



# PolarDB开源版 高级特性



睿创神脑  
RUICHUANGSHENNAO



PolarDB



# Objectives

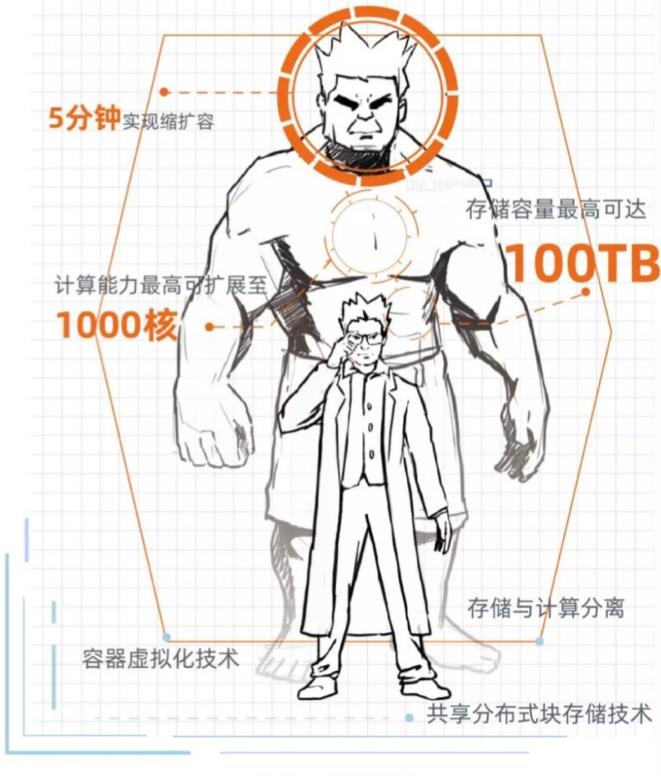
- PolarDB是什么
- 如何安装PolarDB开源版
- PolarDB高级特性原理讲解
- 体验PolarDB高级特性
- 更多数据库高级进阶课程
- 如何提升职场竞争力
- 加入PolarDB开源社区

# PolarDB是什么



## POLARDB侠

分钟级弹性伸缩的阿里云POLARDB



- 阿里云**自研云原生**关系型数据库，  
100%兼容PostgreSQL & MySQL，  
**高度兼容Oracle语法**
- 基于分布式架构和普通PC服务器，提  
供与商用数据库相当的能力，**成本仅  
有1/10**



# PolarDB是什么



阿里云PolarDB多篇论文被国际数据库顶会VLDB、SIGMOD录用

## Towards Cost-Effective and Elastic Cloud Database Deployment via Memory Disaggregation

Yingqiang Zhang<sup>‡</sup>, Chaoyi Ruan<sup>‡,†</sup>, Cheng Li<sup>‡</sup>, Xinjun Yang<sup>‡</sup>, Wei Cao<sup>‡</sup>, Feifei Li<sup>‡</sup>, Bo Wang<sup>‡</sup>, Jing Fang<sup>‡</sup>, Yuhui Wang<sup>‡</sup>, Jingze Huo<sup>‡,†</sup>, Chao Bi<sup>‡,†</sup>  
yingqiang.zyq@alibaba-inc.com, rcy@mail.ustc.edu.cn, chengl7@ustc.edu.cn, {xinjun.y, mingsong.cw, lifeifei, xianguo.wb, hangfeng.fj, yuhui.wyh, jingze.hjz}@alibaba-inc.com, bc23333@mail.ustc.edu.cn  
<sup>‡</sup>Alibaba Group and <sup>†</sup>University of Science and Technology of China

### ABSTRACT

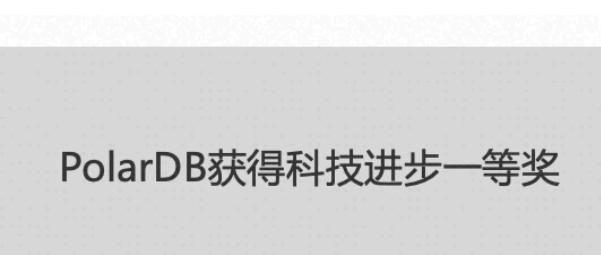
It is challenging for cloud-native relational databases to meet the ever-increasing needs of scaling compute and memory resources independently and elastically. The recent emergence of memory disaggregation architecture, relying on high-speed RDMA network, offers opportunities to build cost-effective and elastic cloud-native databases. There exist proposals to let unmodified applications run transparently on disaggregated systems. However, running relational database kernel atop such proposals experiences notable performance degradation and time-consuming failure recovery, offsetting the benefits of disaggregation.

To address these challenges, in this paper, we propose a novel database architecture called LegaBase, which explores the co-design of database kernel and memory disaggregation. It pushes the memory management back to the database layer for bypassing the Linux I/O stack and re-using or designing (remote) memory access optimizations with an understanding of data access patterns. LegaBase further splits the conventional ARIES fault tolerance protocol to

### 1 INTRODUCTION

With the increasing migration of applications from on-premise data centers to clouds, cloud-native relational databases have become a pivotal technique for cloud vendors. By leveraging modern cloud infrastructures, they provide equivalent or superior performance to traditional databases at a lower cost and higher elasticity. As a consequence, in recent years, major cloud vendors have launched their own cloud-native databases, such as Amazon Aurora[42], Azure Hyperscale [15, 30], Google Spanner [14] and Alibaba PolarDB[9, 10].

However, even the state-of-the-art cloud-native databases still embrace the monolithic server architecture where CPU and memory are tightly coupled. This makes it hard to fulfill the ever-growing and highly elastic resource demands from web-scale applications [36, 38]. For example, analytic queries favor a large amount of memory, which may exceed the capacity of a single machine, and would experience significant performance degradation when the working set does not fit into memory. In addition, the CPU utilization of database instances could be low most of the time [11, 32].



