



DTCC

2017第八届中国数据库技术大会

DATABASE TECHNOLOGY CONFERENCE CHINA 2017

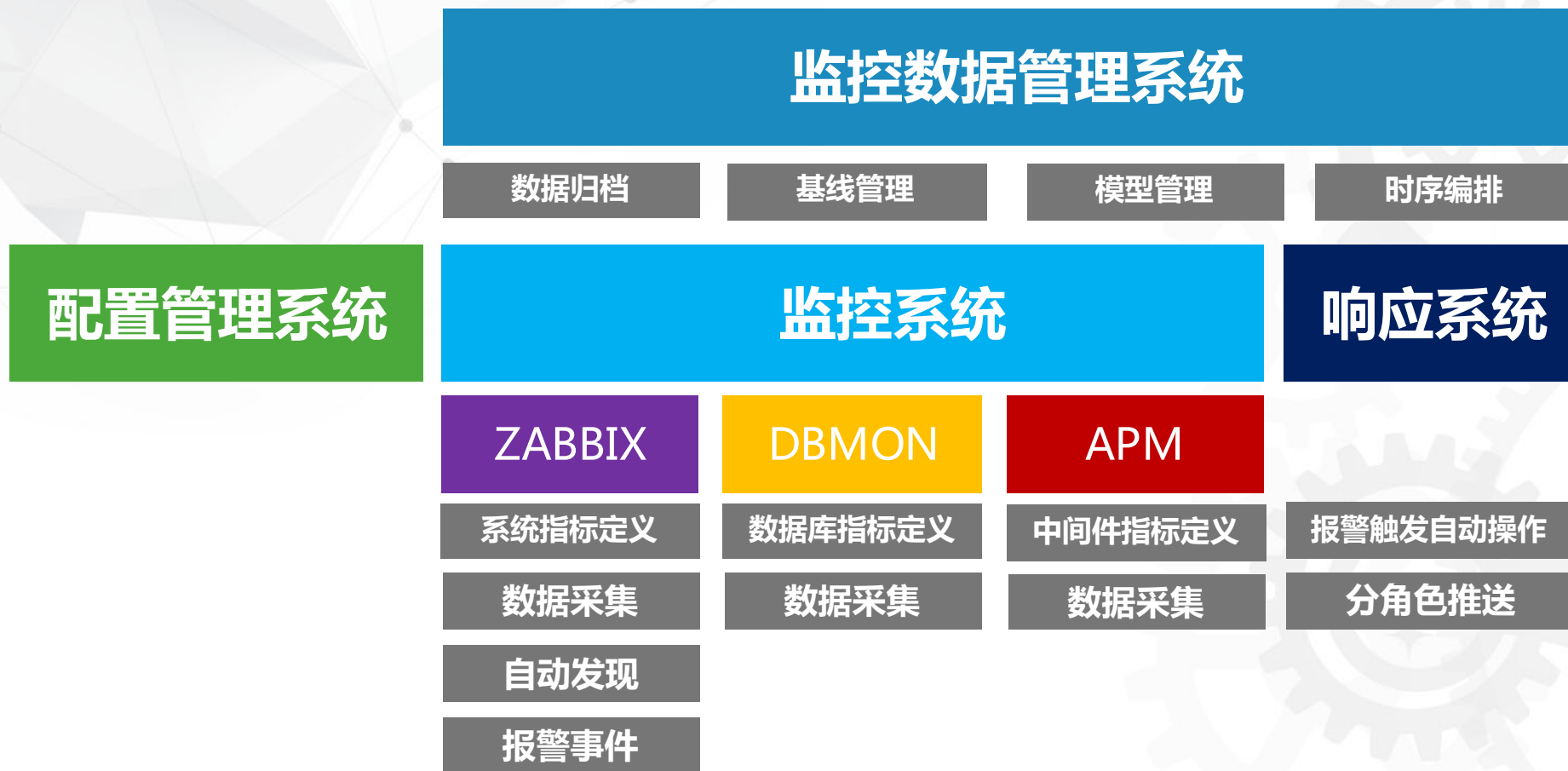
测试环境性能监控与实践

李雁南

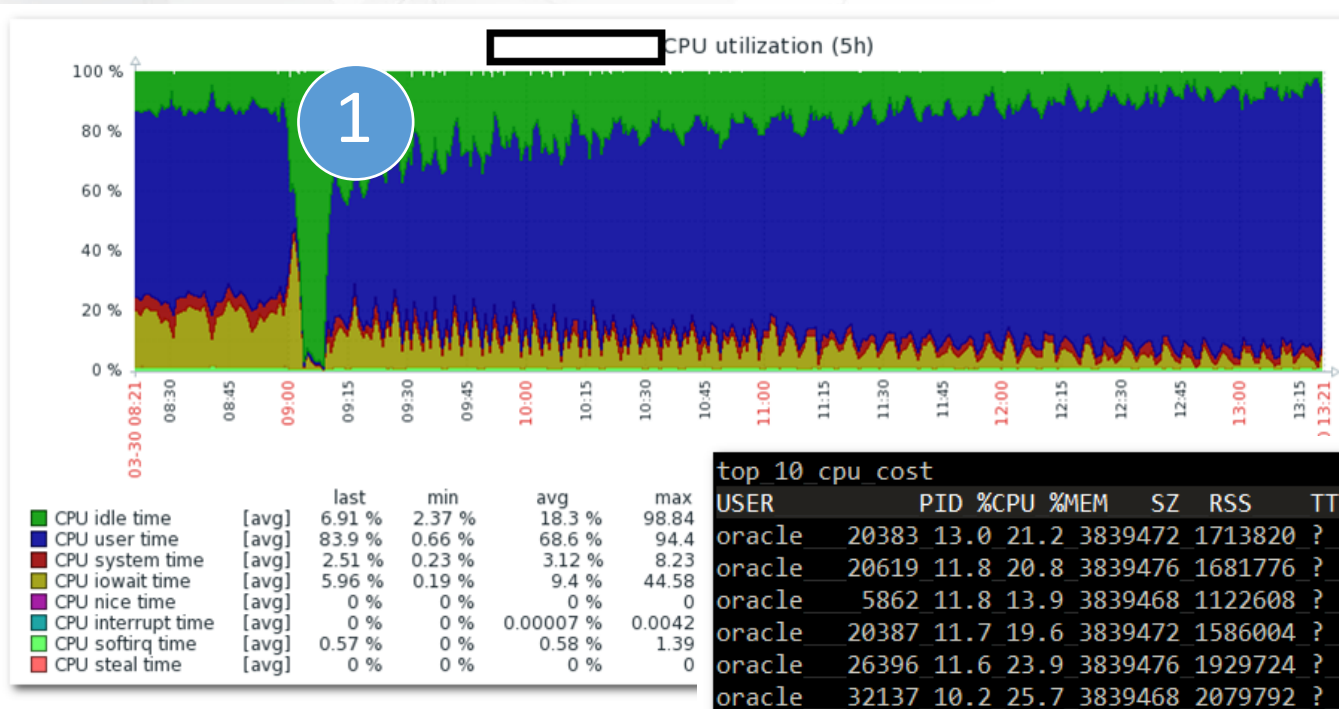
测试环境监控目标

- 不怕出问题
- 不一定立刻发现问题，但是要收集分析问题所需信息
- 要能发现问题，并定位问题的原因

测试环境监控体系



问题分析定位流程



3

APPNAME	ADDR	DBSID	ALL_SESSION	ACTIVE_SESSION	NOT_IDLE_SESSION	EVENT_SESSION
AMC		bamc	77	9	77	1
AMC		bamc	77	9	77	1
AMC		bamc	76	9	76	0
AMC		bamc	77	13	77	1
AMC		bamc	76	12	76	1
AMC		bamc	76	10	76	1
AMC		bamc	76	34	76	4
AMC		bamc	76	35	76	7

4

APPNAME	ADDR	DBSID	EVENT_NAME	SNAP_ID	SQL_ID	TIMES
AMC		BAMC	enq: TX - index contention	12311	7trvwbk3htbhm	23
AMC		BAMC	enq: TX - index contention	12392	0mjuxbn9v5gkq	1
AMC		BAMC	latch: cache buffers chains	12430	gd76jypp5kv5b	1
AMC		BAMC	buffer busy waits	12438	3yhzvj8mdzzr	1
AMC		BAMC	buffer busy waits	12510	bfna00kjuycw	5
AMC		BAMC	enq: TX - index contention	12564	7trvwbk3htbhm	2
AMC		BAMC	enq: TX - index contention	12663	7trvwbk3htbhm	11
AMC		BAMC	latch: cache buffers chains	12725	bfna00kjuycw	1
AMC		BAMC	enq: TX - index contention	12787	b0yvaxc344p3r	28
AMC		BAMC	enq: TX - index contention	12884	b0yvaxc344p3r	2
AMC		BAMC	buffer busy waits	12901	b0yvaxc344p3r	7

