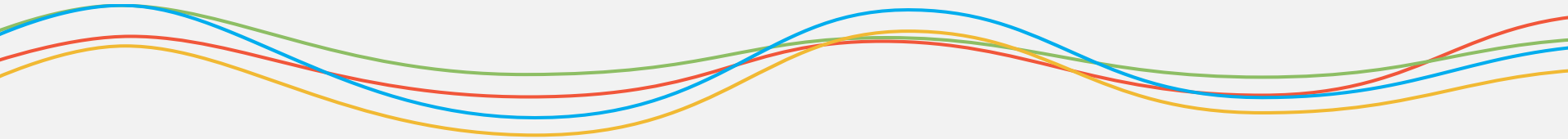
APMCon

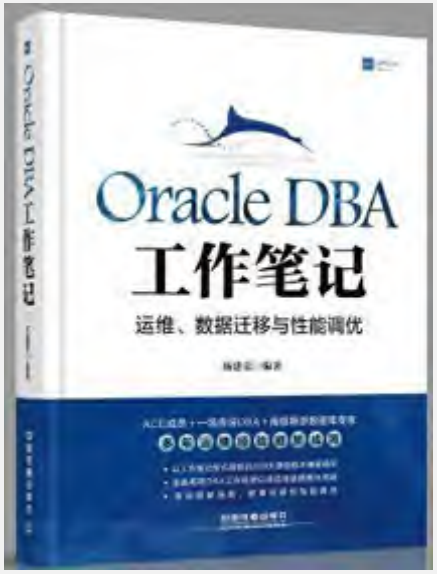
深入解析和定制Oracle优化工具

杨建荣



个人介绍- 杨建荣

- Oracle ACE
- YEP成员
- DBA+联合发起人
- Oracle 10g OCP,OCM ，MySQL OCP
- 对shell , Java有一定的功底
- 《Oracle DBA 工作笔记》作者
- 曾在中国数据库大会,Oracle嘉年华,全球敏捷运维峰会(Gdevops) 演讲
- 每天仍在孜孜不倦的进行技术分享，技术博客共享,已连续坚持800多天



我眼中的一些问题

APMCon

当前问题：实时抓取，分析

- 精细化运维

潜在问题：提前发现，预警

- 半自动化运维，自动化运维

历史问题，遗留问题：数据分析

- 数据分析可视化，智能分析

Oracle优化工具

APMCon

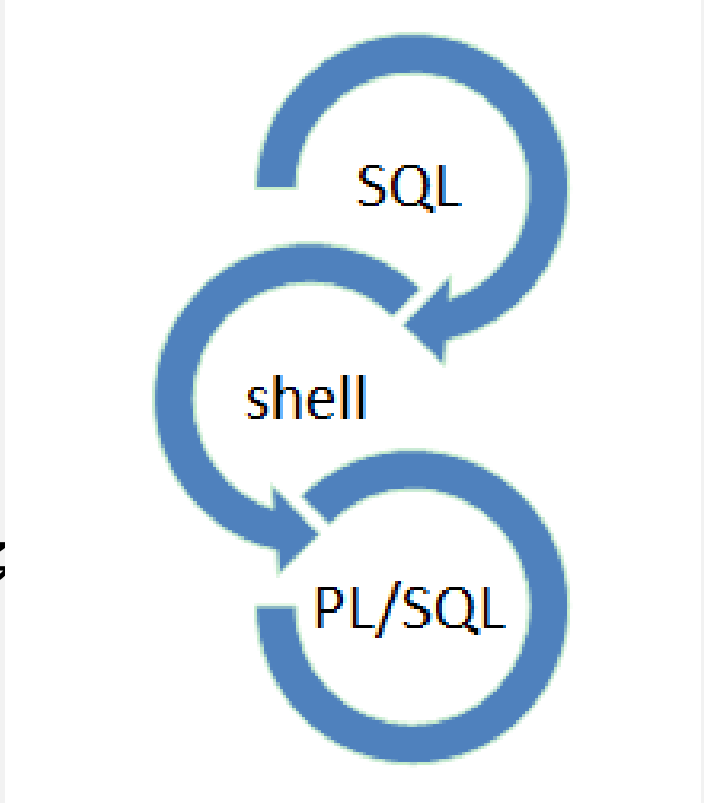
- 工欲善其事必先利其器

- 基于快照
  - ✓ AWR
- 基于会话
  - ✓ ASH
- 基于成本分析
  - ✓ ADDM
- 基于SQL
  - ✓ SQL Tuning
  - ✓ SQL Monitor
  - ✓ SQL Profile