PolarDB获得科技进步一等奖



PolarDB 获科技进步一等奖

国内唯一、连续三年进入Gartner数据库领导者象限





# PolarDB是什么



PolarDB-PG 项目主页地址：

<https://github.com/ApsaraDB/PolarDB-for-PostgreSQL/>

PolarDB-X 项目主页地址：

<https://github.com/polaradb/polardbx-sql>

# PolarDB 技术架构

## 智能代理

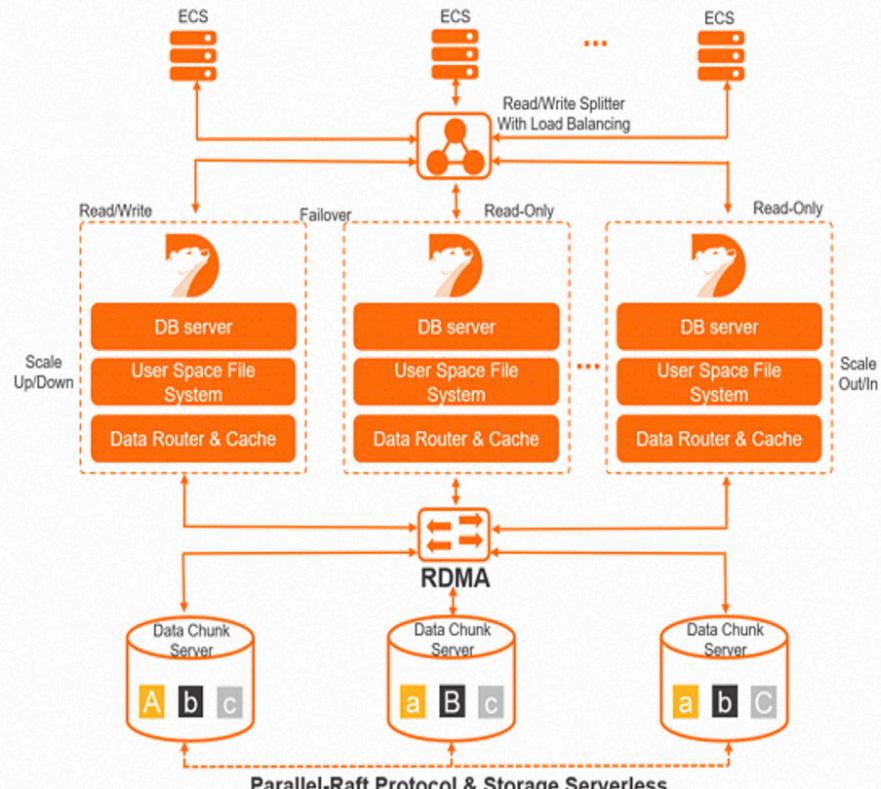
- **读写分离**: 智能分析SQL，并根据SQL自动做读写分离
- **负载均衡**: 自动在多节点做负载均衡并支持自定义业务负载

## 计算节点

- **计算与存储分离**: 每个计算集群一写多读，支持线性扩展，最大可扩展至15个只读节点
- **物理日志复制**: 主备数据复制延迟<1秒，数据分析及时有效
- **并行查询**: 充分利用多核CPU性能，加速业务数据处理
- **高可用检测**: 自动判断节点可用性，主备切换小于30S，会话自动转移，确保数据一致性
- **容器化管理**: K8S资源弹性调度，多租户管理。

## 存储节点

- **存储多副本**: 分布式存储实现多副本强一致，确保RPO=0
- **软硬一体高性能**: 网络、Bypass Kernel的用户态协议协同确保数据库25Gb RDMA高性能
- **秒级备份不加锁**: 利用存储快照实现秒级存储数据备份

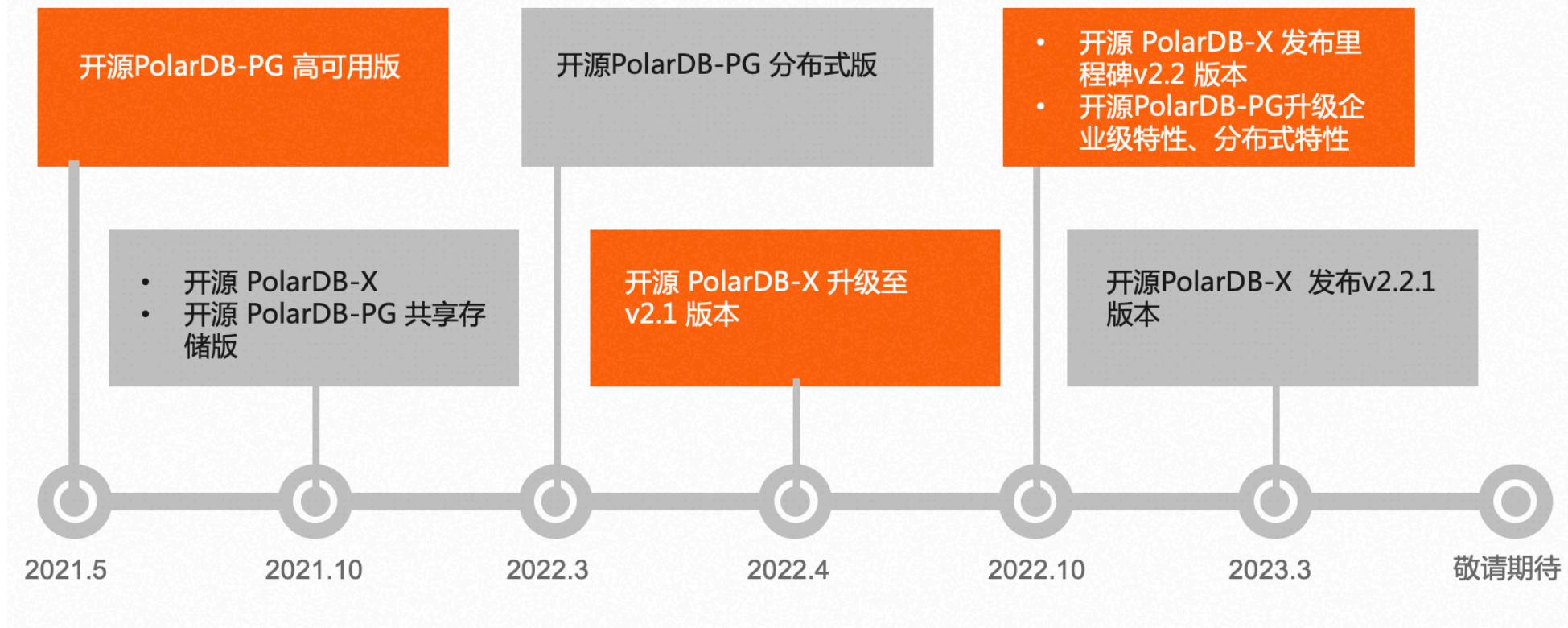




# PolarDB 全面开源时间线



2021年，阿里云把数据库开源作为重要战略方向，正式开源自研核心数据库产品PolarDB





# 如何安装PolarDB开源版



- # 拉取 HTAP PolarDB 镜像

```
docker pull polardb/polardb_pg_local_instance:htap
```

- # 创建运行并进入容器

```
docker run -it --cap-add=SYS_PTRACE --privileged=true --name polardb_pg_htap  
polardb/polardb_pg_local_instance:htap bash
```

- # 测试实例可用性

```
psql -h 127.0.0.1 -c 'select version();'  
version
```

```
-----  
PostgreSQL 11.9 (POLARDB 11.9)  
(1 row)
```

参考 PolarDB 动手实践课 (如何一键本地部署PolarDB for PostgreSQL 版):