生产性能问题

应用连接数高

应用连接不释放

数据库连接风暴

CPU冲高

出现笛卡尔积

数据库锁

批量与联机冲突

索引分裂

全表扫描

联机交易缓慢

异常的等待事件

索引失效

I/O出现瓶颈

内存溢出

数据库性能诊断需求

发现性能问题

困难

- 1、性能测试场景无法完全覆盖功能改造
- 2、功能测试不太关注性能相关内容

没人看

性能分析门槛比较高

看不懂

- 1、性能DBA少
- 2、环境非常多

看不过来

对象

- 1、主要依赖AWR
- 2、商业软件的局限性

没有方法

实践的机会、经验的积累、知识的学习

水平不够

DBA工作的性质

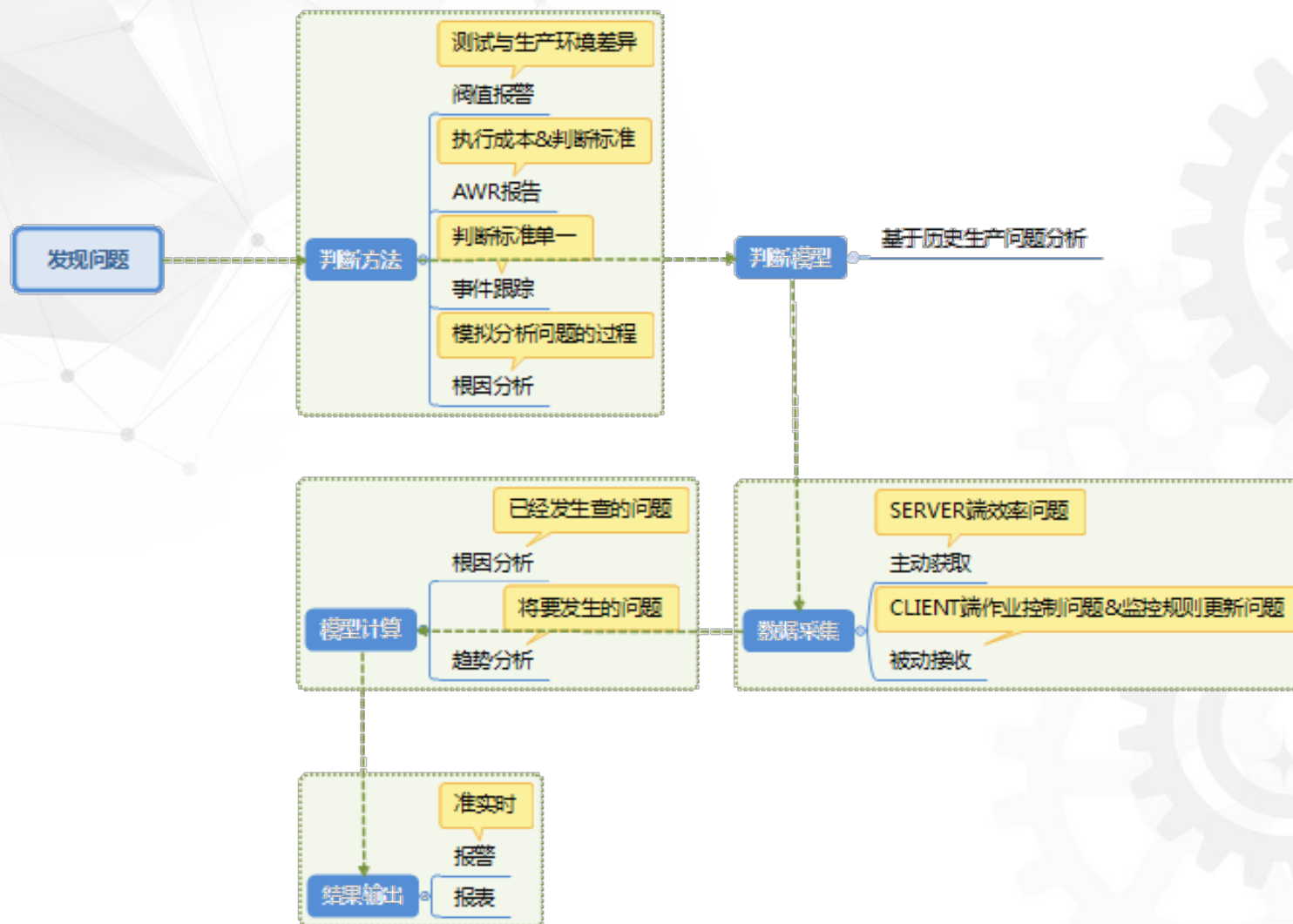
没有时间

适合测试环境特点、人员结构和工作特点

降低参与性能分析的门槛

自主可控，灵活调整的架构

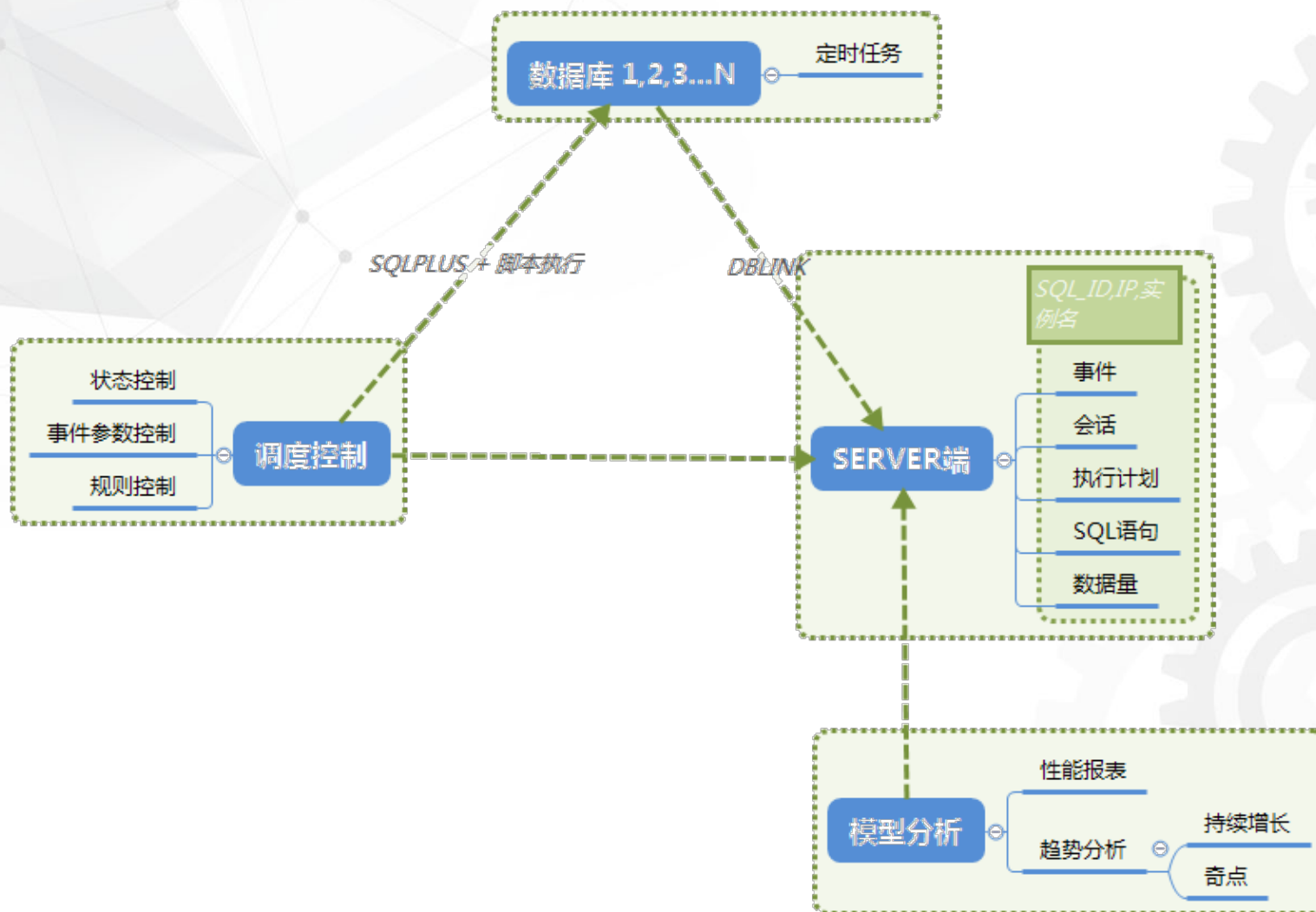
数据库性能诊断流程分析



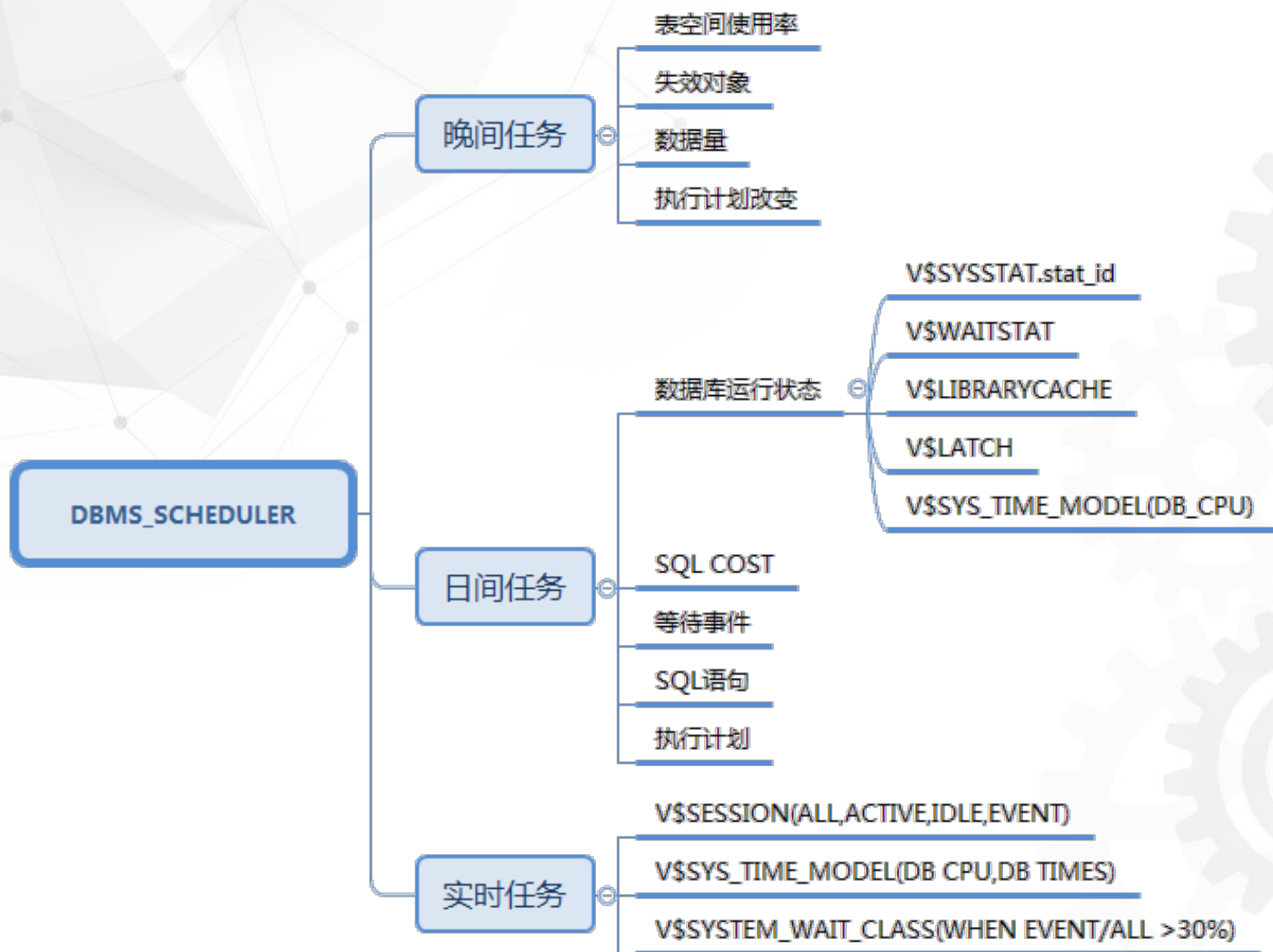
建设诊断模型

明确监控分类	梳理监控内容	分解判断标准	指标代码	选择监控指标	定义报警策略	辅助分析模型
根据监控需求方分类	总结分析存量生产问题，定位问题发生根本原因作为监控内容；根据生产需求，定义监控内容；根据版本改造，定义常规监控内容	分解监控内容中要素，形成单独的监控指标判断标准	与监控指标选择映射，方便制定报警策略	根据判断标准，转化为操作系统、数据库、网络、中间件和批量的可衡量，可发现的指标	当满足以下条件时触发监控报警，提示应用和性能测试人员	监控主体配合其他判断标准，提升监控的准确率，降低误报率，提高分析解决效率
生产问题	C1:连接数高，连接不释放	网络连接数	OS3	操作系统网络连接数	OS3>200	配合操作系统进程检查
	C2:存储过程失效，每次重新解析	数据库等待事件	DB1	cursor:pin S	出现DB1	配合当前时段语句扫描
	C3:索引分裂	数据库等待事件	DB2	enq: TX - index contention	出现 (DB1 & DB2)	配合当前时段语句扫描，操作系统检查
		数据库字典检查	DB9	使用sequence的索引		
	C10:调用次数过高	数据库SQL执行次数检查		exec times	>10000	配合AWR报告
	C4:RAC的GC等待事件	数据库等待事件	DB4	gc%	出现 (DB4)	配合当前时段语句扫描
	C5:异常等待事件	数据库等待事件	DB5	library cache pin	出现 (DB1 & DB5-9)	配合当前时段语句扫描，操作系统检查，中间件检查
