

# Oracle运维中的那些事儿

杨建荣



## 个人介绍- 杨建荣



- Oracle ACE-A 
- YEP成员
- Oracle 10g OCP,OCM
- MySQL OCP
- 对shell , Java有一定的功底
- DBA+联合发起人
- 多年数据库开发和运维经验
- 曾在2015年数据库大会分享关于数据迁移和升级的主题
- 每天仍在孜孜不倦的进行技术分享，通过微信公众号，技术博客共享，已连续坚持620多天

# 演讲思路



# 环境篇

- **DBA**的角色及分工
- 传统行业到互联网行业的转换
- 最终目的：不是比谁更惨，而是能够吸取经验和教训

# DBA的角色及分工



2015

Oracle技术嘉年华

Oracle Technology Carnival 2015

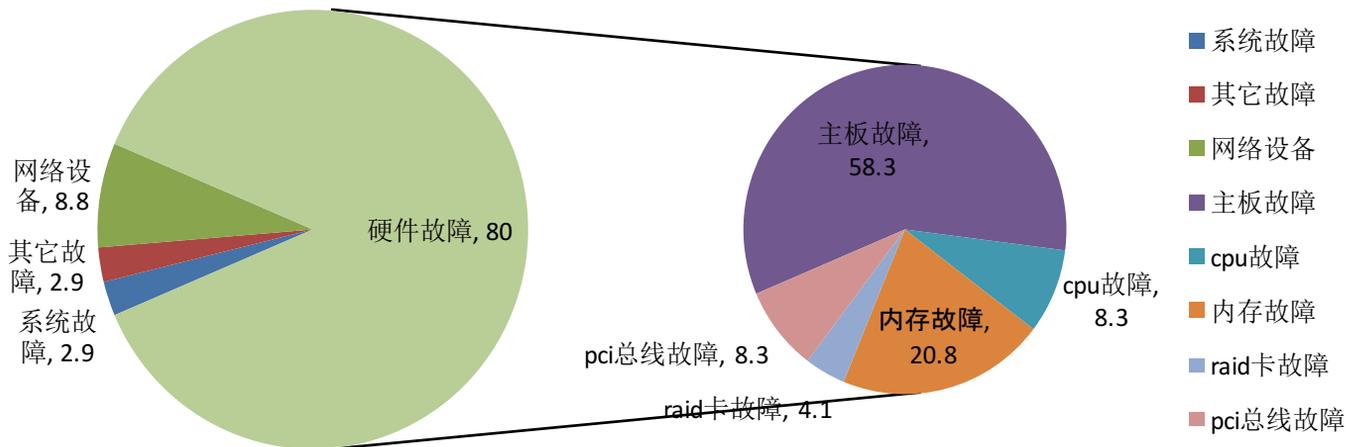
# 传统行业到互联网行业

- **BOSS,秒杀**
- 乙方到甲方
- 透明的硬件平台，硬件选型
- 电信客户的要求，稳定压倒一切
  - 客户UAT测试最长达一年
  - 严格的产品认证
- **11g的rac active/passive**
- **11g的adg 替换传统的BCV**
- **12c的PDB**
  - 保守的特性使用
    - 禁用直方图
    - 产品化极为彻底，很少有创建索引优化sql的场景
    - 禁用回收站
    - 禁用skip scan