简单定制的思路

APMCon

- 先理解工具的工作机制和原理
- 工具的哪些功能需要改进
- 定制思路是否可以复用
- 带来的问题更多还是解决的问题更多
- 尽可能简化，简化配置，简化操作



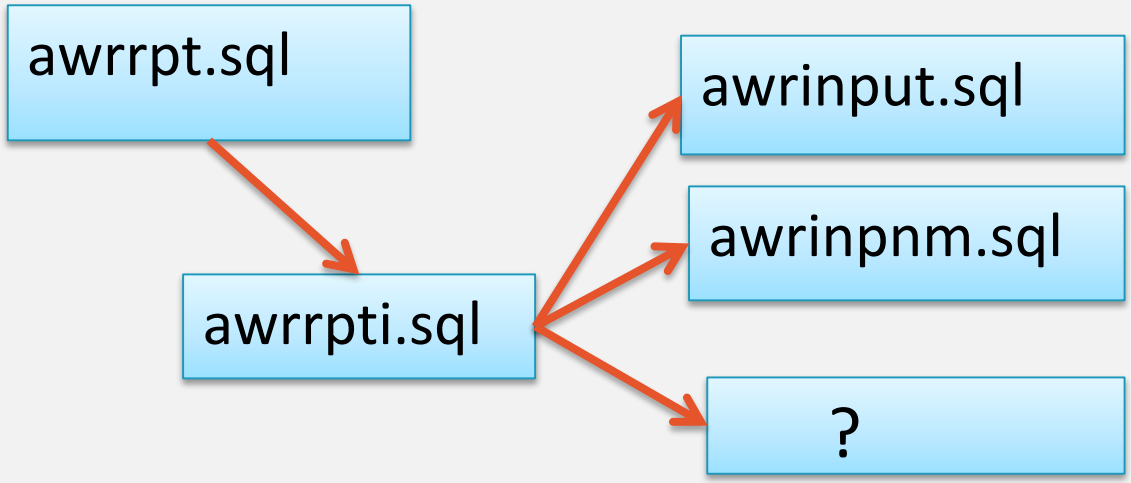
生成一个AWR报告的步骤

APMCon

需求：查看上周某一天早上8:00~9:00之间的一个性能问题

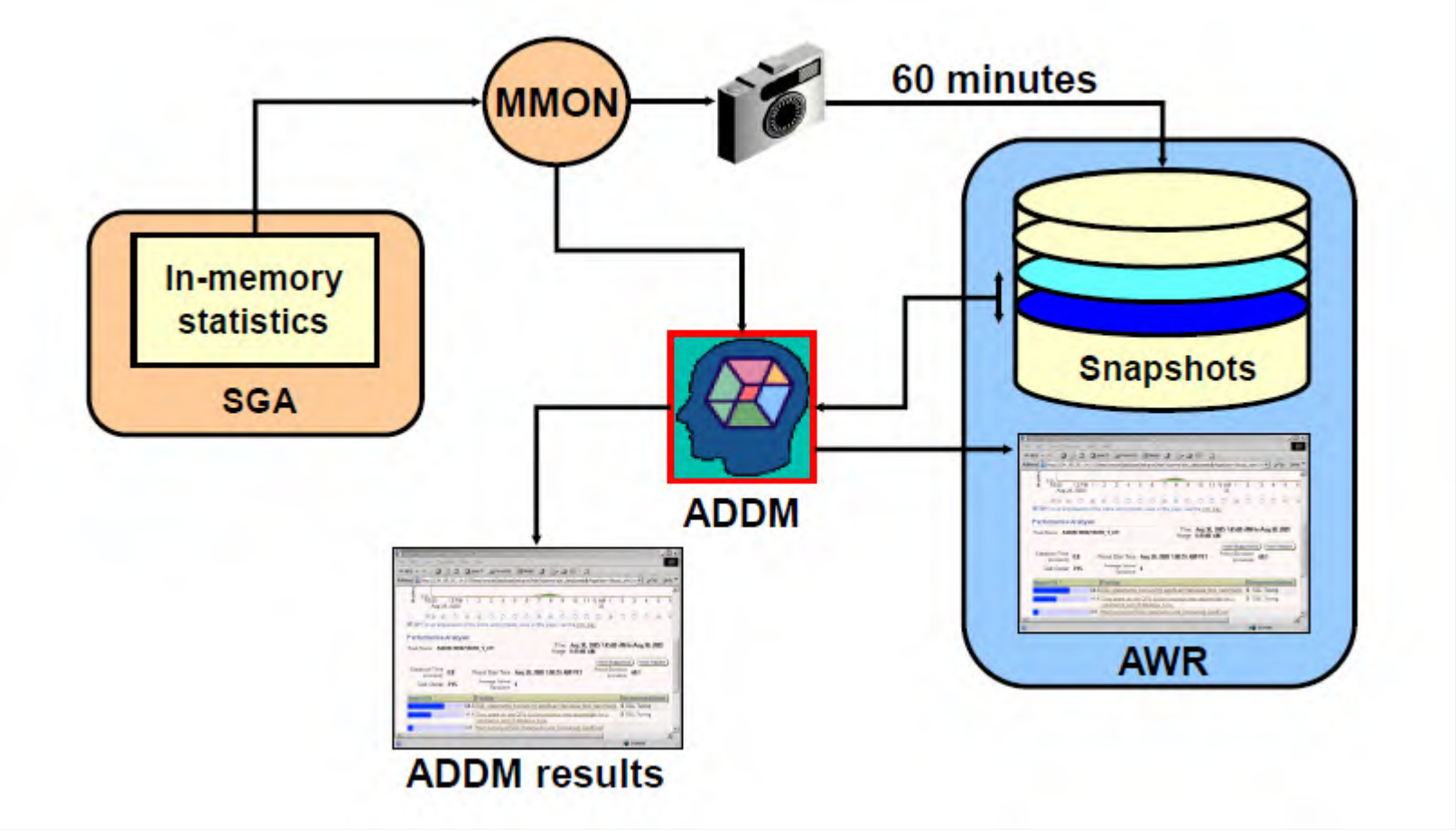
1.查看上周某一个早上8：00～9：00的快照
2. 掐指一算上周那一天到今天的天数
3.然后上下翻屏找快照号
4.找到开始快照号，再找结束快照号
5.定义一个报告的名字

AWR内置的脚本调用关系




```
select output from
table(dbms_workload_repository.&fn_name( :dbid,:inst_num
,:bid, :eid,:rpt_options ));
```