<https://developer.aliyun.com/adc/scenarioSeries/7e5d067521df4552b8fd10e14565200f>

PolarDB官网手册: <https://apsaradb.github.io/PolarDB-for-PostgreSQL/zh/>

# PolarDB高级特性 - ePQ多机并行

## 功能优势

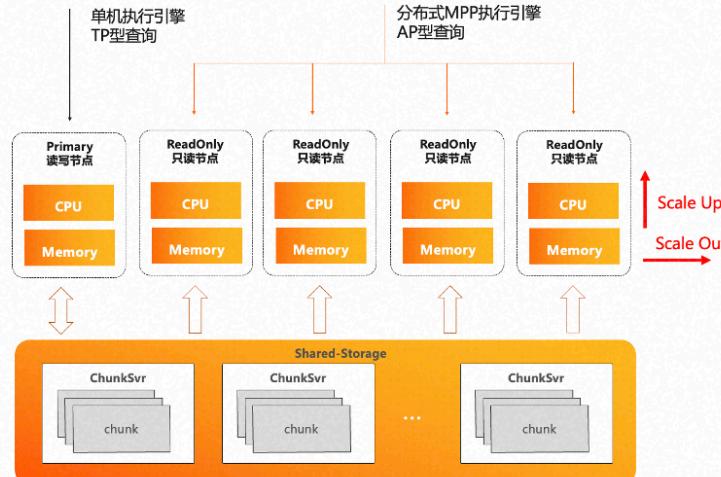
- 支持所有Query、DML、分区表；典型云原生，**支持任意节点发起MPP、单节点任意并行度、MPP执行节点任意划分、存储计算任意倾斜**
- 突破单节点CPU/IO瓶颈**，随业务增长实现线性加速

## 适用场景

- TP/AP需求不同时**：白天集群提供高性能TP服务，晚上集群提供高性能AP服务，如报表类
- TP/AP需求同时**：集群部分机器跑TP，部分机器跑AP。AP数据量在200TB以下，如HTAP场景
- AP需求弹性变化**：AP需求弹性变化，**要求计算节点快速弹性扩缩容**，如实时小数仓场景

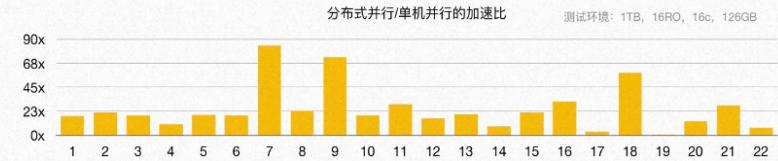
## 典型客户案例

- XX汽车：用于AI训练数据分析；某海运公司：用户时空数据分析

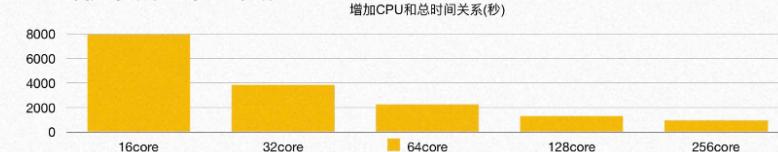


## ePQ测试结果

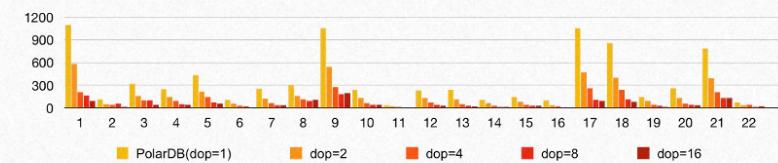
- 256GB，16个PolarDB-PG实例，1TB TPC-H测试结果



## 增加资源的性能变化



TPCH执行时间对比(秒)



## 单机并行 vs 分布式并行查询

相较于单机并行查询：  
在22条TPC-H SQL中，3条查询加速**60+倍**，  
19条查询加速**10+倍**，  
平均加速**23倍**

## 性能随资源扩展线性提升

增加CPU总核心数后（16c-128c），TPC-H总运行时间线性提升，单条SQL执行速度线性提升；



睿创神脑  
RUICHUANGSHENNAO

# PolarDB高级特性 - ePQ多机并行计算



PolarDB



PostgreSQL

4块SATA盘， 10亿条数据71GB的表，开跨节点的并行做group by，102秒出结果：

```
postgres=# explain analyze select n, count(*) from test01 group by n;
```

QUERY PLAN

---

```
PX Coordinator 24:1 (slice1; segments: 24) (cost=0.00..7702.11 rows=9013 width=12) (actual time=101217.270..101219.245 rows=10001 loops=1)
```

```
-> Finalize HashAggregate (cost=0.00..7701.85 rows=376 width=12) (actual time=101210.990..101211.068 rows=444 loops=1)
```

Group Key: n

```
-> PX Hash 24:24 (slice2; segments: 24) (cost=0.00..7701.80 rows=376 width=12) (actual time=99847.258..101210.355 rows=1482 loops=1)
```

Hash Key: n

```
-> Partial HashAggregate (cost=0.00..7701.79 rows=376 width=12) (actual time=101108.066..101108.458 rows=1581 loops=1)
```

Group Key: n

```
-> Partial Seq Scan on test01 (cost=0.00..2195.57 rows=41666408 width=4) (actual time=160.269..93734.425 rows=41690624 loops=1)
```

Planning Time: 6.280 ms

Optimizer: **PolarDB PX Optimizer**

Execution Time: 101265.420 ms

(11 rows)

Time: 101272.064 ms (01:41.272)



# PolarDB高级特性 – 预读



## 堆表预读

堆表预读的参数名为 `polar_bulk_read_size`，功能默认开启，默认大小为128kB。不建议用户自行修改该参数，128kB是贴合PFS的最优值，自行调整并不会带来性能的提升。

关闭功能：

```
ALTER SYSTEM SET polar_bulk_read_size = 0;  
SELECT pg_reload_conf();
```

