



第九届PostgreSQL中国技术大会  
2019 PostgreSQL Conference China

# PG 新特性

阿里云

digonal



# 目录

- PG 12 重点新特性
- 风向标



# PG 12 重点新特性

- 性能增强
  - AM接口
  - 分区表大幅性能提升
  - GiST index include 任意数据组织存储
  - Btree 索引duplicate key存储组织sort by ctid增强性能
  - 自定义统计信息支持多列MCV

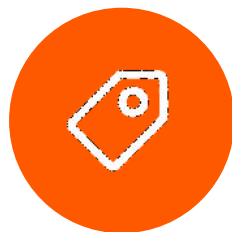


# PG 11+



高性能  
几乎线性提升

- 单表10亿测试
- 平均提速20倍+



自动计算并行度

- 不需要改写SQL
- 自动根据SQL代价启动并行计算



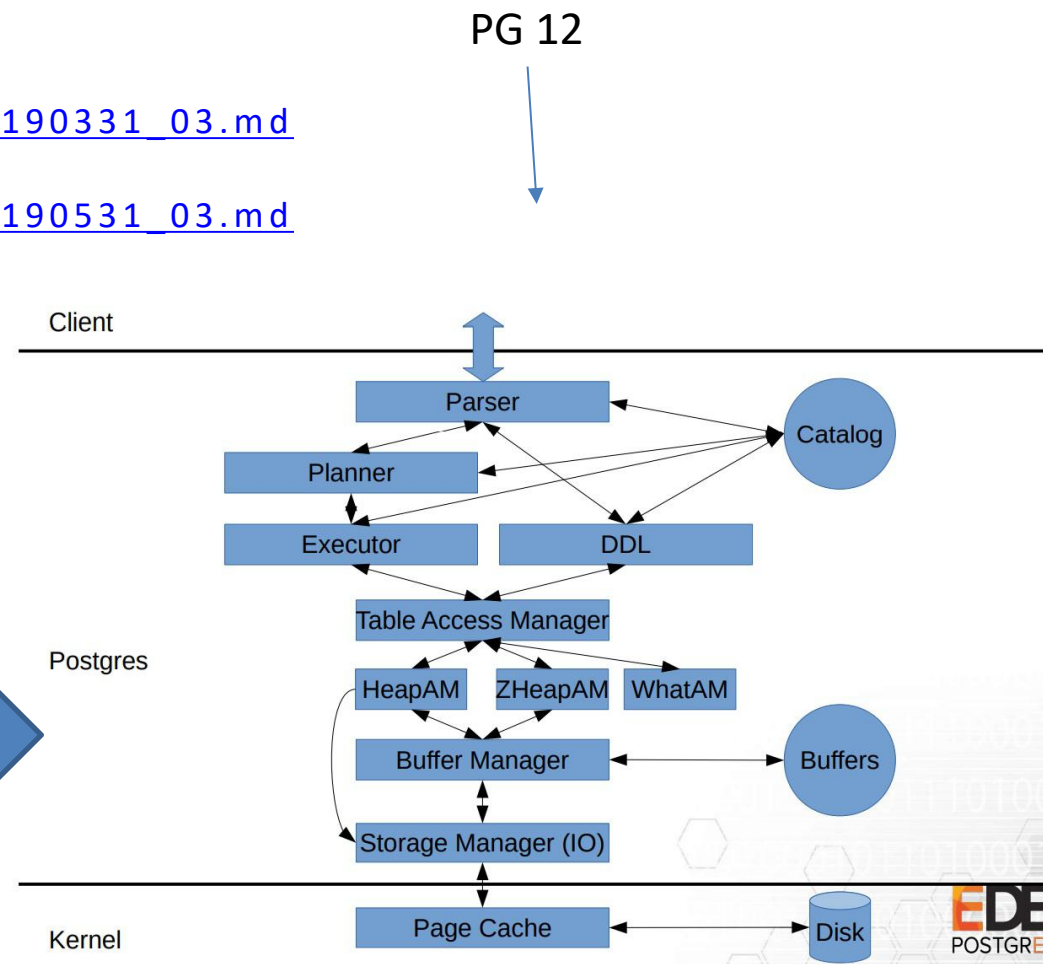
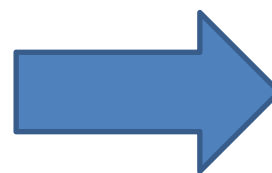
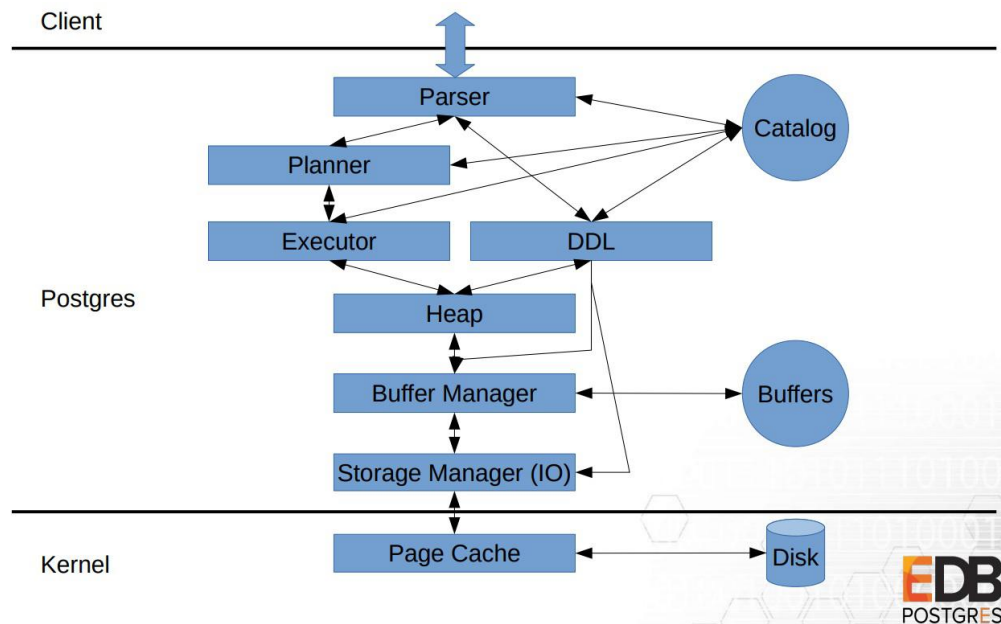
覆盖几乎所有复杂查询

