vpc",
        "cidr": "192.168.0.0/16",
        "vpcUuid": "3f7ce9f8-ABC"
      },
      "resourceUuid": "9a4dc0e9-ABC",
      "task": null,
      "isAutoRenew": false,
      "subnets": [
        {
          "zoneName": "cn-bj-b",
          "subnetId": "sbn-8wABC",
          "name": "hbase-subnet",
          "cidr": "192.168.0.0/24",
          "subnetUuid": "3455e618-ABC"
        }
      ],
      "createTime": "2022-12-19T05:52:09Z",
      "expireTime": null,
      "paymentTiming": "Postpaid"
    }
  ],
  "marker": "-1",
  "isTruncated": false,
  "nextMarker": null,
}
```

```
"maxKeys": 1
}
```

## 改配实例 - 更改TSDB节点规格

更改指定实例的TSDB节点规格。

### 说明

- 只有实例Running状态时才可以进行变配操作。
- 异步接口，可通过调用实例详情接口查看实例状态。

### 请求结构

```
PUT /v{version}/instance/{instanceId}?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "tsdbProxySpec": tsdbProxySpec
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
clientToken	String	是	Query参数	幂等性Token，是一个长度不超过64位的ASCII字符串。
instanceId	String	是	URL参数	指定实例ID。
tsdbProxySpec	String	是	RequestBody参数	改配后实例的tsdb节点规格类型。

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
orderId	String	订单号

### 请求示例

```
PUT /v1/instance/hb-yft8ra?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "tsdbProxySpec": "c4m8"
}
```

### 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 100
{
  "orderId": "cdmkdafk-cdkal-nfj3rj-kse3d"
}

```

## 改配实例 - 添加TSDB节点

为指定实例添加TSDB节点。

### 说明

- 只有实例Running状态时才可以进行变配操作。
- 异步接口，可通过调用实例详情接口查看实例状态。

### 请求结构

```

PUT /v{version}/instance/{instanceId}?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "tsdbProxyCount": tsdbProxyCount
}

```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	Integer	是	URL参数	API版本号
clientToken	String	是	Query参数	幂等性Token，是一个长度不超过64位的ASCII字符串。
instanceId	String	是	URL参数	指定实例ID
tsdbProxyCount	Integer	是	RequestBody参数	改配后实例的tsdb节点数量

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
orderId	String	订单号

### 请求示例

```

PUT /v1/instance/hb-yft8ra?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "tsdbProxyCount": 4
}

```

## 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 100
{
  "orderId":"cdmkdafk-cdkal-nfj3rj-kse3d"
}

```

## 改配实例 - 更改Core节点磁盘大小

更改指定实例的Core节点磁盘大小。

### 说明

- 只有实例Running状态时才可以进行变配操作。
- 异步接口，可通过调用实例详情接口查看实例状态。
- 磁盘扩容单次增加至少60GB。
- 本地盘实例不能进行磁盘扩容。

### 请求结构

```

PUT /v{version}/instance/{instanceId}?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "coreNodeDisks": [{
    "diskType": "diskType",
    "diskSizeInGB": diskSizeInGB
  }]
}

```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
clientToken	String	是	Query参数	幂等性Token，是一个长度不超过64位的ASCII字符串。
instanceId	String	是	URL参数	指定实例Id
coreNodeDisks	List<Disk>	是	RequestBody参数	Core节点磁盘信息

Disk :

参数名称	参数类型	参数描述
diskType	String	节点磁盘类型
diskSizeInGB	Integer	节点磁盘大小

### 返回头域

除公共头域，无其它特殊头域。

## 返回参数

参数名称	类型	描述
orderId	String	订单号

## 请求示例

```
PUT /v1/instance/hb-yft8ra?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "coreNodeDisks": [{
    "diskType": "enhanced_ssd_pl1",
    "diskSizeInGB": 460
  }]
}
```

## 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 100
{
  "orderId": "cdmkdafk-cdkal-nfj3rj-kse3d"
}
```

## 改配实例 - 添加Core节点

为指定实例添加Core节点。

### 说明

- 只有实例Running状态时才可以进行变配操作。
- 异步接口，可通过调用实例详情接口查看实例状态。
- 节点数量单次最多扩容20个。

### 请求结构

```
PUT /v{version}/instance/{instanceId}?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "coreNodeCount": coreNodeCount
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
clientToken	String	是	Query参数	幂等性Token，是一个长度不超过64位的ASCII字符串。
instanceId	String	是	URL参数	指定实例Id
coreNodeCount	Integer	是	RequestBody参数	改配后实例的core节点个数

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
orderId	String	订单号

### 请求示例

```
PUT /v1/instance/hb-yft8ra?resize&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "coreNodeCount": 3
}
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 100
{
  "orderId": "cdmkdafk-cdkal-nfj3rj-kse3d"
}
```

## 更改实例名称

### 请求结构

```
PUT /v{version}/instance/{instanceId}?modifyName HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "instanceName": instanceName
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	指定实例Id
instanceName	String	是	RequestBody参数	实例名，要求： 1) 支持大小写字母、数字、汉字以及-_/等特殊字符，必须以字母开头； 2) 长度限制为1~65。

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

无特殊返回参数

### 请求示例

```
PUT /v1/instance/hb-qNIWTz?modifyName HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "instanceName": "myHBase"
}
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 0
```

## 创建实例

### 前提条件

创建实例需要实名认证，没有通过实名认证的可以前往百度智能云官网控制台中的安全认证下的实名认证中进行认证。

### 说明

- 本接口用于创建相同配置的云数据库 HBase 实例，目前仅支持一次创建一个。
- 创建后付费方式的实例需要账户现金余额大于等于100元。
- 创建接口为异步创建，可通过查询指定实例详情接口查询实例状态。

### 请求结构

```
POST /v{version}/instance?clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "billing":{
    "paymentTiming": paymentTiming,
    "reservation": {
      "reservationLength": reservationLength,
      "reservationTimeUnit": "Month"
    }
  },
  "name": name,
  "engine":engine
  "engineVersion": engineVersion,
  "zkNodeSpec": zkNodeSpec,
  "masterNodeSpec": masterNodeSpec,
  "coreNodeCount": coreNodeCount,
  "coreNodeSpec": coreNodeType,
  "coreNodeDiskType": coreNodeDiskType,
  "coreNodeDiskSizeInGB": coreNodeDiskSizeInGB,
  "coldStorageDir": coldStorageDir,
  "tsdbProxyCount": tsdbProxyCount,
  "tsdbProxySpec": tsdbProxySpec,
  "vpclId": vpclId,
  "tags": [{
    "tagKey": "testKey",
    "tagValue": "testValue"
  }],
  "subnets":[
    {
      "zoneName": zoneName,
      "subnetId": subnetId
    }
  ],
  "resGroupId": "resGroupId"
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必选	参数位置	描述
version	String	是	URL参数	API版本号
clientToken	String	是	Query参数	幂等性Token，是一个长度不超过64位的ASCII字符串。
billing	<a href="#">Billing</a>	是	RequestBody参数	订单、计费相关参数。
name	String	否	RequestBody参数	实例名称
engine	String	是	RequestBody参数	引擎："hbase"
engineVersion	String	是	RequestBody参数	引擎版本："2.0"
zkNodeSpec	String	是	RequestBody参数	ZK节点规格类型
masterNodeSpec	String	是	RequestBody参数	Master节点规格类型
coreNodeCount	Integer	是	RequestBody参数	Core节点数量
coreNodeSpec	String	是	RequestBody参数	Core节点规格类型
coreNodeDiskType	String	否	RequestBody参数	Core节点的磁盘类型
coreNodeDiskSizeInGB	Integer	否	RequestBody参数	Core节点的磁盘大小
coldStorageDir	String	否	RequestBody参数	冷存储的存储目录，举例： bos://bucket/object_prefix/
tsdbProxyCount	Integer	否	RequestBody参数	tsdb proxy节点数量
tsdbProxySpec	String	否	RequestBody参数	tsdb proxy节点规格
vpcl	String	否	RequestBody参数	如果不提供，则选择默认vpc
subnets	List<SubnetModel>	否	RequestBody参数	如果不提供，则选择默认子网
resGroupId	String	否	RequestBody参数	所属资源组ID
tags	List<TagModel>	否	RequestBody参数	实例所属标签

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
orderId	String	订单ID
instances	List<InstanceIdView>	新建的实例的ID

#### InstanceIdView

参数名称	参数类型	参数描述
instanceUuid	String	实例UUID
instanceId	String	实例ID

#### 请求示例

```
POST /v1/instance?clientToken=be31b98c-5e41-4838-9830-9be700de5a20 HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "engine": "hbase",
  "engineVersion": "2.0",
  "coreNodeDiskType": "cloud_hp1",
  "coreNodeDiskSizeInGB": 400,
  "coreNodeCount": 2,
  "zkNodeSpec": "c1m2",
  "masterNodeSpec": "c4m8",
  "vpclId": "vpclId",
  "coreNodeSpec": "c4m16",
  "billing": {
    "paymentTiming": "Prepaid",
    "reservation": {
      "reservationLength": 1,
      "reservationTimeUnit": "Month"
    },
    "autoRenew": {
      "autoRenewTimeUnit": "Month",
      "autoRenewLength": 1
    }
  },
  "tags": [{
    "tagKey": "testKey",
    "tagValue": "testValue"
  }],
  "subnets": [{
    "zoneName": "zoneE",
    "subnetId": "subnetId"
  }],
  "resGroupId": "resGroupId"
}
```

#### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
{
  "orderId": "1cbe9770a8d24afabfbb719997d98d9c",
  "instances": [
    {
      "instanceUuid": "34412f60-5362-4a48-8346-8a1624a93d8d",
      "instanceId": "hb-uxhfnr"
    }
  ]
}
```

## 查询实例详情

### 说明

- 查询指定实例的详细信息

### 请求结构

```
GET /v{version}/instance/{instanceId} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	指定实例Id

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
instanceId	String	实例ID
uuid	String	实例uuid
name	String	实例名称
zkConnAddrs	List<ConnAddrs>	zk链接信息
engine	String	引擎
engineVersion	String	引擎版本
zkNodeSpec	String	ZK节点规格类型
masterNodeSpec	String	Master节点规格类型
coreNodeCount	Integer	Core节点数量
coreNodeSpec	String	Core节点规格类型
coreNodeDiskType	String	Core节点的磁盘类型
coreNodeDiskSizeInGB	Integer	Core节点的磁盘大小
coldStorageDir	String	冷存储的存储目录
status	String	实例状态
masterNodes	List<NodeModel>	Master节点信息
coreNodes	List<NodeModel>	Core节点信息
zkNodes	List<NodeModel>	zk节点信息
tsdbProxyCount	String	tsdb proxy节点数量
tsdbProxySpec	String	tsdb proxy节点规格。仅当tsdbProxyCount > 0时，该参数有意义。
paymentTiming	String	付费方式。 预付费：Prepaid，后付费：Postpaid.
createTime	Datetime	实例创建时间
expireTime	Datetime	实例到期时间
vpc	VpcModel	实例所属VPC
vpclId	String	实例所属VPC的短ID
subnets	List<SubnetModel>	实例所在可用区和子网。
tags	List<TagModel>	实例所属标签
resourceGroups	List<GroupTree>	实例资源所绑定的资源组
webUIAddrs	List<ConnAddr>	webui 链接信息
isAutoRenew	Boolean	true表示已设置自动续费，false为未设置

