

ALIBABA CLOUD

阿里云

云原生数据仓库 AnalyticDB
PostgreSQL 版
快速入门

文档版本：20200914

 阿里云

法律声明

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

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

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
<code>Courier</code> 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。
<i>斜体</i>	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. 开始使用	05
2. 创建实例	06
3. 设置白名单	09
4. 设置账号	11
5. 客户端连接	13

1. 开始使用

云原生数据仓库 PostgreSQL 版（原HybridDB for PostgreSQL）是一种分布式云数据库，由多个节点组成，可提供大规模并行处理数据仓库（MPP）的服务。AnalyticDB for PostgreSQL 基于 Greenplum 开源数据库项目构建，同时构筑新一代向量化计算引擎，OSS 云存储分析，非结构化数据向量检索等。AnalyticDB for PostgreSQL 支持主流 BI 和 ETL 工具，可以同多种云上数据传输和同步工具集成，无缝对接RDS，MaxCompute等数据源。关于 AnalyticDB for PostgreSQL的功能和限制详情，请参见[功能与限制](#)。

要使用 AnalyticDB for PostgreSQL，您需要完成以下操作：

1. 规划实例规格[规格及选型](#)。
2. [创建实例](#)。
3. 配置实例。包括 [设置白名单](#)，[设置账号](#)，[如何切换网络类型?](#)。
4. [客户端访问实例](#)。
5. 进行表设计[数据表管理](#)，表设计需要重点关注
 - [表分布定义](#)，其定义了表数据在节点间的分布策略
 - [表存储格式定义](#)，其定义了采用行存储格式或列存储格式
 - 对于大表，建议考虑设计为分区表，可参照[表分区定义](#)
 - 可以考虑建立索引，提升查询性能，可参照[索引管理](#)
6. 同步或导入数据，请参见[数据写入方式概述](#)及[数据迁移及同步方案综述](#)。
7. 查询及调优，首先需要关注当有大量数据写入或较多更新发生时，及时收集统计信息，确保复杂查询的执行计划最优[使用ANALYZE收集统计信息](#)，其它优化建议可参考[查询性能优化指导](#)。

2. 创建实例

您可以通过云原生数据仓库 AnalyticDB PostgreSQL 版管理控制台创建实例，本文介绍如何在云原生数据仓库 AnalyticDB PostgreSQL 版管理控制台上创建实例。

前提条件

- 已注册阿里云账号。若尚未注册，请前往[阿里云官网](#)进行注册。
- 首次创建 AnalyticDB PostgreSQL 实例需要登录控制台授权创建服务关联角色，有关服务关联角色请参见[服务关联角色](#)。
 - i. 登录[云原生数据仓库 AnalyticDB PostgreSQL 版管理控制台](#)。
 - ii. 单击页面右上角的新建实例。



- iii. 在弹出的创建服务关联角色对话框中单击确定。



- 阿里云账户余额大于等于 100 元。

操作步骤


1. 登录[云原生数据仓库 AnalyticDB PostgreSQL 版管理控制台](#)。
2. 单击页面右上角的新建实例，进入实例购买页面。
3. 选择计费方式。
 - **按量付费**：属于后付费，即按小时扣费。适合短期需求，用完可立即释放实例，节省费用。
 - **包年包月**：属于预付费，即在新建实例时需要支付费用。适合长期需求，价格比按量付费更实惠，且购买时长越长，折扣越多。

