

ALIBABA CLOUD

阿里云

云原生数据仓库 AnalyticDB  
PostgreSQL 版  
实例管理

文档版本：20201120

 阿里云

## 法律声明

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

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

# 通用约定

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

# 目录

1. 按量转预付费	05
2. SQL 审计	06
3. 实例告警	07
4. 重启实例	09
5. 网络连接	10
5.1. 管理外网地址	10
5.2. 释放外网地址	11
6. 版本管理	13
6.1. 版本升级	13
6.2. 4.3版升级6.0版不兼容项检查参考指南	14
6.3. 4.3版本与6.0版本兼容性注意事项	20
6.4. 扩容/升级实例规格	23
7. 标签	25
7.1. 创建实例标签	25
7.2. 删除实例标签	25
7.3. 实例标签筛选	26
8. 释放实例	27

# 1.按量转预付费

购买按量付费的实例后，您可以根据需求将实例改变成包年包月的计费方式。

## 注意事项


- 包年包月的实例无法转成按量付费的实例，在您进行计费方式的转变前请务必考虑清楚，以免造成资源浪费。
- 在计费周期内，包年包月的实例只支持升级配置，不支持降级配置或者释放。
- 变更实例计费方式成功后，实例会即刻按照包年包月的实例计费。
- 按量付费实例变更为包年包月时会产生一个新购订单，您必须完成该订单的支付流程，计费方式的变更才能生效。若未支付或未成功支付，您的[订单管理](#)页面将会出现未完成订单，之后您将无法新购实例或变更实例计费类型。

## 变更条件

- 实例计费类型为按量付费，且实例状态为运行中。
- 实例没有未完成的变更计费方式（即新购）的订单。

## 操作步骤

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 找到目标实例，在操作栏中，单击[按量转包年包月](#)。
3. 在确认订单页面，选择[购买时长](#)，勾选《AnalyticDB for PostgreSQL服务协议》，单击去开通。
4. 在[订单信息](#)页面，确认订单信息，单击[确认支付](#)。

 **说明** 此时系统会生成一个转包年包月的订单。若该订单未支付或作废，将导致您无法进行新购实例或转包年包月的操作。您可以在[订单管理](#)页面支付或作废该订单。

## 2.SQL审计

AnalyticDB for PostgreSQL控制台提供SQL审计功能，您可通过该功能查看SQL明细、定期审计SQL。开通SQL审计功能后，实例性能不会受到影响。

### 背景信息

SQL审计会统计所有DML和DDL操作信息，这些信息是系统通过网络协议分析所得，在SQL查询量较大的时候可能会丢失少量记录。

### 注意事项

- SQL审计记录的保存时间为30天。
- 实例默认关闭SQL审计功能。开启该功能后，实例会产生额外费用，详细收费标准请参见[云数据库AnalyticDB for PostgreSQL详细价格信息](#)。

### 操作步骤

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 在实例列表上方选择实例所在地域。
3. 找到目标实例，单击实例名称。
4. 在实例详情页，单击左侧导航栏的[数据安全性](#)。
5. 选择SQL审计页签，单击开启SQL审计。
6. 在弹出的对话框中单击确定，开启SQL审计功能。
  - 在开始SQL审计功能后，您可通过时间、DB、User、关键字等条件查询SQL信息。
  - 您还可在SQL审计页签，通过关闭SQL审计按键关闭已经开通的SQL审计。

### 相关API

API	描述
<a href="#">ModifySQLCollectorPolicy</a>	开启或关闭指定实例的SQL采集功能

## 3. 实例告警

本文介绍如何通过云监控控制台为AnalyticDB for PG设置报警规则。

### 背景信息

监控报警是通过[阿里云监控](#)产品实现的。通过阿里云监控产品，您可以设置监控项，并在触发监控项的报警规则时，通知报警联系组中的所有联系人。您可以维护报警监控项对应的报警联系组，以便发生报警时，能及时通知到相关联系人。

### 操作步骤

1. 登录[云监控平台](#)
2. 单击左侧导航栏中的事件监控，进入事件监控页面。
3. 在报警规则页签下，单击右上角的创建事件报警，弹出创建/修改事件报警对话框，按照页面提示进行参数配置。