sql

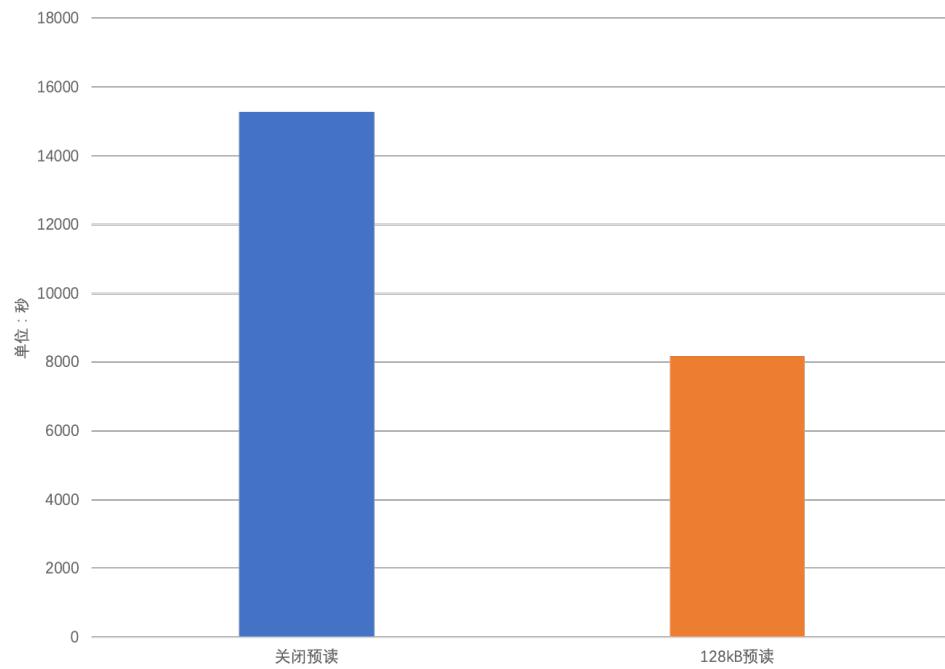
打开功能并设置预读大小为128kB：

```
ALTER SYSTEM SET polar_bulk_read_size = '128kB';  
SELECT pg_reload_conf();
```

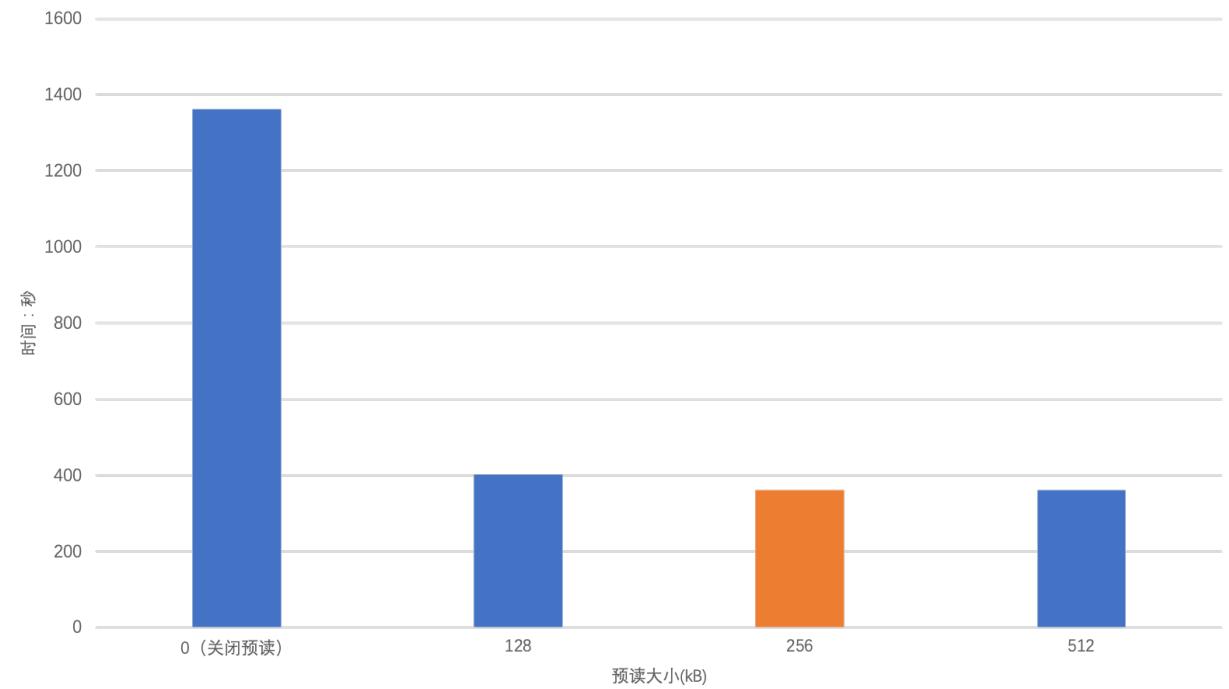
sql

# PolarDB高级特性 – 预读

400GB 表的vacuum性能:



400GB 表的 SeqScan 性能:





# PolarDB高级特性 – 预扩展



## 堆表预展

堆表预展的参数名为 `polar_bulk_extend_size`，功能默认开启，预展的大小默认为4MB。不建议用户自行修改该参数值，4MB为贴合PFS的最低优值。

关闭功能：

```
ALTER SYSTEM SET polar_bulk_extend_size = 0;  
SELECT pg_reload_conf();
```

sql

打开功能并设置预扩展大小为4MB：

```
ALTER SYSTEM SET polar_bulk_extend_size = '4MB';  
SELECT pg_reload_conf();
```

sql



# PolarDB高级特性 – 预扩展



未开启预扩展功能:

Device:	rrqm/s	wrqm/s	r/s	w/s	rMB/s	wMB/s	avgrq-sz	avgqu-sz	await	r_await	w_await	svctm	%util
dm-0	0.00	0.18	3.90	15.25	0.05	0.14	19.72	0.01	0.38	0.72	0.29	0.26	0.49
dm-0	0.00	0.00	80.00	928.00	0.40	5.16	11.31	0.41	0.40	1.51	0.31	0.37	37.50
dm-0	0.00	0.00	79.00	1070.00	0.49	6.27	12.05	0.41	0.35	1.37	0.28	0.31	35.90
dm-0	0.00	0.00	66.00	1019.00	0.34	5.94	11.86	0.39	0.36	1.45	0.29	0.34	36.50
dm-0	0.00	0.00	77.00	1236.00	0.42	22.32	35.47	0.59	0.45	1.42	0.39	0.33	43.30

开启预扩展功能, 写入速度提升4倍:

```
alter system set polar_bulk_extend_size=512;
select pg_reload_conf();
```

rrqm/s	wrqm/s	r/s	w/s	rMB/s	wMB/s	avgrq-sz	avgqu-sz	await	r_await	w_await	svctm	%util
0.00	34.50	44.00	6099.00	0.22	93.07	31.10	0.97	0.16	0.53	0.16	0.07	46.05
0.00	34.50	4.00	6100.00	0.04	93.07	31.24	1.45	0.24	1.00	0.24	0.11	66.55
0.00	23.50	39.50	2919.00	0.18	60.84	42.25	0.53	0.18	0.72	0.17	0.10	29.55
0.00	23.50	1.50	2919.00	0.02	60.84	42.68	0.83	0.28	0.33	0.28	0.16	45.35
0.00	0.00	24.50	8073.50	0.10	108.76	27.53	1.24	0.15	0.55	0.15	0.07	56.60
0.00	0.00	0.50	8073.50	0.00	108.76	27.59	1.96	0.24	1.00	0.24	0.10	79.65

