

# PG+金融架构演进之路

Financial

汪洋

平安科技（深圳）有限公司



平安科技  
PINGAN TECHNOLOGY

汪洋，中国平安集团旗下平安科技数据库技术部总监。近年，对开源数据库技术PostgreSQL产生浓厚兴趣，大力在集团内部进行引入、推广和使用工作。

- ✓ Oracle7 OCP，20+年Oracle经验
- ✓ 2015年开始正式推广PostgreSQL
- ✓ 已有PostgreSQL实例1000+
- ✓ 上线两年零故障





content

1 战略需求

2 战术打击

3 探索前进

# 1 战略需求



## 战略需求

——金融行业IT系统要求

# 金融系统



安全

数据安全

保护客户信息与财富



高可用

架构高可用

保障系统稳定



性能

性能稳定

保证交易体验



推广

推广策略

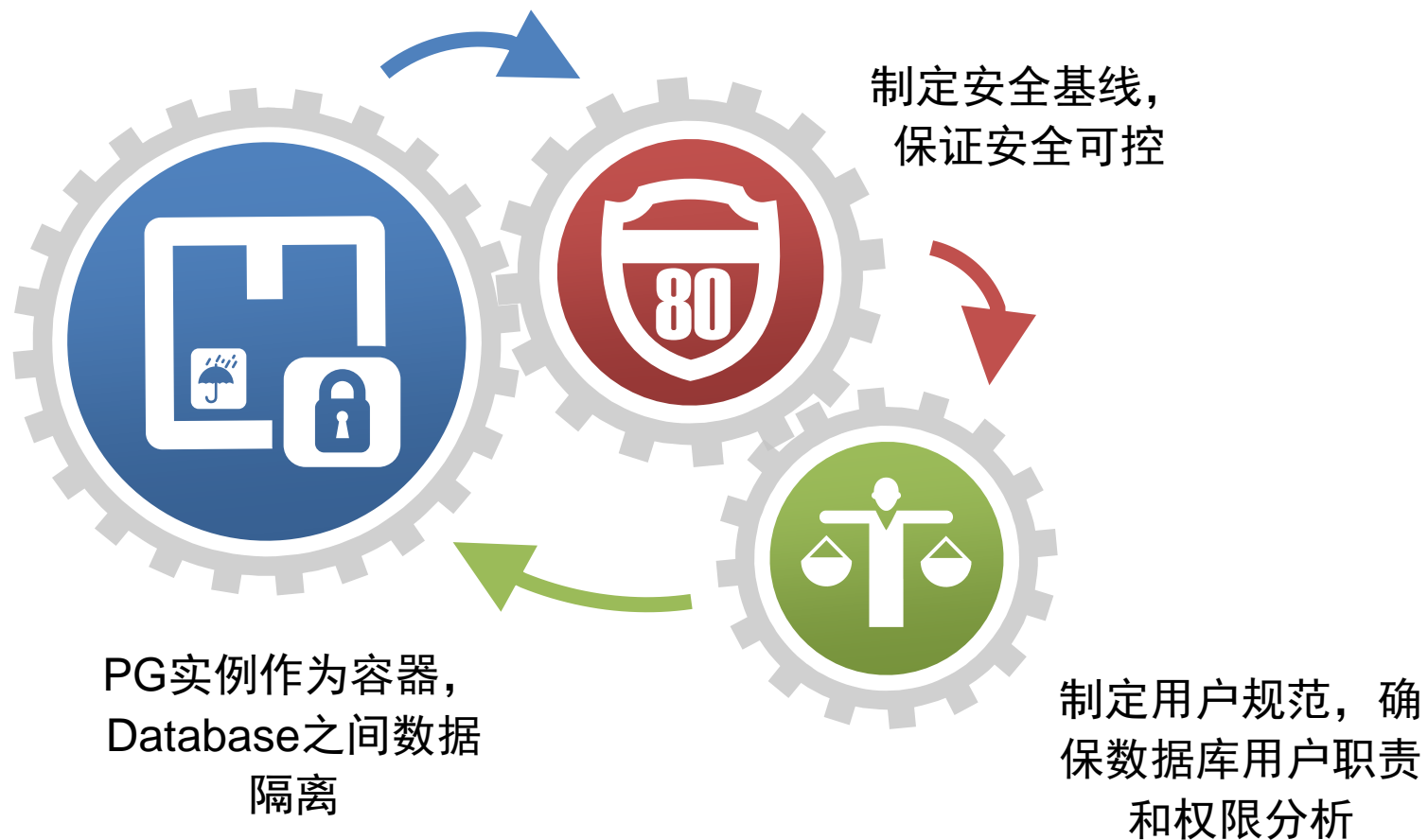
系统平稳过渡





# 战略需求

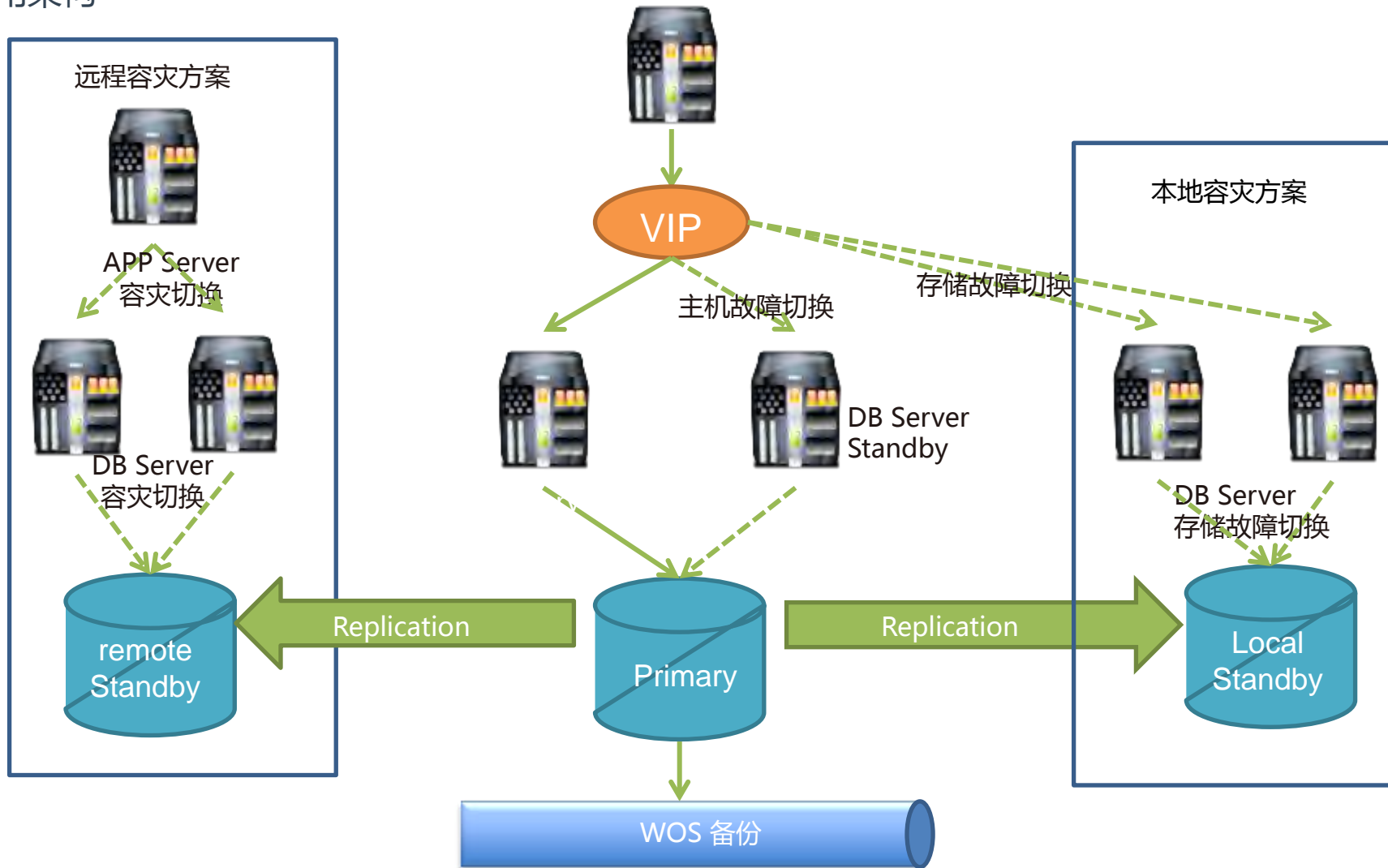
——数据安全





# 战略需求

——高可用架构





## 战略需求

——容灾方案选择



标准的高可用组件，一般用于应对主机故障。也可用于少数实例资源需求突增的临时隔离方案。



本地容灾方案。一般用于生产存储故障切换。也可用于备份和读写分离，减轻生产IO压力。



异地容灾方案。可以使用存储SVC技术快速搭建测试环境。可以使用存储COW技术搭建生产快照库。



**根据实际业务的SLA需求选择不同的容灾方案**

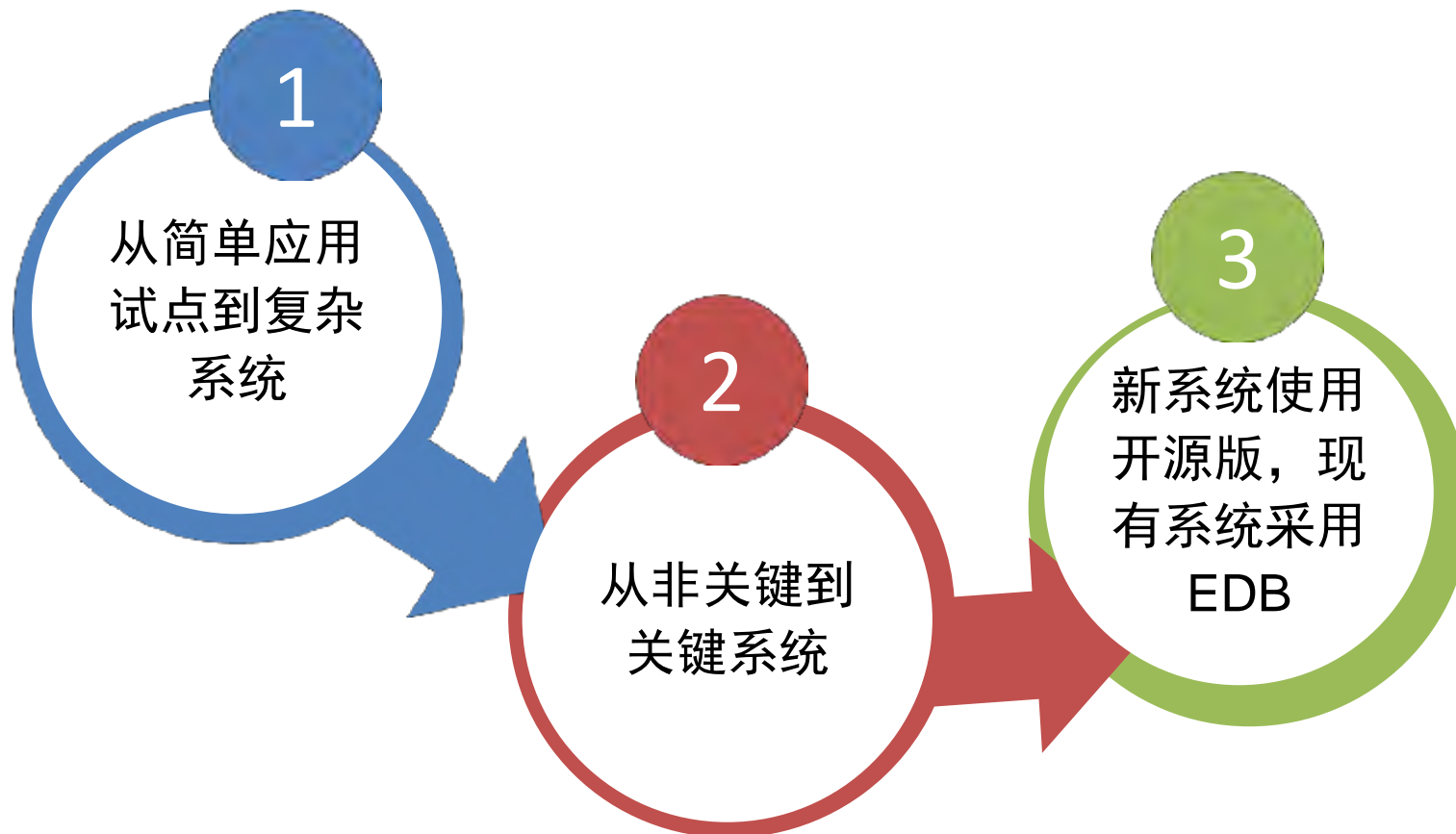






# 战略需求

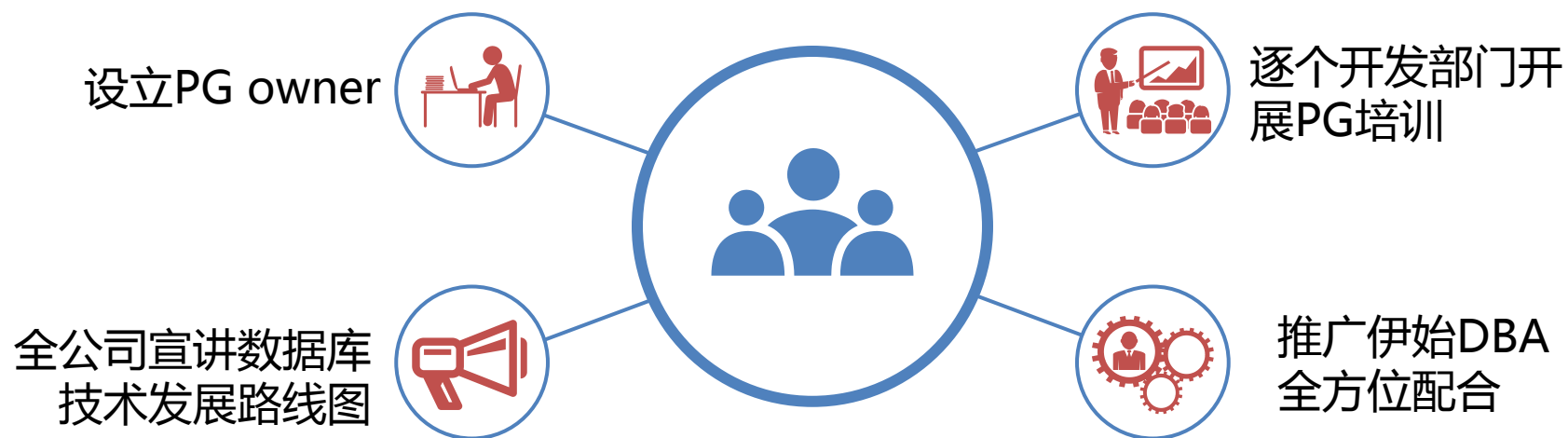
