

FusionDB 文档



目录

目录	2
产品描述	3
产品概述	3
产品特性	3
应用场景	3
产品架构	3
实例规格	4
使用须知	5
典型实践	5
建表最佳实践	5
定期维护任务	8
API参考	8
API简介	8
调用说明	8
API接口说明	11
常见问题	17
常见问题总览	17
使用类问题	18
运维类问题	19
性能类问题	22

产品描述

产品概述

云数据库 FusionDB 是一种在线分布式云数据库，提供了简单、快速、高性能、高可靠的大规模并行处理（MPP）数据仓库服务。云数据库 FusionDB 基于开源Greenplum Database项目开发，提供了完善的监控、安全管理等功能，同时可以轻松实现在线扩展。云数据库 FusionDB 具有良好的兼容性，可以轻松使用PostgreSQL相关生态工具，对海量数据进行分析处理，在使用方便、高效、可靠的同时，让您更专注于业务处理。

产品特性

弹性扩容

云数据库 FusionDB 可根据用户需要，对节点进行弹性扩容，提升整体分析能力，应对业务以及数据增长等场景。通过增加节点可以线性提高系统的存储和计算能力。

高性能

云数据库 FusionDB 基于分布式大规模并行处理MPP框架，支持行列混合存储以及压缩存储，业务可按照需求选择最佳存储方案。结合高并发加载、计算能力，提供高性能处理能力。

安全可靠

云数据库 FusionDB 对数据存储两个副本，支持故障自动转移和修复，结合BCC自身安全防护，提升服务可靠性。通过白名单控制、VPC隔离，也保证了数据的访问安全。

运维方便

用户创建实例后，可快速搭建集群，无需关注集群管理和复杂的维护操作。只需通过控制台操作，即可实现基础的集群管理、监控维护等工作。

应用场景

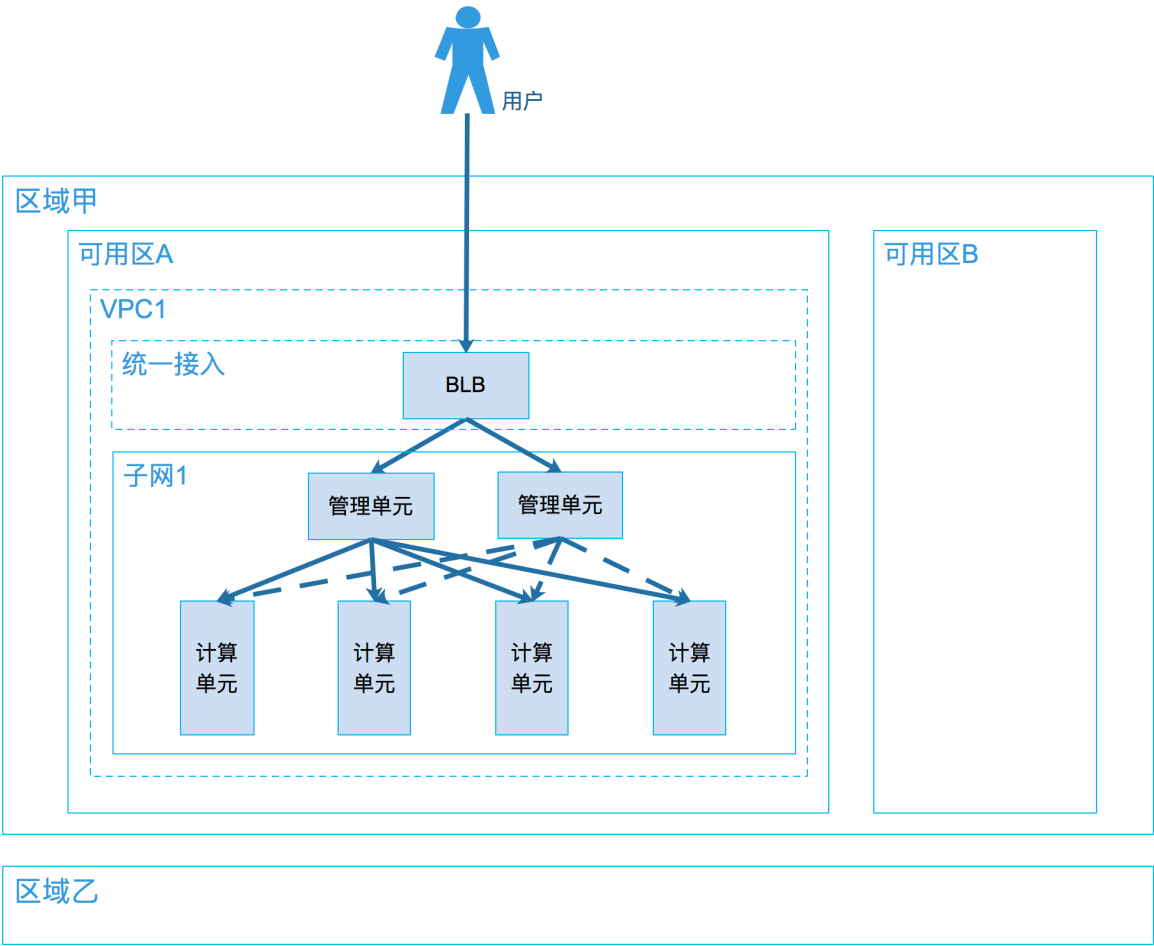
经营分析决策

在金融、电信、零售等领域，需要对订单、流水、财务等业务数据进行汇总分析，以便通过数据掌握公司经营情况以及服务情况，提高决策精度及效率。通过ETL工具将各类数据导入云数据库 FusionDB 中，结合高性能并行处理能力，协助业务快速决策。

海量日志分析

互联网、游戏等领域，各类海量的日志数据，结构化及半结构化数据，通过云数据库 FusionDB 可以使用SQL语句直接对其进行高效分析。

产品架构



实例规格

云数据库 FusionDB 根据计算单元规格不同，允许配置的计算单元个数也不相同，只提供高速SSD存储计算单元，后续开放大容量SATA存储

产品系列	CPU/核	内存/G	存储/G	计算单元个数
小套餐	1	8	80	2、4
中套餐	2	16	160	2、4、8
大套餐	16	128	1280	2个以上