### 参数说明

参数	说明
报警规则名称	设置报警规则的名称，由字母、数字、下划线（_）组成，长度不超过30个字符
事件类型	选择系统事件
产品类型	选择云原生数据库PostgreSQL版
事件类型	选择全部类型
事件等级	选择全部等级
事件名称	选择全部事件 或者 具体的告警事件
资源范围	选择全部资源，任何资源发生相关事件，均会按照配置发送通知给报警联系人。
-	勾选报警通知
联系人组	设置报警联系人组关于报警联系人/报警联系组，请参见 <a href="#">创建报警联系人或报警联系组</a> 和 <a href="#">删除报警联系人或报警联系组</a>
通知方式	设置报警的通知方式

### 告警事件说明

名称	说明
计算组CPU水位超过90%	
计算组内存水位超过85%	

最大计算组存储水位超过80%	
实例服务不可用	实例无法连接或者处理任何请求代表实例不可用。当实例在重启,创建,删除,锁定,升级场景下不会触发告警
长事务数大于等于5	存活时间大于2小时,非idle状态且持有锁的事务定义为长事务
Master节点CPU水位超过90%	
Master节点内存水位超过85%	
实例连接数超过80%	

4. 完成上述参数配置后,单击**确定**创建报警规则。



## 4. 重启实例

您可以通过AnalyticDB for PostgreSQL控制台重启实例。本文将为您介绍如何重启实例。

### 注意

重启实例时，实例将不可用，请在业务低谷时执行该操作。

### 操作步骤

您可以按照下面的步骤来重启实例：

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 选择要操作实例所在的地域。例如，华东1。
3. 单击目标实例右侧操作栏中的**管理**按钮，进入基本信息页面。
4. 单击页面右上角的**重启实例**，并在确认框中单击**确定**。如果您绑定了手机，还需要进行手机验证码验证。

### 注意

重启过程一般耗时3到30分钟，在此过程中该实例不能对外提供服务，请您提前做出调整。当实例重启结束，对应实例恢复运行中状态，您可以正常访问数据库。

完成了上述操作后，您可以回到控制台查看目标实例的运行状态。如果实例重启操作完成，则实例运行状态为运行中，否则为重启中。

### 相关API

API	描述
<a href="#">RestartDBInstance</a>	重启实例

## 5. 网络连接

### 5.1. 管理外网地址

如果您的应用部署在与您的AnalyticDB for PostgreSQL实例在同一地域且网络类型相同的ECS上，则无需申请外网地址。如果您的应用部署在与您的AnalyticDB for PostgreSQL实例在不同地域或网络类型不同的ECS或者阿里云以外的系统上，需申请外网地址，用于对接应用。

 **说明** 只要在同一地域内（可用区可以不同）且网络类型相同的实例，就可以通过内网互通。

#### 应用场景

内外网地址的使用场景如下所示：

- 单独使用内网地址：
  - 适用于应用部署在与您的AnalyticDB for PostgreSQL实例在同一地域的ECS上且实例与ECS的网络类型相同时。
- 单独使用外网地址：
  - 适用于应用部署在与您的AnalyticDB for PostgreSQL实例在不同地域的ECS上时。
  - 适用于应用部署在阿里云以外的系统上时。
- 同时使用内外网地址：
  - 适用于应用中有些模块部署在与您的AnalyticDB for PostgreSQL实例同一地域且网络类型相同的ECS上，有些模块又部署在与您的AnalyticDB for PostgreSQL实例在不同地域的ECS上时。
  - 适用于应用中有些模块部署在与您的AnalyticDB for PostgreSQL实例在同一地域且网络类型相同的ECS上，有些模块又部署在阿里云以外的系统上时。

#### 注意事项

- 在访问数据库前，您需要将访问数据库的IP地址或者IP段加入白名单，操作请参见[设置白名单](#)。
- 使用外网地址会降低实例的安全性，请谨慎选择。为了获得更快的传输速率和更高的安全级别，建议您将应用迁移到与您的AnalyticDB for PostgreSQL在同一地域的ECS上。

#### 申请外网地址操作步骤

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 在实例列表上方选择实例所在地域。
3. 找到目标实例，单击需要创建外网地址的实例ID。
4. 在基本信息页面，单击[申请外网地址](#)，跳转到数据库连接页面。也可直接单击左侧导航的[数据库连接](#)。
5. 在数据库连接页面，单击[申请外网地址](#)。
6. 在弹出的对话框中，单击[确定](#)，生成外网地址。