### 请求示例

```
GET /v1/instance/hb-ABC HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
```

```
Content-Type: application/json
{
  "instanceId": "hb-ABC",
  "uuid": "0946b67d",
  "name": "P9cZKQ",
  "engine": "HBase",
  "engineVersion": "2.0",
  "engineMinVersion": "2.2.7",
  "status": "Running",
  "zkNodeSpec": "c1m2",
  "masterNodeSpec": "c4m8",
  "coreNodeCount": 2,
  "coreNodeSpec": "c4m8",
  "coreNodeDiskType": "cloud_hp1",
  "coreNodeDiskSizeInGB": 400,
  "tsdbProxyCount": 1,
  "tsdbProxySpec": "c4m16",
  "coldStorageDir": null,
  "vpclId": "vpc-bni",
  "orderStatus": null,
  "tags": null,
  "resourceGroups": [
    {
      "name": "默认分组",
      "extra": "",
      "parentUuid": "",
      "groupId": "RESG-wHi",
      "accountId": "06f5",
      "userId": "06f5",
      "createTime": 1614157938000,
      "updateTime": 1614157938000,
      "deleteTime": null,
      "bindTime": "2022-12-19T06:12:36Z"
    }
  ],
  "vpc": {
    "vpclId": "vpc-bni",
    "name": "hbase-vpc",
    "cidr": "192.168.0.0/16",
    "vpcUuid": "3f7ce9"
  },
  "resourceUuid": "9a4dc",
  "webUIAdrs": [
    {
      "connAddr": "master-ABC-1.hbase.baidubce.com",
      "connAddrPort": 16010,
      "isInternal": true
    },
    {
      "connAddr": "http://test.baidu-int.com/hb-ABC/master-ABC-1/16010/master-status",
      "connAddrPort": 0,
      "isInternal": false
    }
  ],
  "task": null,
  "isAutoRenew": false,
  "zkConnAdrs": [
    {
      "connAddr": "zk-ABC-1.hbase.baidubce.com",
      "connAddrPort": 2181,
      "isInternal": false
    }
  ],
}
```

```
{
  "connAddr": "zk-ABC-2.hbase.baidubce.com",
  "connAddrPort": 2181,
  "isInternal": false
},
{
  "connAddr": "zk-ABC-3.hbase.baidubce.com",
  "connAddrPort": 2181,
  "isInternal": false
}
],
"openTSDBAddrs": [
  {
    "connAddr": "192.168.0.1",
    "connAddrPort": 4242,
    "isInternal": false
  }
],
"masterNodes": [
  {
    "nodeId": "master-ABC-1"
  },
  {
    "nodeId": "master-ABC-2"
  }
],
"coreNodes": [
  {
    "nodeId": "core-ABC-1"
  },
  {
    "nodeId": "core-ABC-2"
  }
],
"zkNodes": [
  {
    "nodeId": "zk-ABC-1"
  },
  {
    "nodeId": "zk-ABC-2"
  },
  {
    "nodeId": "zk-ABC-3"
  }
],
"tsdbProxies": [
  {
    "nodeId": "tsdb-ABC-1"
  }
],
"subnets": [
  {
    "zoneName": "zoneB",
    "subnetId": "sbn-8we8",
    "name": "hbase-subnet",
    "cidr": "192.168.0.0/24",
    "subnetUuid": "3455"
  }
],
"createTime": "2022-12-19T05:52:09Z",
"expireTime": null,
"paymentTiming": "Postpaid"
}
```

}

## 释放实例

实例释放后实例所使用的物理资源都被收回，相关数据全部丢失且不可恢复。

### 说明

- 只有付费类型为Postpaid或者付费类型为Prepaid且已过期的实例才可以释放。

### 释放实例 请求结构

```
DELETE /v{version}/instance/{instanceId} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必选	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	待删除的实例Id

**返回头域** 除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
results	List<InstanceResult>	实例操作结果

### 请求示例

```
DELETE /v1/instance/hb-uo05uu HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 0
{
  "results": [
    {
      "instanceId": "hb-uo05uu",
      "code": "Success",
      "message": null
    }
  ]
}
```

## 重启实例

## 说明

- 只有实例Running状态时才可以进行重启操作。

## 重启实例 请求结构

```
PUT /v{version}/instance?restart HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "instancelds": instancelds
}
```

## 请求头域

除公共头域外，无其它特殊头域。

## 请求参数

参数名称	类型	是否必选	参数位置	描述
version	String	是	URL参数	API版本号
instancelds	List<String>	是	RequestBody参数	待重启的实例Id

## 返回头域

除公共头域，无其它特殊头域。

## 返回参数

参数名称	类型	描述
results	List<InstanceResult>	实例操作结果

## 请求示例

```
PUT /v1/instance?restart HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "instancelds":["hb-uo05uu"]
}
```

## 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 100
{
  "results": [{
    "instanceld": "hb-uo05uu",
    "code": "Success",
    "message": null
  }]
}
```

## 白名单相关接口

### 删除白名单

为指定实例删除白名单记录。

#### 说明

- 只有实例状态是Running的实例才支持删除白名单操作。

#### 请求结构

```
PUT /v{version}/instance/{instanceId}?deleteSecurityIps HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "securityIps":[
    "ip1",
    "ip2"
  ]
}
```

#### 请求头域

除公共头域外，无其它特殊头域。

#### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	指定实例Id
securityIps	List<String>	是	RequestBody参数	待删除的白名单IP列表。

#### 返回头域

除公共头域，无其它特殊头域。

#### 返回参数

除公共头域，无其它特殊头域。

#### 请求示例

```
PUT /v1/instance/hb-qNIWTz?deleteSecurityIps HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "securityIps":[
    "192.168.0.1/24",
    "0.0.0.0/0"
  ]
}
```

#### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
```

## 添加白名单

为指定实例添加白名单记录。

### 说明

- 只能针对实例状态是Running的实例添加白名单操作。

### 请求结构

```
PUT /v{version}/instance/{instanceId}?addSecurityIps HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "securityIps":[
    "ip1",
    "ip2"
  ]
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	指定实例Id
securityIps	List<String>	是	RequestBody参数	待添加的白名单IP列表。

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

无特殊返回参数。

### 请求示例

```
PUT /v1/instance/hb-qNIWTz?addSecurityIps HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "securityIps":[
    "192.168.0.1/24",
    "0.0.0.0/0"
  ]
}
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
```

## 获取白名单列表

查询实例白名单列表。

### 说明

- 未被删除的实例都支持查询白名单操作。

### 请求结构

```
GET /v{version}/instance/{instanceId}?describeSecurityIps HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	指定实例ID

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
securityIps	List<String>	白名单分组列表

### 请求示例

```
GET /v1/instance/hb-qNIWTz?describeSecurityIps HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 50
{
  "securityIps":[
    "192.168.0.1/24",
    "0.0.0.0/0"
  ]
}

```

## 付费相关接口

### 取消预付费转后付费

为指定一个开通过**预付费转后付费**的实例关闭**转后付费**(可批量)。

#### 说明

- 仅开通过**预付费转后付费**的实例才支持取消**预付费转后付费**操作。
- 处于变配、创建期间的实例不能进行取消**预付费转后付费**。

### 取消预付费转后付费

#### 请求结构

```

PUT /v{version}/instance?cancelToPostpay&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "instanceIds": [
    "short-id"
  ]
}

```

#### 请求头域

除公共头域外，无其它特殊头域。

#### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号
instanceIds	List<string>	是	RequestBody参数	实例id

#### 返回头域

除公共头域，无其它特殊头域。

#### 返回参数

无特殊返回参数

#### 请求示例

```

PUT /v1/instance?cancelToPostpay&clientToken=be31b98c-5e41-4838-9830-9be700de5a20 HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "instancelds": [
    "hb-a1kabk"
  ]
}

```

### 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 0

```

## 后付费转预付费

使一个后付费实例的付费方式转变为预付费。

### 说明

- 只有后付费方式的实例才支持转预付费操作。
- 处于变配、创建期间的实例不能进行转预付费。
- 在用户付费后，后付费转预付费立即生效。

### 后付费转预付费

#### 请求结构

```

PUT /v{version}/instance/{instanceld}?toPrepay&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "duration": duration
}

```

#### 请求头域

除公共头域外，无其它特殊头域。

#### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号
instanceld	string	是	URL参数	实例id
duration	int	是	RequestBody参数	购买时长，取值范围为：1-9、12、24、36，单位：月。

#### 返回头域

除公共头域，无其它特殊头域。

#### 返回参数

参数名称	类型	描述
orderId	String	订单号

### 请求示例

```
PUT /v{version}/instance/hb-a1kabk?toPrepay&clientToken=be31b98c-5e41-4838-9830-9be700de5a20 HTTP/1.1
HOST: hbase.bj.baidubce.com
Content-Type: application/json
Authorization: Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "duration": 12
}
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 50
{
  "orderId": "aaaad44b0bdc4b8da95bad177d111111"
}
```

## 预付费转后付费

将指定的预付费实例的付费方式转换为后付费（在到期后转变为后付费）。

### 说明

- 仅预付费方式的实例才支持转后付费操作。
- 处于变配、创建期间的实例不能进行转后付费。
- 只有预付费到期后，开通转后付费才生效。

### 预付费转后付费

#### 请求结构

```
PUT /v{version}/instance/{instanceId}?toPostpay&clientToken={clientToken} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

#### 请求头域

除公共头域外，无其它特殊头域。

#### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	String	是	URL参数	API版本号
instanceId	String	是	URL参数	实例ID

#### 返回头域

除公共头域，无其它特殊头域。

## 返回参数

参数名称	类型	描述
orderId	String	订单号

## 请求示例

```
PUT /v{version}/instance/hb-a1kabk?toPostpay&clientToken=be31b98c-5e41-4838-9830-9be700de5a20 HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