当前以中小等套餐为主，如您有其他需求，请您提交工单联系百度工程师为您处理。

产品术语

管理单元

云数据库 FusionDB 中对计算单元进行管理控制的节点，用户不感知

计算单元

云数据库 FusionDB 中对数据存储和计算的实际运行节点，一个集群由多个计算单元组成，通过增加计算单元，可以线性提升集群处理能力

计算单元规格

用户可以购买的计算单元规格，包括CPU、内存、磁盘。不同计算单元规格不同，性能也不同

计算单元个数

用户购买的计算单元数量，最小个数为2个，计算单元规格不同，可购买的个数也不同

使用须知

云数据库 FusionDB 实例创建后需要对用户管理员密码进行修改才能使用，从安全角度考虑，默认情况下不允许任何IP访问，需要配置白名单才可访问。

基本功能点：

- 支持开源Greenplum Database核心功能
- 支持ORCA优化器
- 支持JSON数据类型
- 支持orafunc插件
- 支持在线扩容

限制点：

- 用户管理员账户有创建数据库、创建用户、创建表和schema权限，没有超级用户权限，但有权查看和修改非超级用户的数据，同时可终止非超级用户的连接
- 暂不支持备份恢复
- 暂未开放直接访问百度智能云对象存储BOS中的数据

典型实践

建表最佳实践

分布键

云数据库 FusionDB 在创建表时必须显式指定分布键（DISTRIBUTED RANDOMLY 或 DISTRIBUTED BY （col1, col2,...））。默认情况下，如果创建表时不指定分布键，云数据库 FusionDB 将使用第一个字段作为分布键，这种情况有时会有比较大的风险所以应该坚决杜绝这种情况。

一般来说，均匀为第一大原则，选取更有业务意义的字段，建议选一个常用于Join的字段做分布键。分布键的作用是什么？一者，是为了把数据均匀分散到所有Primary Instance；再者，是为了能够实现Local计算，尽量减少数据的shuffle。比如在做Join时，为了减少数据的重分布，只有当关联字段包含所有分布键字段的情况下，才能实现Local Join，才能保证数据减少发生shuffle。为了提升这种可能性，当然字段越少越好，只有单元素的集合才最容易成为其他集合的子集。

关于随机分布，我们需要有充分的认知，对于不太可能和其他大表进行关联的大表来说，如果我们实在找不到合适的分布键，有时我们可以使用Randomly策略，至少可以确保数据的均匀分布，当其与小表Join时，执行计划会优先选择将小表做Broadcast。分布键的选择是一个比较有技术含量的事情。对于Create时没有指定分布键又没有PK的情况，云数据库 FusionDB 将缺省使用第一个字段作为分布键，这种情况有时会有比较大的风险。

正确实例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name varchar(64)) DISTRIBUTED RANDOMLY;
```

或

```
CREATE TABLE test_table ( createdate DATA, id INT, name varchar(64)) DISTRIBUTED RANDOMLY;
```

错误示例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name varchar(64));
```

AOT

云数据库 FusionDB 中存储数据有2种存储模型：一种是默认类型的Heap表；另一种是appendonly表；

Heap表可以进行循环的Update、Delete和单行Insert操作，也可以进行并发的Update、Delete、Insert。

Appendonly表用于初始加载数据后很少更新，并且后续只是进行批量插入的操作。不推荐对Appendonly表进行单条Insert,Update,Delete操作和并发的Update,Delete操作，并发的Insert是可以进行的。由于对Appendonly进行Update和Delete的空间不会像Heap表那样被重复使用，因此Appendonly表不适合进行频繁的更新操作，Appendonly表适合于一次加载后频繁查询分析的大表。

Appendonly表相比于Heap表在数据存储时页面结构做了优化，每行记录相比于Heap表少20字节的行头信息，因此数据量更小，查询性能更好。因此可结合业务特点选择合适的。

Heap表实例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name varchar(64)) DISTRIBUTED RANDOMLY;
```

AOT表实例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name varchar(64))  
WITH (APPENDONLY=true)<br>DISTRIBUTED RANDOMLY;
```

🔗 行存&列存

行存是数据库存储的传统方式，由多列组合而成的一行连续地存储在磁盘上，因此完整的一行使用单I/O即可从磁盘读取。

列存是每列都会创建一个文件，如果表带有分区，每个分区子表的每列都会创建一个文件。当一个Query仅仅访问带有很多列的列表表中的一小部分列时，其他不相关的列不会备从磁盘检索到，因此I/O的开销相比于行存会小很多。

行存适合于传统的频繁进行更新和插入的交易类型的事务，和对于常用每个Query都涉及记录中的很多列的情况。如果Select列表和Where条件需要数据表中大多数列，则此数据表采用行存最适合。此外，由于行存在灵活性和性能的综合方面有着很好的优势，因此行存适用于通用的或混合作业类型。

列存对于读取操作做了优化，但并未对写操作做优化，一行中的不同列会被写入到磁盘的不同区域。列存对于有很多列的宽表，并且Query只访问这些列中的一小部分集合的典型应用有非常好的性能。

另一方面，列存是把相同数据类型的值存储在一起，因此相比行存占用更少的空间，也使用更少的磁盘I/O。列存表也比行存表也有更好的压缩效果。

列存适合的场景：在数据仓库的分析工作中，对于经常需要对宽表小部分列做聚合计算的场景；有单个列经常被更新并且不修改其他列的场景；

在一张宽表中读完整的一条记录时，列存相比行存需要花费更多的时间，因为每一列在每个segment上都是单独的一个文件。

最终选择行存还是列存，需要结合业务特征做最优的选择。

行存示例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name VARCHAR(64))  
WITH (APPENDONLY=true, ORIENTATION=row)  
DISTRIBUTED RANDOMLY;
```

列存实例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name VARCHAR(64))  
WITH (APPENDONLY=true, ORIENTATION=column)  
DISTRIBUTED RANDOMLY;
```