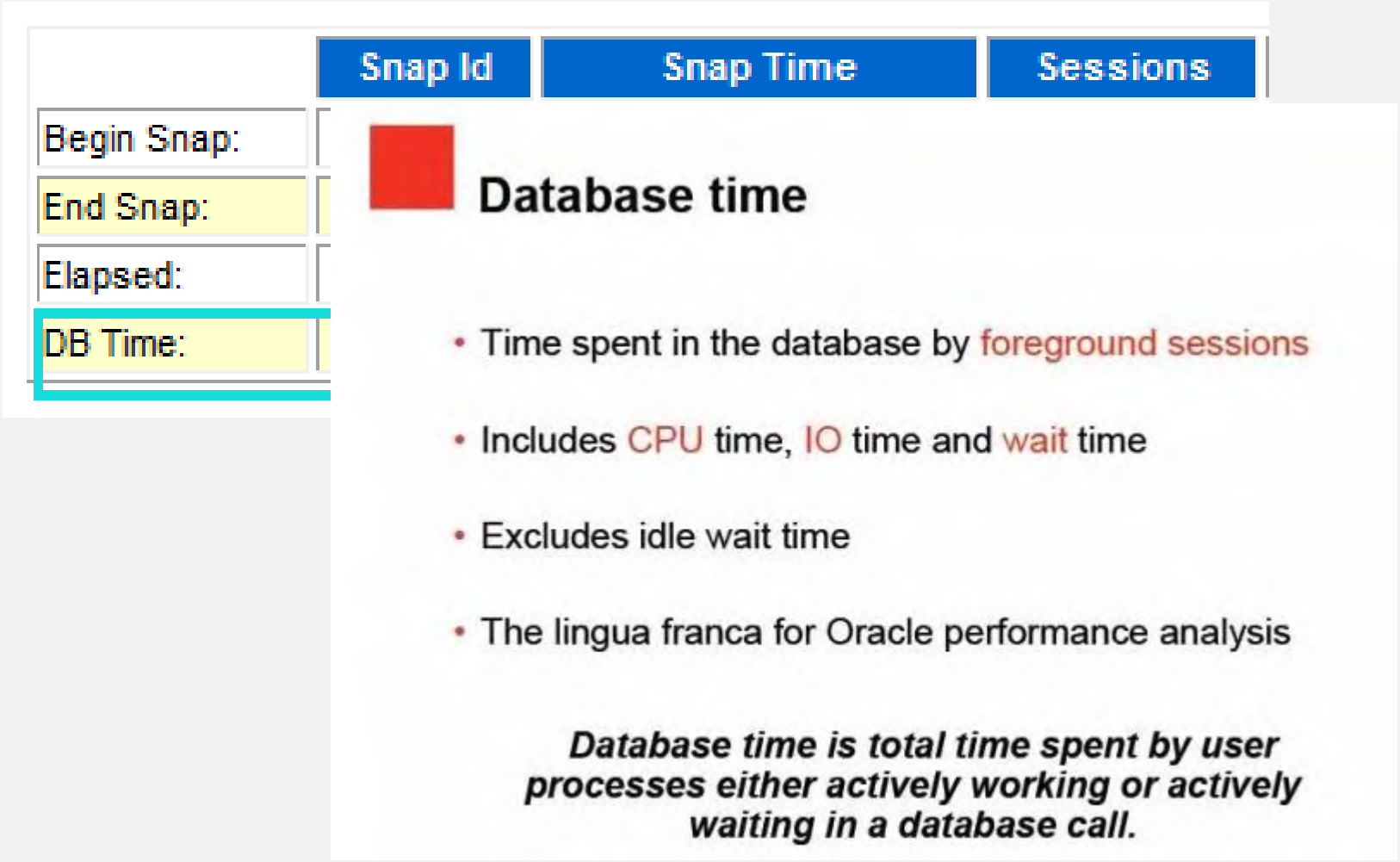
AWR工作原理


MMON
60 minutes
In-memory statistics
SGA
Snapshots
ADDM
AWR
ADDM results

定制AWR的思路

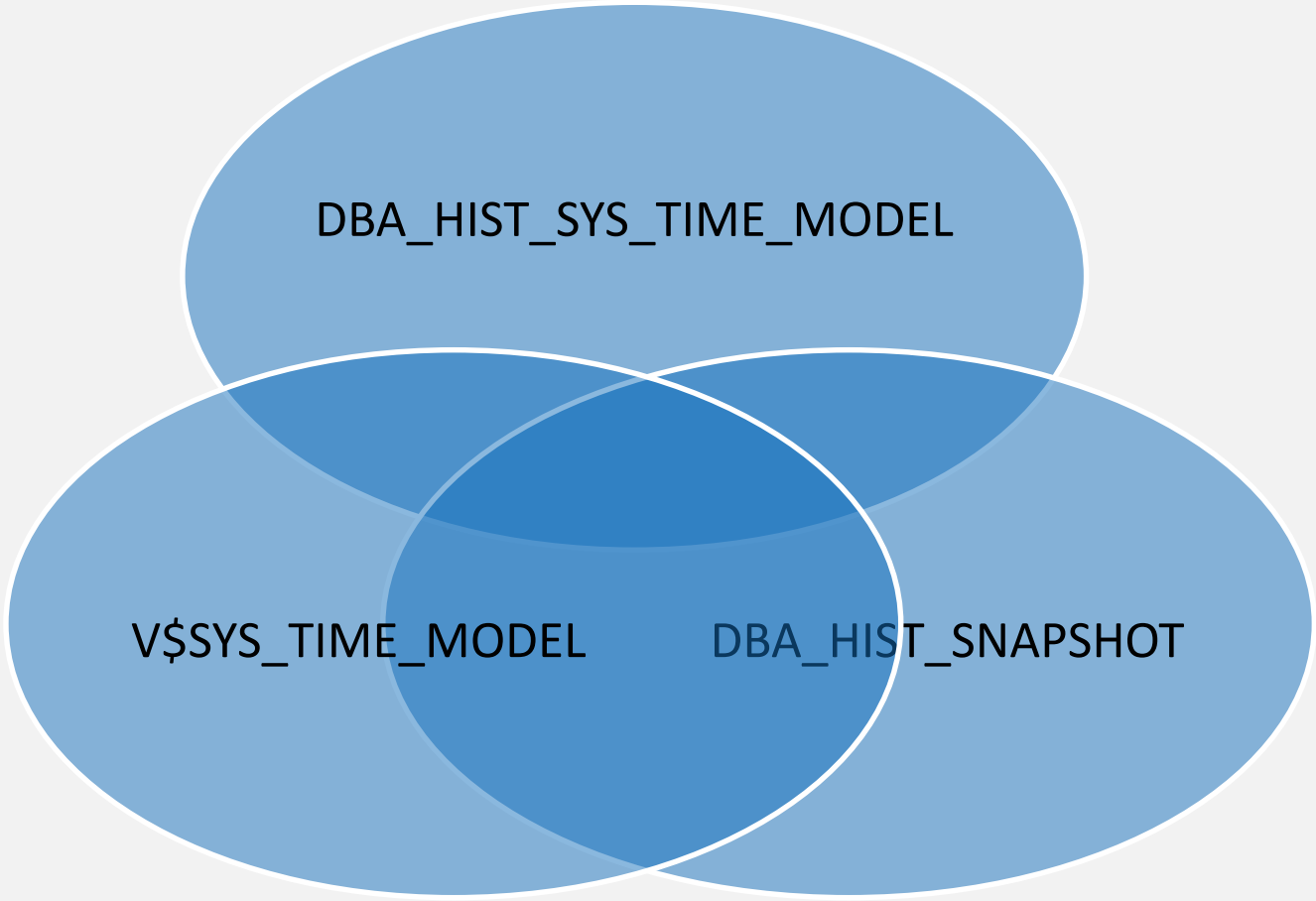
- 先定制快照，定制awr就会顺水推舟
- 小步快走，立等可取
- 快照还有哪些改进之处