生成外网地址后，可在数据库连接页面，通过[释放外网地址](#)按钮释放外网地址。

#### 释放外网地址操作步骤

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。

2. 在实例列表上方选择实例所在地域。
3. 找到目标实例，单击需要释放外网地址的实例ID。
4. 单击导航栏左侧的数据库连接。
5. 在数据库连接页面，单击释放外网地址。  
如未申请外网地址，数据库连接页面仅有申请外网地址按键。
6. 在弹出的对话框中，单击确定，释放外网地址。

## 相关API

API	描述
<a href="#">AllocateInstancePublicConnection</a>	申请实例的外网地址
<a href="#">ReleaseInstancePublicConnection</a>	释放实例的外网连接串

## 5.2. 释放外网地址

申请外网地址之后，如果网络环境发生变化，用户不再需要用外网地址连接实例，可在AnalyticDB for PostgreSQL控制台释放外网地址。释放外网地址之后，请注意更改涉及该地址的应用设置。

在释放外网地址之前，请先阅读以下应用场景。

### 应用场景

内外网地址的使用场景如下所示：

- 单独使用内网地址：
  - 适用于应用部署在与您的AnalyticDB for PostgreSQL实例在同一地域的ECS上且实例与ECS的网络类型相同时。
- 单独使用外网地址：
  - 适用于应用部署在与您的AnalyticDB for PostgreSQL实例在不同地域的ECS上时。
  - 适用于应用部署在阿里云以外的系统上时。
- 同时使用内外网地址：
  - 适用于应用中有些模块部署在与您的AnalyticDB for PostgreSQL实例同一地域且网络类型相同的ECS上，有些模块又部署在与您的AnalyticDB for PostgreSQL实例在不同地域的ECS上时。
  - 适用于应用中有些模块部署在与您的AnalyticDB for PostgreSQL实例在同一地域且网络类型相同的ECS上，有些模块又部署在阿里云以外的系统上时。

### 操作步骤

1. 登录云数据库AnalyticDB for PostgreSQL管理控制台。
2. 在实例列表上方选择实例所在地域。
3. 找到目标实例，在操作栏中，单击管理。
4. 单击导航栏左侧的数据库连接。
5. 在数据库连接页面，单击释放外网地址。  
如未申请外网地址，数据库连接页面仅有申请外网地址按键。

6. 在弹出的对话框中，单击**确定**，释放外网地址。


## 相关API


API	描述
<a href="#">ReleaseInstancePublicConnection</a>	释放实例的外网连接串

## 6. 版本管理

### 6.1. 版本升级


为了更好的满足您的需求，AnalyticDB for PostgreSQL 会不断更新数据库内核版本。在实例创建时，默认使用的是最新版本的数据库内核，当新的版本发布之后，您可以通过小版本升级实例来更新数据库内核版本，从而使用新版本中扩展的功能。本文将介绍如何升级小版本。

 **说明** 当前控制台只支持小版本升级，尚不支持 4.3 到 6.0 的大版本升级。如果您需要进行 4.3 到 6.0 的大版本升级，请先了解[4.3版本与6.0版本兼容性注意事项](#)和[4.3版升级6.0版不兼容项检查参考指南](#)。

 **说明** 实例升级小版本时，会执行实例重启操作，在重启期间实例将不可用，请在业务低谷时执行该操作。

您可以按照如下步骤来升级实例小版本：

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 选择要操作实例所在的地域。例如，华东1。
3. 单击目标实例右侧操作栏中的**管理**
4. 单击页面右上角的**小版本升级**，并在确认框中单击**确定**。如果您绑定了手机，还需要进行手机验证码验证。

 **说明** 小版本升级过程一般耗时3到30分钟，在此过程中该实例不能对外提供服务，请您提前做好调整。当实例结束，对应实例恢复运行中状态，您可以正常访问数据库。

完成了上述操作后，您可以回到控制台查看目标实例的运行状态。如果实例小版本升级操作完成，则实例运行状态为**运行中**，否则为**版本升级中**。小版本升级时会检查您的实例当前的版本号，如果您的实例已经是当前最新的小版本，实际的升级和重启实例过程将会被跳过。

#### 查看当前版本信息方法

您可以通过连接实例，执行 `show rds_release_date;` 来查看当前实例的release的小版本号。

```
postgres=# show rds_release_date;
 rds_release_date
-----
20190603
(1 row)
```