## 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
Content-Length: 50
{
  "orderId": "aaaad44b0bdc4b8da95bad177d111111"
}
```

## 附录

[对象定义](#)

[InstanceModel](#)

参数名称	类型	描述
instanceId	string	实例ID
name	string	实例名称
engine	string	引擎
engineVersion	string	引擎版本
zkNodeSpec	string	ZK节点规格类型
masterNodeSpec	string	Master节点规格类型
coreNodeCount	int	Core节点数量
coreNodeSpec	string	Core节点规格类型
coreNodeDiskType	string	Core节点的磁盘类型
coreNodeDiskSizeInGB	int	Core节点的磁盘大小
coldStorageDir	string	冷存储的存储目录
vpId	string	实例所属VPC的短ID
status	string	实例状态
tsdbProxyCount	int	tsdb proxy节点数量
tsdbProxySpec	string	tsdb proxy节点规格。仅当tsdbProxyCount > 0时，该参数有意义。
paymentTiming	string	付费方式。 预付费：Prepaid，后付费：Postpaid.
createTime	Datetime	实例创建时间
expireTime	Datetime	实例到期时间
vpc	VpcModel	实例所属VPC
subnets	List<SubnetModel>	实例所在可用区和子网。
tags	List<TagModel>	实例所属标签
resourceGroups	List<GroupTree>	实例资源所绑定的资源组
isAutoRenew	boolean	true表示已设置自动续费，false为未设置

#### NodeModel

参数名称	类型	描述
nodeId	string	节点ID

#### ConnAddr

参数名称	类型	描述
connAddr	string	地址
connAddrPort	int	端口
isInternal	boolean	内网连接

#### VpcModel

参数名称	类型	描述
vpclId	string	vpc的id
name	string	名称
cidr	string	网段及子网掩码
vpcUuid	string	vpc的Uuid

#### SubnetModel

参数名称	参数类型	参数描述
subnetId	string	子网短ID
zoneName	string	子网所属可用区。

#### TagModel

参数名称	参数类型	参数描述
tagKey	string	标签键
tagValue	string	标签值

#### InstanceResult

参数名称	参数类型	参数描述
instanceId	string	实例id
code	string	结果代码
message	string	信息

#### 类型编码定义

##### coreNodeDiskType

编码	描述
enhanced_ssd_pl1	增强型SSD
cloud_hp1	通用型SSD
hp1	高性能云磁盘
local_nvme	本地盘

##### engine

编码	描述
HBase	HBase

##### engineVersion

编码	描述
2.0	HBase 2.x

##### engineMinVersion

编码	描述
2.2.7	HBase 2.2.7

## ☞ 状态编码定义

### ☞ InstanceStatus

编码	描述
Creating	创建中
CreateFailed	创建失败
Running	运行中
Starting	(充值/续费) 启动中
Stopping	(过期/欠费) 停机中
Expired	实例已过期或已欠费
Rebooting	实例重启中
SpecsChanging	实例改配中
SpecsChangeFailed	实例改配失败

## ☞ 订单相关

### ☞ Billing

状态	类型	描述
paymentTiming	String	付费方式。 预付费：Prepaid，后付费：Postpaid.
reservation	Reservation	保留信息，支付方式为后支付时不需要设置，预支付时必须设置
autoRenew	AutoRenew	自动续费信息，支付方式为后支付时不需要设置

### ☞ Reservation

状态	类型	描述
reservationLength	int	时长，取值范围为[1,2,3,4,5,6,7,8,9,12,24,36]
reservationTimeUnit	String	时间单位，取值范围为["Month"]，当前仅支持按月

### ☞ AutoRenew

状态	类型	描述
autoRenewLength	int	到期后续费的时长（月），取值范围为[0,1,2,3,4,5,6,7,8,9,12,24,36]。0表示关闭自动续费，否则为开启自动续费。
autoRenewTimeUnit	String	时间单位，取值范围为["month"]，当前仅支持按月

## 备份接口

### 删除实例备份

#### 说明

- 删除指定实例备份文件

#### 请求结构

```
DELETE /v{version}/instance/{instanceId}/backup/{backupId} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号
instanceId	string	是	URL参数	指定实例ID
backupId	string	是	URL参数	备份ID

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

无

### 请求示例

```
DELETE /v1/instance/hb-ABC/backup/backup-abc123 HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
```

## 查看实例备份详情

### 说明

- 查询指定实例的指定备份的详情信息

### 请求结构

```
GET /v{version}/instance/{instanceId}/backup/{backupId} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号
instanceId	string	是	URL参数	指定实例Id
backupId	string	是	URL参数	备份ID

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
id	string	备份ID
target	string	存储源
backupAllTable	boolean	是否备份所有的表
tables	string	创建备份时，指定的表信息
mode	string	备份模式，取值参见 API参考/备份接口/附录/BackupRecord_mode
status	string	备份状态，取值参见 API参考/备份接口/附录/BackupRecord_status
result	string	备份结果，取值参见 API参考/备份接口/附录/BackupRecord_result
errMsg	string	备份失败原因
startTime	string	备份开始时间，格式为yyyy-MM-ddTHH:mm:ssZ (UTC时间)
endTime	string	备份结束时间，格式为yyyy-MM-ddTHH:mm:ssZ (UTC时间)
size	int64	备份大小，单位:字节
description	string	备注信息

注：tables是用户创建备份时指定的表信息（backupAllTable=false时）。如果创建备份时设置backupAllTable=true，则备份详情中tables为空。

### 请求示例

```
GET /v1/instance/hb-ABC/backup/backup-abc123 HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
{
  "backupAllTable": true,
  "description": "",
  "endTime": "2023-02-24T02:42:12Z",
  "errMsg": "",
  "id": "backup-TcWW9N",
  "mode": "Manual",
  "result": "SUCCEED",
  "size": 10912,
  "startTime": "1970-01-01T00:00:00Z",
  "status": "FINISHED",
  "tables": "",
  "target": "bos://backup-01/hb-axv123/backup-TcWW9N/"
}

```

## 修改备份描述信息

### 说明

- 修改指定备份的描述信息
- 仅允许对有效备份（即status为FINISHED且result为SUCCEED的备份）进行此操作

### 请求结构

```

PUT /v{version}/instance/{instanceId}/backup/{backupId} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "description": "xxxx"
}

```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号，当前取值1
instanceId	string	是	URL参数	指定实例Id
backupId	string	是	URL参数	指定备份Id
description	string	否	RequestBody参数	备份描述信息

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

无

### 请求示例

```
PUT /v1/instance/hb-abc123/backup/backup-qwe123 HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-
21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "description": "xxxx"
}
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
```

## 创建备份

### 说明

- 为指定实例创建备份。
- 实例须处于运行中状态。

### 请求结构

```
PUT /v{version}/instance/{instanceId}?backup HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "backupAllTable":false,
  "tables":"default:*",
  "description": "xxx"
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号, 当前取值1
instanceId	string	是	URL参数	指定实例Id
target	string	否	RequestBody参数	备份的存储源, 不填写则备份到实例的备份空间。当前不允许用户指定target。
backupAllTable	boolean	是	RequestBody参数	是否对所有表进行备份。
tables	string	否	RequestBody参数	如果backupAllTable为false, 则需要指定备份的表。格式为namespace:table, 其中namespace为命名空间, table是表名, 支持*表示备份所有表。 例: default:table1,namesapce1:table2。 如果需要备份多个表, 则用逗号 (,) 隔开。
description	string	否	RequestBody参数	备份备注信息

### 返回头域

除公共头域, 无其它特殊头域。

### 返回参数

参数名称	类型	描述
backupId	string	备份ID

### 请求示例

```
PUT /v1/instance/hb-abc123?backup HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "target": "bos://bucket/object_prefix",
  "backupAllTable":false,
  "tables":"default:*,namesapce1:table2",
  "description": "xxx"
}
```

此请求表示对命名空间default的所有表、命名空间namesapce1的table2表进行备份。

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
{
  "backupId": "backup-xxxxxx"
}
```

### 修改备份策略

#### 说明

- 修改指定实例的备份策略

- 自动备份只支持全量备份，不支持指定表备份

### 请求结构

```
PUT /v{version}/instance/{instanceId}/backupPolicy HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "autoBackup": "ON",
  "maxBackupCnt": 10,
  "preferredBackupPeriod": "Monday",
  "preferredBackupTime": "03:00Z"
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号，当前取值1
instanceId	string	是	URL参数	指定实例Id
maxBackupCnt	int	是	RequestBody参数	最大的备份保存数量
autoBackup	string	是	RequestBody参数	自动备份状态，取值：[ON,OFF]。ON：开启，OFF：关闭。仅当autoBackup为ON时，preferredBackupPeriod和preferredBackupTime有意义。
preferredBackupPeriod	string	否	RequestBody参数	当autoBackup为ON时，须填写此参数。自动备份周期，只能是以下7种取值： Monday：周一； Tuesday：周二； Wednesday：周三； Thursday：周四； Friday：周五； Saturday：周六； Sunday：周日。 说明：暂时只支持一周备份一次。
preferredBackupTime	string	否	RequestBody参数	当autoBackup为ON时，须填写此参数。自动备份开始时间（UTC）。格式为HH:mmZ。

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

无

### 请求示例

```

PUT /v1/instance/hb-abc123/backupPolicy HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
{
  "autoBackup": "ON",
  "maxBackupCnt": 10,
  "preferredBackupPeriod": "Monday",
  "preferredBackupTime": "03:00Z"
}

```

### 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json

```

## 查询备份策略

### 备份功能说明

1. Status为FINISHED且Result为SUCCEED的备份称为有效备份。以下操作仅适用于有效备份：用于恢复数据库；修改备份描述信息；查看备份的数据库信息；删除备份。非有效备份不能进行上述操作。
2. 备份策略中backupUsedCapacity为实例所有有效备份大小之和。
3. maxBackupCnt是一个实例的有效备份保存数量最大值（含自动备份和用户主动备份）。当实例成功备份（含自动备份和用户主动备份）后，如果有效备份数已经大于maxBackupCnt，则系统将删除最早的有效备份，直至有效备份数变为maxBackupCnt。
4. backupCapacityQuota是一个实例的备份空间总量，如果某次备份操作的备份文件大于剩余的备份空间（即 backupCapacityQuota - backupUsedCapacity），则该次备份操作将失败。
5. 删除的备份文件不可恢复（含系统删除和用户主动删除的备份）。

### 说明

- 查询指定实例的备份策略

### 请求结构

```

GET /v{version}/instance/{instanceId}/backupPolicy HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string

```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号，当前取值1
instanceId	string	是	URL参数	指定实例Id

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
maxBackupCnt	int	最大的备份保存数量
backupCapacityQuota	double	备份空间总量。单位：GB。
backupUsedCapacity	double	已使用空间大小。单位：GB。
autoBackup	string	自动备份状态，取值：[ON,OFF]。ON：开启，OFF：关闭。仅当autoBackup为ON时，preferredBackupPeriod和preferredBackupTime有意义。
preferredBackupPeriod	string	自动备份周期，只能是以下7种取值：Monday：周一；Tuesday：周二；Wednesday：周三；Thursday：周四；Friday：周五；Saturday：周六；Sunday：周日。说明：暂时只支持一周备份一次。
preferredBackupTime	string	自动备份开始时间（UTC时间）。

### 请求示例

```
GET /v1/instance/hb-abc123/backupPolicy HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
{
  "autoBackup": "OFF",
  "maxBackupCnt": 3,
  "preferredBackupPeriod": "Wednesday",
  "preferredBackupTime": "01:00Z",
  "backupCapacityQuota": 100000000,
  "backupUsedCapacity": 0
}
```

### 查询备份列表

#### 说明

- 查询指定实例的备份列表（含已删除备份）

#### 请求结构

```
GET /v{version}/instance/{instanceId}/backup?marker={marker}&maxKeys={maxKeys} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

#### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号，当前取值1
instanceId	string	是	URL参数	指定实例Id
marker	string	否	Query参数	批量获取列表的查询的起始位置，是一个由系统生成的字符串
maxKeys	int	否	Query参数	每页包含的最大数量，可选10/20/30，默认10

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
marker	string	批量获取列表的查询的起始位置，是一个由系统生成的字符串
maxKeys	int	每页包含的最大数量
isTruncated	boolean	true表示后面还有数据，false表示已经是最后一页
nextMarker	string	获取下一页所需要传递的marker值。当isTruncated为false时，该域不出现
result	List<BackupRecord>	备份信息列表。参见 API参考/备份接口/附录/BackupRecord