说明 按量付费实例可以转为包年包月实例。包年包月实例无法转为按量付费实例。

4. 设置以下参数。

参数	说明
地域和可用区	实例所在的地理位置。购买后无法更换地域。 <ul style="list-style-type: none"> 请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。 请确保实例与需要连接的ECS实例创建于同一个地域，否则它们无法通过内网互通，只能通过外网互通，无法发挥最佳性能。
可用区	可用区是地域中的一个独立物理区域，不同可用区之间没有实质性区别。您可以选择将实例与ECS实例创建在同一可用区或不同的可用区。
网络类型	专有网络，也称为VPC（Virtual Private Cloud）。VPC是一种隔离的网络环境，安全性和性能均高于传统的经典网络。需要事先创建与实例在同一地域下的VPC和交换机，具体步骤请参见 创建专有网络 。
专有网络	选择专有网络VPC的ID。
专有网络交换机	选择专有网络下的交换机。
实例资源类型	<ul style="list-style-type: none"> 存储弹性模式：支持独立磁盘扩容，支持在线平滑扩容。 存储预留模式：不支持独立磁盘扩容，不支持在线平滑扩容。
引擎版本	可选择6.0标准版、6.0向量增强版。6.0向量增强版支持非结构化向量数据的检索和分析能力，适用于以图搜图，声纹识别，语义文本检索，电商商品图片分析等业务场景。请参见 向量分析概述 。
节点规格（segment）	计算资源单位，不同的计算组规格有不同的存储空间和计算能力。关于规格详情，请参见云原生数据仓库AnalyticDB PostgreSQL的 规格及选型 。
节点数量（segment）	所购买的“节点”数量，最小单位为2个，节点个数的增加可以线性地提升性能。
存储磁盘类型	支持ESSD云盘、高效云盘。ESSD云盘存取效率较高，高效云盘价格比较便宜。
单节点存储容量	实例中每个节点独享的存储空间大小。

- 完成设置后，单击**立即购买**。
- 在确认订单页面，勾选《AnalyticDB for PostgreSQL服务协议》，然后单击去开通完成订单支付。
- 用户可在实例列表页面查看新建实例。

 **说明** AnalyticDB for PostgreSQL数据库初始化需要一定时间，待实例列表中的实例运行状态显示为运行中，才可进行后续操作。

相关API

API	描述
CreateDBInstance	创建实例


3. 设置白名单

在启用实例前，您必须先修改白名单。为保障数据库的安全稳定，请将需要访问数据库的 IP 地址或者 IP 段加入白名单。

背景信息

访问 AnalyticDB for PostgreSQL 数据库有如下三种场景：

- 外网访问 AnalyticDB for PostgreSQL 数据库。
- 内网访问 AnalyticDB for PostgreSQL 数据库。请确AnalyticDB for PostgreSQL 和 ECS 网络类型一致。
- 内外网同时访问 AnalyticDB for PostgreSQL 数据库。请确AnalyticDB for PostgreSQL 和 ECS 网络类型一致。

 **注意** 关于设置网络类型，请参见[如何切换网络类型？](#)。

操作步骤

1. 登录[云原生数据仓库PostgreSQL版管理控制台](#)。
2. 选择目标实例所在地域。
3. 单击目标实例的 ID，进入实例基本信息页面。
4. 在实例菜单栏中，选择数据安全性，进入数据安全性页面。
5. 在白名单设置标签页中，单击 default 白名单分组后的修改，进入修改白名单分组页面。

 **注意**

您也可以单击 default 白名单分组后的清空，删除默认分组中的白名单，然后单击添加白名单分组新建自定义分组。

6. 删除“组内白名单”中的默认白名单 127.0.0.1，然后填写自定义白名单。参数说明如下所示：
 - **分组名称**：2~32 个字符，由小写字母、数字或下划线组成，开头需为小写字母，结尾需为字母或数字。默认分组不可修改，且不可删除。
 - **组内白名单**：填写可以访问数据库的 IP 地址或者 IP 段，各 IP 地址或者 IP 段间用英文逗号分隔。
 - 白名单功能支持设置 IP 地址（如 10.10.10.1）或者 IP 段（如 10.10.10.0/24，表示 10.10.10.X 的 IP 地址都可以访问数据库）。
 - % 或者 0.0.0.0/0 为允许任何 IP 访问。

 **注意**

该设置将极大降低数据库安全性，如非必要请勿使用。

- 新建实例设置了本地环回 IP 地址 127.0.0.1 为默认白名单，禁止任何外部 IP 访问本实例。

7. 单击确定，添加白名单。

后续操作

正确使用白名单可以让 AnalyticDB for PostgreSQL 得到高级别的访问安全保护，建议您定期维护白名单。后续操作中，您可以单击分组名称后的修改修改已有分组，或者单击删除删除已有的自定义分组。