- 表扫描
- 索引扫描
- SQL 过滤
- JOIN、子查询
- 聚合、自定义函数
- 分区表、外部表
- 自定义聚合、函数
- 排序、建表、物化、创建索引



# AM接口

- [https://github.com/digoal/blog/blob/master/201903/20190331\\_03.md](https://github.com/digoal/blog/blob/master/201903/20190331_03.md)
- [https://github.com/digoal/blog/blob/master/201905/20190531\\_03.md](https://github.com/digoal/blog/blob/master/201905/20190531_03.md)







# 存储引擎扩展

- zedstore(列存)
- zheap(支持回滚段)

```
postgres=# SELECT AVG(i199) FROM (select i199 from layout offset 0) x; -- heap
          avg
-----
500000.500000000000000
(1 row)

Time: 4679.026 ms (00:04.679)

postgres=# SELECT AVG(i199) FROM (select i199 from zlayout offset 0) x; -- zedstore
          avg
-----
500000.500000000000000
(1 row)

Time: 379.710 ms
```



# 分区表-大量分区性能提升

## • 1000个分区、469倍

- <https://github.com/digooz/postgresql-partitioning-benchmark>
- <https://github.com/digooz/postgresql-partitioning-benchmark>

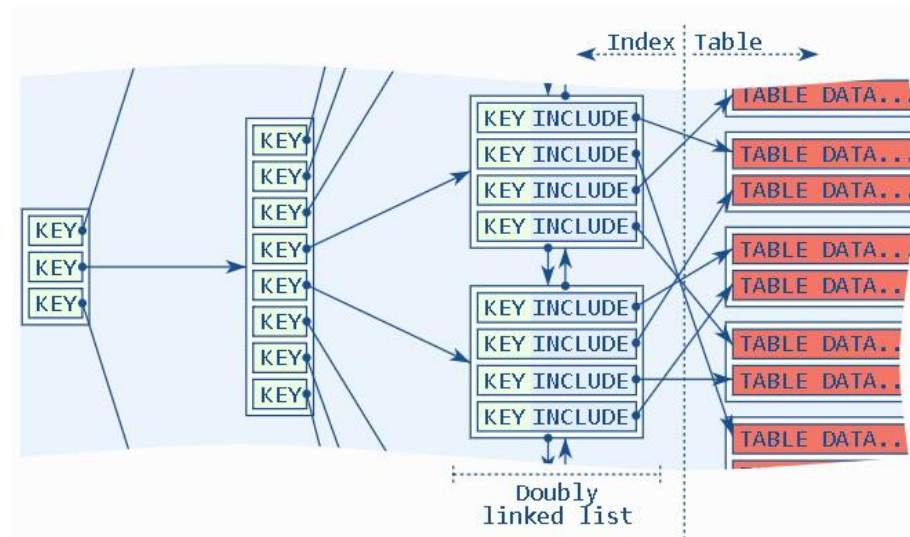
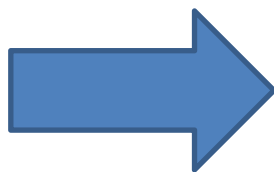
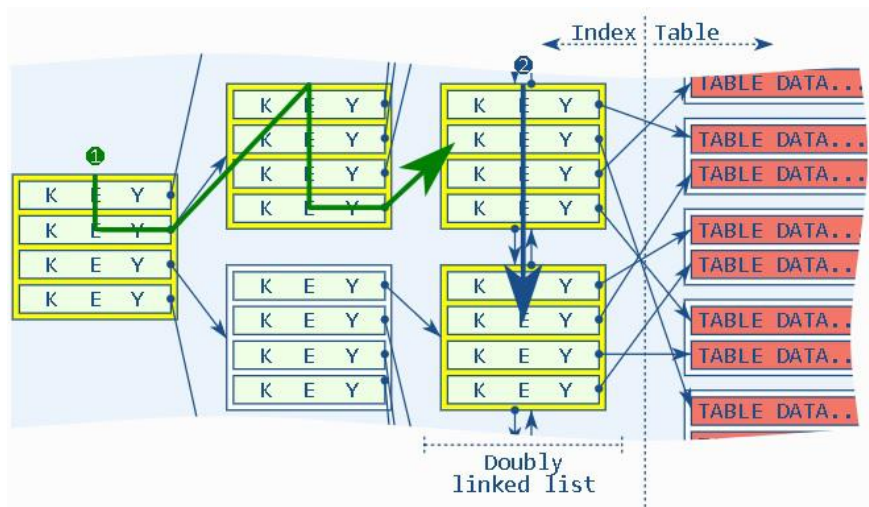
case	PG 11	PG 12 beta1	性能提升倍数
单表查询 qps	1161671	1160909	持平
单表upsert qps	317379	332552	持平
分区表(1024分区)查询 qps	1163	545602	469倍
分区表(1024分区)upsert qps	2885	246627	85倍

## • 痛点:

# Index include

- 数据离散存储，查询多条时IO多，查询慢，占用内存多，memcpy效率低。
- 聚集存储（b+t）只有一种存储形态，聚集KEY必须是PK，二级索引效率低，实用性差。
- PG 11 Btree, PG 12 GiST support index include

```
• create index idx_t1_1 on t1 (id) include(c1,c2,c3,info,crt_time);
```