重要的监控项:DB time



定制快照的数据基础

APMCon



得到DB time的快照列表

APMCon

BEGIN_SNAP	END_SNAP	SNAPDATE	DURATION_MINS	DBTIME
36343	36344	27 Aug 2015 04:00	60	85
36344	36345	27 Aug 2015 05:00	60	118
36345	36346	27 Aug 2015 06:00	60	150
36346	36347	27 Aug 2015 07:00	60	180
36347	36348	27 Aug 2015 08:00	60	137
36348	36349	27 Aug 2015 09:00	60	140
36349	36350	27 Aug 2015 10:00	60	67
36350	36351	27 Aug 2015 11:00	60	8

定制AWR更多的福利

APMCon

awrsqrpt.sql

dbms_workload_repository.awr_sql_report_html

sh genawrsqltext.sh 12315 12316 <SQL_ID>

强大的AWR Format

APMCon

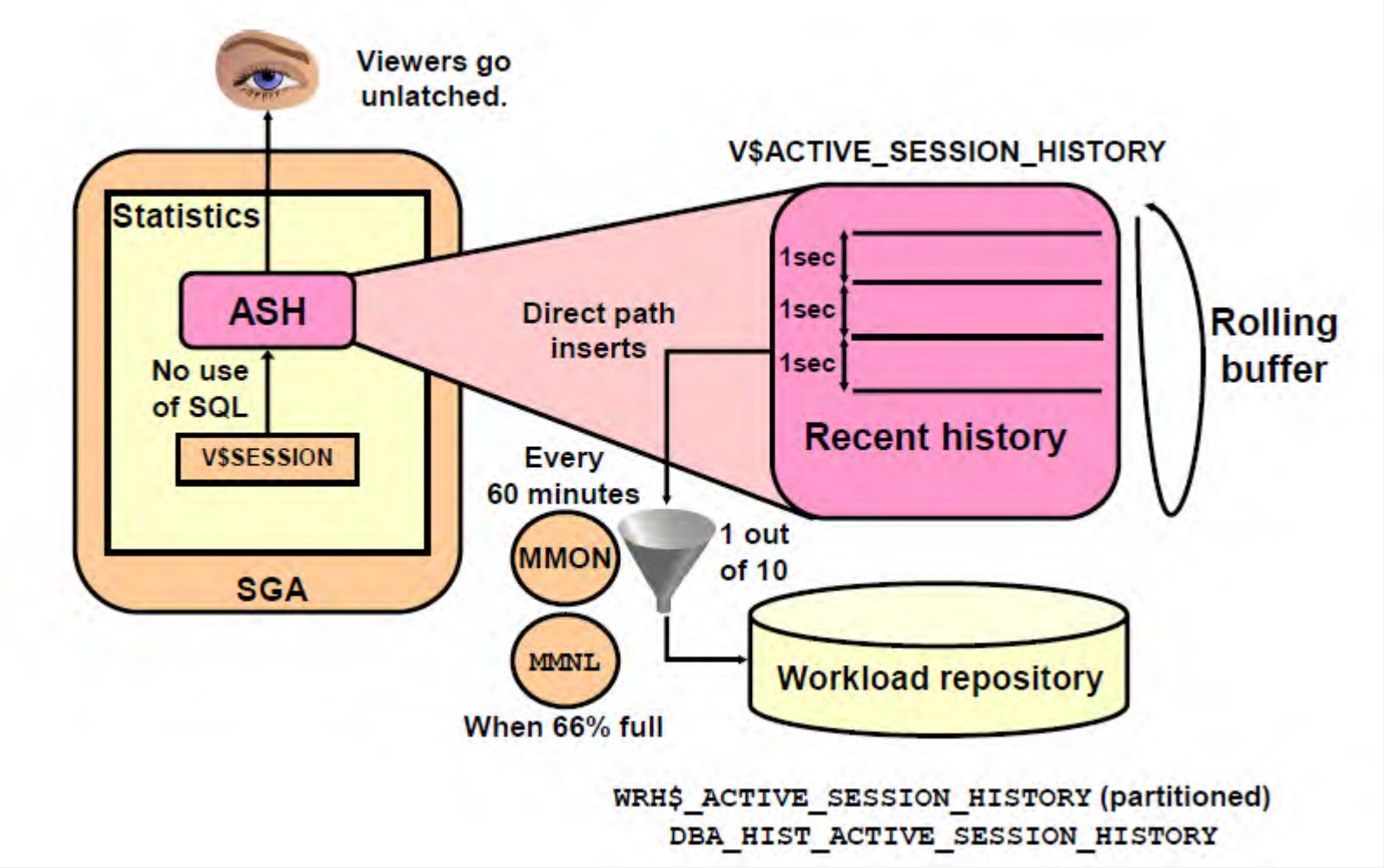
Overview | Top SQL Combined | Top Events | Exadata Stats | Memory | I/O Graphs | Observations (14) | About