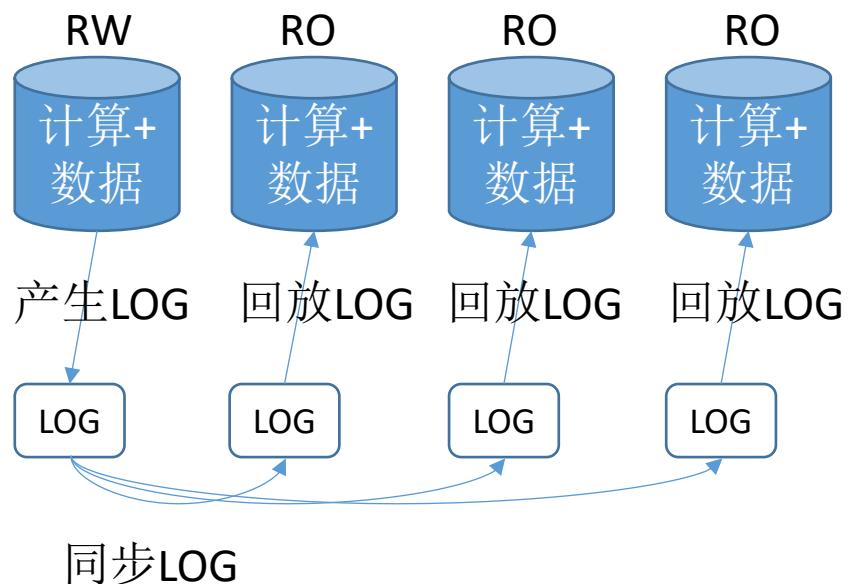
# PolarDB高级特性 – 一写多读

传统架构:

每个数据库实例需要一份数据, 通过日志进行同步.

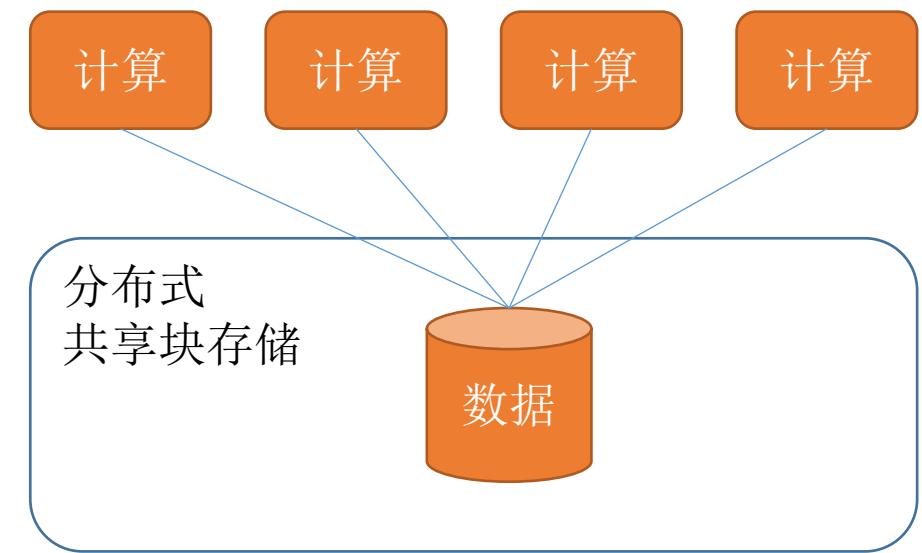
缺点:

- 成本高
- 同步延迟较高
- 扩展不方便

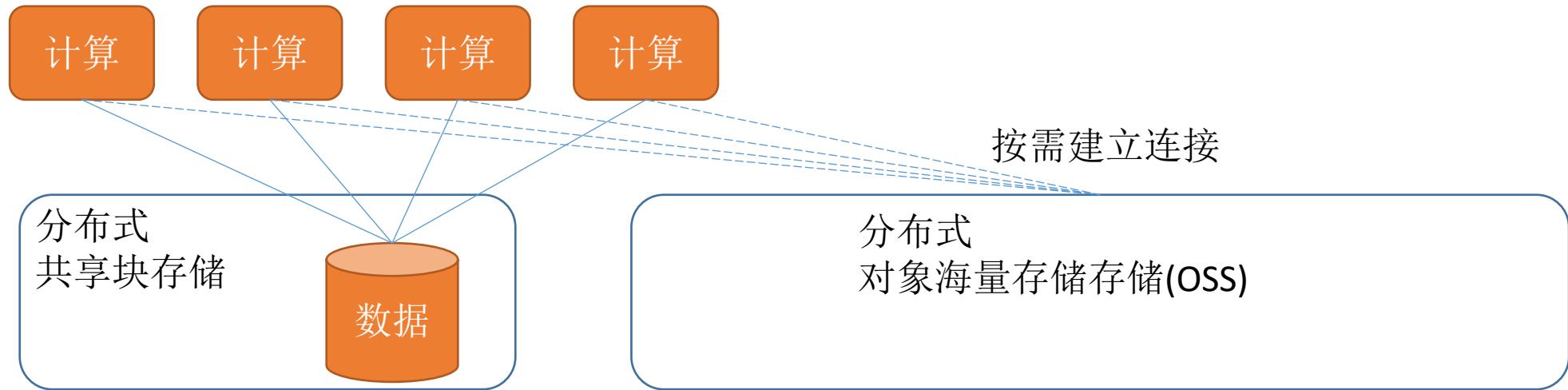


PolarDB:

一份数据, 多个计算节点



# PolarDB高级特性 – 冷热分离



本地存储:  
延迟低、带宽高、速度快、贵、  
无法直接跨实例分享数据

OSS存储: 延迟高、带宽一般、速度一般、便宜  
可以跨实例分享数据.  
近乎无限存储空间



# PolarDB高级特性 – 冷热分离



```
# 创建插件
create extension oss_fdw;
# 创建 server
CREATE SERVER ossserver FOREIGN DATA WRAPPER oss_fdw OPTIONS
    (host 'oss-cn-hangzhou.aliyuncs.com' , id 'xxx', key 'xxx', bucket 'mybucket');
# 创建 OSS 外部表
CREATE FOREIGN TABLE ossexample
    (date text, time text, open float,
     high float, low float, volume int)
    SERVER ossserver
    OPTIONS (filepath 'osstest/example.csv', delimiter ',',
             format 'csv', encoding 'utf8', PARSE_ERRORS '100');
# 创建表, 数据就装载到这张表中
create table example
    (date text, time text, open float,
     high float, low float, volume int);
# 数据从 ossexample 装载到 example 中。
insert into example select * from ossexample;
```



# PolarDB高级特性 – 冷热分离



```
# 可以看到
```

```
# oss_fdw 能够正确估计 OSS 上的文件大小，正确的规划查询计划。  
explain insert into example select * from ossexample;  
          QUERY PLAN
```