——推广策略





# 战略需求

——人员培养策略





# 战略需求

——PG其他优势

- **容易转型**

PG很多设计理念和Oracle很类似，无论是运维还是开发人员，学习周期较短

- **强大特性**

数据库自身的特性很强，例如支持复杂SQL查询以及强大的过程性语言支持，利于Oracle向PG的迁移

- **容易扩展**

提供多种的数据类型支持以及扩展选件，例如JSON、PostGIS、FDW等，意味着可以简化或者整合企业内部的不同类型数据库产品，降低运维成本

- **快速部署**

轻量，可以快速部署，缩短系统上线周期，加快企业产品推向市场的时间



## 2 战术打击



## 战术打击

——制定规范



### 架构规范

机器选型

操作系统版本

操作系统用户配置

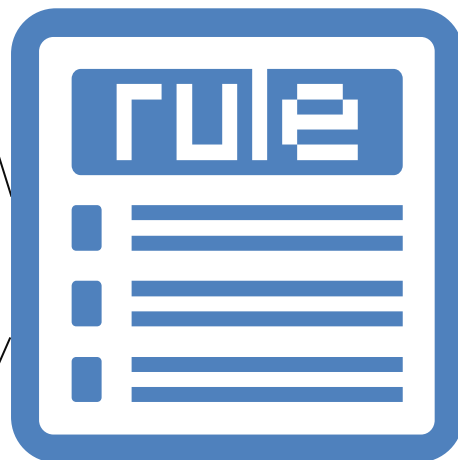
存储卷的规划