相关API

API	描述
DescribeDBInstanceIPArrayList	查询允许访问实例的IP名单
ModifySecurityIps	修改白名单

4. 设置账号

本文档将介绍如何在 AnalyticDB for PostgreSQL 的实例中创建账号及重置密码。

背景信息

AnalyticDB for PostgreSQL实例支持两种数据库账号：高权限账号和普通账号。

- 高权限账号拥有所有数据库的所有操作权限。
- 普通账号拥有已授权数据库（owner）的所有操作权限。

 **说明** 操作权限包括SELECT、INSERT、UPDATE、DELETE、TRUNCATE、REFERENCES、TRIGGER。

权限设置，请参见[用户权限管理](#)。

在使用云原生数据仓库PostgreSQL版之前，需要AnalyticDB for PostgreSQL实例中创建初始账号用来连接数据库。用户无法在控制台创建其他账号，但是登录到数据库后可通过SQL语句创建其他账号。请参见[使用SQL语句创建账号](#)。

创建初始账号

注意

- 初始账号创建后，无法删除该账号。
- 初始账号为高权限账号。

1. 登录[云原生数据仓库PostgreSQL版管理控制台](#)。
2. 选择目标实例所在地域。
3. 单击目标实例的 ID，进入实例实例基本信息页面。
4. 在实例菜单栏中，选择帐号管理，进入帐号管理页面。
5. 单击创建初始帐号，进入创建帐号页面。
6. 填写数据库账号和密码，然后单击确定。
 - 数据库账号：2~16 个字符，由小写字母、数字或下划线组成，开头需为字母，结尾需为字母或数字，如 *user4example*。
 - 密码：8~32 个字符，由大写、小写、数字或特殊字符中的三类字符组成。
 - 确认密码：输入与密码一致的字段。

使用SQL语句创建账号

- 创建高权限账号

```
create role admin0 WITH LOGIN ENCRYPTED PASSWORD '111111' rds_superuser;
```

- 创建普通账号


```
create role test1 WITH LOGIN ENCRYPTED PASSWORD '111111';
```

重置密码

在使用 AnalyticDB for PostgreSQL 过程中，如果忘记数据库账号密码，可以通过 [AnalyticDB for PostgreSQL 数据库管理控制台](#) 重新设置密码。

 **注意** 为保障数据安全，建议您定期更换密码。

1. 登录[云原生数据仓库PostgreSQL版管理控制台](#)。
2. 选择目标实例所在地域。
3. 单击目标实例的 ID，进入实例实例基本信息页面。
4. 在实例菜单栏中，选择帐号管理，进入帐号管理页面。
5. 单击需要管理账号后的重置密码，进入修改账号页面。
6. 输入新密码并确认新密码后，单击确定。


 **注意** 密码有 8~32 个字符，由大写、小写、数字或特殊字符中的三类字符组成。建议不要使用曾经用过的密码。

相关API

API	描述
CreateAccount	创建账户
DescribeAccounts	查询数据库账户信息
ModifyAccountDescription	修改数据库账户的备注名
ResetAccountPassword	重置账户密码

5. 客户端连接

云原生数据仓库PostgreSQL版完全兼容 PostgreSQL 消息协议，可以直接使用支持 PostgreSQL 消息协议的工具，例如命令行 psql、libpq、JDBC、ODBC、psycopg2等；图形化工具pgAdmin（注：ADB PG 4.3版本只支持 pgAdmin III 1.6.3及之前版本，ADB PG 6.0支持最新的pgAdmin 4版本），DBeaver, Navicat, 阿里云 DMS 数据管理服务等也支持 AnalyticDB for PostgreSQL实例的管理和开发。

 **说明** AnalyticDB for PostgreSQL 4.3版本 基于 PostgreSQL 8.3 内核；AnalyticDB for PostgreSQL 6.0版本 基于 PostgreSQL 9.4内核。

DMS 控制台

数据管理（Data Management Service，简称DMS）支持MySQL、SQL Server、PostgreSQL、PPAS、Petadata等关系型数据库，DRDS等OLTP数据库，AnalyticDB、DLA等OLAP数据库和MongoDB、Redis等NoSQL的数据库管理。它是一种集数据管理、结构管理、用户授权、安全审计、数据趋势、数据追踪、BI图表、性能与优化和服务器管理于一体的数据管理服务。