---

```
Insert on example (cost=0.00..1.60 rows=6 width=92)  
  -> Foreign Scan on ossexample (cost=0.00..1.60 rows=6 width=92)  
        Foreign OssFile: osstest/example.csv.  
        Foreign OssFile Size: 728  
(4 rows)
```

```
# 表 example 中的数据写出到 OSS 中。
```

```
insert into ossexample select * from example;  
explain insert into ossexample select * from example;  
          QUERY PLAN
```

---

```
Insert on ossexample (cost=0.00..16.60 rows=660 width=92)  
  -> Seq Scan on example (cost=0.00..16.60 rows=660 width=92)  
(2 rows)
```

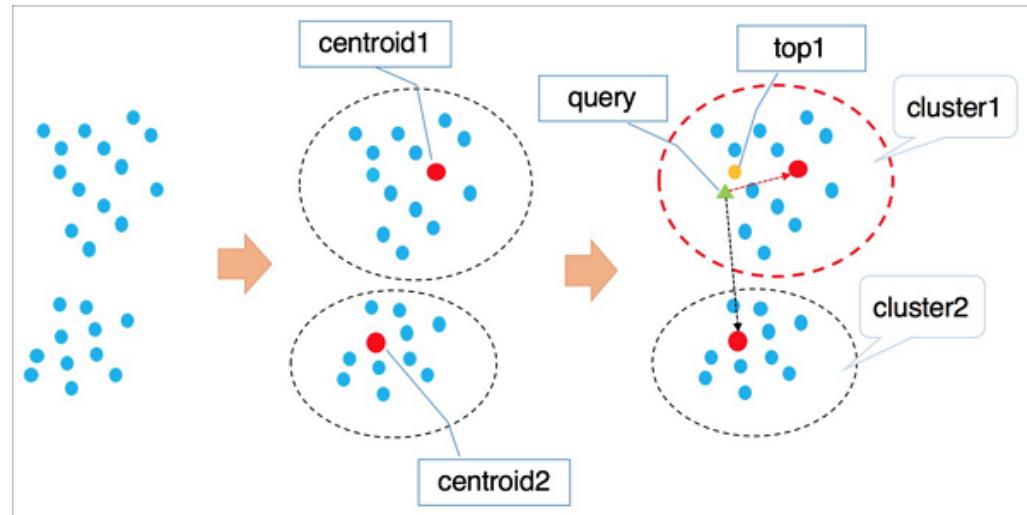
参考手册:

[https://help.aliyun.com/document\\_detail/118532.html](https://help.aliyun.com/document_detail/118532.html)

# PolarDB高级特性 – 向量检索

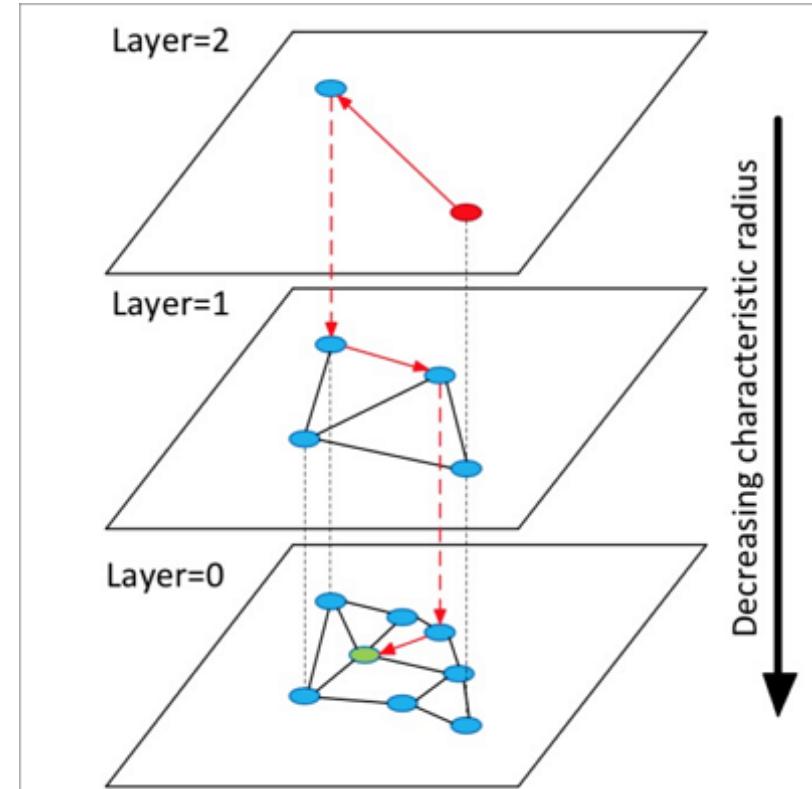
适合AI的向量相似搜索, 需要向量索引的加速:  
图像识别、聊天机器人、相似论文筛选、...

ivfflat:



参考文档: [https://help.aliyun.com/document\\_detail/446755.html](https://help.aliyun.com/document_detail/446755.html)

hnsw:





# 体验PolarDB高级特性



PolarDB for PostgreSQL 版动手实践实验:

<https://developer.aliyun.com/adc/scenarioSeries/7e5d067521df4552b8fd10e14565200f>

开发者社区 > 云起实验室 > 实践系列课 > PolarDB for PostgreSQL 动手实践

## PolarDB for PostgreSQL 动手实践

云原生数据库 PolarDB for PostgreSQL（简称 PolarDB-PG）是一款阿里云自主研发的云原生关系型数据库产品，100% 兼容 PostgreSQL，具有极致弹性、毫秒级延迟、HTAP 的能力和高可靠、