			DB6	enq:US - contention		
			DB7	latch: row cache objects		
			DB8	read by other session		
			DB3	latch: cache buffers chains		
	C6:全表扫描	访问类型 数据库字典检查 数据量检查	DB1	cursor: pin S	(DB11 & DB12) + DB11	配合当前时段语句扫描，操作系统检查，中间件检查
			DB10	TABLE ACCESS FULL		
			DB11	没有索引的表		
			DB12	测试数据量与生产差异大 数据量持续增长		
生产需求	C7:移行操作	执行时间	T1	时间满足投产时间窗口	T1 + (DB1-8) + DB13	无
		数据库等待事件	DB1-8	没有发生异常等待事件		
		表空间使用率	DB13	数据库表空间没有异常增长		
	C8:资源变化	操作系统监控	CPU	CPU使用率	max (OS1)+avg (OS1)	配合当前时段语句扫描，中间件检查
			MEM	内存使用率	max (OS2)+avg (OS2)	
			NET	网络连接数	max (OS3)+avg (OS3)	
			EVENT	系统报警次数	SUM (EVENT_LOG)	
			OS4	文件系统使用率	max (OS4)+avg (OS4)	
	C9:高风险语句	语句扫描	EVENT	异常等待事件	出现 (DB1-8)	配合当前时段语句扫描，操作系统检查，中间件检查
			ACCESS	异常访问方式	出现TABLE ACCESS FULL, TABLE ACCESS STORAGE FULL	
			JOIN	异常表连接	出现NESTLOOPS, MERGE JOIN CARTESIAN	
			SESS	等待事件session占比	DB15>33%	
			COST	SQL开销大	DB16>1000	
			EXEC	SQL执行时间长	DB17>5分钟	
版本改造	架构改造	语句扫描，操作系统监控	DB1-17, OS1-4	无异常	全量报警策略	无
	重点交易改造					
	批量时间	批量执行时间	P1	满足生产批量时间窗口	max (P1), avg (P1)	配合当前时段语句扫描，操作系统检查
	出现系统层面报警	一段时间内的报警数量	C8_SYS	一段时间内的报警数量	num (C8_SYS)	配合数据库检查以及日志
	数据量变化	操作系统监控	DB12	数据量峰值	max (DB12)	无