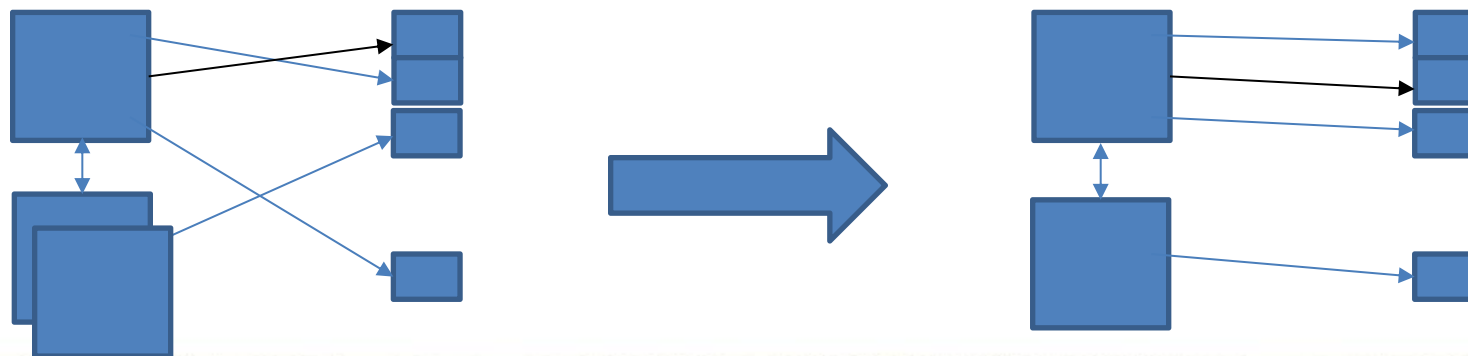






# Btree, duplicate keys sort by ctid

- [https://github.com/digoal/blog/blob/master/201911/20191128\\_03.md](https://github.com/digoal/blog/blob/master/201911/20191128_03.md)
- 1、duplicate keys值按ctid排序存储，减少索引页split。原来是按duplicate group无序存储。
- 2、提高查询效率（有序scan）
- 3、降低索引存储空间。
- 4、vacuum时，scan table，存储垃圾tuple ctids，存满vacuum memory(maintenance mem)，scan index回收垃圾index tuples。当回收整个duplicate keys，vacuum产生的wal更少(涉及的index page更少))。





# 自定义统计信息支持多列MCV

- Where a = ? And b=? Or c=? And d=? ...
- 评估选择性

单列柱状图

单列柱状图

简单粗暴：  
单列概率相乘、相加



多列柱状图

概率更加精准

单列柱状图



# PG 12 重点新特性

- 功能

- Reindex Concurrently 不堵塞写操作
- 日志采样
- **SQL/JSON path language**
- Generated columns
- Case-insensitive and accent-insensitive grouping and ordering
- Checksum offline 修改， pg\_checksum插件支持online开关checksum



# 日志采样

- [https://github.com/digoal/blog/blob/master/201904/20190405\\_09.md](https://github.com/digoal/blog/blob/master/201904/20190405_09.md)
- `log_statement_sample_rate`，当设置了`log_min_duration_statement`时，如果`log_statement_sample_rate`也设置了，它表示百分之多少的超时SQL被记录。
- `log_transaction_sample_rate`，不管其他任何设置，它表示百分之多少的事务被记录duration。（事务为最小粒度单位，如果一个事务被触发了记录duration，这个事务中的所有SQL都会被记录。）





# SQL/JSON path language

- 新增类型jsonpath
- <https://www.postgresql.org/docs/12/functions-json.html#FUNCTIONS-SQLJSON-PATH>

Example JSON	Example Query	Result
<code>{"x": [2.85, -14.7, -9.4]}</code>	<code>+ \$.x.floor()</code>	<code>2, -15, -10</code>
<code>{"x": [2.85, -14.7, -9.4]}</code>	<code>- \$.x.floor()</code>	<code>-2, 15, 10</code>
<code>[2]</code>	<code>2 + \$[0]</code>	<code>4</code>