如您需要知道当前版本是否是最新版本，或者查看版本特性，可至[版本发布记录](#)查看发布记录。

#### 相关API

API	描述
UpgradeDBInstance	升级内核小版本

## 6.2. 4.3版升级6.0版不兼容项检查参考指南

云原生数据仓库AnalyticDB PostgreSQL（简称ADB PG）集群从4.3版本升级到6.0版本时存在一些不兼容项，需要您手工进行预处理。本文以Linux环境下连接ADB PG 4.3版实例为例，介绍了通过执行shell脚本检查常规不兼容项的方法。Windows环境可参考脚本SQL语句执行检查。

### 注意事项

业务SQL、自定义存储过程或函数、自定义视图不包含在检查项目内，您需要根据实际情况在ADB PG 6.0实例中进行验证。

### shell脚本参考

```
#!/bin/bash
#
# Copyright (c) 2020, Alibaba Group, Inc.
#
# Description: check unsupported items before upgrade instance.
# Usage:   sh 4x_to_6x_check.sh <PGHOST> <PGPORT> <PGUSER> <PGPASSWORD>
# CheckList:
#   1) 检查实例版本
#   2) 检查libraries
#   3) 检查表分布键
#   4) 检查表字段类型
#   5) 检查扩展模块
# Notice: 如果提示错误信息，请人工修改数据库对应项。
#
if [[ $# -lt 4 ]]; then
    echo "Usage: $0 <PGHOST> <PGPORT> <PGUSER> <PGPASSWORD>"
    exit 1
fi
export PGHOST=$1
export PGPORT=$2
export PGUSER=$3
export PGPASSWORD=$4
db_ver=`psql -d postgres -c "copy (select version()) to stdout"`
db_names=`psql -d postgres -c "copy (select soddname from gp_toolkit.gp_size_of_database) to stdout"`
db_names=${db_names}
db_len=${#db_names[@]}
```

```
unsupport6x_ext="(fastann,'feature_extractor','varbitx)\"
unsupport6x_disted_type="(money','interval)\"
unsupport6x_type="(unknown)\"
# 检查实例版本
check_version()
{
    echo \"
    echo $db_ver
    echo \"
    echo '***** check base version...\"
    base_time=`date -d \"2020-08-31\" +%s`
    db_verdate=${db_ver##*compiled on}
    seconds=`date -d \"$db_verdate\" +%s`
    if [[ $seconds -lt $base_time ]]; then
        echo 'ERROR: please upgrade minor version...\"
    else
        echo 'pass.....\"
    fi
}
# 检查libraries
check_libraries()
{
    echo \"
    echo '***** check untransferred libraries...\"
    count=`psql -d postgres -c \"copy (select count(1) from pg_catalog.pg_library) to stdout\"`
    if [[ $count -gt 0 ]]; then
        psql -d postgres -c \"select name,laname language from pg_catalog.pg_library;\"
        echo \"WARN: please transfer libraries manually...\"
    else
        echo 'pass.....\"
    fi
}
# 检查表分布键
check_table_did()
{
    echo \"
    echo '***** check unsupported table distributedId types...\"
    count=0
    if [[ $db_ver == *8.2*4.3* ]]; then
        for ((i=0; i<$db_len; ++i)); do
            col=\"select count(1) from pg_catalog.pg_class pg_catalog.pg_attribute pg_catalog.pg_type pg_catal
```

```

sql= `select count(1) from pg_catalog.pg_class c,pg_catalog.pg_attribute a,pg_catalog.pg_type t,pg_catal
og.gp_distribution_policy p where
  a.atttypid=t.oid and a.attrelid=c.oid and p.localoid=c.oid and a.attnum=any(p.attnums) and a.attnum>0
and t.typname in $unsupported6x_disted_type"
count1=`psql -d ${db_names[$i]} -c "copy ($sql) to stdout"`
count=$((count + count1))
if [[ $count1 -gt 0 ]]; then
  sql="select '${db_names[$i]}' dbname,n.nspname schema,c.relname table_name,a.attname distributed_
field,t.typname field_type from
  pg_catalog.pg_namespace n,pg_catalog.pg_class c,pg_catalog.pg_attribute a,pg_catalog.pg_type t,pg_c
atalog.gp_distribution_policy p
  where a.atttypid=t.oid and n.oid=c.relnamespace and a.attrelid=c.oid and p.localoid=c.oid and a.attnum
=any(p.attnums) and a.attnum>0 and t.typname in $unsupported6x_disted_type order by schema,table_nam
e;"
  psql -d ${db_names[$i]} -c "$sql"
  fi
done
fi
if [[ $count -gt 0 ]]; then
  echo 'ERROR: please alter table distributedId types manually...'
else
  echo 'pass.....'
fi
}
# 检查表字段类型
check_table_ftype()
{
  echo ""
  echo '***** check unsupported table field types...'
  count=0
  if [[ $db_ver == *8.2*4.3* ]]; then
    for ((i=0; i<$db_len; ++i)); do
      sql="select count(1) from pg_catalog.pg_class c,pg_catalog.pg_attribute a,pg_catalog.pg_type t where
a.atttypid=t.oid and a.attrelid=c.oid and a.attnum>0 and t.typname in $unsupported6x_type"
      count1=`psql -d ${db_names[$i]} -c "copy ($sql) to stdout"`
      count=$((count + count1))
      if [[ $count1 -gt 0 ]]; then
        sql="select '${db_names[$i]}' dbname,n.nspname schema,c.relname table_name,a.attname field_name,t
.typname field_type from
        pg_catalog.pg_namespace n,pg_catalog.pg_class c,pg_catalog.pg_attribute a,pg_catalog.pg_type t
        where a.atttypid=t.oid and n.oid=c.relnamespace and a.attrelid=c.oid and a.attnum>0 and t.typname in