hide zero stats | show all

Convert to CSV

Statistic	Total	per Second	per Trans
cell physical IO interconnect *bytes*	922.2 GB	1.5 GB	43.8 MB
physical read IO requests	8,874,418	14,770.26	411.35
physical read *bytes*	896.1 GB	1.5 GB	42.5 MB
physical read total IO requests	7,670,502	12,766.51	355.54
physical read total *bytes*	906.6 GB	1.5 GB	43.0 MB
physical read total multi block requests	931,184	1,549.83	43.16
physical *reads*	896.1 GB	1.5 GB	42.5 MB
physical *reads* cache	59.6 GB	101.5 MB	2.8 MB
physical *reads* cache prefetch	29.8 GB	50.8 MB	1.4 MB
physical *reads* direct	836.5 GB	1.4 GB	39.7 MB
physical *reads* direct (lob)	185.8 MB	316.6 KB	8.8 KB
physical *reads* direct temporary tablespace	10.9 GB	18.5 MB	528.3 KB
physical *reads* prefetch warmup	0 B	0 B	0 B
physical write IO requests	239,746	399.02	11.11
physical write *bytes*	13.6 GB	23.1 MB	659.2 KB
physical write total IO requests	288,566	480.28	13.38
physical write total *bytes*	15.6 GB	26.5 MB	756.5 KB
physical write total multi block requests	26,021	43.31	1.21
physical writes	1,777,673	2,958.70	82.40
physical writes direct	1,679,971	2,796.08	77.87
physical writes direct (lob)	26,882	44.74	1.25
physical writes direct temporary tablespace	1,615,477	2,688.74	74.88
physical writes from cache	97,702	162.61	4.53
physical writes non checkpoint	1,772,473	2,950.04	82.16

ASH的工作原理


Viewers go unlatched.
V$ACTIVE_SESSION_HISTORY
Statistics
ASH
No use of SQL
V$SESSION
SGA
Direct path inserts
1sec
1sec
1sec
Recent history
Rolling buffer
Every 60 minutes
MMON
1 out of 10
MMNL
When 66% full
Workload repository
WRH$_ACTIVE_SESSION_HISTORY (partitioned)
DBA_HIST_ACTIVE_SESSION_HISTORY

定制ASH

APMCon

- AWR的一个不是缺点的缺点：后知后觉
- ASH的定制如法炮制
- dbms_workload_repository.ash_report_html
- sh genashhtml.sh 20151208010000 20151208020000

定制ADDM

APMCon

- dbms_advisor.create_task 生成一个任务
- dbms_advisor.set_task_parameter设置参数属性
- dbms_advisor.execute_task来执行任务
- dbms_advisor.get_task_report(‘$TASK_NAME’ , ‘TEXT’ , ‘TYPICAL’) 输出最终的报告
- 基于快照，基于shell一键定制

定制SQL Monitor

APMCon

SQL Plan Monitoring Details (Plan Hash Value=3078438663)

Id	Operation	Name	Estimated Rows	Cost	Active Period (8137s)	Execs	Rows	Memory
0	UPDATE STATEMENT					1		
1	UPDATE	TEST_CON_NEW				1	0	
-> 2	FILTER					1	221	
-> 3	FILTER					1	2730	
-> 4	TABLE ACCESS BY INDEX ROWID	TEST_CON_NEW	215	17		1	2730	
-> 5	INDEX SKIP SCAN	IND_VIP_NEW_MIX	215	14		1	2731	
-> 6	SORT AGGREGATE		1			2731	2730	
7	FILTER					2731	8292	
8	TABLE ACCESS BY INDEX ROWID	VIP_RECHARGE_LOG	1	2		2731	8292	
-> 9	INDEX RANGE SCAN	IDX_VIP_CHARGE_DATE	2	1		2731	4G	
10	SORT AGGREGATE		1			221	221	
11	FILTER					221	3863	
12	TABLE ACCESS BY INDEX ROWID	VIP_RECHARGE_LOG	1	2		221	3863	
13	INDEX RANGE SCAN	IDX_VIP_CHARGE_DATE	2	1		221	2G	