🔗 数据压缩

考虑到随着业务的发展，数据量存储的压力，对于数据量大一些的冷数据可以考虑采用压缩。只要压缩级别不是太高，都可以有效的降低存储空间，并且对查询性能不会造成太大的影响。常用的压缩选项是zlib5，但是如果提升到zlib9，可能压缩效果会提升10%+，但访问性能会成倍下降。有时候quicklz也是不错的选择，但其压缩效果不如zlib，当然访问性能也会比zlib好一些。具体生产场景可以结合业务场景做一些对比测试，如果访问是I/O瓶颈可以采用压缩存储，使用CPU来换I/O，并可以测试不同的压缩算法和级别以获得最满意的压缩选项。列压缩的压缩率比行压缩的更好，但应与表尺寸一并考虑。

对于大表来说，采用压缩存储，既节省空间也节省IO资源。长远来看还可降低阵列卡和磁盘的故障率。只有AOT表才能使用使用表压缩，Heap不能使用压缩。

压缩示例（使用Zlib方式，压缩级别为5）

```
CREATE TABLE test_table ( createdate DATA, id INT, name VARCHAR(64))
WITH (APPENDONLY=true, ORIENTATION=row, COMPRESSTYPE=zlib, COMPRESSLEVEL=5)
DISTRIBUTED RANDOMLY;
```

分区表

云数据库 FusionDB 的分区原理与其他数据库无异，合适的分区可以减少磁盘I/O操作加速查询效率，但是建议是不要把表分区设计的过于零碎，一般来说一个表的分区数量，几十、上百个就差不多了，不宜过多，对应冷数据可以做分区合并并压缩存储已节省存储空间。分区之间数据无需均匀。具体的分区策略是否合理，需要根据具体情况来评估。

分区表最好选择日期字段做为分区键，按月或者按天分区，具体可以根据业务的使用而决定。

另外，一般不建议用多级分区，多级分区维护困难，同时数据文件的数量也会剧增。

按月分区示例：

```
CREATE TABLE test_table ( createdate DATA, id INT, name VARCHAR(64))
WITH (APPENDONLY=true, ORIENTATION=row, COMPRESSTYPE=zlib, COMPRESSLEVEL=5)
DISTRIBUTED RANDOMLY
PARTITION BY range(createdate)
(
PARTITION m201801 START (date '2018-01-01') INCLUSIVE END (date '2018-02-01') EXCLUSIVE,
PARTITION m201802 START (date '2018-02-01') INCLUSIVE END (date '2018-03-01') EXCLUSIVE,
PARTITION m201803 START (date '2018-03-01') INCLUSIVE END (date '2018-04-01') EXCLUSIVE,
PARTITION m201804 START (date '2018-04-01') INCLUSIVE END (date '2018-05-01') EXCLUSIVE,
PARTITION m201805 START (date '2018-05-01') INCLUSIVE END (date '2018-06-01') EXCLUSIVE,
PARTITION m201806 START (date '2018-06-01') INCLUSIVE END (date '2018-07-01') EXCLUSIVE,
PARTITION m201807 START (date '2018-07-01') INCLUSIVE END (date '2018-08-01') EXCLUSIVE,
PARTITION m201808 START (date '2018-08-01') INCLUSIVE END (date '2018-09-01') EXCLUSIVE,
PARTITION m201809 START (date '2018-09-01') INCLUSIVE END (date '2018-10-01') EXCLUSIVE,
PARTITION m201810 START (date '2018-10-01') INCLUSIVE END (date '2018-11-01') EXCLUSIVE,
PARTITION m201811 START (date '2018-11-01') INCLUSIVE END (date '2018-12-01') EXCLUSIVE,
PARTITION m201812 START (date '2018-12-01') INCLUSIVE END (date '2019-01-01') EXCLUSIVE,
DEFAULT PARTITION other
);
```

数据类型

在建表时，各字段的数据类型的选择也是比较讲究的：

1. 使用相同数据类型的字段进行表Join操作，如果数据类型不一致，那么云数据库 FusionDB 需要对字段进行动态类型转化以确保字段可以比较，另外，由于字段之间类型不匹配，很可能会导致系统选用错误的Join方式，从而查询性能严重下降。因此在建表时需兼顾各表中字段类型的兼容。
2. 选择正确的数据类型。比如一个日期字段，可以设置为date、varchar等类型，但除非是特殊要求，否则最为合适的是选用date类型。因为在云数据库 FusionDB 中，如果使用日期字段作为过滤条件，date类型比较效率比varchar类型的比较效率要高；再者无法针对varchar类型对表进行基于时间的分区，这对大表的优化造成影响。

定期维护任务

在云数据库 FusionDB 中建议设置定期空洞回收的维护任务。

回收空洞

云数据库 FusionDB 使用MVCC机制，数据的更新操作（INSERT、UPDATE、DELETE、ALTER TABLE ADD COLUMN ）不会直接修改原数据，而是标记数据为不可用，然后插入新数据。随着不断的更新数据，数据库中不可用的数据会越来越多，造成系统性能下降，并占用大量磁盘空间，因此需要定期进行空洞回收。

VACCUUM

Vacuum都会将空洞重新标记为可用，而且不会锁表，表进行Vacuum期间不影响读写。由于Vacuum对数据库其他操作影响不大，所以建议对于频繁更新的表最好两个小时执行一次；如果更新是每天一次批量进行的，可以在每天批量更新后做一次。

VACUUM FULL

VACUUM FULL会将表中所有的空洞都进行回收，但会锁表，VACUUM FULL期间表无法读写，并且会导致CPU，磁盘IO增加。所以建议每周做一次VACUUM FULL操作。

API参考

API简介

云数据库 FusionDB 功能说明

云数据库 FusionDB，作为一套完整的分布式数据库系统，提供创建、删除、查看等实例管理，账号管理、白名单管理、监控等功能。

容许的操作

对象	可进行的操作
实例	创建、删除、查看信息
账号	查看、修改密码
IP白名单	查看、修改

调用说明

通用说明

API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。

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](#)。

🔗 幂等性

当调用某些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://bcc.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。