### 开发规范

规范SQL写法

避免重复踩坑



### 用户规范

角色分工

最小权限原则

保障数据安全

Schema使用



### 安全基线

角色密码复杂度

禁用默认用户

不使用默认端口

审计DDL

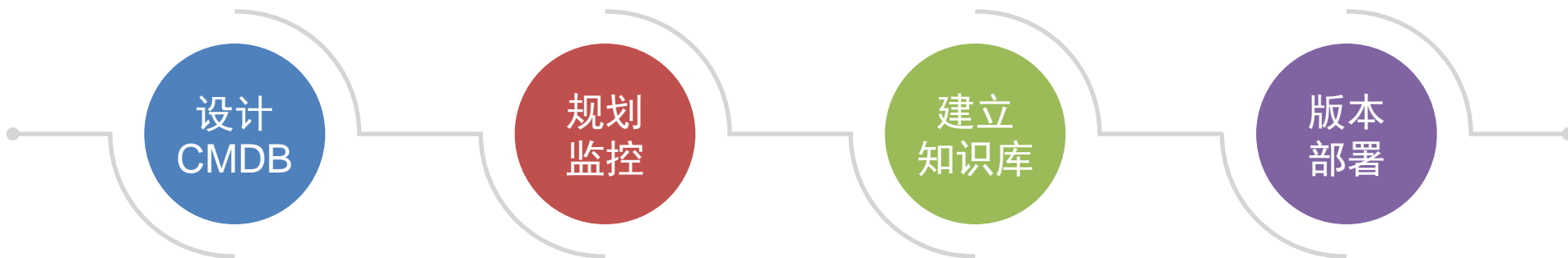
下线审计





## 战术打击

——其他准备



是监控，自动化  
的基础

根据数据库特点，  
规划监控点。

知识与经验总结，  
告警操作指引，  
告警自愈脚本源

持续集成，自动  
部署



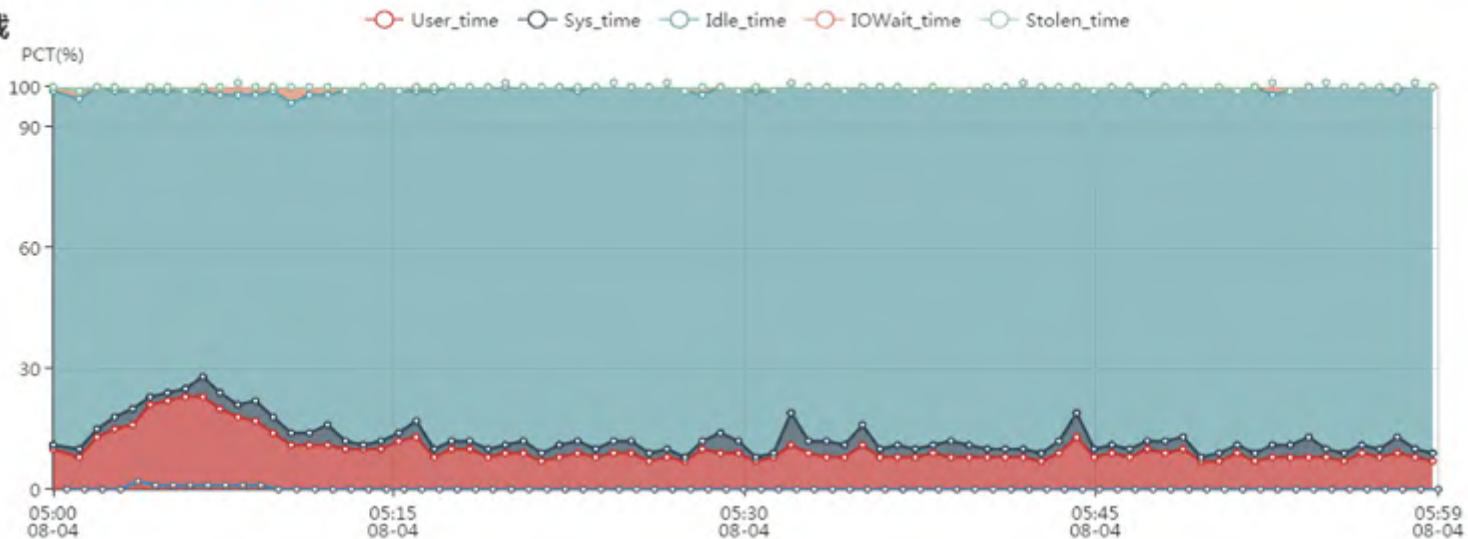


## 战术打击

——优化，PGAWR

### Operating System Statistics

#### CPU 负载



### Database Statistics

Database	Tps	Hit Rate	Logical IO/s	Physical IO/s	Rollback/s	Deadlocks	Total read time(s)	Total write time(s)	DB Size	Inc Size
users	0.54	97.00	16.65	0.38	0.00	0	103	0	1033 MB	208 kB
paebd	0.07	99.00	23247.27	29.42	0.00	0	20492	1	263 GB	21 MB

### Background writer stats

Checkpoints Timed	Checkpoints Req	Buffers Checkpoint	Buffers Clean	Maxwritten Clean	Buffers Backend	Buffers Alloc
12	0	76982	0	0	2720	4227