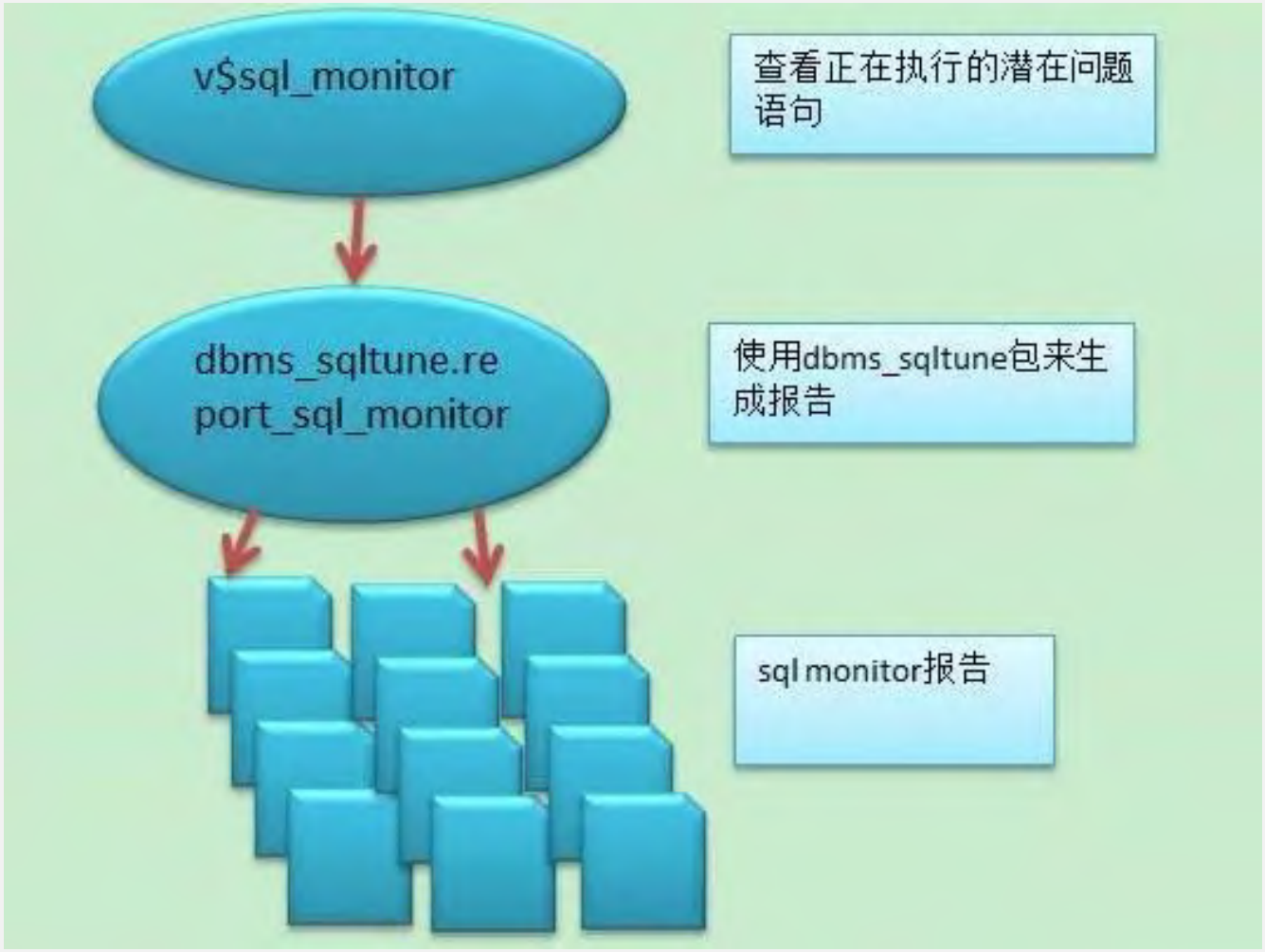
```
SELECT dbms_sqltune.report_sql_monitor(
sql_id => '$1',
report_level => 'ALL',
type=>  'HTML'
) comm  from dual;
```

SQL Monitor的改进点

APMCon

- OSM中的v$sql_monitor类似v$session的机制，数据刷新
- 自动生成sql monitor的html或text 格式报告
- 报告生成频率的控制
- 对报告进行筛查，得到报告中的top SQL
- 对历史报告进行深度分析

SQL Monitor和数据分析结合起来



定制SQL Profile

APMCon

优点：

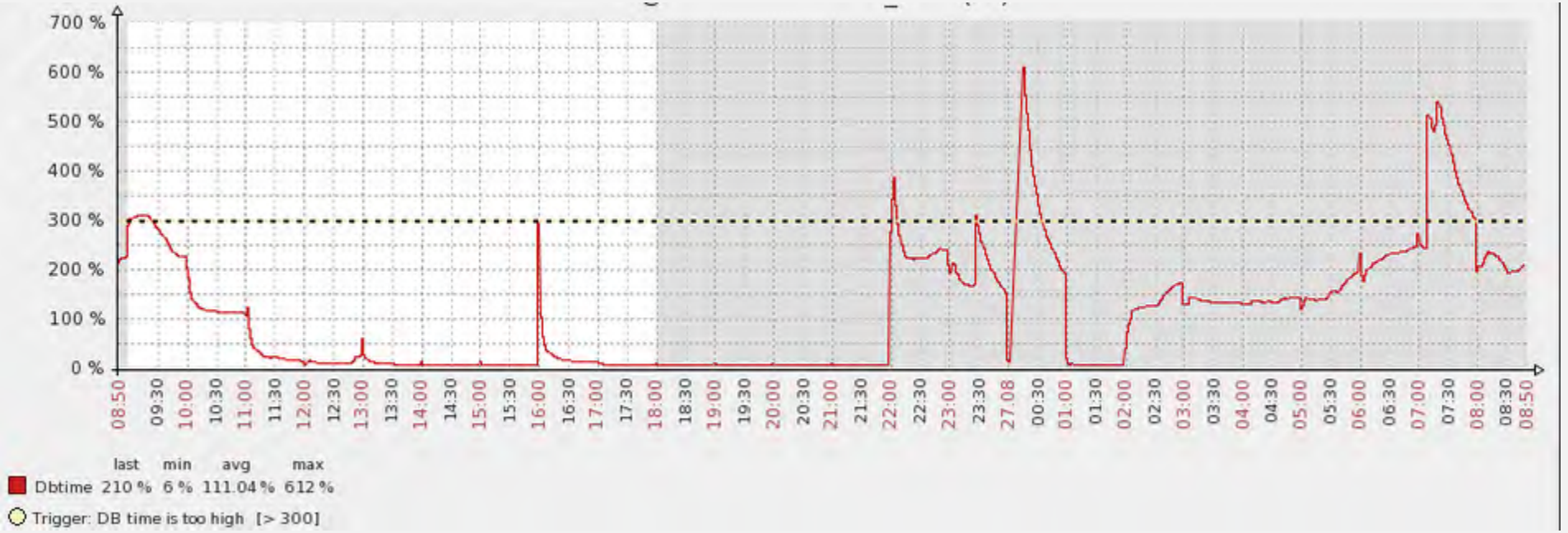
- 在线修改，替换执行计划，无需修改SQL
- Oracle内部(ST CoE)也定制了SQLT