```



```
$unset support6x_type order by schema,table_name;"
    psql -d ${db_names[$i]} -c "$sql"
fi
done
fi
if [[ $count -gt 0 ]]; then
    echo 'ERROR: please alter table field types manually...'
else
    echo 'pass.....'
fi
}
# 检查扩展模块
check_extensions()
{
    echo ""
    echo '***** check unsupported extensions...'
    count=0
    if [[ $db_ver == *8.2*4.3* ]]; then
        for ((i=0; i<$db_len; ++i)); do
            count1=`psql -d ${db_names[$i]} -c "copy (select count(1) from pg_catalog.pg_extension where extname
in $support6x_ext) to stdout"`
            count=$((count + count1))
            if [[ $count1 -gt 0 ]]; then
                psql -d ${db_names[$i]} -c "select '${db_names[$i]}' dbname,extname,extversion from pg_catalog.pg_ext
ension where extname in $support6x_ext;"
            fi
        done
    fi
    if [[ $count -gt 0 ]]; then
        echo 'WARN: please drop useless extensions manually...'
        echo 'REF DROP EXTENSION SQL: drop extension <name>'
    else
        echo 'pass.....'
    fi
}
check_version
check_libraries
check_table_did
check_table_ftype
check_extensions
```

## 参数说明

参数	说明
<PGHOST>	ADB PG 4.3版实例的连接地址
<PGPORT>	ADB PG 4.3版实例的端口号
<PGUSER>	连接ADB PG 4.3版实例的用户名
<PGPASSWORD>	上述用户名对应的密码

## 操作步骤

1. 运行以下命令在Linux设备中安装postgresql客户端。

```
sudo yum install postgresql
```

2. 查看Linux设备外网地址。登录AnalyticDB PostgreSQL控制台，将Linux设备的外网IP地址添加至ADB PG 4.3版实例的白名单中，详情请参见设置白名单。
3. 使用Linux设备连接待检查不兼容项的ADB PG 4.3版实例。

```
psql -h <PGHOST> -p <PGPORT> -U <PGUSER>
```

4. 将shell脚本编辑保存为如4x\_to\_6x.sh的脚本文件，运行以下命令执行检查不兼容项的脚本文件4x\_to\_6x\_check.sh。

```
sh 4x_to_6x_check.sh <PGHOST> <PGPORT> <PGUSER> <PGPASSWORD>
```

5. 根据提示信息，修改数据库中对应的不兼容项，修改完毕后，请再次执行脚本查看结果是否通过检查。

## 检查结果参考

检查通过情况（运行脚本后全部显示“pass”，或者不包含“ERROR”信息）

```
***** check base version...
pass.....
***** check untransferred libraries...
pass.....
***** check unsupported table distributedId types...
pass.....
***** check unsupported table field types...
pass.....
***** check unsupported extensions...
pass.....
```