本章节将为您介绍如何使用DMS登录云数据库 AnalyticDB for PostgreSQL。

1. 登录**AnalyticDB for PostgreSQL 控制台**。
2. 创建实例，具体操作请参见**创建实例**；若已完成创建操作，请单击目标实例ID进入实例详情页。
3. 创建账号，具体操作请参见**设置账号**。

 **说明** 该账号用于登录DMS，每个实例只能创建一个。

4. 单击实例详情页右上角**登录数据库**。



5. 在RDS数据库登录页面输入用户名和密码，单击**登录**。
6. 若有页面弹出如下提示，请根据当前情况选择相应设置。
 - **设置所有实例**：当前用户下所有的实例都会被添加DMS的IP地址，以后不需要重复操作该步骤。
 - **设置本实例**：只有当前登录的实例添加DMS的IP地址。
 - **不设置**：不设置DMS的IP地址，该操作将无法使用DMS登录数据库。



命令行 psql

psql 是 Greenplum 中比较常用的工具，提供了丰富的命令，其二进制文件在 Greenplum 安装后的 BIN 目录下。使用步骤如下所示：

1. 通过如下任意一种方式进行连接：

- 连接串的方式

```
psql "host=yourgpdbaddress.gpdb.rds.aliyuncs.com port=3432 dbname=postgres user=gpdbaccount password=gpdbpassword"
```

- 指定参数的方式

```
psql -h yourgpdbaddress.gpdb.rds.aliyuncs.com -p 3432 -d postgres -U gpdbaccount
```

参数说明：

- -h：指定主机地址。
- -p：指定端口号。
- -d：指定数据库（默认的数据库是 postgres），
- -U：指定连接的用户。
- 可以通过 `psql --help` 查看更多选项。在 psql 中，可以执行 `\?` 查看更多 psql 中支持的命令。

2. 输入密码，进入 psql 的 Shell 界面。psql 的 Shell 界面如下：

```
postgres=>
```

参考文档

- 关于的 psql 的更多使用方法，可参见 Greenplum 社区文档 [gp6 psql](#)。
- ADB PG 也支持 PostgreSQL 的 psql 命令，使用时请注意细节上的差异。详情参见 PostgreSQL 社区文档 [“PostgreSQL 8.3.23 Documentation — psql”](#)。

下载方式

对于 RHEL (Red Hat Enterprise Linux) 和 CentOS 版本 6 和 7 平台，可以通过以下地址进行下载，解压后即可使用：

- ADBPG 4 客户端工具下载：
 - RHEL 6 或 CentOS 6 平台，请单击 [ADBPG_client_package_el6](#) 进行下载。
 - RHEL 7 或 CentOS 7 平台，请单击 [ADBPG_client_package_el7](#) 进行下载。
- ADBPG 6 客户端工具下载：
 - RHEL 6 或 CentOS 6 平台，请单击 [ADBPG_client_package_el6](#) 进行下载。
 - RHEL 7 或 CentOS 7 平台，请单击 [ADBPG_client_package_el7](#) 进行下载。

使用方式

将客户端工具压缩包下载到本地后，以 ADBPG 6 客户端工具 CentOS 7 平台为例，其使用方式如下：

1. 在客户端工具压缩包目录下，对其解压：

```
tar -xzf adbpg_client_package.el7.x86_64.tar.gz
```

2. 解压后切换到 bin 目录下，执行命令：

```
cd adbpg_client_package/bin
```

3. bin 目录下包括客户端工具 psql、pg_dump 等，按照各工具参考文档，执行命令行。
 - 使用 psql 连接客户端的方式参考 [命令行 psql](#)。
 - pg_dump 是 PostgreSQL 的逻辑备份工具，其使用方式可参考 [命令行 pg_dump](#)。