缺点：

- 性能下降的不定时炸弹
- 规划化，标准化，遗留问题

性能优化案例1

APMCon

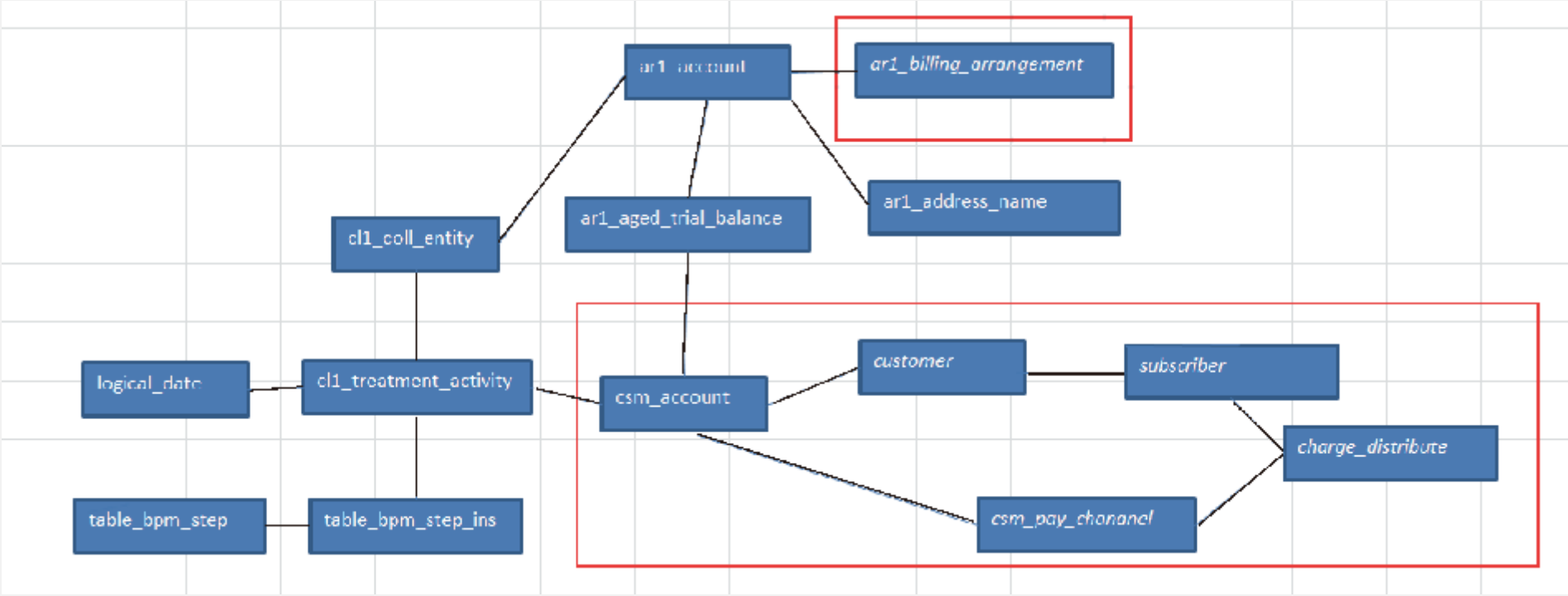


性能优化结果

APMCon

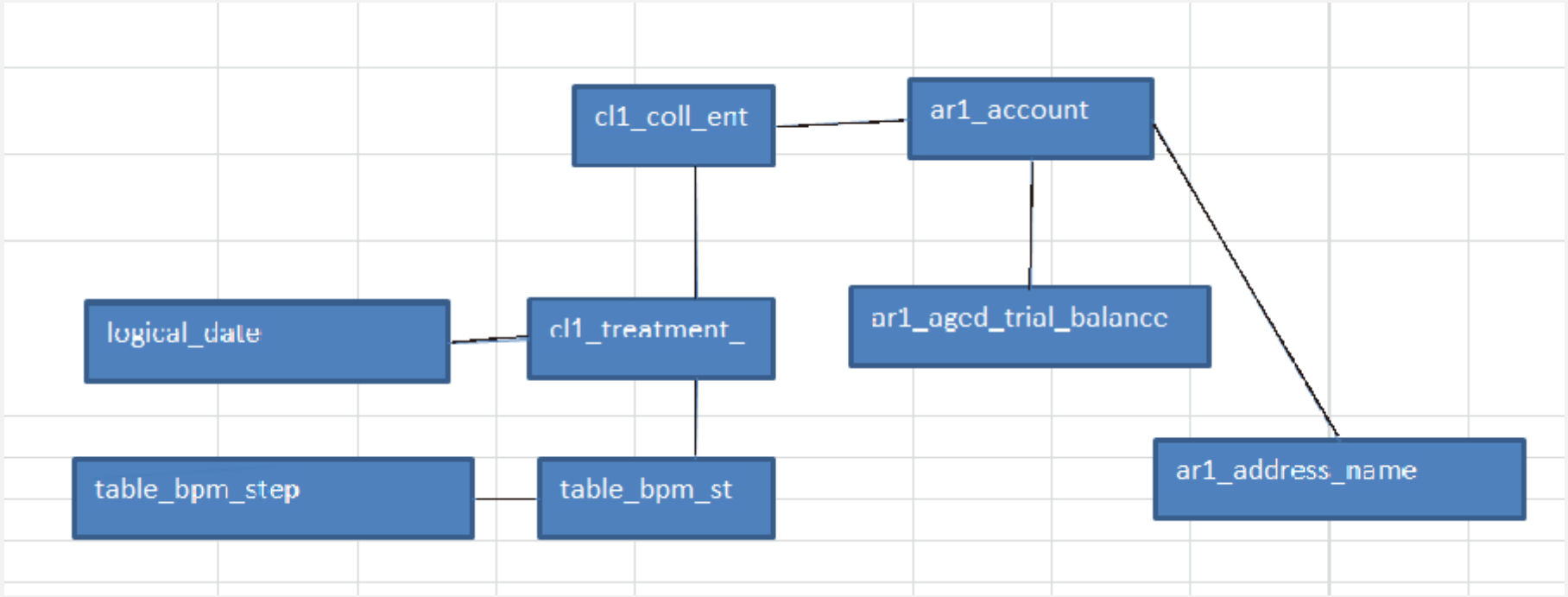
性能优化案例2