存在不兼容项的检查情况（提示ERROR信息的必须修改）

```

PostgreSQL 8.2.15 (Greenplum Database 4.3.99.00 build dev) compiled on May 2 2020 09:35:15
***** check base version...
ERROR: please upgrade minor version...
***** check untransferred libraries...
  name | language
-----+-----
select_1 | plpgsql
(1 row)
WARN: please transfer libraries manually...
***** check unsupported table distributedId types...
 dbname | schema | table_name | distributed_field | field_type
-----+-----+-----+-----+-----
 adbpg | public | test1 | id | money
(1 row)
ERROR: please alter table distributedId types manually...
***** check unsupported table field types...
 dbname | schema | table_name | field_name | field_type
-----+-----+-----+-----+-----
 adbpg | public | test2 | name | unknown
(1 row)
ERROR: please alter table field types manually...
***** check unsupported extensions...
 dbname | extname | extversion
-----+-----+-----
 adbpg | varbitx | 1.0
(1 row)
WARN: please drop useless extensions manually...
REF DROP EXTENSION SQL: drop extension <name>
    
```

提示信息	修改方式
ERROR: please upgrade minor version...	登录AnalyticDB PostgreSQL控制台将实例进行小版本升级，详情请参见版本升级。
WARN: please transfer libraries manually...	提示4.3库中用到了library，这些library不会自动迁移，升级后您需要手工进行迁移。
ERROR: please alter table distributedId types manually...	提示4.3库中存在不兼容的表分布键，需要您在4.3库中手动修改。
ERROR: please alter table field types manually...	提示4.3库表存在不兼容的字段类型，需要在4.3库中手动修改。

提示信息	修改方式
WARN: please drop useless extensions manually...	提示4.3库中存在不兼容的扩展模块，这些模块不会迁移，如果用到需要考虑修改涉及模块的表/存储过程等，如果没用可考虑删除。

## 6.3. 4.3版本与6.0版本兼容性注意事项

本文介绍了ADB PG 4.3版本与6.0版本在兼容性上的注意事项，如果您需要将实例从4.3版本升级到6.0版本，请根据实际情况对本文提及的部分进行相应的修改。

### 优化器


名称	V4.3	V6.0
默认优化器	Legacy优化器	ORCA优化器

4.3版本与6.0版本都支持Legacy优化器和ORCA优化器，可根据需求选择使用，有关优化器的选择，请参见[两种优化器的选择](#)。

### 转义符

- 在6.0版本中，字符串的转义字符 `\`，将作为字符串本身，不再表示转义符。
- 可执行以下语句保持和4.3版本语义一致，但不建议这样操作。

```
set standard_conforming_strings = off;
```

 **说明** 上述语句仅对session级别生效。如果需要对集群生效，请[提交工单](#)，由ADB PG后台技术人员进行设置。

### 类型转换

- 6.0版本中不再支持 `YYYYMMDDHH24MISS` 格式的字符串自动强转为时间戳类型。您可以通过系统内建函数 `to_timestamp/to_char`，进行 `YYYYMMDDHH24MISS` 格式的字符串和时间戳之间的转换。
- 6.0版本不再支持数字类型隐式转换成TEXT类型。如果4.3版本中有这类语句，升级到6.0版本后，您可以通过增加Function支持这类场景。比如：

```
create or replace function substr(numeric, integer, integer) returns text as $$
select substr($1::text, $2, $3);
$$ language sql IMMUTABLE strict;
```

```
create or replace function pg_catalog.btrim(str numeric) returns text as $$
return $_[0];
$$ language plperl IMMUTABLE strict;
```

```
create or replace function to_date(timestamp, text) returns date as $$
select to_date($1::text,$2);
$$ language sql IMMUTABLE strict;
```

```
create or replace function to_date(integer, text) returns date as $$
select to_date($1::text,$2);
$$ language sql IMMUTABLE strict;
```

- 如果有SQL或Function需要隐式转换成TEXT类型的场景，会受到影响，需要手工改写。

## 外部表错误日志

6.0版本不支持在 `create external table` 或 `copy` 语句时，使用 `into error_table` 子句。您可以使用系统函数代替 `into error_table` 子句的功能。

```
gp_read_error_log('$external_table')
gp_truncate_error_log('$external_table')
```