🔗 日期与时间规范

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2014-06-01表示2014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2014-06-01T23:00:10Z表示UTC时间2014年6月1日23点0分10秒。

🔗 公共参数

🔗 公共请求参数

通用请求参数是指每个接口都需要使用到的请求参数。

参数名	参数类型	是否必须	参数描述
Authorization	String	Yes	认证Access Key
If-Match	String	No	-
If-None-Match	String	No	-
x-bce-request-id	String	No	1) 所有请求都应该唯一地对应一个ID，用于标识该请求。 2) requestId可用于问题定位、性能分析等等多个场景。所有的日志都应该带有requestId以便后续分析。 3) requestId使用 UUID version 4 ，暂时由各服务自行生成，后续考虑统一由BFE生成
x-bce-date	String	No	请求时间，格式参考 时间与日期

🔗 公共返回参数

每次接口调用请求，无论成功失败，都包含x-bce-request-id HTTP头，调用成功系统同时会返回业务数据。对单一资源的查询请求会返回HTTP ETag头。

示例：

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
ETag: xxx-xxx-xx
Content-Type: application/json
{
  "cluster" : {
    "clusterId": "xxxjxjkp1fimpb",
    "clusterName": "xxxtest",
    "engineVersion": "5.x",
    "endpoint": {
      "port": 6600,
      "address": "xxx.xxxjxjkp1fimpb.bce.baidu.com"
    },
    "region": "bj",
    "instanceClass": "db1.large",
    "allocatedMemoryInMB": 32768,
    "allocatedStorageInGB": 320,
    "usedStorageInMB": 0,
    "instanceStatus": "available",
    "lockMode": "unlock",
    "eipStatus": "closed",
    "publiclyAccessible": false,
    "instanceCreateTime": "2017-05-15 07:05:56",
    "sliceAmount": 1,
    "instanceAmount": 2,
    "subInstanceAmount": 4,
    "azone": "default",
    "vpcId": "98c700ed-fed4-4d4e-bdd0-a17f64ead477",
    "subnetId": {
      "zoneA" : "66051292-093c-49df-8648-c18742f0c392"
    },
    "sliceList": [
      {
        "sliceId": "slice946t28kt1k",
        "usedStorageInMB" : 10,
        "status" : "available",
        "instanceIds": [
          {
            "role": "master",
            "instanceId": "5a63a4c5-5d7c-44ac-89bd-cded7ac2ec44",
            "instanceShortId": "dr-asdf1",
            "status" : "available",
            "usedStorageInMB" : 10
          },
          {
            "role": "standbymaster",
            "instanceId": "a9419e43-f3c0-4527-a995-9a27478af22c",
            "instanceShortId": "dr-asdf2",
            "status" : "available",
            "usedStorageInMB" : 10
          },
          {
            "role": "slave",
            "instanceId": "a9419e43-f3c0-4527-a995-9a27478af22c",
            "instanceShortId": "dr-asdf2",
            "status" : "available",
            "usedStorageInMB" : 10
          }
        ]
      }
    ]
  }
}
```

```
}  
}
```

返回结果

调用 API 服务后返回数据采用统一格式，返回的 HTTP 状态码为 2xx，代表调用成功；

返回 4xx 或 5xx 的 HTTP 状态码代表调用失败；

调用成功返回的数据格式为 [JSON](#)。

调用成功

调用成功返回结果中包含具体的业务数据，示例如下：

```
{  
  "instance": {  
  }  
}
```

调用失败

调用接口出错后，将不会返回结果数据。调用方可根据错误代码表来定位错误原因。当调用出错时，HTTP 请求返回一个 4xx 或 5xx 的 HTTP 状态码，返回的消息体中是具体的错误代码及错误信息和requestId。

示例:

```
{  
  "requestId": "7869616F-7A68-6977-656E-406261696475",  
  "code": "invalidAction",  
  "message": "The action or operation requested is invalid. Verify that the action is typed correctly."  
}
```

API接口说明

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。

获取实例列表

请求方法	API
GET	/json-api/v1/cluster

请求参数

参数名称	参数类型	是否必须	参数描述
< 公共请求参数 >	-	-	参考 公共请求参数

返回参数

参数名称	参数类型	参数描述
< 公共返回参数 >	-	参考 公共返回参数
clusters	List	由cluster组成的数组，cluster详细参数参考下述API

请求示例

```
GET /json-api/v1/cluster HTTP/1.1
HOST: xxx.baidu.com
Authorization: bce-auth-v1/rdsak/2014-07-03T06:13:13Z/1800/x-bce-date/{signature}
```

返回示例