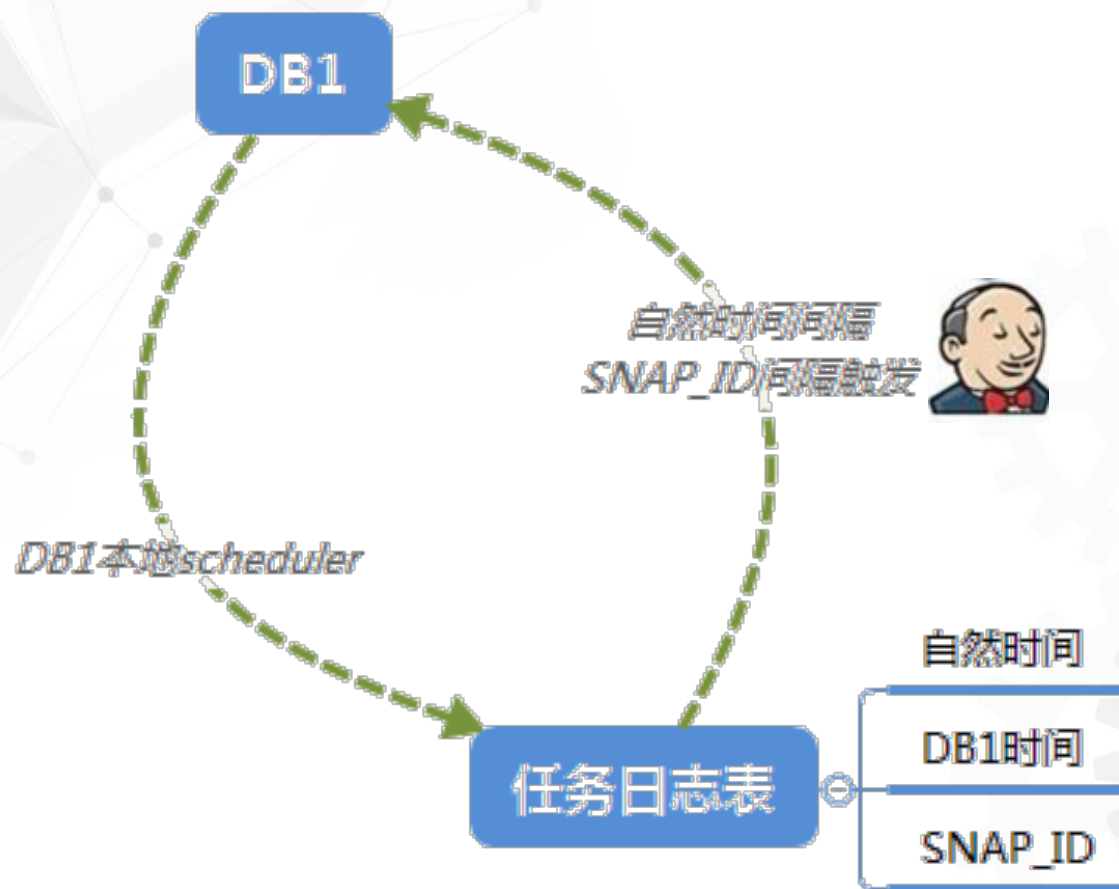
数据库性能诊断流程



客户端定时任务



问题：非自然时间的同步



报表输出

明确监控内容	梳理监控内容	分解判断标准	指标代码	选择监控指标	定义报警策略	辅助分析模型	统计次数
根据监控需求方分类	存量生产问题根本原因；生产需求的监控内容；版本改造常规监控内容	分解监控内容中要素，形成单独的监控指标判断标准	与监控指标选择映射，方便制定报警策略	根据判断标准，转化为的可衡量，可发现的指标	当满足以下条件时触发监控报警，提示应用和性能测试人员	需配合的其他辅助分析指标	出现次数和统计值
生产问题	C1:连接数高，连接不释放	数据库连接数	C1	数据库session连接数	值较大,且EVENT值大	数据库session连接数检查	[TOTAL:53 ACTIVE:12 EVENT:0]
	C2:存储过程失效，每次重新解析	数据库等待事件	C2	cursor:pin S	出现C2	当前时段语句扫描	[0]
	C3:索引分裂		C3	enq: TX - index contention	出现C3	使用sequence的索引	[4]
	C4:RAC的GC等待事件		C4	gc%	出现(DB4)	当前时段语句扫描	[12932]
	C10:SQL调用次数多		C10	executions_delta	>10000	当前时段语句扫描	[37844]
	C5:异常等待事件	数据库等待事件	C5_1	library cache pin	出现C2,C5	当前时段语句扫描，操作系统检查，中间件检查	[24]
			C5_2	enq:US - contention			[0]
			C5_3	latch: row cache objects			[10]
			C5_4	read by other session			[0]
			C5_5	latch: cache buffers chains			[64]
			C2	cursor: pin S			[0]
	C6:全表扫描	访问类型	C6_1	TABLE ACCESS FULL	出现C6	当前时段语句扫描，操作系统检查，中间件检查	[21334]
		数据字典检查	C6_2	没有索引的表			[0]
		数据量检查	C6_3	测试数据量不到生产的1/3			[31]
			C6_4	全表扫描对象数据量持续增长			[3]