### 请求示例

```
GET /v1/instance/hb-abc123/backup HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863ffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```

HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
{
  "marker": "-1",
  "isTruncated": false,
  "nextMarker": null,
  "maxKeys": 10,
  "result": [{
    "backupAllTable": true,
    "description": "test_0223_7",
    "endTime": "2023-02-23T11:33:18Z",
    "errMsg": "",
    "id": "backup-MKPAjE",
    "mode": "Manual",
    "result": "SUCCEED",
    "size": 10912,
    "startTime": "1970-01-01T00:00:00Z",
    "status": "FINISHED",
    "tables": "",
    "target": "bos://hb-backup-01/hb-abc123/backup-MKPAjE/"
  },
  {
    "backupAllTable": true,
    "description": "",
    "endTime": "2023-02-24T02:42:12Z",
    "errMsg": "",
    "id": "backup-TcWW9N",
    "mode": "Manual",
    "result": "SUCCEED",
    "size": 10912,
    "startTime": "1970-01-01T00:00:00Z",
    "status": "FINISHED",
    "tables": "",
    "target": "bos://hb-backup-01/hb-abc123/backup-TcWW9N/"
  }
]
}

```

## 附录

### 对象定义

**BackupRecord** | 参数名称 | 类型 | 描述 | |-----|-----|-----|-----| id |string| 备份ID | target | string | 备份存储源 | backupAllTable| boolean | 是否备份所有的表 | tables | string | 创建备份时，指定的表信息 | mode | string | 备份模式，取值参见 BackupRecord\_mode | status | string | 备份状态，取值参见 BackupRecord\_status | result | string | 备份结果，取值参见 BackupRecord\_result | errMsg | string | 备份失败原因 | startTime| string | 备份开始时间，格式为yyyy-MM-ddTHH:mm:ssZ (UTC时间) | endTime | string | 备份结束时间，格式为yyyy-MM-ddTHH:mm:ssZ (UTC时间) | size |int64| 备份大小，单位:字节 | description | string | 备注信息 |

**RestoreRecord** | 参数名称 | 类型 | 描述 | |-----|-----|-----|-----| id |string| 恢复记录ID | instanceId|string | 备份所属实例ID | backupId|string | 备份ID | restoreAllTable| boolean | 是否恢复所有的表 | tables | string | 恢复数据库时，指定的表信息 | status | string | 恢复状态，取值参见 RestoreRecord\_status | result | string | 恢复结果，取值参见 RestoreRecord\_result | errMsg | string | 恢复失败原因 | startTime| string | 恢复开始时间，格式为yyyy-MM-ddTHH:mm:ssZ (UTC时间) | endTime | string | 恢复结束时间，格式为yyyy-MM-ddTHH:mm:ssZ (UTC时间) |

### 编码定义

**BackupRecord\_mode**

编码	描述
Automated	系统备份
Manual	手动备份

**BackupRecord\_status**

编码	描述
INIT	初始化
BACKUPING	备份中
FINISHED	已完成
DELETING	删除中
DELETED	已删除
TO_BE_DELETED	待删除

**BackupRecord\_result**

编码	描述
UNKOWN	未知
SUCCEED	成功
FAILED	失败

**RestoreRecord\_status**

编码	描述
INIT	初始化
RESTORING	恢复中
FINISHED	已完成

**RestoreRecord\_result**

编码	描述
UNKOWN	未知
SUCCEED	成功
FAILED	失败

**查看实例恢复记录****说明**

- 查询指定实例的指定恢复记录的详情信息

**请求结构**

```
GET /v{version}/instance/{instanceId}/restoreRecord/{restoreRecordId} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

**请求头域**

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号
instanceId	string	是	URL参数	指定实例ID
restoreRecordId	string	是	URL参数	恢复记录ID

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
id	string	恢复记录ID
instanceId	string	备份所属实例ID
backupId	string	备份ID
restoreAllTable	boolean	是否恢复所有的表
tables	string	恢复数据库时，指定的表信息
status	string	恢复状态，取值参见 API参考/备份接口/附录/RestoreRecord_status
result	string	恢复结果，取值参见 API参考/备份接口/附录/RestoreRecord_result
errMsg	string	恢复失败原因
startTime	string	恢复开始时间，格式为yyyy-MM-ddTHH:mm:ssZ（UTC时间）
endTime	string	恢复结束时间，格式为yyyy-MM-ddTHH:mm:ssZ（UTC时间）

### 请求示例

```
GET /v1/instance/hb-ABC/backup/backup-abc123 HTTP/1.1
HOST: hbase.bj.baidubce.com
ContentType: application/json
Authorization: bce-auth-v1/f81d3b34e48048fbb2634dc7882d7e21/2017-11-21T04:17:29Z/3600/host/74c506f68c65e26c633bfa104c863fffac5190fdec1ec24b7c03eb5d67d2e1de
```

### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
{
  "backupId": "backup-TcWW9N",
  "endTime": "2023-02-24T03:02:51Z",
  "errMsg": "",
  "id": "restore-dnjKiO",
  "restoreAllTable": false,
  "result": "SUCCEED",
  "startTime": "2023-02-24T03:02:16Z",
  "status": "FINISHED",
  "tables": "ns2:table1/ns3:table"
}
```

### 查询恢复记录列表

## 说明

- 查询指定实例的恢复记录列表

## 请求结构

```
GET /v{version}/instance/{instanceId}/restoreRecord?marker={marker}&maxKeys={maxKeys} HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

## 请求头域

除公共头域外，无其它特殊头域。

## 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号，当前取值1
instanceId	string	是	URL参数	指定实例Id
marker	string	否	Query参数	批量获取列表的查询的起始位置，是一个由系统生成的字符串
maxKeys	int	否	Query参数	每页包含的最大数量，可选10/20/30，默认10

## 返回头域

除公共头域，无其它特殊头域。

## 返回参数

参数名称	类型	描述
marker	string	批量获取列表的查询的起始位置，是一个由系统生成的字符串
maxKeys	int	每页包含的最大数量
isTruncated	boolean	true表示后面还有数据，false表示已经是最后一页
nextMarker	string	获取下一页所需要传递的marker值。当isTruncated为false时，该域不出现
result	List<RestoreRecord>	备份信息列表。参见 <a href="#">API参考/备份接口/附录/RestoreRecord</a>

## 请求示例

```
GET /v1/instance/hb-abc123/restoreRecord HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
```

## 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
{
  "nextMarker": nextBackupId,
  "marker": currentBackupId,
  "maxKeys": maxPageCount,
  "isTruncated": is_truncated,
  "restoreRecords": [{
    "backupId": "backup-TcWW9N",
    "endTime": "2023-02-24T02:58:21Z",
    "errMsg": "",
    "id": "restore-eOydUW",
    "restoreAllTable": true,
    "result": "SUCCEED",
    "startTime": "2023-02-24T02:56:47Z",
    "status": "FINISHED",
    "tables": ""
  },
  {
    "backupId": "backup-TcWW9N",
    "endTime": "2023-02-24T03:02:51Z",
    "errMsg": "",
    "id": "restore-dnjKi0",
    "restoreAllTable": false,
    "result": "SUCCEED",
    "startTime": "2023-02-24T03:02:16Z",
    "status": "FINISHED",
    "tables": "ns2:table1/ns3:table"
  }
  ]
}
```

## 恢复数据库

### 说明

- 指定一个备份文件，进行数据库恢复操作
- 实例须处于运行中状态

### 请求结构

```
PUT /v{version}/instance/{instanceId}?restore HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "backupId": "backup-xxx"
}
```

### 请求头域

除公共头域外，无其它特殊头域。

### 请求参数

参数名称	类型	是否必须	参数位置	描述
version	string	是	URL参数	API版本号，当前取值1
instanceId	string	是	URL参数	指定实例ID
backupId	string	是	RequestBody参数	备份ID
restoreAllTable	boolean	否	RequestBody参数	恢复所有数据，默认值为True
tables	string	否	RequestBody参数	<p>如果restoreAllTable为False，则需要指定恢复的表。</p> <p>①如果恢复至当前表，填写的格式为namespace1:table1</p> <p>②如果恢复至其他表，按[备份表/恢复表]的格式填写，如namespace1:table1/namespace2:table2，则备份文件中的备份表将被恢复到当前实例的恢复表。如果需要恢复多个表，则用逗号(,)隔开。</p> <p>注意以下情况时备份会失败：恢复表的命名空间(namespace2)不存在。</p>

### 返回头域

除公共头域，无其它特殊头域。

### 返回参数

参数名称	类型	描述
restoreRecordId	string	恢复记录ID

### 请求示例

```
PUT /v1/instance/hb-abc123?restore HTTP/1.1
HOST: hbase.bj.baidubce.com
Authorization: authorization string
{
  "backupId": "backup-xxx"
}
```