6个实验 157 6小时 ★★★★★ 9.6分

已加入课程



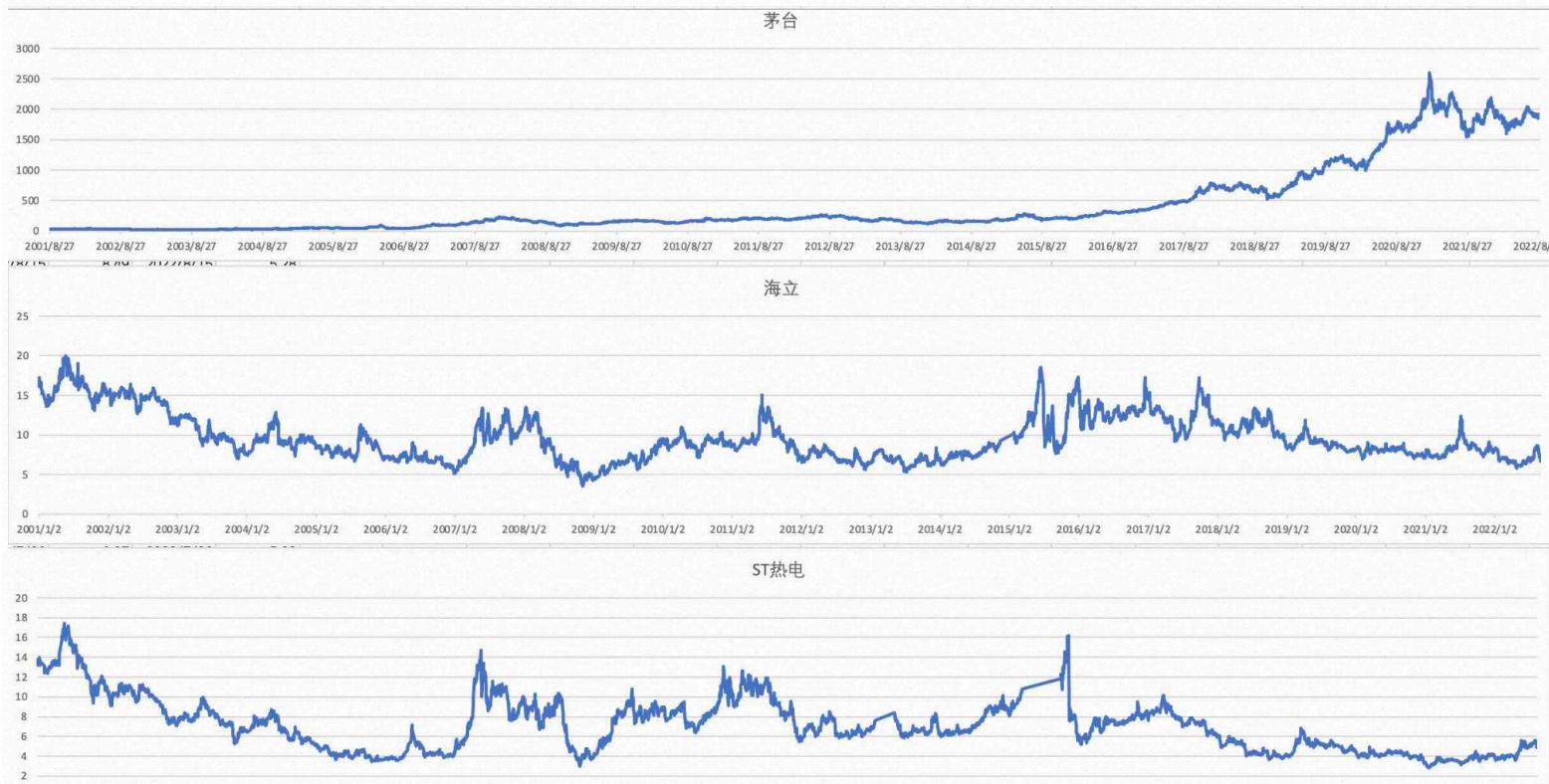
# 体验PolarDB高级特性



练习,用PolarDB证明巴菲特的投资理念

[https://github.com/digoal/blog/blob/master/202209/20220908\\_02.md](https://github.com/digoal/blog/blob/master/202209/20220908_02.md)

[https://github.com/digoal/blog/blob/master/202209/20220909\\_01.md](https://github.com/digoal/blog/blob/master/202209/20220909_01.md)





# 更多数据库进阶课程



- 1.PG手册: <https://www.postgresql.org/docs/current/index.html>
- 2.PG开源项目: <https://git.postgresql.org/gitweb/?p=postgresql.git;a=summary>
- 3.PolarDB-PG手册: <https://apsaradb.github.io/PolarDB-for-PostgreSQL/>
- 4.PolarDB-PG开源项目: <https://github.com/ApsaraDB/PolarDB-for-PostgreSQL>
- 5.PolarDB-X手册: <https://doc.polardbx.com/>
- 6.PolarDB-X开源项目: <https://github.com/polardb/polardbx-sql>
- 7.PolarDB-X知乎官方号: <https://www.zhihu.com/org/polardb-x>
- 8.PolarDB 开源社区B站视频号: <https://space.bilibili.com/2070322174>
- 9.PolarDB 高手课: <https://developer.aliyun.com/learning/course/1258>
- 10.PolarDB-PG 内核从入门到精通: <https://developer.aliyun.com/learning/topic/database>
- 11.PolarDB 开源学习图谱: <https://www.aliyun.com/database/openpolardb/activity>
- 12.免费动手实验: <https://developer.aliyun.com/adc/>
- 13.免费云资源: <https://click.aliyun.com/m/1000371820/>
- 14.其他资料:  
<https://www.interdb.jp/pg/index.html>  
<https://postgrespro.com/community/books/internals>  
<https://github.com/digoal/blog/blob/master/README.md>



# 精品课程



## 数据库内核解读课程

本系列课程将面向DBA、高校学生、内核爱好者，介绍PG内核架构、各模块基本原理、用法、代码实现。

阿里云  
开源学堂

### PostgreSQL 数据库内核解读系列

第三讲：PostgreSQL 存储管理(一)

于巍（漠雪）  
阿里云数据库开源社区 Maintainer

时间  
07/08 (周五) 16:00~17:00

个人简介：  
13年数据库内核开发和架构设计经验，目前主要负责横向拉通阿里云数据库各产品通用技术和架构演进，标准化，开源技术等工作。

## 数据库内核从入门到精通

阿里云数据库专家携手高校教师带你从零开始系统化地学习数据库理论知识，由浅入深提升数据库实践能力。

### 校企合作系列课程 《数据库内核从入门到精通》 正式开讲！

阿里云开发者社区、PolarDB开源社区、武汉大学联合出品

## 公开课：PolarDB高手课

本系列课程将对PolarDB开源技术和实践、上云方法论进行一个整体的解读。

### 王远 · PolarDB 高手课

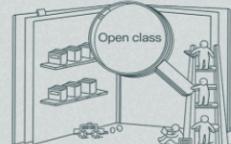
大师论道，数据库最佳实践一通百通

#### 你将获得

- 揭秘数据库技术发展趋势与机遇
- 5个数据库快速上云要点实操
- 读懂 PolarDB-X 与 PolarDB-PG 核心原理
- 掌握 PolarDB-X 与 PolarDB-PG 实践方案