报表输出

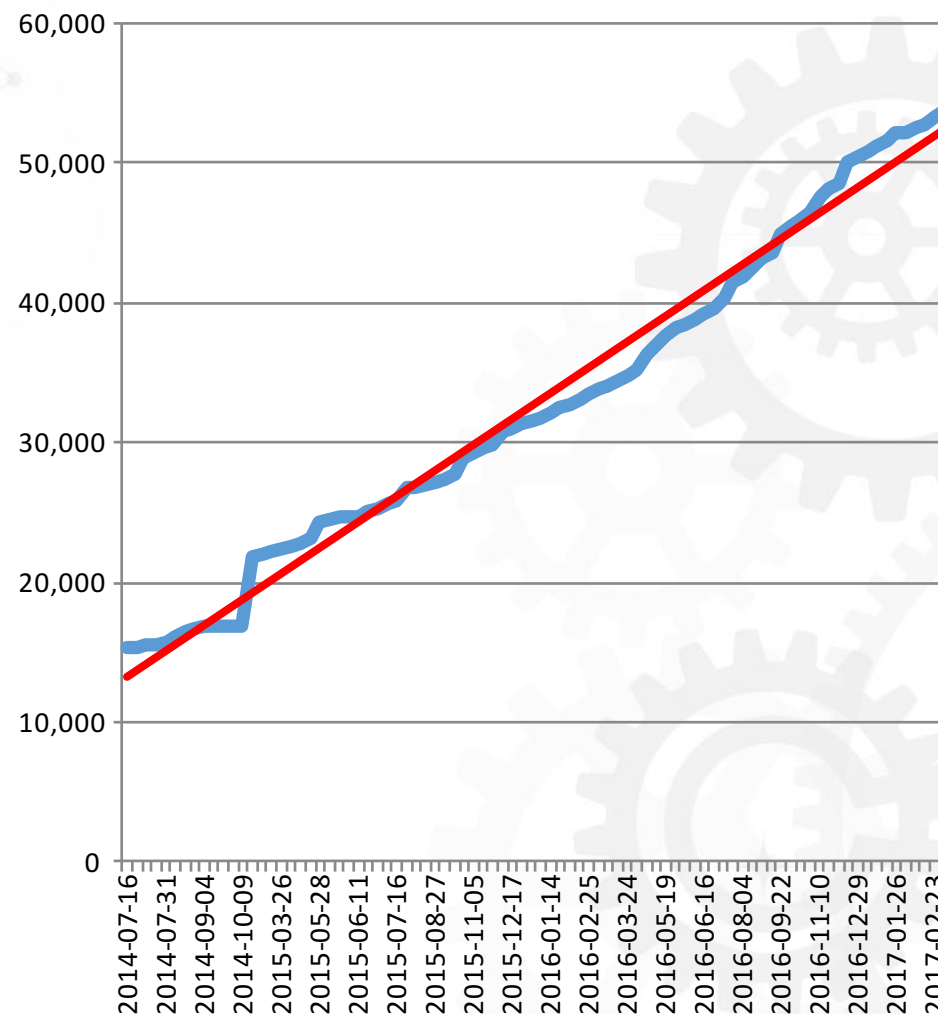
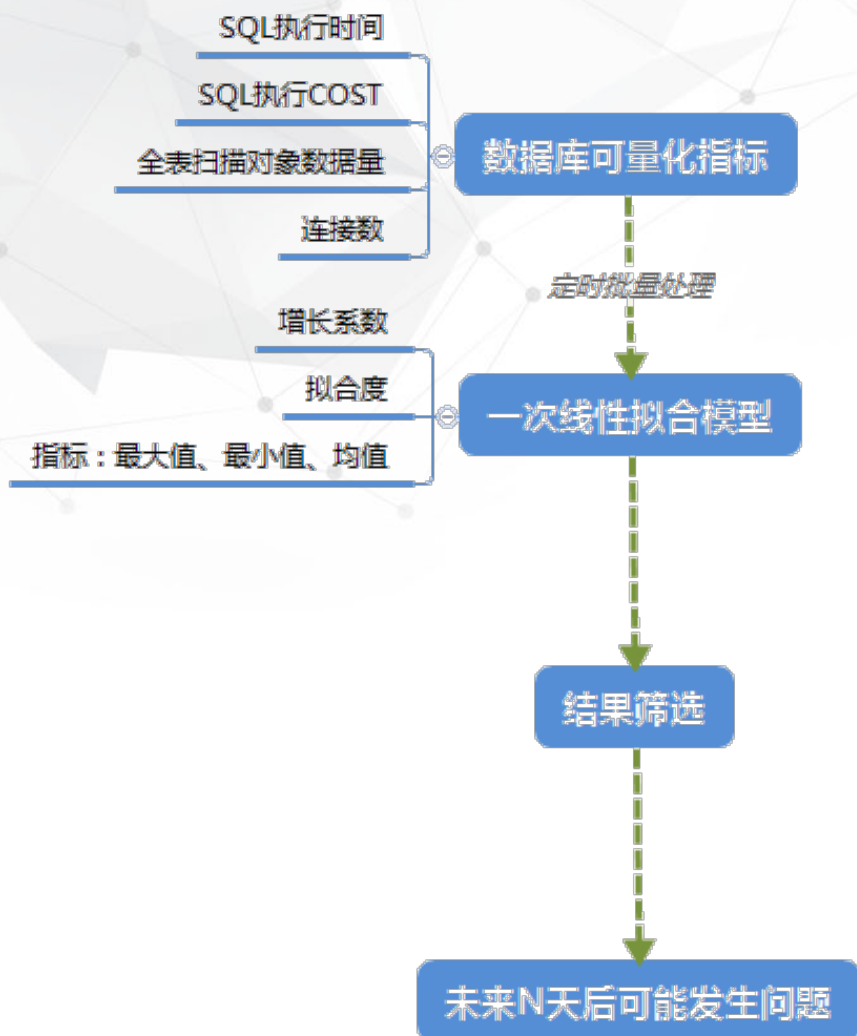
ALL enq: TX - index contention gc library cache pin enq:US - contention latch: row cache objects read by other session
latch: cache buffers chains cursor: pin S

应用名	ip	实例名	等待事件	SQL_ID	发生次数
CIIS		erm01	cursor: pin S	1rvj38hyjm2u2	2
CIIS		erm01	cursor: pin S	c1muxkmw2dcfj	6
CIIS		erm01	cursor: pin S wait on X	2b4a450zdwuzv	2

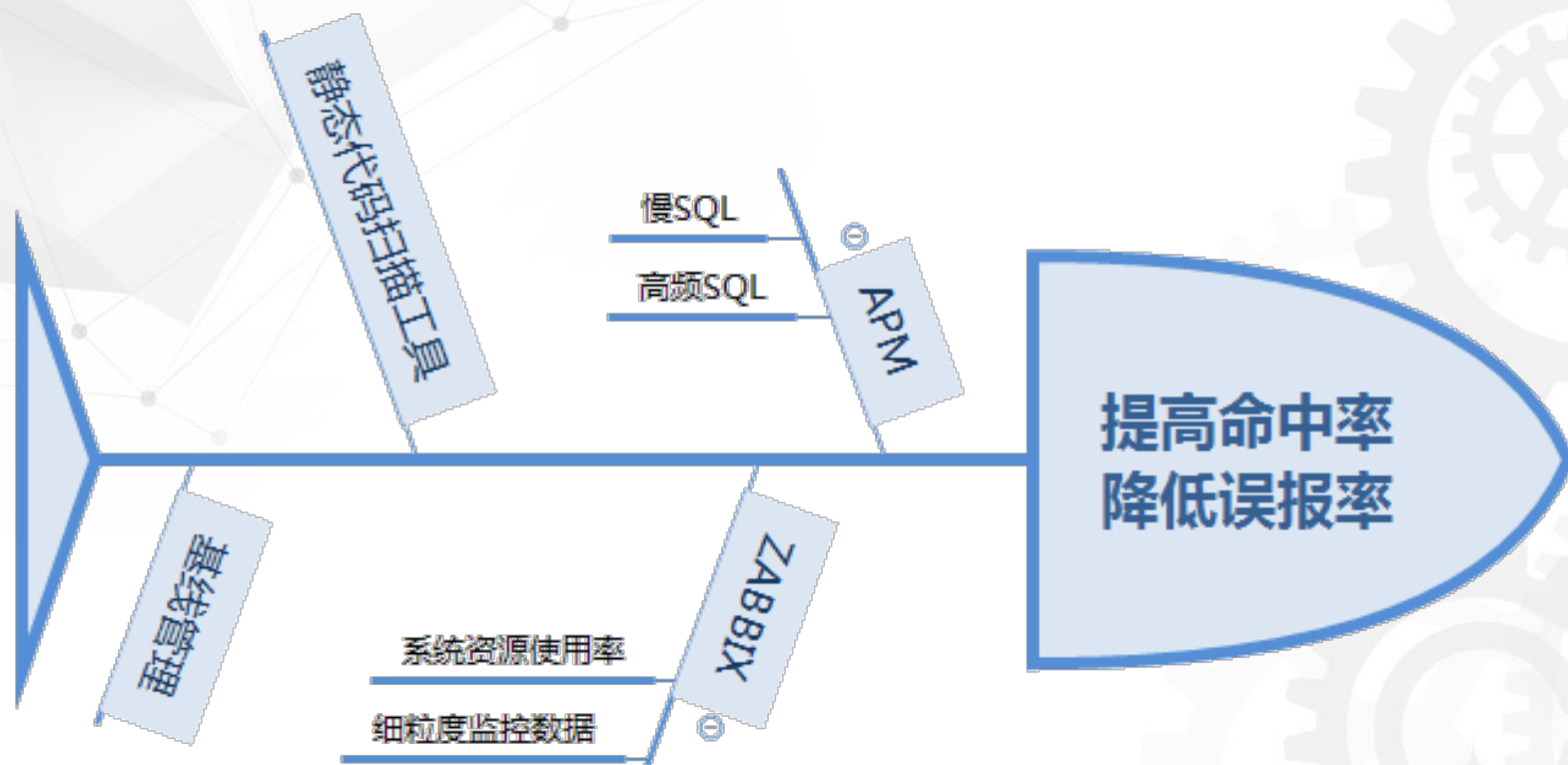
SQL文本

```
SELECT qry_time, qry_batno, request_no, user_id, user_name, organ_id, organ_name, system_id, request_type,
qry_type, mps_flag, reason, reason_desc, crs_reason, crs_reason_desc, condition, NAME, catofpaid, pid, pid18,
cur_status, rpt_status, err_desc, s_msgt_mtd, s_host_no, s_thread_no, r_host_no, r_thread_no, s_num, crs_catofpaid,
START_TIME, TASK_LOCK,
CERTCHK_FLAG, CERT_SMADDR_FLAG, TIMEDECAY, CERT_REQUEST_NO, CERT_ERRDESC FROM ( SELECT