```
HTTP/1.1 200 OK
x-bce-request-id: 7869616F-7A68-6977-656E-406261696475
Content-Type: application/json
[
  "clusters" : [
    {
      "clusterId": "xxxjxkp1fimpb",
      "clusterName": "xxxtest",
      "endpoint": {
        "port": 6600,
        "address": "xxx.xxxjxkp1fimpb.baidu.com",
        "vnetIp"  : "192.0.0.4",
        "inetIp" : "180.0.0.4",
      },
      "region": "bj",
      "instanceClass": "db1.large",
      "engineVeresion": "5.x",
      "allocatedMemoryInMB": 32768,
      "allocatedStorageInGB": 320,
      "usedStorageInMB": 0,
      "instanceStatus": "available",
      "lockMode": "unlock",
      "eipStatus": "closed",
      "publiclyAccessible": false,
      "instanceCreateTime": "2017-05-15 07:05:56",
      "sliceAmount": 1,
      "instanceAmount": 2,
      "subInstanceAmount": 4,
      "azone": "zoneA",
      "vpclId": "98c700ed-fed4-4d4e-bdd0-a17f6433d471",
      "subnetId": {
        "zoneA" : "66051292-093c-49df-8648-c1f744f0c302"
      },
    }
  ]
]
```

🔗 获取实例详情信息

获取集群信息，包括集群ID、集群名、连接信息、计算节点数、计算节点套餐规格（内存、存储空间）、集群状态、锁定模式、创建时间、区域等信息。

请求方法	API
GET	/json-api/v1/cluster/{clusterId}

请求参数

参数名称	参数类型	是否必须	参数描述
< 公共请求参数 \>	-	-	参考 公共请求参数

返回参数

参数名称	参数类型	参数描述
< 公共返回参数 >		参考 公共返回参数
cluster	Object	一个cluster对象

Cluster对象

参数名称	参数类型	参数描述
clusterId	String	集群id
clusterName	String	集群实例名字
endpoint	Object	集群入口信息： { "port": 6600, //数据库端口 "address": "xxx.xxxjxjkp1fimpb.baidu.com" , //数据库连接地址 "vnetIp" : "192.0.0.4", //vnet IP地址 "inetIp" : "180.0.0.4", //EIP地址 }
instanceClass	String	实例规格
allocatedMemoryInMB	Int	单个存储节点的内存
allocatedStorageInGB	Int	单个存储节点的空间
usedStorageInMB	Int	总体已用存储空间
instanceStatus	String	集群状态: creating;available;rebooting;backuping;renaming;failed;restoring;deleting;deleted
lockMode	String	集群锁定状态: unlock;lockExpiration;lockRestoration;lockDiskQuota
instanceCreateTime	String	集群创建时间
publiclyAccessible	bool	是否开放外网访问权限
eipStatus	String	公网访问开放状态
region	String	集群区域
azone	String	可用区
vpclId	String	vpc id信息
subnetId	object	子网id信息 { "zoneA":"5087d05d-3818-4dc6-b87d-8810a921c453" }
azone	String	可用区
sliceAmount	int	分片数，该值为"1"
instanceAmount	int	计算节点数
subInstanceAmount	int	单个节点上主实例个数
engineVersion	string	云数据库 FusionDB 版本："5.x"

请求示例

```
GET /json-api/v1/cluster/xxmxiaoziwen0
HTTP/1.1HOST: xxx.baidu.com
Authorization: bce-auth-v1/rdsak/2014-07-03T06:13:13Z/1800/x-bce-date/{signature}
```

返回示例

```
{
  "cluster" : {
    "clusterId": "xxjxjkp1fimpb",
    "clusterName": "xxtest",
    "engineVersion": "5.x",
    "endpoint": {
      "port": 6600,
      "address": "xxx.xxjxjkp1fimpb.baidu.com",
      "vnetIp" : "192.0.0.4",
      "inetIp" : "180.0.0.4",
    },
    "region": "bj",
    "instanceClass": "db1.large",
    "allocatedMemoryInMB": 32768,
    "allocatedStorageInGB": 320,
    "usedStorageInMB": 0,
    "instanceStatus": "available",
    "lockMode": "unlock",
    "eipStatus": "closed",
    "publiclyAccessible": false,
    "instanceCreateTime": "2017-05-15 07:05:56",
    "sliceAmount": 1,
    "instanceAmount": 2,
    "subInstanceAmount": 4,
    "azone": "default",
    "vpclId": "98c700ed-fad4-424e-bdd0-a17f64ead471",
    "subnetId": {
      "zoneA" : "66051292-493c-4ddf-8648-c18742f0c302"
    },
  },
  "sliceList": [
    {
      "sliceId": "slice946t28kt1k",
      "usedStorageInMB" : 10,
      "status" : "available",
      "instanceIds": [
        {
          "role": "master",
          "instanceId": "5a63a4c5-5d7c-44ac-89bd-cded7ac2ec44",
          "instanceShortId": "dr-asdf1",
          "status" : "available",
          "usedStorageInMB" : 10
        },
        {
          "role": "standbymaster",
          "instanceId": "a9419e43-f3c0-4527-a995-9a27478af22c",
          "instanceShortId": "dr-asdf2",
          "status" : "available",
          "usedStorageInMB" : 10
        },
        {
          "role": "slave",
          "instanceId": "a9419e43-f3c0-4527-a995-9a27478af22c",
          "instanceShortId": "dr-asdf2",
          "status" : "available",
          "usedStorageInMB" : 10
        }
      ]
    }
  ]
}
```

```
    }
  ]
}
]
}
}
```

创建实例

请求方法	API
POST	/json-api/v1/cluster

创建集群，需要选定计算节点数、计算节点套餐等信息。

请求参数

参数名称	参数类型	是否必须	参数描述
< 公共请求参数 >	-	-	参考 公共请求参数
orderId	String	Yes	订单号
clusterName	string	Yes	集群名字
instanceClass	string	Yes	套餐名称
allocatedMemoryInMB	Int	Yes	单个节点内存大小
instanceAmount	int	Yes	计算节点数
subInstanceAmount	int	Yes	单个节点上主实例个数（当前对于小套餐，此参数值是1；中套餐，此参数值是2）
vpId	String	Yes	vpId
subnetId	Object	Yes	子网id信息 { "zoneA": "5087d05d-3818-4dc6-b87d-8810a921c453" }
azone	string	Yes	可用区
engineVersion	string	Yes	云数据库 FusionDB 版本："5.x"

返回参数

参数名称	参数类型	参数描述
< 公共返回参数 >	-	参考 公共返回参数
clusters	List	由cluster对象组成的数组

请求示例

```
POST /json-api/v1/cluster HTTP/1.1
HOST: xxx.baidu.com
Authorization:bce-auth-v1/rdsak/2014-07-03T06:13:13Z/1800/x-bce-date/{signature}

{
  "orderId": "20170607012312400345676",
  "clusterAmount": 1,
  "clusterParameters":{
    "clusterName": "xxxtest",
    "engineVersion": "5.x",
    "instanceClass": "db1.large",
    "allocatedMemoryInMB": 32768,
    "instanceAmount": 2,
    "subInstanceAmount": 4,
    "vpclId": "98c700ed-fed4-4d56-bdd0-a17f64ead471",
    "subnetId": {
      "azoneA": "66033292-093c-49d4-8648-c18742f0c302"
    }
  }
}
```

返回示例

请求成功：