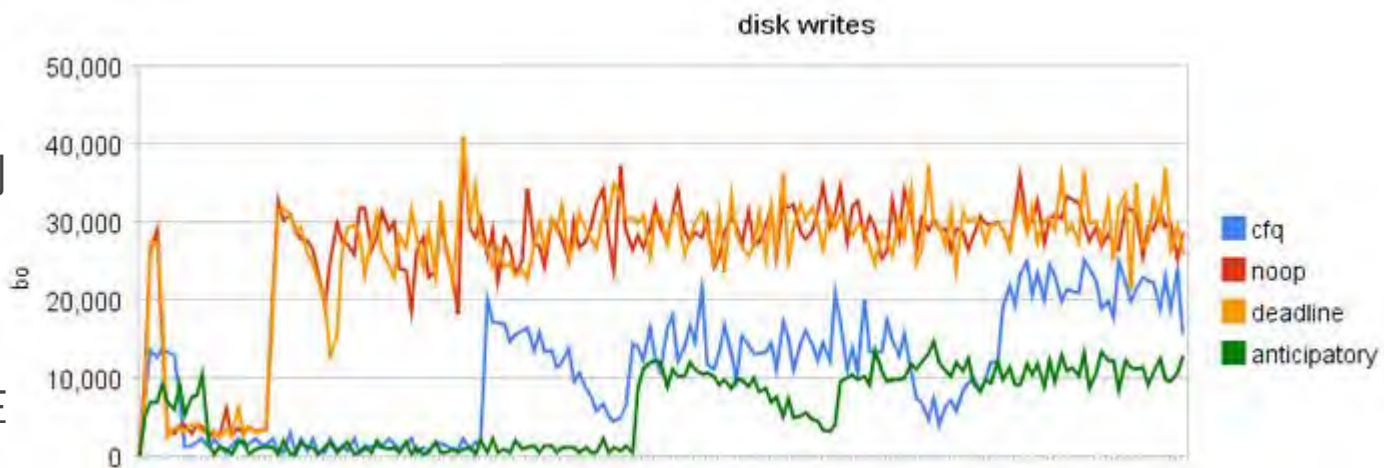
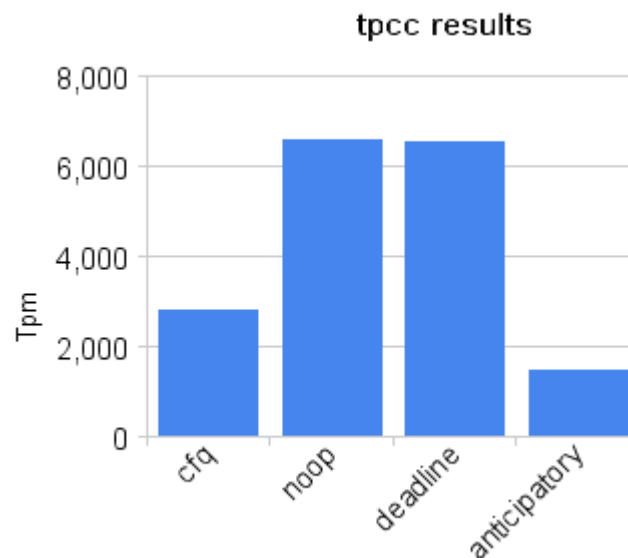




## 战术打击

——优化，IO Scheduler

- 1、闪存采用cfq调度算法的性能是最差的，需要调整调度算法。
- 2、在随机读和顺序读的情况下：  
noop/deadline无明显性能区别。cfq这种调度方式，相比noop/deadline情况下，iops下降严重，await和svctm时间增加。
- 3、在随机写的情况下：3类调度方式性能差别不是非常大，对比：deadline > noop > cfq
- 4、在顺序写的情况下：3类调度方式性能存在一定差别，对比：noop > deadline > cfq