a.qry_time, a.batch_no AS qry_batno, a.request_no, a.user_id, a.user_name, a.organ_id, a.organ_name, a.system_id,
a.request_type, a.qry_type, a.mps_flag, a.reason, a.reason_desc, a.crs_reason, a.crs_reason_desc, a.condition,
a.NAME, a.catofpaid, a.pid, a.pid18, a.cur_status, a.rpt_status, a.err_desc, a.s_msgt_mtd, a.s_host_no,
a.s_thread_no, a.r_host_no, a.r_thread_no, a.s_num, a.crs_catofpaid, b.request_order, START_TIME, TASK_LOCK,
CERTCHK_FLAG, CERT_SMADDR_FLAG, TIMEDECAY, CERT_REQUEST_NO, CERT_ERRDESC, ROW_NUMBER
() OVER (PARTITION BY b.request_order ORDER BY a.batch_no) AS rn FROM qry_per a, d_request_type b WHERE
( a.CUR_STATUS = '080' AND (s_msgt_mtd = '1' or s_msgt_mtd = '0' or s_msgt_mtd = '2') ) AND (a.TASK_LOCK =
'false' or a.TASK_LOCK is null) AND a.s_host_no = :1 AND a.request_type = b.request_type) WHERE rn < 11 ORDER
BY request_order
```