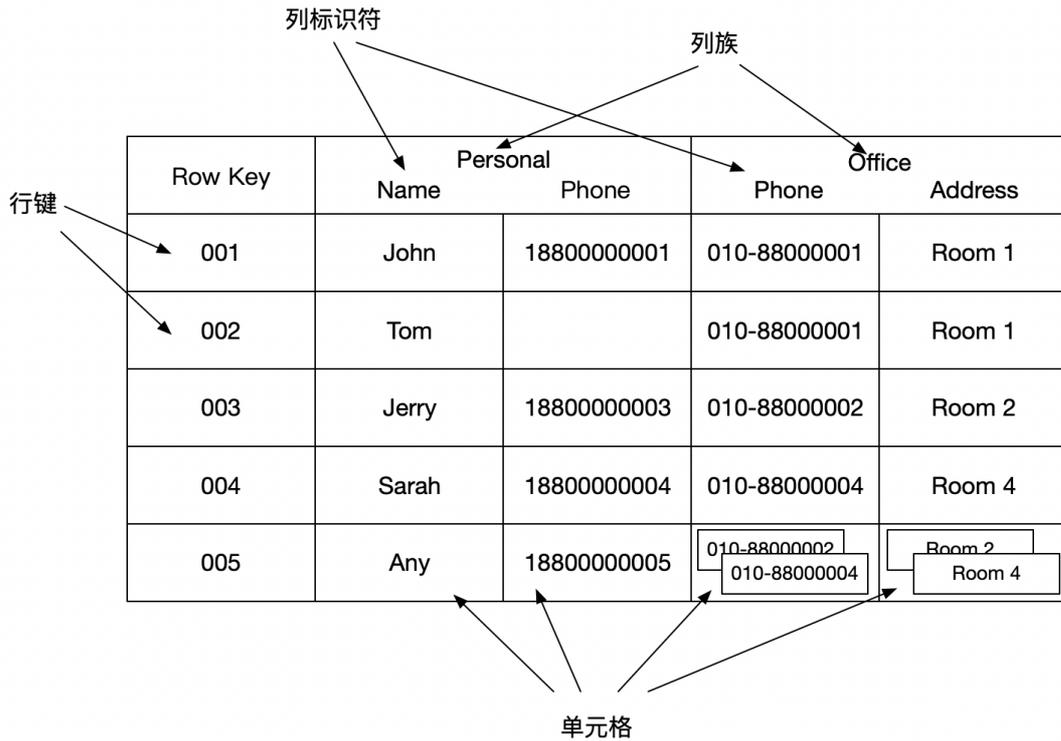
### 返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 946002ee-cb4f-4aad-b686-5be55df27f09
Content-Type: application/json;charset=UTF-8
{
  "restoreRecordId": "restore-xxxxx"
}
```

## 开发指南

### 数据模型

在HBase中，数据存储在有行和列的表中。这与关系数据库（RDBMS）的术语相似，但是更恰当的方式是将HBase表视为一种多维映射。



## 相关术语

### 命名空间 (Namespace)

命名空间是一组表 (Table) 的集合，类似于关系型数据库中的Database概念。这一概念有助于多租户场景下的数据和资源的隔离。

### 表 (Table)

一张HBase表由许多行 (row) 数据组成。

### 行 (Row)

一行数据由一个行键 (row key) 以及一个或多个值 (value) 构成。在存储数据时，行按照行键的字母序排列。因此，行键的设计非常重要：相关的行最好能够被分配到相近的位置。一个常见的例子是网站地址。如果你的行键设置成网站地址，那么你应当将网站地址逆序存储（例如org.apache.www, org.apache.mail等）。这种行键设计能够将同属于apache.org的网址存放在相近的位置。

### 列 (Column)

一列数据由一个列族 (column family) 以及一个列标识符 (column qualifier) 组成；二者通常由一个:符号相连（例如column\_family:column\_qualifier），表示一列数据。

### 列族 (Column Family)

列族表示物理上存储在一起的一组列 (column) 和他们的值。这么做的原因通常是性能。每个列族中都有一组存储属性，例如其是否该换存在内存中，其数据如何压缩存储，或是其行键的编码的方式为何，等等。一张表格中的每行都拥有相同的列族，但是某一行也许只会有一些列族上存储内容，而另一些列族上为空。

### 列标识符 (Column Qualifier)

列标识符与列族一起指定了一列数据。尽管创建表时列族的数量与名称就固定了，但是列标识符可以动态增加或删除。即使在同一个列族中，行与行之间也可能拥有不同的列标识符。

- 单元格 (Cell)

一个单元格指一个由 <(行, 列族, 列标识符) => (时间戳, 值)> 构成的数据结构。一个单元格可以有多个版本的值, 由时间戳区分。

- 时间戳 (Timestamp)

每个值都带有一个时间戳, 指代某个特定版本。默认情况下, 时间戳指的是当数据写入某个RegionServer时RegionServer上的本地时间, 但是用户也可以在写入数据时指定时间戳。

## 概念视图

本示例表格由[BigTable论文](#)第二页中的示例修改得到。

行键	时间戳	列族 contents	列族 anchor	列族 people
"com.cnn.www"	t9		anchor:cnnsi.com = "CNN"	
"com.cnn.www"	t8		anchor:my.look.ca = "CNN.com"	
"com.cnn.www"	t6	contents:html = "<html>…"		
"com.cnn.www"	t5	contents:html = "<html>…"		
"com.cnn.www"	t3	contents:html = "<html>…"		
"com.example.www"	t5	contents:html = "<html>…"		people:author = "John Doe"

需要注意的是, 表格中的空白代表该行在这个列族上没有数据。需要注意的是, 表格中的空格在HBase中不占用空间, 实际上也并不存在。下面这个json代表了与表格相同的信息。

注意: 这个json只是作为示例使用, 他并不是严格意义上绝对准确的。

```
{
  "com.cnn.www": {
    contents: {
      t6: contents:html: "<html>..."
      t5: contents:html: "<html>..."
      t3: contents:html: "<html>..."
    }
    anchor: {
      t9: anchor:cnnsi.com = "CNN"
      t8: anchor:my.look.ca = "CNN.com"
    }
    people: {}
  }
  "com.example.www": {
    contents: {
      t5: contents:html: "<html>..."
    }
    anchor: {}
    people: {
      t5: people:author: "John Doe"
    }
  }
}
```

## 物理视图

尽管在概念视图的示例中，HBase中的数据看起来以行为单位存储，但实际上他们按照列族存储。一个新的列标识符可以随时被加入已经存在的列族存储中。

行键	时间戳	列族 anchor
"com.cnn.www"	t9	anchor:cnnsi.com = "CNN"
"com.cnn.www"	t8	anchor:my.look.ca = "CNN.com"

如上表所示，没有值的空白单元格完全没有被存储，自然也不占用物理存储空间。对这些空白单元格的查询不会返回任何值。但是，如果用户在查询时不提供时间戳，HBase会返回特定列的最新值。在HBase中，同一单元格的不同版本数据按照时间戳降序排列，因此HBase只需要返回第一个找到的值即可。

## 数据模型操作

HBase共有四个常用的数据模型操作，分别为Get、Put、Scan和Delete。这些操作的操作对象都是一张表。

### Get

获取特定行的一些值。

### Put

在表中插入新行（如果行不存在）或者更新行中的某些值（如果行存在）。

### Scan

使用迭代器（iterator）访问一些行。

### Delete

从表中删除特定行。

## 数据排列顺序

对于所有的数据模型操作，HBase都会按照一个特定顺序返回数据。数据会首先按行排列，随后按列族排列，接着按列标识符排列，最后按时间戳降序排列。

## 列元数据

在列族之外，HBase并不会在某些地方存储列的元数据（如列标识符等）。因此，HBase可以支持列标识符的动态修改，以及支持同一列族中拥有大量不同的列标识符。用户需要自己妥善处理并保管列族中的列标识符，保证其符合用户预期。如果用户不慎遗失了列元数据，那么重新获得该数据的方式只有scan整个列族并重新构建元数据。

## Join

HBase不支持，或者说不像RDBMS那样支持Join操作。如同数据模型操作中所提到的，HBase支持两种读操作：Get和Scan。

但是这不意味着用户无法实现Join操作。事实上，用户可以使用四种常用数据模型操作手动实现定制化的Join操作。

## ACID

ACID，即原子性（Atomicity）、一致性（Consistency）、隔离性（Isolation）与持久性（Durability）。关于HBase的ACID特性，请参考Lars Hofhansl的[ACID in HBase](#)。

如果您想了解更多关于HBase数据模型的详细内容，请参阅[HBase官方文档](#)。

## Schema设计原则

### Schema创建

HBase schema（模式）可以通过Apache HBase Shell或者Java API Admin来创建或者更新。进行ColumnFamily（列族）修改时，必须禁用表，例如：

```
Configuration config = HBaseConfiguration.create();
Admin admin = new Admin(config);
TableName table = TableName.valueOf("myTable");
admin.disableTable(table);
HColumnDescriptor cf1 = ...;
admin.addColumn(table, cf1);
HColumnDescriptor cf2 = ...;
admin.modifyColumn(table, cf2);
admin.enableTable(table);
```

### Schema更新

当表或者列族被修改时(比如region大小、block大小)，这些更改会在下一次major合并、StoreFiles重新写入时生效。

### Schema设计经验

HBase存在许多种不同的数据集，具有不同的访问模式和服务层级的要求。因此，以下经验法则只是概述。

- 目标region的大小限制在10GB到50GB之间。
- 限制cell的大小在10MB之内，如果使用的是mob类型，限制在50MB之内。否则，考虑把cell的数据存储在HDFS中，并在HBase中存储指向该数据的指针。
- 典型的scheme每张表包含1到3个列族。HBase表设计不应当和RDBMS表设计类似。
- 对于拥有1或2个列族的表来说，50-100个region是比较合适的。注意region是列族的连续段。
- 保持列族名称尽可能短。每个值都会存储列族的名称(忽略前缀编码)。它们不应该像在典型的RDBMS中一样具有自我记录和描述性。
- 如果正在存储基于时间的机器数据或者日志信息，并且rowkey是基于设备ID或者服务ID+时间，最终可能会出现这样一种情况，即更旧的数据region在某个时间段后永远不会有额外写入。在这种情况下，最终会存在少量的活动region和大量不会再有新写入的region。

### HBase列族数量

HBase目前不适合两个或三个列族以上的任何项目，因此应保持模式中的列族数量较少。目前，flushing和compactions是基于每个region进行的，因此如果一个列族承载大量数据带来flushing，则即使它们携带的数据量很小，相邻的族也将被flushing。当存在许多列族时，flushing和compactions可能会产生一堆不必要的I/O。

如果单个表中存在多个列族，请注意基数（即行数）不用相差太大。如果ColumnFamilyA有100万行而ColumnFamilyB有10亿行，则ColumnFamilyA的数据可能会分布在许多区域（和RegionServers）中。这使得ColumnFamilyA的大规模扫描效率降低。

### 版本数

#### 最大版本数

HBase通过HColumnDescriptor给各列族配置要存储的最大行版本数。最大版本的默认值为1。这是一个重要的参数，因为HBase不会覆盖一个值，只会时间戳来区分值的不同版本。早期的版本会在major compaction过程中被删除。根据应用需求，可以增加或减少最大版本的数量。不建议将最大版本的数量设置为非常高的级别（例如，数百或更多），除非这些旧值非常珍贵，因为这将大大增加StoreFile的大小。

## ☞ 最小版本数

与最大行版本数一样，HBase通过HColumnDescriptor按列族配置要保留的最小行版本数。最小版本的默认值为0，表示该功能已禁用。行版本参数的最小数量与生存时间参数一起使用，并且可以与行版本参数的数量组合，以允许配置诸如“保留最后T分钟的数据，最多N个版本，但至少保留M个版本”（其中M是最小行版本数的值， $M < N$ ）。

## ☞ 支持的数据类型

HBase通过Put和Result操作支持“byte-in/bytes-out”接口，因此任何可以转换为字节数组的都可以存储为值。输入可以是字符串、数字、复杂对象、图像，只要它们可以呈现为字节。值的大小存在实际限制。HBase中的所有行都符合数据模型，包括版本控制。在进行设计时要考虑到这一点，以及列族的块大小。

## ☞ 生存时间

列族可以设置TTL（Time To Live）长度（以秒为单位），HBase将在超时后自动删除行。TTL设置适用于行的所有版本。在HBase中TTL时间为UTC时区。仅包含过期的行的存储文件，会在minor compaction时删除。将

`hbase.store.delete.expired.storefile`设置为false可禁用此功能。将最小版本数设置为0以外的值也会禁用此功能。

最新版本的HBase还支持按每个cell（单元）设置生存时间。cell TTL使用Mutation # setTTL作为变更请求的一个属性提交。如果设置了TTL属性，则它将应用于被操作更新的所有cell。cell TTL处理和列族TTL之间存在两个显著差异：

- cell TTL以毫秒而不是秒为单位表示。
- cell TTL不能超过列族TTL设置的有效时间。

# Rowkey设计原则

## ☞ hotspotting

HBase中的行由Rowkey（行键）按字典顺序排序。此设计优化了扫描（scan），将相关的行或将要一起读取的行存储在彼此附近。设计不良的行键会导致hotspotting发生。当大量客户端流量（traffic）指向群集的一个节点或少数几个节点时，hotspotting就会发生。此流量可能表示读取、写入或其他操作。流量超过负责托管该区域的单个机器的极限，就会导致性能下降并可能使得区域不可用。这也可能对同一RegionServer托管的其他区域产生负面影响，因为该主机无法为请求的负载提供服务。为充分、均匀地利用集群，须良好的设计数据访问模式。

为了防止hotspotting写入，设计Rowkey时应该尽量使数据被写入群集中的多个区域，除非确实有必要使数据写在同一个区域里。下面描述了一些避免hotspotting的常用技术及其优缺点。

## ☞ salting

在这里，salting与加密无关，是指将随机数据添加到行键的开头。具体而言，salting是向行键添加随机分配的前缀，以使其排序与其他方式不同。所有可能的前缀数对应于数据的区域数。如果有一些“热”行键模式在其他更均匀分布的行中反复出现，则Salting可能会有所帮助。考虑以下示例，该示例显示salting可以在多个RegionServers之间分散写入负载，同时也说明了其对读取的负面影响。

### *Salting 示例*

假设有以下行键列表，表按每个字母对应一个区域来分割。前缀“a”对应一个区域，前缀“b”对应另一个区域。在此表中，以“f”开头的行都在同一区域中。此示例关注具有以下键的行：