#### 王远 (惊玄)

阿里云  
数据库技术架构部负责人  
资深技术专家



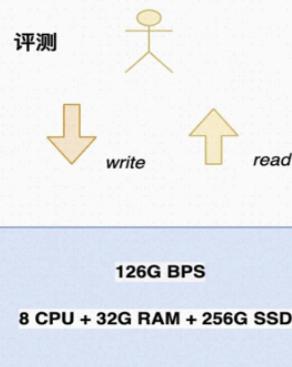


# 开发者赛事



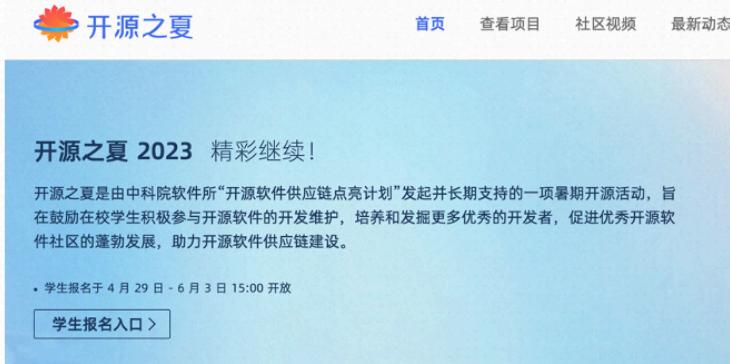
## 第四届全球数据库大赛：分布式 NewSQL性能挑战

本次比赛以使用PolarDB分布式版本为主，以分布式NewSQL为背景，围绕数据库索引的多维查询。



## 开源之夏2023:高校学生的编程比赛

鼓励在校学生积极参与开源软件的开发维护，培养和发掘更多优秀的开发者，促进优秀开源软件社区的蓬勃发展，助力开源软件供应链建设。



## 阿里巴巴编程之夏：高校学生的编程比赛

面向全球18岁及以上本科、硕士、博士高校学生的技术普惠计划，它旨在鼓励高校学生深度参与开源开发活动，激励学生以第一视角感受开源世界的魅力。





# 如何提升职场竞争力



## 学生痛点：

- 1 压力大：学分、毕业设计、考研、就业
- 2 静不下心来深入钻研
- 3 不了解企业业务场景和用人需求
- 4 圈子小（老师，同学，亲人），酒香（优秀学员）也怕巷子深
- 5 简历单薄，缺乏竞争力

# 如何提升职场竞争力

## 什么样的人才具有竞争力?

- 热点行业 +



行业趋势、市场规模

- 先进思维 +



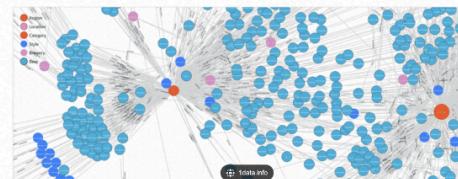
(公理体系+哲科)  
职业天花板、跨行业、跨角色、跨舒适区能力

- 先进生产力工具 +



冷兵器 vs 原子弹

- 圈子(个人影响力) +



连接,被连接,强连接,贵人  
(《超级版图》)

- 战绩

图片来源:互联网

知行合一



理论+实证主义



# 如何提升职场竞争力



- 1 获得内核研发技能
  - 《数据库内核研发从入门到精通》 <https://developer.aliyun.com/article/1128441>
    - 成果: 学完可以进行云原生分布式数据库功能研发
- 2 获得数据库通识和应用技能
  - 《PG&PolarDB开源数据库工作室》课程
    - 成果: 掌握数据库原理、应用开发、实践技能
- 3 掌握企业应用场景和实战技能
  - 免费云资源: <https://click.aliyun.com/m/1000371820/>
  - 免费实验室: <https://developer.aliyun.com/adc/>
- 4 获得技能证明 (但不要止步于考证)
  - 参与考试认证 (训练营、数据库人才认证)
  - 学习和考试图谱: <https://www.aliyun.com/database/openpolardb/activity/>



# 如何提升职场竞争力



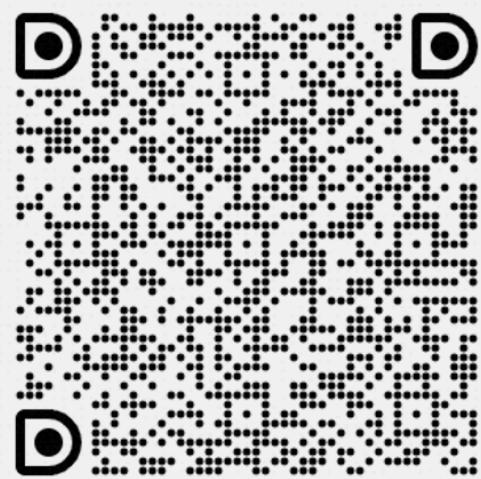
- 5 进入开源数据库圈子 (拓展人脉, 专家直通问答, 提升自己的连接数)
  - 微信群、钉钉群、PolarDB开源工作室、合作生态伙伴、合作社区等 ... 数十万人群(行业领袖,企业CTO,架构师,开发者等)
- 6 参加分享, 提升个人影响力, 变成社区明星, 提升自己的被连接数 以及 强连接数
  - 参与开源meetup分享、开源官网发表文章、在线分享(开源学堂、开源训练营等)、评选数据库之星 ...
- 7 参加比赛; 加入SIG, 参加开源项目代码贡献(顺便完成毕业设计)
  - 组队参与编程之夏、开源之夏、黑客松等比赛活动(赢钱赢名气);
  - 参与PolarDB开源兴趣小组和开源项目代码贡献;
  - 把自己变成金字塔尖人才, 为简历镀金, 争取“保送”机会;
- 8 优秀人才内部推荐、实习机会



# 加入PolarDB开源社区



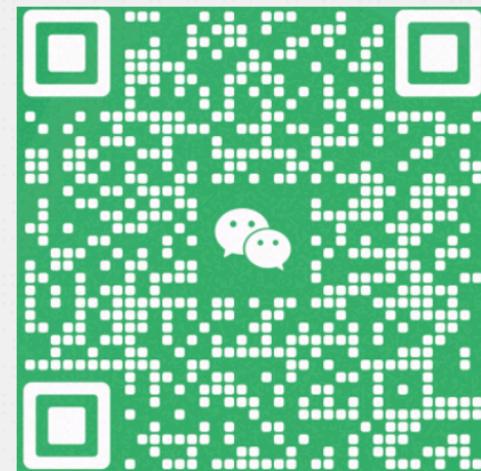
PolarDB-X 开源交流钉钉群



PolarDB-PG 开源交流钉钉群



PolarDB微信公众号



内核交流微信



# 总结

- PolarDB是什么
- 如何安装PolarDB开源版
- PolarDB高级特性原理讲解
- 体验PolarDB高级特性
- 更多数据库高级进阶课程
- 如何提升职场竞争力
- 加入PolarDB开源社区



# 练习

- 1、观察多机并行查询对大表查询性能的提升效果
- 2、观察预读带来的大表查询seqscan和vacuum性能提升
- 3、观察预扩展对数据批量写入带来的性能提升
- 4、连接到不同的计算节点, 观察每个计算节点的数据同步延迟



睿创神脑  
RUICHUANGSHENNAO