问题：测试与生产的数据量差异



提高命中率、降低误报率的方法

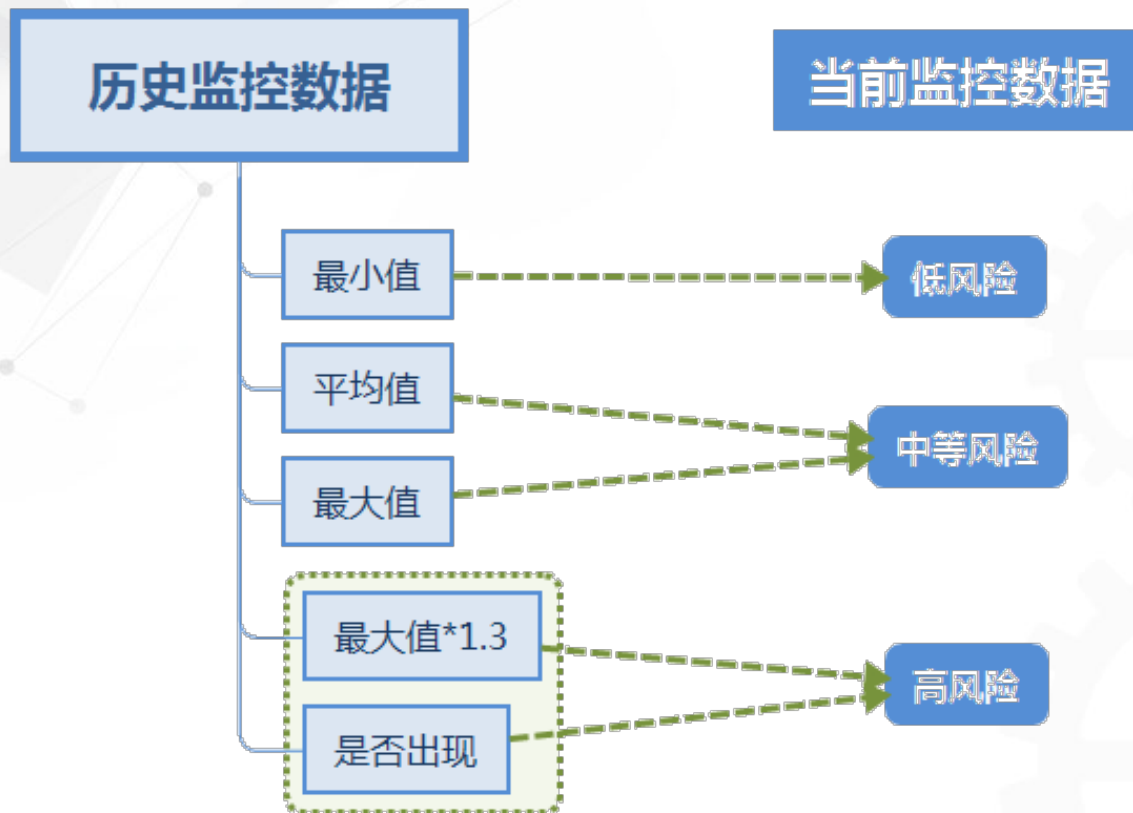


测试覆盖率问题：静态代码扫描



处理绑定变量的方法？ORA-00904

测试与生产资源差异问题：基线管理



与操作系统监控整合的问题：ZABBIX

▼ 1 个附件



15663002.log

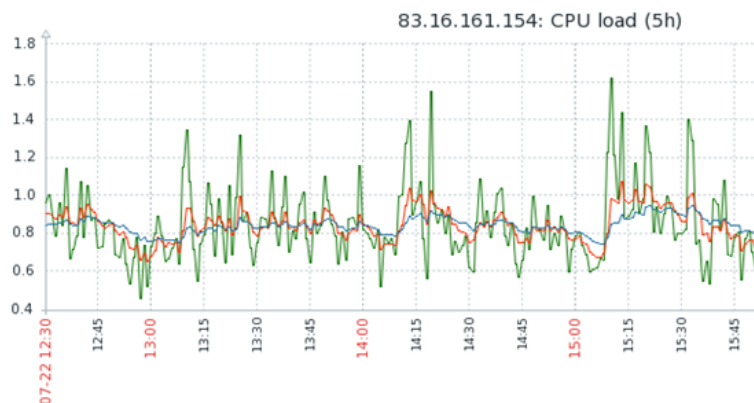
性能检查

CPU idle time (system.stat[cpu,id]): 3.65 %

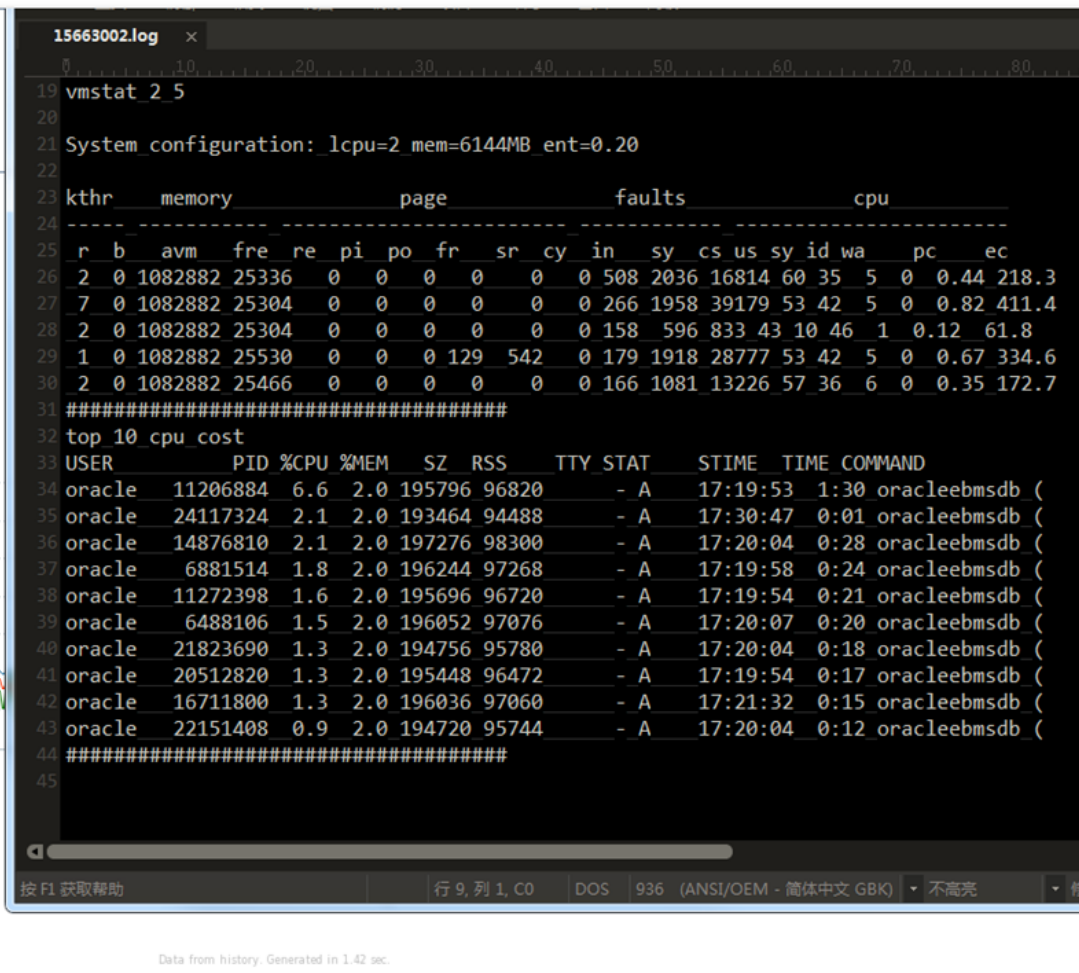
当前性能统计请参考附件中内容，最近1小时运行情况请参考图片

历史情况请登陆[http://\[redacted\]/monitor/](http://[redacted]/monitor/)查看

问题和建议请联系李雁南



		last	min	avg	max
■ Processor load (1 min average per core)	[avg]	0.4517	0.4517	0.8528	1.62
■ Processor load (5 min average per core)	[avg]	0.6443	0.6443	0.8519	1.07
■ Processor load (15 min average per core)	[avg]	0.7826	0.7404	0.8504	0.9494
● Trigger: Processor load is too high on 83.16.161.154 [> 5]					



从交易的角度看数据库：APM

PurePath 构成项 错误				
PurePath 树 (仅显示相关节点)				
方法	参数	总执行时...	细分	类
prepareCall(String sql)	{call pkg_...	0.08	cpu (100.0%)	WSJdbcConnection
doPost(HttpServletRequest)		11640.56	io (94.0%)	CtpRequestServlet
execute()	{call pkg_...	9676.03	io (100.0%)	WSJdbcPreparedStatement