```
{
  "clusters": [
    {
      "clusterId": "xxxm59yzsvk81",
      "clusterName": "xxxtest",
      "endpoint": {
        "port": 6600,
        "address": "xxx.xxxjxkp1fimpb.baidu.com",
        "vnetIp": "192.0.0.4",
        "inetIp": "180.0.0.4",
      },
      "region": "bj",
      "instanceClass": "db1.large",
      "allocatedMemoryInMB": 32768,
      "allocatedStorageInGB": 320,
      "usedStorageInMB": 0,
      "instanceStatus": "creating",
      "lockMode": "unlock",
      "eipStatus": "closed",
      "publiclyAccessible": false,
      "instanceCreateTime": "2017-06-07 05:56:57",
      "sliceAmount": 1,
      "instanceAmount": 2,
      "subInstanceAmount": 4,
      "azone": "azoneA",
      "vpclId": "983740ed-fed4-4d4e-bdd0-a17f64ead471"
    }
  ]
}
```

请求失败：


```
{
  "requestId":"35663333-3033-3766-3536-373564633661",
  "code":"ClusterInstancesExist",
  "message":"Cluster Instance is exist."
}
```

删除实例

请求方法	API
DELETE	/json-api/V1/cluster/{clusterId}

集群当前lockMode必须为lockExpiration。

请求参数

参数名称	参数类型	是否必须	参数描述
< 公共请求参数 >	-	-	参考 公共请求参数

返回参数

参数名称	参数类型	参数描述
< 公共返回参数 >	-	参考 公共返回参数

请求示例

```
DELETE /json-api/v1/cluster/testclusterId HTTP/1.1
HOST: xxx.baidu.com
Authorization: bce-auth-v1/rdsak/2014-07-03T06:13:13Z/1800/x-bce-date/{signature}
```

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
```

常见问题

常见问题总览

使用类问题

- [申请了云数据库 FusionDB 实例，默认账号的密码是什么？](#)
- [云数据库 FusionDB 可以通过SQL语句或客户端工具进行数据库账号的创建和删除吗？](#)
- [如何使用列存和压缩？](#)
- [如何使用表分区？](#)
- [如何查看数据库启动时间？](#)

运维类问题

- [如何通过工具或者代码访问数据库？](#)
- [脚本怎样免密连接数据库实例？](#)
- [如何查看数据库的参数信息？](#)
- [如何修改参数？](#)
- [如何查看当前数据库的连接信息？](#)
- [无法连接云数据库 FusionDB，如何解决？](#)
- [如何查看某个database的空间占用？](#)
- [如何查看某个schema的空间占用？](#)
- [如何查看一个表的空间占用？](#)
- [在云数据库 FusionDB 删除数据库时，提示有连接在使用该库，该怎么解决？](#)
- [在云数据库 FusionDB 中运行一个SQL语句，一直不返回结果，运行时间远超预估时间，怎么处理？](#)
- [删除数据后无法释放空间，如何处理？](#)
- [常见异常问题排查和处理方法](#)

🔗 性能类问题

- [索引建立有什么建议吗？](#)
- [怎样选择表的分布键？](#)
- [对于建表有什么建议呢？](#)
- [有哪些SQL优化建议？](#)

使用类问题

🔗 申请了云数据库 FusionDB 实例，默认账号的密码是什么？

默认账号是useradmin，初始密码不提供，需要到控制台->账号管理中修改密码后使用。

🔗 云数据库 FusionDB 可以通过SQL语句或客户端工具进行数据库账号的创建和删除吗？

用户可以使用useradmin账号通过SQL语句或客户端工具进行数据库账号的创建和删除以及给账号授权，管理控制台不提供该功能。

🔗 如何使用列存和压缩？

在云数据库 FusionDB 中创建表时，默认使用行式存储，并且不启用压缩。如需使用列存和压缩功能，您必须在建表时指定列存和压缩选项。

例如，可以在建表语句中加入以下子句，来启用列存和压缩功能。 `with (APPENDONLY=true, ORIENTATION=column, COMPRESSTYPE=zlib, COMPRESSLEVEL=5, BLOCKSIZE=1048576, OIDS=false)` 一般情况下，推荐使用列存和压缩功能，尤其是在包含较多复杂查询，或需要降低存储成本的场景。

🔗 如何使用表分区？

对于数据库中的事实表以及一些比较大的表，建议使用表分区。使用表分区功能，方便您定期进行数据删除（通过 `alter table drop partition` 命令可删除整个分区的数据）和导入（使用交换分区的方式，即 `alter table exchange partition` 命令可以加入新数据分区）。

云数据库 FusionDB 支持 Range Partition（范围分区）、List Partition（列表分区）。但是，Range Partition 只支持利用数值或时间类型的字段来分区。以下是一个使用 Range Partition 的表的例子

```
create table edw_dws.dws_customer_detail_decrypt_new(like edw_dws.dws_customer_detail_decrypt) WITH
(appendonly=true,COMPRESSTYPE=ZLIB,COMPRESSLEVEL=5) Distributed randomly PARTITION BY RANGE(stat_dt) (
PARTITION y2017 START (date '2017-01-01') INCLUSIVE END (date '2018-01-01') EXCLUSIVE , PARTITION m201801
START (date '2018-01-01') INCLUSIVE END (date '2018-02-01') EXCLUSIVE , DEFAULT PARTITION other );
```

🔗 如何查看数据库启动时间？

```
select pg_postmaster_start_time();
```

运维类问题

🔗 如何通过工具或者代码访问数据库？

云数据库 FusionDB 支持多种方式进行实例的连接，主要包含如下方式：

1. pgadmin是开源图形化管理工具，可以对监控数据、执行计划等图形化展示；
2. psql是一种命令行的连接方式；
3. C/C++ 程序可以使用ODBC接口；
4. Java程序可以使用JDBC接口；
5. Perl脚本语言可以使用DBI连接接口；
6. python通过模块psycopg2来实现; 以psql客户端为例介绍连接实例的方法，其他客户端可参见此方法。

使用 psql 客户端运行如下命令，根据提示输入密码，连接数据库。

```
psql -U username -h hostname -p port dbname Password for user username: psql (8.3.23) Type "help" for help.
dbname=#
```

参数说明如下：

- username：初始账号用户名
- hostname：实例地址
- port：实例端口号
- dbname：要连接的数据库名称 如：psql -U myuser -h FusionDB.xxxxxx.whlocal.com -p 6600 dbname