<code>["abc", "abd", "aBdC", "abdacb", "babc"]</code>	<code>\$[*] ? (@ like_regex "^ab.*c" flag "i")</code>	<code>"abc", "aBdC", "abdacb"</code>
<code>["John Smith", "Mary Stone", "Bob Johnson"]</code>	<code>\$[*] ? (@ starts with "John")</code>	<code>"John Smith"</code>
<code>{"x": [1, 2], "y": [2, 4]}</code>	<code>strict \$.* ? (exists (@ ? (@[*] &gt; 2)))</code>	<code>2, 4</code>
<code>[-1, 2, 7, "infinity"]</code>	<code>\$[*] ? ((@ &gt; 0) is unknown)</code>	<code>"infinity"</code>



# 风向标

- PG DNA

- License
- 社区化开源数据库
- 模块化设计，Easy extendable
- 社区化运作有组织有纪律
- Major sponsors组成-健壮性
- 底盘（可靠性、安全性、sql兼容性、可用性、易用性、生态）



# 风向标

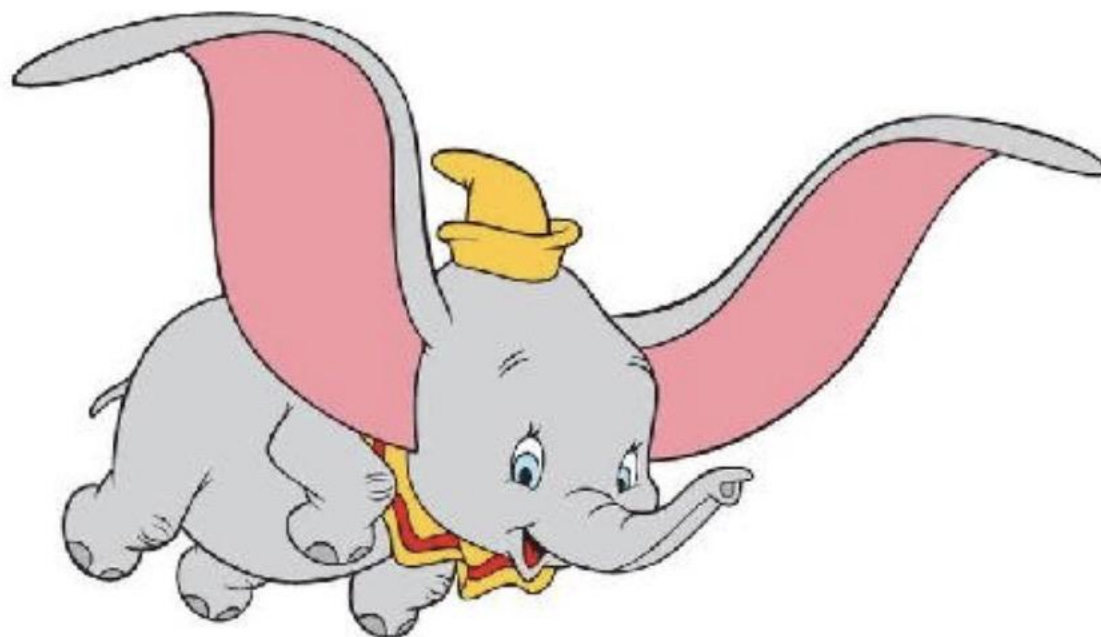
- 大环境
  - 拥抱开源，全球化趋势
  - 自主可控



# 风向标

- 风向、机会

- RAC
- NewSQL
- 垂直化市场功能
- SaaS(ERP,。。。)
- 工具生态（同步、迁移、...）
- Support、培训







# 谢谢

PG|POLARDB技术…

部门



该群属于“阿里云开发者社区”部门群，仅组织内部成员可以加入，如果组织外部人员收到此分享，需要先申请加入该组织。