APMCon



性能优化结果

APMCon



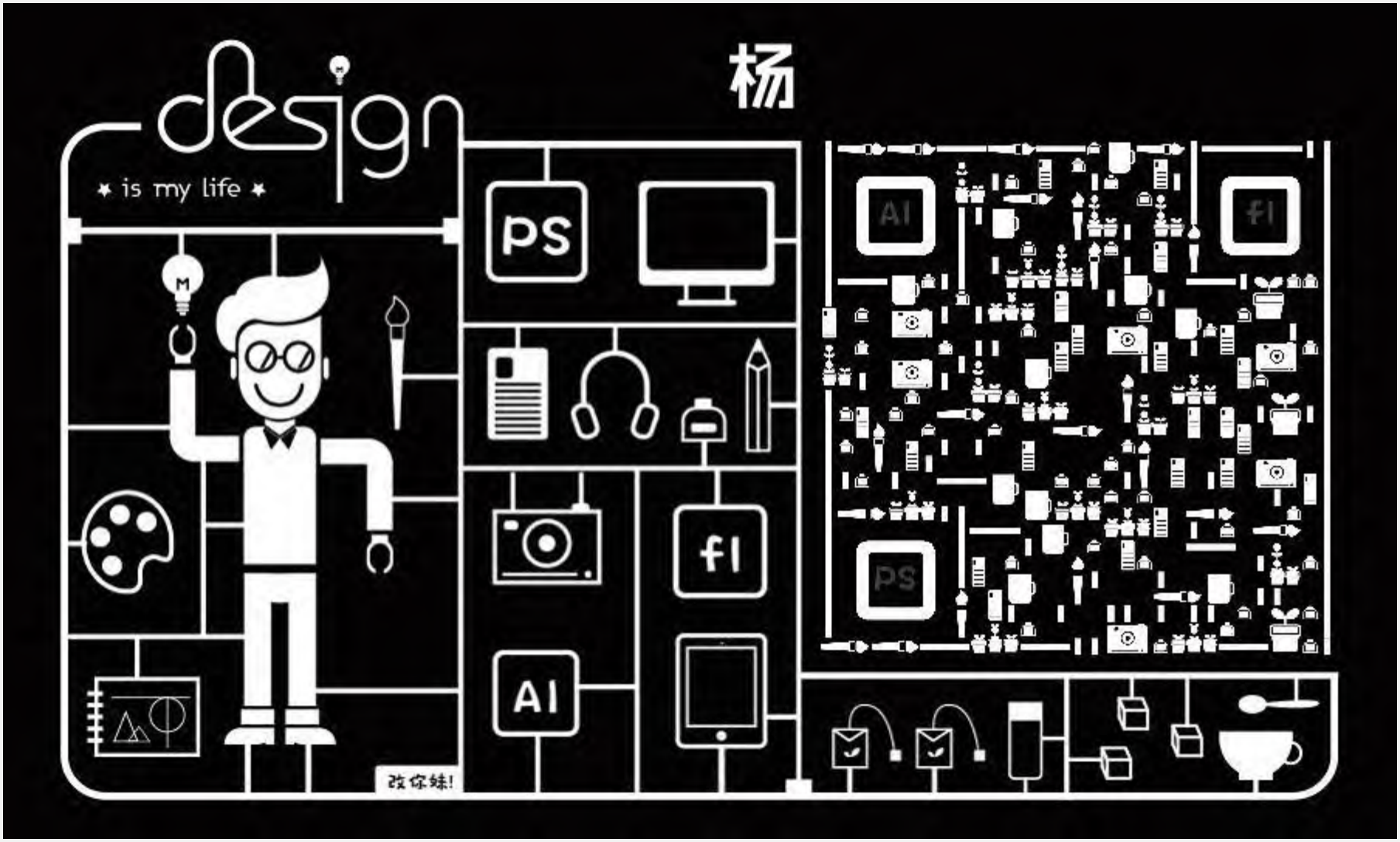
更多的定制和方向

APMCon

- 精细化运维
- 自动化运维
- 数据分析和优化结合
- 有时候需要半自动化,基于安全

个人微信公众号，欢迎关注

APMCon



THANK YOU