## 战术打击

——Checkpoint优化

### 大幅减少归档WAL log

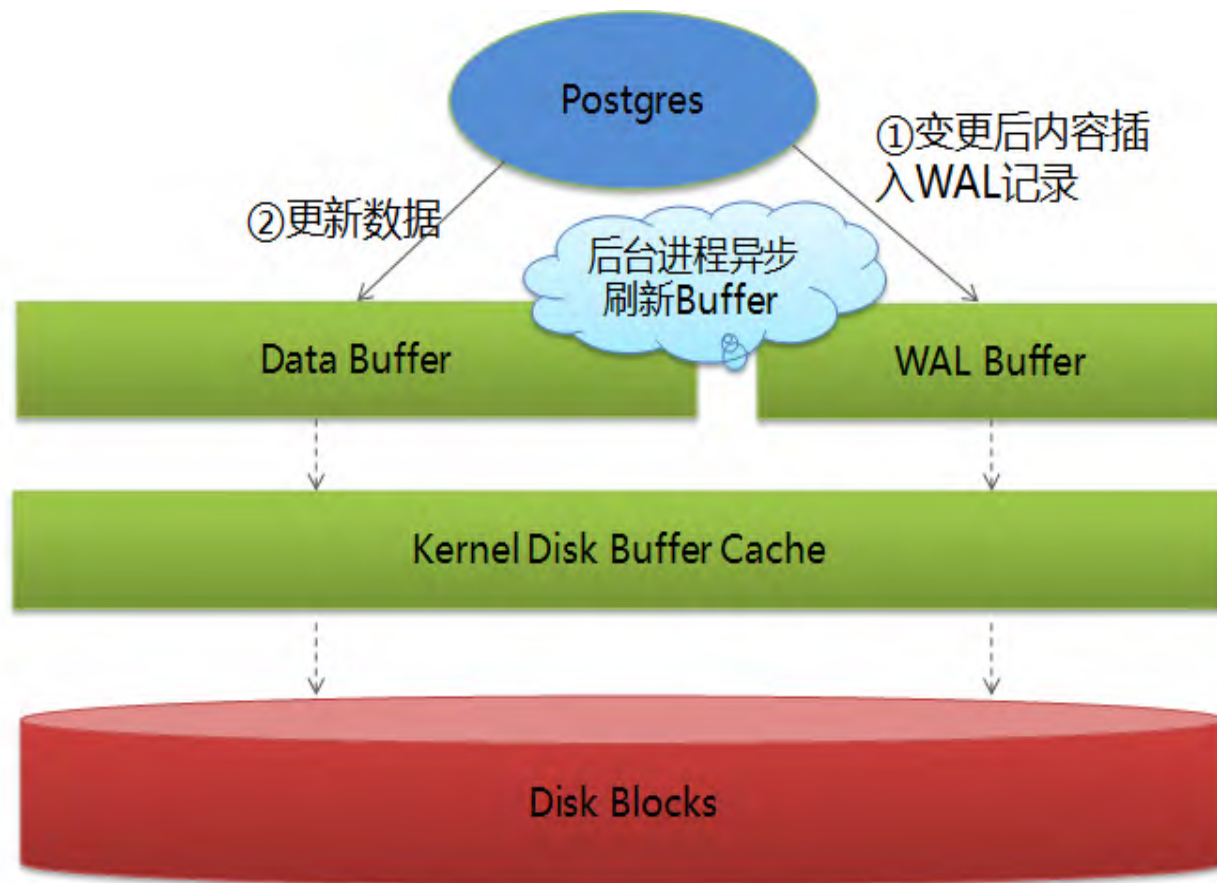
Checkpoint会触发full page write，过于频繁的Checkpoint会产生大量额外的WAL log

### 降低IO

频繁的Checkpoint会加大IO的压力

### 建议值

Checkpoint\_segments参数在128左右，能支持大部分的应用





## 战术打击

——优化, SQL/DB Audit



### 主动预防

标准化  
规范化  
智能化  
产品化

SQL AUDIT

### DB AUDIT

变更追踪  
违规审查  
统计优化



### 改进优化





## 战术打击

——优化，SQL/DB Audit规则

类型	分类	规则号	检查点	说明
SQL Audit	通用	SQLA-DML-G001	禁止使用select *	禁止使用select *，必须明确选择所需的列，每句SQL扣0.5分。
		SQLA-DML-G002	禁止出现重复查询子句	可以使用with as替换子句来提升SQL语句执行效率，每句SQL扣0.5分。
	更新	SQLA-DML-U001	禁止出现order by子句	防止DML过程出现不必要的排序，每句SQL扣0.5分。
		SQLA-DML-U002	必须出现where子句	防止DML过程出现全表锁，每句SQL扣1分。
	查询	SQLA-DML-S001	禁止嵌套select子句	防止出现select子句的嵌套子查询，避免出现性能问题，每句SQL扣0.5分。
	删除	SQLA-DML-D001	必须出现where子句	防止DML过程出现全表锁，每句SQL扣1分。
		SQLA-DML-D002	禁止出现order by子句	防止DML过程出现不必要的排序，每句SQL扣0.5分。
	PLAN	PG-PLAN-I001	全表扫描	对于大于200M的表全表扫描，执行计划里有Seq Scan操作，每次扣2分。
		PG-PLAN-S001	慢查询	执行时间超过1秒的SQL，每次扣1分。
DB Audit	TABLE	TAB_102	禁止unlogged	create table语句参数不能包含unlogged选项, unlogged会影响数据库的灾备和恢复。
		TAB_108	禁止增加字段同时设置默认值	加字段不能直接加default值，如： alter table t11 add yy varchar(2) default 'y';
	INDEX	IDX_102	必须非阻塞创建索引	CREATE INDEX需要加上CONCURRENTLY关键字：CREATE CONCURRENTLY INDEX
	VIEW	VIEW_104	禁止视图中使用order by	视图中禁止使用order by 子句，因为视图是在各个程序中共享使用，为避免不必要的性能消耗，建议在使用视图时再进行排序。