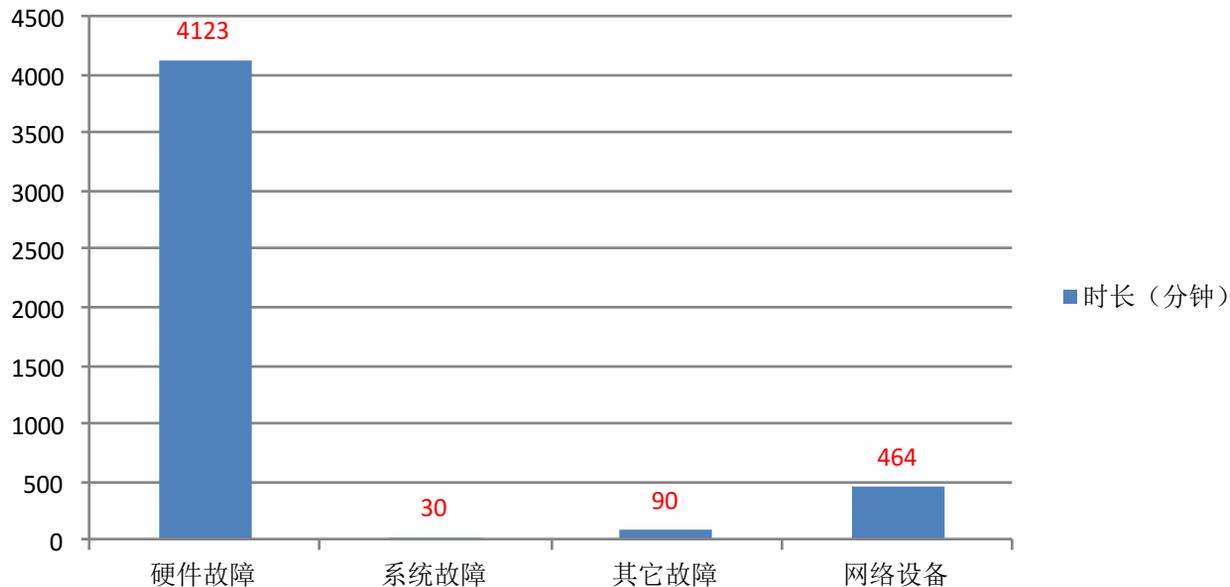
# 监控篇

- select 'Oracle is alive' from dual; 监控是否合理
- grid control ,em12c VS zabbix+orabbix
- 监控盲点
  - 11g中备库的v\$flash\_recovery\_area\_usage的自动归档删除策略
  - 11g备库在mount阶段
  - Asm使用空间的监控
- 硬件监控

## 各故障类型占比 (%)

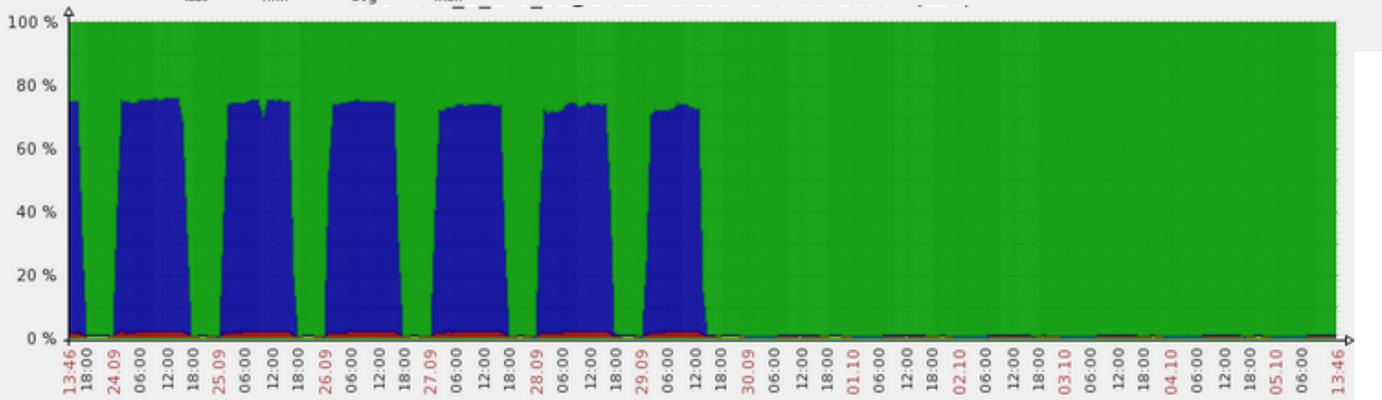
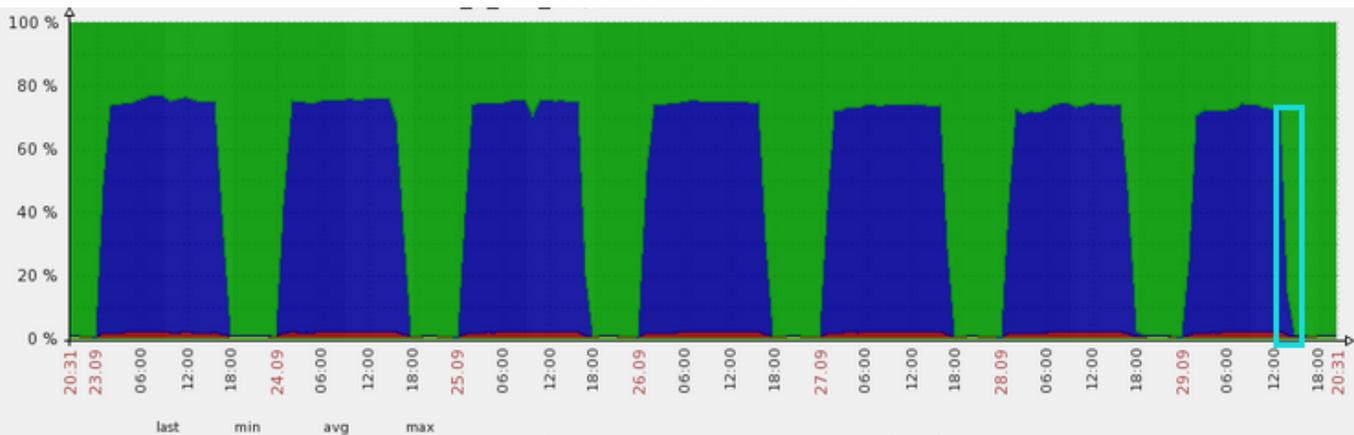