🔗 脚本怎样免密连接数据库实例？

1. 在调用脚本的用户的home目录下新建“.pgpass”文件，并写文件内容为：“hostname:port:database:username:password”。
eg: 以work用户调用脚本，则在/home/work/目录下新建“.pgpass”文件，文件内容为：“FusionDB.xxxxxx.whlocal.com:6600:postgres: myname:bfechib578VTF”（如果是相连接不同的数据库，可以写多条，每行一条）。
2. 给文件的修改使用权限为0600：chmod 0600 .pgpass。
3. 使用“psql -U myuser -h FusionDB.xxxxxx.whlocal.com -p 6600 dbname”登录数据库不需要输入密码（只有home目录下配有“.pgpass”文件的用户才可以免密码进入）。

🔗 如何查看数据库的参数信息？

连接数据库实例后执行SQL语句show all可查看数据库的参数信息。

如何修改参数？

目前，本产品不支持全局地修改参数，但是用户可以在连接内修改参数（具体参考 云数据库 FusionDB 的参数修改限制）。可以使用命令 `ALTER ROLE <用户名> SET <参数名> = <参数值>` 来修改参数。该参数修改只针对指定的用户生效。

如何查看当前数据库的连接信息？

连接数据库实例后执行SQL语句 `select * from pg_stat_activity` 可查看连接信息，结果集会显示出当前连接的数据库名，用户，IP地址，连接开始时间，查询的语句等。

无法连接云数据库 FusionDB，如何解决？

情况1:

问题描述

连接数据库实例，报错如下：FATAL: remaining connection slots are reserved for non-replication superuser connections

原因

这是由于连接数满了导致。

解决办法

1. 查看当前连接数的限制 `show max_connections;`
2. 查看当前数据库的连接 `select * from pg_stat_activity;`
3. 使用如下SQL终止连接 `SELECT pg_terminate_backend(pid);` 其中pid指进程号，具体是pg_stat_activity视图中pid字段的值。

情况2:

问题描述

连接数据库实例报错如下 psql: FATAL: no pg_hba.conf entry for host "10.94.20.24", user "xxx", database "fsgrim", SSL off

原因

授权问题，未添加ip白名单导致。

解决办法

rds控制台->账号管理中添加ip白名单 10.94.20.24 即可。

如何查看某个database的空间占用？

连接数据库实例后执行以下SQL语句可查看数据库的空间占用情况（单位MB）。

```
select datname,pg_size_pretty(pg_database_size(datname)) from pg_database;
```

如何查看某个schema的空间占用？

查询一个Schema下面的所有表的总大小（单位MB，包括索引和数据）；`select schemaname`

```
,round(sum(pg_total_relation_size(schemaname||'.'||tablename))/1024/1024) "Size_MB" from pg_tables where  
schemaname='<schemaname>' group by 1;
```

如何查看一个表的空间占用？

查询一张非分区表的总大小（单位为MB，包含表的索引和数据）：`select pg_size_pretty(pg_total_relation_size('.'));` 查询分区

表所有分区的总大小（单位MB，包含表的索引和数据）：`select`

```
schemaname,tablename,round(sum(pg_total_relation_size(schemaname || '.' || partitiontablename))/1024/1024) "MB" from  
pg_partitions where schemaname=' ' and tablename=' ' group by 1,2;
```

🔗 在云数据库 FusionDB 删除数据库时，提示有连接在使用该库，该怎么解决？

问题描述

在云数据库 FusionDB 删除数据库提示如下：

```
ERROR: database "test" is being accessed by other users DETAIL: There is 1 other session using the database.
```

原因

这是因为当前有连接在使用该库。

解决办法

在删除数据库之前首先断开所有连接该库的连接。 查看所有连接该库的连接：`SELECT * FROM pg_stat_activity WHERE datname = 'db_name';` 确定以上的连接是否可以断开，若是则执行：`SELECT pg_terminate_backend(pid) FROM (select pid from pg_stat_activity where datname = 'db_name' ) t;` 断开所有连接后再执行：`drop database db_name;`

🔗 在云数据库 FusionDB 中运行一个SQL语句，一直不返回结果，运行时间远超预估时间，怎么处理？

分析原因

数据库中更有可能发生死锁。

通过使用以下语句查看系统当前执行语句中是否含有waiting字段为t的语句：`SELECT * FROM pg_stat_activity;` 更加直观地可以使用 `SELECT * FROM pg_stat_activity WHERE waiting = 't';` 查询锁状态的执行语句。

解决办法

首先查看数据库当前运行最长时间的任务，看看是否阻塞了其他SQL语句；`select * from pg_stat_activity where current_query != '' order by query_start asc limit 1;` 如果当前存在大量的任务处于锁状态，应该终止相应的长事务进程：`SELECT pg_terminate_backend(pid)` 其中pid指进程号，具体是pg_stat_activity视图中pid字段的值。

🔗 删除数据后无法释放空间，如何处理？

问题描述

在云数据库 FusionDB 使用delete语句清空表数据或者删除部分数据后，表的大小无变化，所占用的磁盘空间无法释放。 **分析原因**

在云数据库 FusionDB 中，使用 DELETE 语句并不会真正的删除数据记录，而是简单地将其标记为不可见，并不会释放数据所占用的空间；同样，UPDATE语句也不会真正的修改原数据，而是将原数据标记为不可见，然后将更新后的数据记录插入到表尾。所以频繁进行删除更新操作的表会越来越大。

解决办法

1. 如果想清空表数据，建议直接使用TRUNCATE语句实现，该语句会直接删除表的数据文件，所以会马上释放空间；
2. 使用VACUUM table_name;回收不可见的空间，但只是简单地回收空间并且令其可以再次使用，空间并不返回到操作系统（在大多数情况下）；
3. 如果想真正释放不可见的空间，可以使用VACUUM FULL table_name;来实现，但该方法会对表加上排它锁，以及会消耗一部分资源，建议在数据库维护阶段操作。

🔗 常见异常问题排查和处理方法

🔗 事务被终止

在执行 SQL 时，偶尔会遇到如下错误：

```
current transaction is aborted, commands ignored until end of transaction block.
```