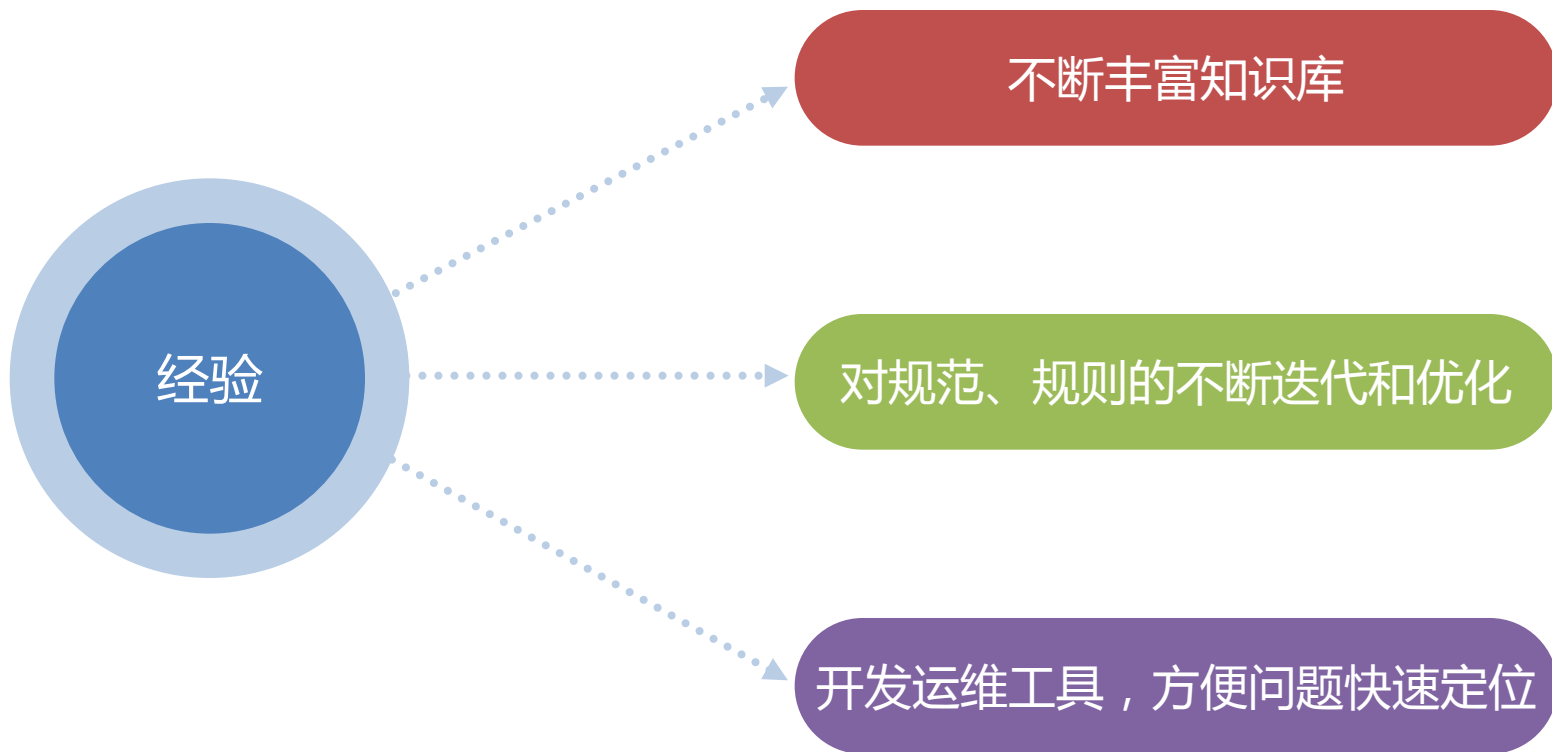


# 3 探索前进



## 探索前进

——积累运维经验





## 探索前进

——技术预研

并行计算

A

等待事件

B

Redis权限管理

C

Pg\_stat\_activity增加queryid

D



# Thanks!

## Q & A