## 数据类型

- NUMERIC类型的文件存储格式变化，会影响到磁盘空间使用；
- MONEY类型由32位改为64位，会影响磁盘空间使用；
- 6.0版本不支持以下数据类型作为分布键：
  - abstime
  - reltime
  - tinterval
  - money
  - anyarray

## 关键字

6.0版本新增、修改、删除了部分关键字。（数据库对象的命名，不可以与关键字冲突）。

关键字不同的分类，含义不一样。可以通过如下命令，查看所有关键字和具体分类。（4.3 和 6.0 版本均适用）。

```
select * from pg_get_keywords();
```

## 关键字分类说明

catcode	catdesc	说明
U	unreserved	不保留，可以用于任何对象（视图、表、函数、类型、索引、字段、类型等）。

catcode	catdesc	说明
C	unreserved (cannot be function or type name)	不保留，但是不可用于函数、类型名。
T	reserved (can be function or type name)	保留，但是可用于函数、类型名。
R	reserved	保留，不可用于任何对象名称。

#### 4.3版和6.0版本分类不同的关键字列表如下：

关键字	v4.3	v6.0
<b>between</b>	reserved	unreserved (cannot be function or type name)
<b>collation</b>	无	reserved (can be function or type name)
<b>concurrently</b>	unreserved	reserved (can be function or type name)
<b>convert</b>	unreserved (cannot be function or type name)	无
<b>filter</b>	reserved	unreserved
<b>lateral</b>	-	reserved
<b>new</b>	reserved	无
<b>off</b>	reserved	unreserved
<b>old</b>	reserved	无
<b>percentile_cont</b>	unreserved (cannot be function or type name)	无
<b>percentile_disc</b>	unreserved (cannot be function or type name)	无
<b>range</b>	reserved	unreserved
<b>reindex</b>	unreserved	reserved
<b>rows</b>	reserved	unreserved
<b>sort</b>	reserved	reserved (can be function or type name)
<b>variadic</b>	无	reserved

## 系统表

系统表在4.3版和6.0版中有部分区别，如果您的业务逻辑引用了以下系统表，请进行修改否则将引起报错。

V4.3	V6.0	说明
pg_class.reltoastidid	无	已删除
pg_stat_activity.procpid	pg_stat_activity.pid	进程号
pg_stat_activity.current_query	pg_stat_activity.state pg_stat_activity.query	拆分为两列，state标识后端状态，query标识当前执行的请求
gp_distribution_policy.attrnums	gp_distribution_policy.distkey	修改列名，且数据类型变int2vector
sesion_level_memory_consumption.__gp_localid sesion_level_memory_consumption.__gp_masterid	无	已删除
pg_filespace pg_filespace_entry	无	已删除

## 内置函数的参数

在6.0版本中，一些内置函数的参数发生了变化。

V4.3	V6.0	说明
int4_avg_accum(bytea, integer)	int4_avg_accum(bigint[], integer)	无
string_agg(expression)	string_agg(expression, delimiter)	把一个表达式变成字符串

## 其他用法

名称	V4.3	V6.0
left()函数	不支持	支持
update分布键	不支持	支持

## 6.4. 扩容/升级实例规格

在使用云数据库 AnalyticDB for PostgreSQL 过程中，随着您的数据量和计算量的动态增长，一些计算资源如CPU、磁盘、内存以及数据处理节点数量将成为数据处理速度的瓶颈。为了支持实例的动态扩展，AnalyticDB for PostgreSQL 提供在线扩容实例规格的功能，目前暂不支持对实例的降级操作。本文介绍了升级实例规格的相关操作。

## 查看当前实例规格

AnalyticDB for PostgreSQL 实例集群规格包括计算节点规格和计算节点数据量，各种计算节点规格详细说明见[规格及选型](#)。

您可以通过以下的步骤来查看当前的实例规格：

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 选择要操作实例所在的地域。例如，华东1。
3. 单击列表中的实例名称，进入基本信息页面。

在基本信息页面中显示了当前的实例规格，包括计算节点规格，计算节点数量（注：原规格定义模式计算组规格、计算组详情、计算组数量等也会一同显示）。

AnalyticDB for PostgreSQL 目前提供两类规格：

- 高性能，规格名称以高性能SSD开始，基于 SSD 存储，提供更好的 I/O 能力，带来更高的性能。
- 高容量，规格名称以大存储HDD开始，基于HDD存储，提供更大、更实惠的空间，满足更高的存储需要。