```
foo0001
foo0002
foo0003
foo0004
```

现在，希望将它们分布在四个不同的地区，可以使用四种不同的salt：a，b，c和d。在这种情况下，每个字母前缀的都对应不同的区域。应用salt后，将改为使用以下4个rowkey。由于现在将数据写入了四个不同的区域，理论上写入时的吞吐量将是所有数据写入到同一区域时的四倍。

```
a-foo0003
b-foo0001
c-foo0004
d-foo0002
```

然后，如果新添加另一行，将给它随机分配四个可能的salt值中的一个，并靠近其中一个现有行。

```
a-foo0003
b-foo0001
c-foo0003
c-foo0004
d-foo0002
```

由于前缀的分配是随机的，因此如果要按字典顺序检索行，则需要执行更多工作。通过这种方式，salting增加写入的吞吐量，但在读取操作的开销会变大。

## 🔗 Hashing

除了随机分配前缀之外，可以使用单向Hashing，使给定行salting时始终使用同一的前缀，从而将负载分散到RegionServer，也允许读取时可以预测。使用确定性哈希允许客户端重建完整的rowkey并使用Get操作正常检索该行。

### *Hashing示例*

考虑上述salting示例中给出的情况，可以改为使用单向Hashing得到行键为foo0003的行，并且可预测其前缀“a”。然后，要检索该行，须先知道它的键。还可以进一步优化，比如使某些键对始终对应着特定区域。

## 🔗 反转键

第三个防止hotspotting的技巧是反转固定长度或可数的键，使最常更改的部分（最低有效位数）位于第一位。这个方法有效地使行键随机化，但牺牲了行的排序属性。

## 🔗 简化行与列

在HBase中，值是作为一个cell（单元）保存在系统之中的，伴随着它的行、列名和时间戳。如果行名和列名很大（特别是比单元的值还要大时），那么可能会遇到一些特别的状况。在HBase的StoreFile中，有一个用于随机访问而保留的索引，如果访问一个单元的坐标过大、占用很大的内存，则该索引会被用尽。针对这个问题，可以增大块的大小，也可以设置较小的行名和列名。在这里，微小的“低效”也不能忽略，因为列族、属性和Rowkey都可能在数据中多次重复。

## 🔗 列族

使列族名尽可能小，最好是一个字符（例如“d”表示数据/默认值）。

## 🔗 属性

尽管详细的属性名称（例如“myVeryImportantAttribute”）易于阅读，但是较短的属性名称（例如“via”）更适合存储在HBase中。

## 🔗 Rowkey长度

设置Rowkey应该尽可能的短，使它们更适用于数据访问（如Get和Scan）。在设计rowkey时需要权衡。

## 🔗 字节模式

long类型是8个字节。8个字节内存储的最大无符号数18446744073709551615。如果将此数字存储为字符串（假设每个字符有一个字节），则需要将近3倍的字节。

```

// long
//
long l = 1234567890L;
byte[] lb = Bytes.toBytes(l);
System.out.println("long bytes length: " + lb.length); // returns 8

String s = String.valueOf(l);
byte[] sb = Bytes.toBytes(s);
System.out.println("long as string length: " + sb.length); // returns 10

// hash
//
MessageDigest md = MessageDigest.getInstance("MD5");
byte[] digest = md.digest(Bytes.toBytes(s));
System.out.println("md5 digest bytes length: " + digest.length); // returns 16

String sDigest = new String(digest);
byte[] sbDigest = Bytes.toBytes(sDigest);
System.out.println("md5 digest as string length: " + sbDigest.length); // returns 26

```

不过，使用二进制表示将使数据在代码之外难以阅读。例如，这是在新增一个值时在shell中看到的内容：

```

hbase(main):001:0> incr 't', 'r', 'f:q', 1
COUNTER VALUE = 1

hbase(main):002:0> get 't', 'r'
COLUMN                                CELL
f:q                                     timestamp=1369163040570, value=\x00\x00\x00\x00\x00\x00\x00\x01
1 row(s) in 0.0310 seconds

```

## 🔗 倒序时间戳

数据库处理中的一个常见问题是快速找到最新版本的值。使用倒序时间戳作为键的一部分对解决这个问题很有帮助。该技术将  $(\text{Long.MAX\_VALUE} - \text{timestamp})$  附加到键的末尾，例如：`[键][reverse_timestamp]`。

执行 `Scan [key]` 并获取第一条记录，就可以找到表中 `[key]` 的最新值。此技术可以替代版本号的使用，其意图是“永久”（或很长时间）保留所有版本。同时，通过使用 `Scan`，可以快速获取其他版本。

## 🔗 Rowkey和列族

Rowkey的作用域为列族。因此，同样的Rowkey可以存在于同一个表的不同列族中。

## 🔗 Rowkey不可改变

Rowkey是无法更改的。唯一在一个表中“更改”Rowkey的方法是删除行然后重新插入。这是一个常见的问题，因此一开始（以及插入大量数据之前）就需要确保rowkey是正确的。

## 🔗 RowKey与区域split的关系

如果预分割（pre-split）了表，那么了解rowkey如何在区域边界上分布是很重要的。为说明这一点，可以考虑用十六进制字符作为键的关键位置（例如，“0000000000000000”到“fffffffffffffff”）。通过 `Bytes.split` 分割这些键的范围，这样会分得10个区域。

```

48 48 48 48 48 48 48 48 48 48 48 48 48 48 48 // 0
54 -10 -10 -10 -10 -10 -10 -10 -10 -10 -10 -10 -10 -10 // 6
61 -67 -67 -67 -67 -67 -67 -67 -67 -67 -67 -67 -67 -68 // =
68 -124 -124 -124 -124 -124 -124 -124 -124 -124 -124 -124 -124 -126 // D
75 75 75 75 75 75 75 75 75 75 75 75 75 75 72 // K
82 18 18 18 18 18 18 18 18 18 18 18 18 18 14 // R
88 -40 -40 -40 -40 -40 -40 -40 -40 -40 -40 -40 -40 -44 // X
95 -97 -97 -97 -97 -97 -97 -97 -97 -97 -97 -97 -97 -102 // _
102 102 102 102 102 102 102 102 102 102 102 102 102 102 102 // f

```

问题是所有数据都将堆积在前两个区域和最后一个区域，从而产生“块状”区域问题。要了解原因，须参阅ASCII表。有意义的值是[0-9]和[a-f]，在ASCII表中，'0'是第48号，'f'是第102号；第58号到第96号之间存在巨大的间隙，这个区间内的值不会出现在键空间（keyspace）中，因此中间区域将不会被使用。为了预分割该示例中的键空间，需要自定义分割。

方法1：预分割表（pre-splitting table）通常是较好的方法，但pre-split时须注意使所有的区域在键空间中有所对应。虽然示例演示的是十六进制键空间的问题，但在其他键空间也是同样的道理。

方法2：尽管通常不可取，但只要所有区域都能在键空间中有所对应，十六进制的键也可以与预分割表配合使用。

以下是十六进制键预分区的示例：

```

public static boolean createTable(Admin admin, HTableDescriptor table, byte[][] splits)
throws IOException {
    try {
        admin.createTable( table, splits );
        return true;
    } catch (TableExistsException e) {
        logger.info("table " + table.getNameAsString() + " already exists");
        // the table already exists...
        return false;
    }
}

public static byte[][] getHexSplits(String startKey, String endKey, int numRegions) {
    byte[][] splits = new byte[numRegions-1][];
    BigInteger lowestKey = new BigInteger(startKey, 16);
    BigInteger highestKey = new BigInteger(endKey, 16);
    BigInteger range = highestKey.subtract(lowestKey);
    BigInteger regionIncrement = range.divide(BigInteger.valueOf(numRegions));
    lowestKey = lowestKey.add(regionIncrement);
    for(int i=0; i < numRegions-1;i++) {
        BigInteger key = lowestKey.add(regionIncrement.multiply(BigInteger.valueOf(i)));
        byte[] b = String.format("%016x", key).getBytes();
        splits[i] = b;
    }
    return splits;
}

```

## HBase SQL服务

### HBase SQL(Phoenix)使用说明

[🔗 HBase SQL\(Phoenix\)使用说明](#)

[🔗 Phoenix简介](#)

Phoenix是HBase的开源SQL服务，支持OLTP与操作分析。Phoenix提供了具有完整ACID事务功能的标准SQL与JDBC API。更多信息请参阅[Apache Phoenix官方文档](#)。

[🔗 前提条件](#)

通过内网连接时，请确认已购买与HBase实例在相同专有网络VPC的BCC服务器，并将BCC服务器的IP地址在HBase中添加为白名单。

- 若未购买BCC服务器，具体操作请参见[创建BCC实例](#)。
- 若未配置白名单，具体操作请参见[设置白名单](#)。

## 🔗 操作步骤

1. 下载[Phoenix客户端](#)并将其放入BCC服务器中。
2. 在数据库连接信息中，复制该实例的ZK链接地址。



3. 在 `$PHOENIX_HOME/bin/hbase-site.xml` 中新增配置项，其中 `$ZK_IP1`, `$ZK_IP2`, `$ZK_IP3`，需使用实际实例的zk地址替换。

```
<property>
  <name>hbase.zookeeper.quorum</name>
  <value>$ZK_IP1,$ZK_IP2,$ZK_IP3</value>
</property>
```

5. 在BCC中进入 `$PHOENIX_HOME/bin/` 文件夹并运行如下命令，其中 `$ZK_IP1`, `$ZK_IP2`, `$ZK_IP3`，需使用实际实例的zk地址替换。

```
./sqlline.py $ZK_IP1,$ZK_IP2,$ZK_IP3
```

出现形如 `0: jdbc:phoenix:zk-xxxxx-1.hbase.bj.baidubce>` 的消息即为配置成功。

## HBase SQL(Phoenix)二级索引

### 🔗 HBase SQL (Phoenix) 二级索引

#### 🔗 简介

Phoenix支持多种二级索引功能。

#### 🔗 全局索引

全局索引适用于读负载较重的场景。对于全局索引，索引的所有性能损失都发生在数据写入时。我们在写入时拦截数据表更新（DELETE、UPSERT VALUES和UPSERT SELECT），构建索引更新，然后向所有相关索引表发送更新。在读取时，Phoenix将选择要速度最快的索引表，并像遍历任何其他HBase表一样直接遍历它。当一个索引被用于某个查询时，会保证该查询中所有被引用的列均可在该索引中找到。

全局索引创建语句如下：

```
CREATE TABLE DATA_TABLE (  
  A VARCHAR PRIMARY KEY,  
  B VARCHAR,  
  C INTERGER  
);  
  