该错误表示，当前事务被终止，执行的命令被数据库回滚，一般是事务中有 SQL 执行报错，导致当前事务回滚，如果不是 SQL 的问题，可能是网络问题，或者集群中某个节点太慢导致系统不稳定，一般重新运行即可。

🔗 网络异常错误

在执行sql时，偶尔会遇到如下错误：

```
Error on receive from seg2 slice1 yq01-dba-rimgemini-gp-s03.yq01:40001 pid=1987: server closed the connection unexpectedly
```

子节点执行报错，服务连接异常终止，可以重新运行试试。如果不能解决问题，则可能 segment 上出现了问题，要检查 segment 的日志文件进行问题排查。

🔗 内存不足

内存不足报错：

```
insufficient memory reserved for statement
```

一般是当前系统运行了太多 SQL，或者是某一个 SQL 消耗太大，占用了大量的内存导致其他 SQL 运行报错，一般的做法是

1. 减少并发 SQL 数量，比如从业务层面控制，或者从资源队列的层面控制；
2. 查看pg_stat_activity表中哪个 SQL 执行时间比较长，一般消耗资源多的也是执行时间相对比较长的，将找到的 SQL 终止掉。

🔗 不能对分布键执行 Update

对表进行 Update 的时候报错：

```
ERROR: Cannot parallelize an UPDATE statement that updates the distribution columns
```

分布键不能被更新，因为这会导致数据重分布，如果分布键一定要更新，解决办法就是新建一张表，然后将原始表数据导入到新表（可在导入的时候边导入边处理）中，从而实现 update 的操作。

🔗 占用的资源超过了资源队列限制

当用户执行的 SQL 超过了资源队列限制的大小，会报如下的错误：

```
statement require more resource than resource queue allows
```

此时 SQL 不能执行，出现这个报错一般是 SQL 写的有问题，检查下是否 SQL 过于复杂，或者产生了笛卡尔积，解决方法如下：

- 增加该用户所在的资源队列的大小限制
- 使用超级用户执行该 SQL，超级用户没有资源队列的限制
- 改写 SQL，将复杂的 SQL 拆分成多个小 SQL 执行，减少一次 SQL 的消耗 检查 SQL 是否正确，优化 SQL 使用更少的资源

性能类问题

🔗 索引建立有什么建议吗？

云数据库 FusionDB 中支持两类索引，分别是B树索引和位图索引。但在云数据库 FusionDB 中不推荐使用索引，原因有三个：

1. 云数据库 FusionDB 一般需要访问大数据集，并且云数据库 FusionDB 可以通过分区、并行执行等方式提升查询效率。索引的作用不能有效发挥。
2. 拖慢数据入库的性能。因为数据仓库经常需要导入大量数据，这个时候需要调整索引结构，导入速度极差。
3. 需要占用额外的磁盘存储空间。因此，在OLAP场景中的大多数情况不建议使用。比如频繁插入更新的字段、重复度高的字段均不建议创建索引。少数场景可以考虑建立索引：比如经常需要按照类似主键的字段从几十亿的数据中就查数百条记录，这种场景，索引能够有效地提升查询性能；或者有少数表希望能像OLTP场景记录信息，数据量较小，可以创建主键索引保证数据的唯一。

☞ 怎样选择表的分布键？

在考虑分布策略的时候首要原则就是要保证数据在每个节点尽可能分布均匀，其次是尽可能减少数据重分布操作带来的资源消耗。所以提出了以下三个原则。

1. 如果后续的SQL没有group by或者join等会引起数据重分布的操作，此时选择随机分布策略；
2. 如果需要经常做group by或者join这样的操作，我们就要慎重考虑分布键，先考虑下做join操作的联合建或者group by字段是否可以使数据分布均匀。
3. 如果找不到这种分布键，就选择随机分布。还有一点需要特别注意的是，如建表没有指定具体的分布策略，GP将会默认选择主键作为分布键，如果没有指定主键，将会默认选择第一列作为分布键。很容易导致严重的数据倾斜。

☞ 对于建表有什么建议呢？

1. 分布键选取正确，按照上个问题进行慎重选择。
2. 存储模型的选择：一种是默认类型的Heap表，Heap表可以进行循环的Update、Delete和单行insert操作，也可以进行并发的Update、Delete、Insert；另一种是Appendonly表，Appendonly表用于初始加载数据后很少更新，并且后续只是进行批量插入的操作。
3. 行存&列存的选择：默认使用行式存储。行存适合于传统的频繁进行更新和插入的交易类型的事务，和对于常用每个Query都涉及记录中的很多列的情况。如果Select列表和Where条件需要数据表中大多数列，则此数据表采用行存最适合；列存适用于宽表。1) 对于经常需要对小部分列做聚合计算的场景。2) 有单个列经常被更新并且不修改其他列的场景。
4. 是否数据压缩：考虑到随着业务的发展，数据量存储的压力，对应数据量大一些的冷数据可以考虑采用压缩。只要压缩级别不是太高，都可以有效的降低存储空间，并且对查询性能不会造成太大的影响。常用的压缩选项是zlib5，但是如果提升到zlib9，可能压缩效果会提升10%+，但访问性能会成倍下降。
5. 是否分区：云数据库 FusionDB 的分区原理与其他数据库无异，合适的分区可以减少磁盘I/O操作加速查询效率，但是建议是不要把表分区设计的过于零碎，一般来说一个表的分区数量，几十个就差不多了，不宜过多，对应冷数据可以做分区合并并压缩存储已节省存储空间。分区之间数据无需均匀。

☞ 有哪些SQL优化建议？

- 数据类型转换，必须要显式转换CAST('string' as 类型)；
- 用LEFT JOIN代替 NOT IN；
- 尽量使用外连接的方式代替嵌套子查询；
- 查询记录数过多时，限制返回条数，如limit 100；
- 两个表之间的join字段类型必须一致；
- join字段尽可能的是表的分布键，以避免数据重分布；
- 使用group by代替distinct；
- 尽量避免创建索引，且索引对于多表的join是没有性能提升的。