## 扩容实例规格


当确认需要升级实例规格后，您可以按照下面的步骤进行实例规格的升级扩容：

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 选择要操作实例所在的地域。例如，华东1。
3. 单击目标实例右侧操作栏中的升级按钮，进入[确认订单](#)页面。
4. 选择合适的计算节点规格和计算节点数量，点击去开通按钮。

目前，AnalyticDB for PostgreSQL 支持不同的计算节点规格和计算节点数量组合，详情参见[规则组合速查表](#)。在选择新的计算节点规格和计算节点数量组合时，要满足以下约束：

- 新旧计算节点规格必须为同一存储类型节点规格，而且新的计算节点规格  $\geq$  旧的计算节点规格
- 如果新的计算节点规格 = 旧的计算节点规格，新的计算节点数量  $>$  旧的计算节点数量

在选择计算节点规格和计算节点数量的时候，除了满足以上约束，您还需要对自己的数据量和计算量进行评估，从而选择合适的计算节点规格和计算节点数量组合，详情可参考[如何选择实例规格](#)。

 **注意** 根据您数据量的不同，实例规格升级的过程大约需要至少30分钟到数十个小时不等，与表数量、分区表数量、索引数量、是否压缩、总数据量、实例规格等因素相关。在此过程中，为了保证数据的一致性，您实例只对外提供只读服务，并且会闪断两次，请您提前做出调整。当升级实例规格结束，对应实例恢复运行中状态，您可以正常访问数据库，而且实例的数据库内核版本自动升级为最新。

在完成了上述操作之后，您可以回到控制台，查看目标实例的运行状态。如果实例规格升级操作完成，则实例运行状态为运行中，否则为升降级中。



## 7. 标签

### 7.1. 创建实例标签

如果您有大量实例，可以通过给实例绑定标签，对实例进行分类管理。每个标签由一对键值组成，您可以通过键值，对实例进行二级分类。

#### 限制说明

- 每个实例最多可以绑定20个标签，且标签键必须唯一。相同的标签键会被覆盖。
- 每次最多设置50个实例进行批量标签绑定。
- 不同地域的标签信息是独立的。
- 任一标签在解绑后，如果没有绑定任何实例，则该标签会被删除。

#### 操作步骤

1. 登录[云原生数据仓库PostgreSQL版控制台](#)
2. 选中实例列表菜单，实例表格有一列显示“标签”，鼠标悬浮在图标上进行设置，如下图所示。  
□
3. 单击**编辑标签**，输入标签的键键和值，单击**确定**，如下图所示。



4. 填写完所有要绑定的标签后，单击**确认**完成绑定。

### 7.2. 删除实例标签

如果实例调整或者不再需要标签，您可以删除该实例的标签。

#### 限制说明

- 每次解绑的标签数量不能超过20个。
- 任一标签在解绑后，如果没有绑定任何实例，则该标签会被删除。

#### 操作步骤

1. 登录[云原生数据仓库PostgreSQL版控制台](#)。
2. 选择某一实例，鼠标悬浮在“标签”图标上，点击“编辑标签”。
3. 单击要删除的标签后的X删除标签，如下图所示。  
□
4. 单击**确认**,完成操作。

## 7.3. 实例标签筛选

实例绑定标签后，您可以根据标签筛选实例。


1. 登录[云原生数据仓库PostgreSQL版控制台](#)
2. 单击实例列表表格左上角的**标签按钮**，选择标签的键和值筛选实例，如下图所示。  
□

## 8. 释放实例

根据业务要求，您可以手动释放按量付费的实例。


### 前提条件

实例类型为按量付费实例。

 **说明** 您可以随时释放按量付费的实例；包年包月的实例不能主动删除或释放，到期后自动释放。

### 操作步骤

1. 登录[云数据库AnalyticDB for PostgreSQL管理控制台](#)。
2. 在实例列表上方选择实例所在地域。
3. 找到目标实例，在操作栏中，单击**管理**。
4. 在**基本信息**页面，单击运行状态右侧的**释放**按钮。
5. 在弹出的对话框中勾选**您确定释放实例？**，然后单击**确定**，释放实例。

 **注意** 释放后的实例不可恢复，请谨慎操作。

### 相关API

API	描述
<a href="#">DeleteDBInstance</a>	释放实例