另外，adbpg 也支持 docker 形态的工具包，如下所示。你可以在 [这里](#) 找到对应平台 docker 安装地址。

```
# 运行对应于 adbpg 4.x 版本的工具镜像。
docker run -idt --name=adbpgcli aliadbpg/adbpgcli:v4.3.0
docker exec -it adbpgcli /bin/bash -l

# 运行对应于 adbpg 6.x 版本的工具镜像。
docker run -idt --name=adbpgcli aliadbpg/adbpgcli:v6.3.0
docker exec -it adbpgcli /bin/bash -l

[root@adbpgcli /]# psql --help
```

pgAdmin

pgAdmin 是 PostgreSQL 图形客户端，可以直接用于连接 AnalyticDB for PostgreSQL。详情参见 [官网](#)。AnalyticDB for PostgreSQL 4.3 版本基于 PostgreSQL 8.3 版本内核，因此必须使用 pgAdmin III 1.6.3 或之前的版本才能连 AnalyticDB for PostgreSQL 4.3（pgAdmin 4 当前不支持）。AnalyticDB for PostgreSQL 6.0 版本基于 PostgreSQL 9.4 版本，支持最新的 pgAdmin 4。

您可以从 PostgreSQL 官网下载 [pgAdmin III 1.6.3](#) 或 [pgAdmin 4](#)。pgAdmin 支持各种平台，例如 Windows、MacOS 和 Linux。其它图形客户端，详情参见 [图形客户端工具](#)。

操作步骤

1. 下载安装支持的 pgAdmin 版本。
2. 选择文件 > 新增服务器，进入配置连接窗口。

3. 填写配置信息，如下图所示：



4. 单击确定，即可连接到 AnalyticDB for PostgreSQL。

JDBC

用户需要使用 PostgreSQL 官方提供的 JDBC。下载方法如下：

- 单击 [这里](#)，下载 PostgreSQL 的官方 JDBC，下载之后加入到环境变量中。
- 也可采用 Greenplum 官网提供的工具包，详情请参见“[Greenplum Database 4.3 Connectivity Tools for UNIX](#)”或“[Greenplum 6.3 Greenplum Client](#)”。

代码示例

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
public class gp_conn {
    public static void main(String[] args) {
        try {
            Class.forName("org.postgresql.Driver");
            Connection db = DriverManager.getConnection("jdbc:postgresql://mygpdbpub.gpdb.rds.aliyuncs.com:3432/postgres","mygpdb","mygpdb");
            Statement st = db.createStatement();
            ResultSet rs = st.executeQuery("select * from gp_segment_configuration;");
            while (rs.next()) {
```



```
System.out.print(rs.getString(1));
System.out.print(" | ");
System.out.print(rs.getString(2));
System.out.print(" | ");
System.out.print(rs.getString(3));
System.out.print(" | ");
System.out.print(rs.getString(4));
System.out.print(" | ");
System.out.print(rs.getString(5));
System.out.print(" | ");
System.out.print(rs.getString(6));
System.out.print(" | ");
System.out.print(rs.getString(7));
System.out.print(" | ");
System.out.print(rs.getString(8));
System.out.print(" | ");
System.out.print(rs.getString(9));
System.out.print(" | ");
System.out.print(rs.getString(10));
System.out.print(" | ");
System.out.println(rs.getString(11));
}
rs.close();
st.close();
} catch (ClassNotFoundException e) {
    e.printStackTrace();
} catch (SQLException e) {
    e.printStackTrace();
}
}
```

详细文档，请参见 [“The PostgreSQL JDBC Interface”](#)。

Python

Python 连接 Greenplum 和 PostgreSQL 采用的库是 `psycopg2`。使用步骤如下：

1. 安装 `psycopg2`。在 CentOS 下，有如下三种安装方法：
 - 执行如下命令：`yum -y install python-psycopg2`
 - 执行如下命令：`pip install psycopg2`

- 从源码安装：

```
yum install -y postgresql-devel*
wget http://initd.org/psycopg/tarballs/PSYCOPG-2-6/psycopg2-2.6.tar.gz
tar xf psycopg2-2.6.tar.gz
cd psycopg2-2.6
python setup.py build
sudo python setup.py install
```

2. 安装后，设置 PYTHONPATH 环境变量，之后就可以引用，如下所示：

```
import psycopg2
sql = 'select * from gp_segment_configuration;'
conn = psycopg2.connect(database='gpdb', user='mygpdb', password='mygpdb', host='mygpdbpu
b.gpdb.rds.aliyuncs.com', port=3432)
conn.autocommit = True
cursor = conn.cursor()
cursor.execute(sql)
rows = cursor.fetchall()
for row in rows:
    print row
conn.commit()
conn.close()
```

会得到类似以下的结果：