## 各故障类型影响累计时长（分钟）



# 硬件故障的反思

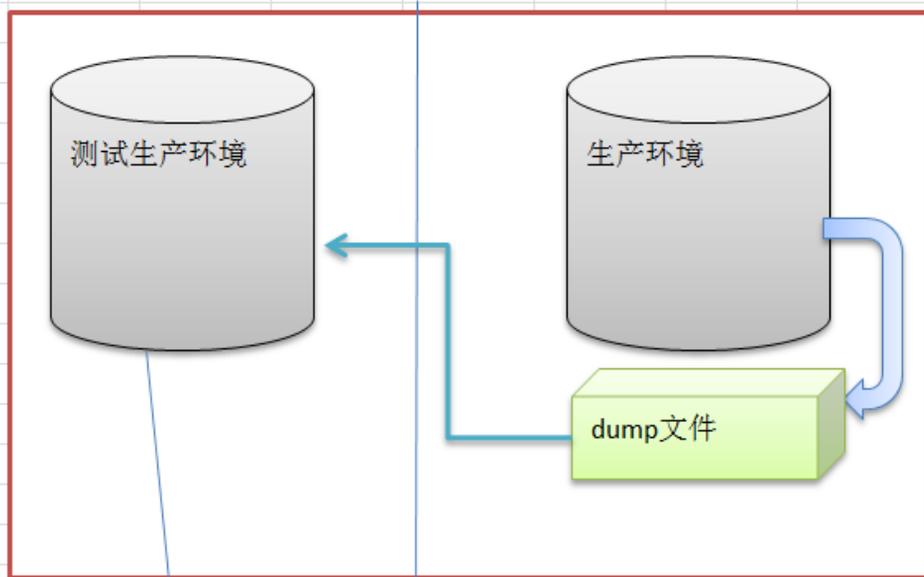
- 监控维度要全
- 监控粒度要细
- 不能等待问题降临，要直面迎击
  - 100多台服务器，硬件巡检12台有磁盘坏块
- 没有条件，不能坐以待毙，要敢于DIY
  - DB time抖动的分析
- 做历史问题，遗留问题的终结者



	last	min	avg	max
CPU idle time	98.79 %	20.22 %	74.59 %	99.59 %
CPU user time	0.48 %	0.18 %	24.34 %	76.29 %
CPU system time	0.28 %	0.11 %	0.77 %	4.36 %
CPU iowait time	0.35 %	0.0021 %	0.24 %	6.31 %
CPU nice time	0 %	0 %	0.000014 %	0.04 %

# 运维管理

- 日常维护
  - 使用空格的技巧
  - 空格的惨案: `ifconfig -a6` vs `ifconfig -a 6`
  - 简单至极的维护案例, `system`表空间满
- 不可忽略的细节
  - 外部表 `exp`
  - `silent`模式下的`responsefile, responseFile`
- 合理利用新特性
  - `sql monitor`一条运行14天的`sql`语句
- 少踩一些坑
  - `drop datafile`导致的备库bug
- 灵活变通, 省事省力
  - 5T数据和1M数据的灵活切换