CREATE INDEX B_IDX ON DATA_TABLE(B) INCLUDE C;
```

## 本地索引

本地索引适用于写负载较重的场景。与全局索引一样，Phoenix将在查询时自动选择是否使用本地索引。使用本地索引时，索引数据和表数据将被存储在同一服务器上，从而防止写入期间的任何网络开销。即使查询没有被完全覆盖，本地索引也可以被使用（即Phoenix通过数据表的点查询自动检索索引中没有的列）。

与全局索引不同，在4.8.0版本之前，表的所有本地索引都存储在单独的共享表中。从4.8.0开始，Phoenix将所有本地索引数据存储在单一数据表中的单独隐藏列族中。当使用本地索引进行读取时，用户必须首先检查region的数据，因为无法预先确定本地索引确切位置。因此，读取操作会产生一些额外开销。

关于Phoenix二级索引更多的内容请参照[Phoenix二级索引官方文档](#)。

## HBase SQL(Phoenix)快速开始

### 本地索引

### Phoenix入门

在开始之前，请确认您配好了Phoenix环境。如果您还没有配置好Phoenix环境，请参阅[HBase SQL\(Phoenix\)使用说明](#)进行配置。

### 快速开始

1. 创建us\_population.sql文件：

```
CREATE TABLE IF NOT EXISTS us_population (  
  state CHAR(2) NOT NULL,  
  city VARCHAR NOT NULL,  
  population BIGINT  
  CONSTRAINT my_pk PRIMARY KEY (state, city));
```

2. 创建us\_population.csv文件：

```
NY,New York,8143197  
CA,Los Angeles,3844829  
IL,Chicago,2842518  
TX,Houston,2016582  
PA,Philadelphia,1463281  
AZ,Phoenix,1461575  
TX,San Antonio,1256509  
CA,San Diego,1255540  
TX,Dallas,1213825  
CA,San Jose,912332
```

3. 将数据写入Hbase:

```
./psql.py <zookeeper_quorum> us_population.sql us_population.csv
```

4. 运行Phoenix client并查看数据是否写入成功：

```
l0: jdbc:phoenix:zk-k8iadj-1.hbase.bj.baidubce> select * from us_population;
```

STATE	CITY	POPULATION
AZ	Phoenix	1461575
CA	Los Angeles	3844829
CA	San Diego	1255540
CA	San Jose	912332
IL	Chicago	2842518
NY	New York	8143197
PA	Philadelphia	1463281
TX	Dallas	1213825
TX	Houston	2016582
TX	San Antonio	1256509

```
10 rows selected (0.051 seconds)
```

## HBase时序引擎

### OpenTSDB概述

#### OpenTSDB概述

#### OpenTSDB时序引擎

OpenTSDB是一个具有可伸缩性的，基于HBase的分布式时间序列数据库，其数据存储存储在HBase上。OpenTSDB的主要用途是从计算机系统中（如网络设备，操作系统，应用程序等）采集与保存监控数据，并且使用户可以十分容易地访问与绘制这些数据。

#### OpenTSDB架构

OpenTSDB由一个时序守护进程（TSD）以及一组命令行程序组成。用户与OpenTSDB的交互主要通过运行一个或多个TSD来实现。每个TSD是独立的，无状态的，因此用户可以根据需要运行任意多个TSD来处理任何负载。每个TSD都使用HBase来存储和检索时间序列数据。OpenTSDB的数据模式针对时间序列的快速聚合进行了高度优化，以最大限度地减少存储空间消耗。TSD的用户不需要直接访问底层存储。用户可以通过简单的telnet风格协议、HTTP API或简单的内置GUI与TSD通信。所有通信都发生在同一端口上。

#### 数据规范

一条OpenTSB记录包含以下数据：

- \* metric - 时间序列的通用名称，如sys.cpu.user，stock.quote或env.probe.temp。
- \* timestamp - Unix/POSIX Epoch时间戳，以秒或毫秒为单位，定义为自1970年1月1日00:00:00 UTC 时间以来经过的秒数。目前仅支持正时间戳。
- \* value - 在时间序列的给定时间戳中存储的数值。这可以是整数或浮点值。
- \* tag (s) - 由tagk（键）和tagv（值）组成的键/值对。每个数据点必须至少有一个标记。
- \* data point - 时间序列数据点，表示某个metric在某个时间点的数值。

#### 系统表简介

在集群开启OpenTSDB后，OpenTSDB会在集群中创建4张OpenTSDB系统表，用于存储数据。请不要尝试修改这4张系统表的内容，因为这可能导致OpenTSDB不可用。

OpenTSDB系统表如下所示：

表名	说明
OPENTSDB.DATA	用于存储OpenTSDB的数据点
OPENTSDB.UID	用于存储metric, tag等元数据的UID映射关系
OPENTSDB.TREE	用于存储metric的结构信息
OPENTSDB.META	用于存储时间序列索引和元数据

连接与使用OpenTSDB

## 连接与使用OpenTSDB

### 开启OpenTSDB

用户创建新的云HBase数据库实例时，可以选择开启OpenTSDB时序引擎。

可用区：

时序引擎： 关

存储方式：

开启时序引擎后，用户可以在下方配置时序引擎节点的规格和数量。需要注意的是，开启时序引擎节点后，无法再关闭或扩容。请谨慎配置。

#### 时序引擎（OpenTSDB）节点配置

规格：

数量：

最小节点数量1个，最大数量为20个

温馨提示：OpenTSDB设置后不支持关闭或者扩容，请谨慎配置！

用户同样可以在创建集群后在实例列表页查看并开启时序引擎。

OpenTSDB	标签	资源分组	操作
未启用		默认分组	管理 重启 查看监控 更多
启用		默认分组	管理 重启 查看监控
未启用		默认分组	管理 重启 查看监控

增加节点

磁盘扩容

**开启OpenTSDB**

编辑标签

释放

### 连接OpenTSDB

开启OpenTSDB后，用户可以在实例详情的数据库连接页面查找OpenTSDB连接地址。用户可以使用HTTP API或telnet风格API连接并使用OpenTSDB。具体API使用方式可以参考[OpenTSDB官方文档](#)。

## 连接信息

● 为了安全连接，新创建的集群被默认设置为无法访问。通过 vpc 内网访问，您需将 HBase 的客户端 IP 地址加入白名单，才能成功连接 HBase 集群。 [设置白名单>](#)

ZK 连接地址 (VPC内网) : [REDACTED]

OpenTSDB连接地址 (内网) : [REDACTED]

访问连接: [使用HBase Shell](#) [使用JAVA API](#)

## 修改OpenTSDB节点配置

用户开启OpenTSDB服务后可以在实例列表页修改时序引擎节点的规格和数量。

OpenTSDB	标签	资源分组	操作
启用	-	默认分组	管理 重启 查看监控 更多
未启用	[REDACTED]	默认分组	管理 重启 查看监控

增加节点

磁盘扩容

OpenTSDB扩容

编辑标签

释放

10条/页 < 1 > 30

## 常见问题

## 如何选择规格

云数据库HBase支持用户根据业务的QPS/TPS、存储量、读写请求、延迟及稳定性等相关需求，自由选择节点规格和数量、存储类型和容量。

在产品规格选定后，仅支持扩容core节点数量和存储容量，暂不支持变更core节点的存储类型或各节点规格，因此建议您做好使用前评估。如果需要产品选型、容量评估的帮助，您可以联系客户经理或者提工单咨询，获得更专业的评估建议。

### 集群版各节点说明

分布式部署版本，支持多副本存储、架构高可用，支持监控告警并提供SLA保障，适合业务生产、性能测试等场景，具体提供的规格和建议如下：

#### 集群版ZK节点规格

ZK节点主要负责协调服务，它提供了配置管理、分布式锁、集群服务等功能，集群版ZK节点默认3个，规格要求较低，固定1核2GB即可满足。

#### 集群版Master节点规格

1. 集群版Master节点默认2个，实现高可用，避免单点故障。Master节点上会部署HBase的Master，如果规格偏低，会极大限制HBase集群性能。
2. Master节点规格可以根据所需的core节点数量来选择，中低规格推荐4C16GB。如果集群中表及region数量非常多，则应基于推荐配置适当选择更高的配置。
3. Master节点规格共提供7种，4核8GB/4核16GB/8核16GB/8核32GB/16核32GB/16核64GB/32核64GB，更多规格信息参见[实例节点规格](#)，规格选择建议如下：

Core节点数量	Master节点规格
≤ 3	4C16G
4-7	8C16G
8-15	8C32G
16-31	16C32G 或者以上
≥32	16C64G 或者以上

### Core节点规格

1. Core节点单次可选2-200个，购买成功后支持扩容节点数量，暂不支持变更节点规格。Core节点即HBase的RegionServer，如果规格偏低，可能影响业务稳定性和增大读写延迟。
2. Core节点规格和数量可以根据业务请求量、请求的大小来选择，低规格建议8C32GB，中规格建议16C64GB，请求较多或负载较大时建议32C128GB。
3. Core节点规格共提供7种，8核16GB/16核32GB/32核64GB/4核16GB/8核32GB/16核64GB/32核128GB，更多规格信息参见[实例节点规格](#)。

### 时序引擎节点规格

1. 时序引擎节点单次可选0-20个，购买成功后支持扩容节点数量与提升节点规格。时序引擎节点即OpenTSDB的部署节点，如果规格偏低，可能影响业务稳定性和增大读写延迟。
2. 时序引擎节点规格和数量可以根据业务请求量、请求的大小来选择，低规格建议4核16GB，请求较多或负载较大时建议32核128GB。
3. 时序引擎节点规格和数量建议与Core节点规格和数量相匹配。例如，两个4核16GB的时序引擎节点算力与两个8核16GB的Core节点提供的HBase算力大致相等，能够最大化产品性能。
4. 时序引擎节点规格共提供4种，4核16GB/8核32GB/16核64GB/32核128GB，更多规格信息参见[实例节点规格](#)。

### 存储类型和容量

#### 存储类型：

提供云磁盘3种，通用型SSD，增强型SSD，高性能云磁盘。存储类型一旦选定，不支持修改。存储总量：可选范围为400-32765GB。

#### 规格选择建议

对延迟要求高的在线业务推荐使用增强型SSD；对延迟要求一般的在线/准在线业务推荐使用通用型SSD；

QPS	Core规格和数量建议	说明	节点平均挂载存储建议
0~1000	2台4C16GB	推荐的最低规格，低负载访问场景。每台Core节点上的region数量不宜超过100个。	400G
1000~2w	2~3台 8C32GB	通常推荐的规格，中低负载访问场景。每台Core region不宜超过400个。	800G
2w ~3w	8C32GB/16C32GB/16C64GB 或者以上	根据请求量的大小选择Core节点的数量；对数据延迟要求高的在线业务，应选择内存大的规格。	1000G
3w以上	3w以上32C64GB 或者以上	并发QPS非常高的业务，选择CPU核多的规格。	1200G

## 通用问题

### 一般问题

### ☞ 云数据库HBase能提供什么能力？

是基于Apache HBase提供的分布式、可伸缩、全托管的KeyValue数据存储服务，服务可提供高性能的随机读写能力，适用于海量结构化数据、半结构化数据以及时序数据的存储和查询应用。

### ☞ 为什么要选择云数据库HBase服务？

相较于自建HBase集群，云数据库HBase开箱即用，整体成本更优，服务高可用，高性能，全托管免运维，节约用户时间与精力，可更加专注业务本身。

### ☞ 错误排查

#### ☞ 购买失败？

大多数场景是因为您购买的区域的已无库存。您需要等待一段时间或者尝试购买其他地域。如果急用，请提交工单。

#### ☞ 购买实例成功，但连接不上？

因为新建集群因为安全的考虑，默认是无法访问的，您需要检查下是否将访问云主机加入到【白名单】中，详见[白名单设置](#)。

### ☞ 计费相关

#### ☞ 购买方式

目前云数据库HBase仅支持按需后付费计费方式。

- 对于按需计费方式，系统会按小时扣减费用，只要您账户上有足够余额可抵扣产生费用，就可以一直使用服务，所以请您特别注意保证账户资金充足。

#### ☞ 如何收费

请参考[计费说明-计费项与价格](#)。

#### ☞ 欠费处理

- 当您的账户余额不足导致系统扣费失败时，您的服务资源将会进入7天的保留期，保留期间不计费，保留期内服务资源将被冻结而无法使用，但资源和数据会为您保留。
- 保留期内如您仍未给账户充值，则保留期到期后服务资源和数据会被删除，且无法恢复。
- 在保留期到期之前，您可进行账户充值操作。充值成功后，会优先扣减欠款，并同时解冻服务资源，您可以继续正常使用服务。

## 典型实践

### 数据迁移

#### 使用Snapshot工具进行数据迁移

**注意：**为了保证数据安全，如果您需要使用Snapshot工具迁移您的数据至云数据库HBase产品，请提交工单，我们会为您开放相关接口

#### ☞ Snapshot工具介绍

CopyTable工具是HBase自带的一个数据迁移工具，可以帮助用户在集群间进行数据迁移。

#### ☞ 迁移优势

- Snapshot工具是HBase内建工具，可以很好地支持用户在自建HBase集群与云数据库HBase产品间的数据迁移。

- Snapshot操作较为轻量级，对源集群影响较小。

#### 🔗 注意事项

- Snapshot工具更多地被用于停机迁移。

#### 🔗 迁移示例

1. 购买一台BCC，在BCC中下载并配置HBase shell工具包，令其能够链接源集群
2. 运行hbase shell，在源集群中使用以下指令生成snapshot：