```
(1, -1, 'p', 'p', 's', 'u', 3022, '192.168.2.158', '192.168.2.158', None, None)
(6, -1, 'm', 'm', 's', 'u', 3019, '192.168.2.47', '192.168.2.47', None, None)
(2, 0, 'p', 'p', 's', 'u', 3025, '192.168.2.148', '192.168.2.148', 3525, None)
(4, 0, 'm', 'm', 's', 'u', 3024, '192.168.2.158', '192.168.2.158', 3524, None)
(3, 1, 'p', 'p', 's', 'u', 3023, '192.168.2.158', '192.168.2.158', 3523, None)
(5, 1, 'm', 'm', 's', 'u', 3026, '192.168.2.148', '192.168.2.148', 3526, None)
```

libpq

libpq 是 PostgreSQL 数据库的 C 语言接口，用户可在 C 程序中通过 libpq 库访问 PostgreSQL 数据库并进行数据库操作。在安装了 Greenplum 或者 PostgreSQL 之后，在其 lib 目录下可以找到其静态库和动态库。

相关案例请参见 [这里](#)，此处不再列举。

关于 libpq 详情，请参见 [“PostgreSQL 9.4.10 Documentation — Chapter 31. libpq - C Library”](#)。

ODBC

PostgreSQL 的 ODBC 是基于 LGPL（GNU Lesser General Public License）协议的开源版本，可以在 [PostgreSQL 官网](#) 下载。

操作步骤

1. 安装驱动。

```
yum install -y unixODBC.x86_64
yum install -y postgresql-odbc.x86_64
```

2. 查看驱动配置。

```
cat /etc/odbcinst.ini
# Example driver definitions
# Driver from the postgresql-odbc package
# Setup from the unixODBC package
[PostgreSQL]
Description = ODBC for PostgreSQL
Driver      = /usr/lib/psqlodbcw.so
Setup       = /usr/lib/libodbcpsqlS.so
Driver64    = /usr/lib64/psqlodbcw.so
Setup64     = /usr/lib64/libodbcpsqlS.so
FileUsage   = 1
# Driver from the mysql-connector-odbc package
# Setup from the unixODBC package
[MySQL]
Description = ODBC for MySQL
Driver      = /usr/lib/libmyodbc5.so
Setup       = /usr/lib/libodbcmyS.so
Driver64    = /usr/lib64/libmyodbc5.so
Setup64     = /usr/lib64/libodbcmyS.so
FileUsage   = 1
```

3. 配置 DSN，将如下代码中的 **** 改成对应的连接信息。

```
[mygpdb]
Description = Test to gp
Driver = PostgreSQL
Database = ****
Servername = ****.gpdb.rds.aliyuncs.com
UserName = ****
Password = ****
Port = ****
ReadOnly = 0
```

4. 测试连通性。

```
echo "select count(*) from pg_class" | isql mygpdb
+-----+
| Connected!          |
|                    |
| sql-statement      |
| help [tablename]   |
| quit               |
|                    |
+-----+
SQL> select count(*) from pg_class
+-----+
| count      |
+-----+
| 388        |
+-----+
SQLRowCount returns 1
1 rows fetched
```

5. ODBC 已连接上实例，将应用连接 ODBC 即可，具体操作请参见 [这里](#) 和 [C# 连接到 PostgreSQL](#)。

其他参考信息

图形客户端工具

AnalyticDB for PostgreSQL 用户可以直接使用 Greenplum 支持的其它图形化客户端工具，包括

- [pgadmin III \(1.6.3\)](#)
- [pgAdmin 4](#)
- [dbeaver](#)
- [Navicat Premium](#)
- [Navicat For PostgreSQL](#)
- [SQL Workbench](#)

Greenplum 客户端参考

Greenplum 官网也提供了一个安装包，包含 JDBC、ODBC 和 libpq，用户可方便地安装和使用，详情参见 [Greenplum 官方文档](#)。

参考文档

- [Greenplum 4.3 文档](#)
- [Greenplum 6.0文档](#)
- [PostgreSQL psqLODBC](#)
- [PostgreSQL ODBC 编译](#)
- [Greenplum ODBC 下载](#)
- [Greenplum JDBC 下载](#)