socketRead0(FileDe		9665.55	io (100.0%)	SocketInputStream
getObject(int)		553.84	cpu io (66.0%)	WSJdbcCallableStatement
socketRead0(FileDe		553.84	cpu io (66.0%)	SocketInputStream
getObject(int)		73.15	cpu io (66.0%)	WSJdbcCallableStatement
socketRead0(FileDe		73.15	cpu io (66.0%)	SocketInputStream
getObject(int)		501.55	cpu io (66.0%)	WSJdbcCallableStatement
socketRead0(FileDe		501.55	cpu io (66.0%)	SocketInputStream
getObject(int)		658.33	cpu io (66.0%)	WSJdbcCallableStatement
socketRead0(FileDe		647.86	cpu io (66.0%)	SocketInputStream
getReportContent(Map		94.05	cpu io (66.0%)	ReportDownLoadViewer
log4j [ERROR]	logType=-...	-		procedureLogger

最慢组件

com.ibm.ws.rsadapter.jdbc.WSJdbcPreparedStatement/executeQuery

调用次数

81

持续时间(s)

19.668

占比(%)

85.217

努力方向：更加高效智能

机器学习

- 利用问题SQL的监控指标作为训练样本
- 频繁项集+关联分析算法
- 回归算法



2017年第八届中国数据库技术大会
DATABASE TECHNOLOGY CONFERENCE CHINA 2017

SequeMedia
威拓传媒

IT168.com

ITPUB

ChinaUnix



THANKS

SequeMedia
盛拓传媒

IT168.com

ITPUB

ChinaUnix