```
snapshot 'snapshot_example', 'snapshot_example_backup_1'
```

3. 在BCC上运行以下代码，其中\$SOURCE\_MASTER1代表源集群的master节点地址，\$PEER\_MASTER1代表目标集群的master节点地址。

```
hbase org.apache.hadoop.hbase.snapshot.ExportSnapshot -snapshot snapshot_example_backup_1 -copy-from
hdfs://$SOURCE_MASTER1:8020/apps/hbase/data -copy-to
hdfs://$PEER_MASTER1.hbase.baidubce.com:8020/apps/hbase/data -mappers 16 -bandwidth 1024 -overwrite
```

4. 修改BCC中的hbase-site.xml，令BCC链接目标集群。
5. 在目标集群中运行以下命令，生成新表：

```
clone_snapshot 'snapshot_example_backup_1','snapshot_example'
```

6. 查看迁移结果。

关于Snapshot工具的更多详细信息，请参照官方文档[Snapshot](#)。

## 使用CopyTable工具进行数据迁移

#### 🔗 CopyTable工具介绍

CopyTable工具是HBase自带的一个数据迁移工具，可以帮助用户在集群间进行数据迁移。

#### 🔗 迁移优势

- CopyTable工具是HBase内建工具，可以很好地支持用户在自建HBase集群与云数据库HBase产品间的数据迁移。
- CopyTable工具底层使用Hadoop MapReduce进行数据迁移，速度较快。

#### 🔗 注意事项

- CopyTable工具更多地被用于停机迁移
- CopyTable工具使用HBase API进行数据迁移，对集群吞吐量可能有影响

#### 🔗 迁移示例

1. 购买一台BCC，在BCC中下载并配置HBase shell工具包，令其能够链接目标集群
2. 在目标集群建表。目标表配置与源集群表配置一致：

```
create 'copytable_example','f1'
```

3. 修改BCC中的hbase-site.xml，令BCC链接源集群。
4. 在源集群中运行以下命令，进行迁移，其中\$ZK1,\$ZK2,\$ZK3使用目标集群的真实Zookeeper地址代替：

```
hbase org.apache.hadoop.hbase.mapreduce.CopyTable --peer.adr=$ZK1,$ZK2,$ZK3:2181:/hbase copytable_example
```

5. 连接目标集群，查看迁移结果。

关于CopyTable工具的更多详细信息，请参照官方文档[CopyTable](#)。

## 不停机迁移

**注意：**为了保证数据安全，如果您需要不停机迁移您的数据至云数据库HBase产品，请提交工单，我们会为您开放相关接口

### 🔗 工具介绍

**Snapshot** Snapshot的相关信息可参照[使用Snapshot工具进行数据迁移](#)。**Replication** Replication是一种集群复制机制，允许使用源集群的预写日志（WAL）来传播更改，从而使两个集群的状态保持一致。Replication的常见应用场景诸如容灾备份和恢复/数据离线处理等。

Replication的更多详细信息可参照官方文档[Cluster Replication](#)。**VerifyReplication** VerifyReplication工具被用于确认Replication的状态。具体介绍可参照官方文档[VerifyReplication](#)。

### 🔗 迁移优势

- 迁移时基本不需要停机

### 🔗 迁移示例

1. 购买一台BCC，在BCC中下载并配置HBase shell工具包，令其能够链接源集群与目标集群。
2. 确保目标集群内没有与待迁移表格相同名字的表；确保两个集群能够互相连通。
3. 在源集群中执行添加目标集群，并开启replication。其中\$PEER\_ZK表示目标集群的Zookeeper地址。

```
##### here '100' is peer_cluster_id
add_peer '100', CLUSTER_KEY => '$PEER_ZK1,$PEER_ZK2,$PEER_ZK3:2181:/hbase', SERIAL => true

##### table_name='t1', column_family_name='f1'
alter 't1',{NAME=>'f1',REPLICATION_SCOPE=>'1'}
```

3. 在源集群使用Snapshot工具，将存量数据import到目标集群中。
4. 等待一段时间，让目标集群处理完积压的WAL。
5. 使用VerifyReplication工具查看两个集群是否保持同步。

```
hbase org.apache.hadoop.hbase.mapreduce.replication.VerifyReplication -mappers 10 -bandwidth 1024 $peer_cluster_id
$table_name
```

6. 两个集群基本保持同步后，关闭源集群的写权限，等待两个集群完全同步后开启目标集群的写权限，迁移完成。

# 服务等级协议SLA

## 云数据库HBase服务等级协议SLA

协议生效时间：2022年10月28日

本服务等级协议（Service Level Agreement，简称“SLA”）规定了百度智能云向客户提供的云数据库 HBase 服务可用性等级指标及赔偿方案。

### 1. 定义

服务周期：一个服务周期为一个自然月。

服务周期总分钟数：按照单云数据库HBase实例服务周期内的总天数 $\times$ 24（小时） $\times$ 60（分钟）计算。

服务不可用分钟数：在某一分钟内，用户所有试图与指定的云数据库HBase实例建立连接的连续尝试均失败，则视为该分钟内该云数据库 HBase 实例服务不可用。在一个服务周期内云数据库HBase实例不可用分钟数之和即服务不可用分钟数。

月度服务费：为客户在一个服务周期（即自然月）中就单云数据库HBase实例所支付的服务费用总额，如客户一次性支付多个月份的服务费，则将按照所购买的月数分摊计算月度服务费。

### 2. 服务可用性

#### 2.1 服务可用性计算公式

服务可用性以单个云数据库 HBase 系统实例为维度，按照如下方式计算：

服务可用性 = (服务周期总分钟数 - 服务不可用分钟数) / 服务周期总分钟数 \* 100%

#### 2.2 服务可用性承诺

百度智能云的云数据库 HBase 承诺在每个服务周期内的服务可用性不低于99.9%，如云数据库HBase 未达到上述服务可用性承诺，客户可以申请获得赔偿。

赔偿范围不包括以下原因所导致的服务不可用：

1. 百度智能云预先通知用户后进行系统维护所引起的，包括合理升级、变更、停机、割接、维修和模拟故障演练；
2. 任何百度智能云所属设备以外的网络、设备故障或配置调整引起的。
3. 客户的应用程序受到黑客攻击而引起的。
4. 客户维护不当或保密不当致使数据、口令、密码等丢失或泄漏所引起的。
5. 客户的疏忽或由客户授权的操作所引起的。
6. 因热点问题导致的服务不可用。
7. 客户未遵循百度智能云产品使用文档或使用建议引起的，如客户在控制台、API或者CLI等控制方式对云数据库HBase 实例进行操作引起的不可用。
8. 由于客户所安装软件或者其他非百度智能云直接运营的第三方软件或者配置引起的错误。
9. 任何百度智能云所属设备以外的网络、设备故障或配置调整引起的。
10. 不可抗力以及意外事件引起的。
11. 其他非百度智能云原因所造成的不可用。

### 3. 赔偿方案

### 3.1 赔偿标准

对于单云数据库HBase 实例，如服务可用性低于99.9%，可按照下表中的标准获得赔偿，赔偿方式仅限于用于购买云数据库HBase产品的代金券，且赔偿总额不超过未达到服务可用性承诺当月客户就该云数据库HBase 实例支付的单实例月度服务费（不含用代金券抵扣的费用）。

服务可用性	赔偿代金券金额
低于99.9%但等于或高于99.00%	月度服务费用的15%
低于99.00%但等于或高于95.00%	月度服务费用的30%
低于95.00%	月度服务费用的100%

### 3.2 赔偿申请时限

客户可以在每月第五（5）个工作日后对上个月没有达到可用性的云数据库 HBase 实例提出赔偿申请。赔偿申请必须限于在云数据库 HBase没有达到服务可用性的相关月份结束后两（2）个月内提出。超出申请时限的赔偿申请将不被受理。

### 4. 其他

1. 在法律法规允许的范围内，百度智能云负责对本协议进行解释说明。
2. 本协议一经公布立即生效，百度智能云有权对本SLA条款做出修改。如本SLA条款有任何修改，百度智能云将以网站公示或发送邮件的方式通知您。如您不同意百度智能云对SLA所做的修改，您有权停止使用云数据库 HBase 服务，如您继续使用云数据库HBase 服务，则视为您接受修改后的SLA。
3. 本协议项下百度智能云数据库HBase 服务对于用户所有的通知均可通过网页公告、站内信、电子邮件、手机短信或其他形式等方式进行；该类通知于发送之日视为已送达收件人。
4. 本协议的订立、执行和解释及争议的解决均应适用中国法律并受中国法院管辖。如双方就本协议内容或其执行发生任何争议，双方应尽量友好协商解决；协商不成时，任何一方均可向北京市海淀区人民法院提起诉讼。
5. 本协议构成双方对本协议之约定事项及其他有关事宜的完整协议，除本协议规定的之外，未赋予本协议各方其他权利。
6. 如本协议中的任何协议无论因何种原因完全或部分无效或不具有执行力，本协议的其余协议仍应有效并且有约束力。
7. 关于用户约束条款，详见《[用户服务协议](#)》中的"用户的权利与义务"相关条款内容。
8. 关于服务商免责条款，详见《[用户服务协议](#)》中的"免责声明"相关条款内容。