100,'bbbb'  
200,'aaaa'  
300,'cccc'

100,'aaaa'



# 优化篇

- 不定时的性能炸弹sqlprofile
- 不重启的调优 硬件不到位 遗留问题 不重视 预算不足
- 临时提交的变更4个小时的pl/sql和1分钟的sql
- sqlldr加载性能问题的分析
- 容易被忽略的备库调优
  
- 案例
- 数据库无法登录的案例

# 优化篇案例

- \$ sqlplus / as sysdba  
ORA-09817: Write to audit file failed.  
**Linux-x86\_64 Error: 28: No space left on device**
- audit 占用空间很小,sysdba 登录会自动写 audit

MON	DA	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22
08	27	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	1	0	0	1	0	0	0
08	28	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0	1	0	1
08	29	1	0	0	0	1	0	0	0	1	0	0	1	0	0	1	0	1	0	1	0	0	1	0
08	30	1	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	0
08	31	0	1	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0	1	1	0	0	1	0
09	01	0	1	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	2	5	7	12	14	17
09	02	23	27	29	32	35	38	40	44	47	50	52	56	60	61	62	62	70	73	74	78	81	82	87
09	03	95	94	98	102	106	107	108	109	116	118	120	74	19	0	0	0	0	0	0	0	0	0	0

## 优化案例

Elapsed Time (s)	Executions	Elapsed Time per Exec (s)	%Total	%CPU	%IO	SQL Id	SQL Mod
931.73	14,409	0.06	29.39	99.77	0.00	<a href="#">c95g9rc1hw48j</a>	JDBC Thin
233.07	1,018	0.23	7.35	99.97	0.00	<a href="#">aswj2snnfgv26</a>	JDBC Thin
222.75	187	1.19	7.03	99.99	0.00	<a href="#">c6p9qpbxcmfys</a>	JDBC Thin

<a href="#">c95g9rc1hw48j</a>	update cs_mongo_sync_id set max_id = :1 where sync_id_type = :2
-------------------------------	---

Executions	Rows Processed	Rows per Exec	Elapsed Time (s)	%CPU	%IO	SQL Id	SQL Mod
14,684	14,684	1.00	3.39	94.7	.7	<a href="#">98kzt71wqz5gg</a>	JDBC Thin
14,409	78,329,332	5,436.14	931.73	99.8	0	<a href="#">c95g9rc1hw48j</a>	JDBC Thin

# 优化案例

- **对比测试:**
  - 运行3分钟，日志切换14次（存在冗余数据）
  - 运行7分钟，日志切换0次（删除冗余数据）

# 开发,业务篇

- 定制awr,ash,addm
- 熟练掌握开发技能
  - 数据库连接满的原因分析
- 熟练掌握业务
  - 通过业务优化sql
  - 分区规则，业务和技术的脱钩

# 业务和技术的脱钩案例

• SEGMENT_NAME	PARTITION_NAME	SEGEMENT_BYTES
• CHARGE	P120_C100	1
CHARGE	P120_C10	438
CHARGE	P120_C20	1
CHARGE	P120_C30	1
CHARGE	P120_C40	1

- 分区规则

- **partition by range(period\_key,customer\_key)**

- partition P41\_C100 values less than (41,100)

- partition p120\_c10 values less than (120,10)

- **PARTITION\_NAME HIGH\_VALUE TS\_NAME**

• P41_C90	41, 90	DATAH01
• P41_C100	41, 100	DATAH01
• P120_C10	120, 10	DATAH01
• P120_C20	120, 20	DATAH01
• P120_C30	120, 30	DATAH01

# 业务和技术的脱钩案例

- SQL> select \*from test partition(p120\_c10);
- PERIOD\_KEY CUSTOMER\_KEY
- -----
- 57 99
- 57 150
- 120 5
- 119 50
- 120 5

# 展望

## – 对DBA更高的要求，更高的挑战

- 开发技能
- 数据架构
- 一专多能
- 新技术前沿
- 全栈DBA

## – 展望

- 自动化运维
- 精细化运维

