

openGauss训练营

WDR报告和性能调优



目录

- openGauss集群性能指标体系
 - 集群性能指标体系及应用
 - 核心指标矩阵
- WDR
- 性能调优案例分析

openGauss集群性能指标体系

➤ 系统级：集群级别或节点级别指标

- OS
- Instance Time
- Memory
- Session, Thread
- Events, Utility

➤ 对象级：数据库对象指标

- Database、Table、Index
- File
- Lock

➤ 应用级：表征应用负载性能

- Transaction、Statement
- Active session Profile
- Slow query
- Full SQL trace

系统级-OS

- OS_RUNTIME, OS_THREADS
- 获取实时的CPU时间, LOAD, 内存消耗信息 (类似top), 用于判断当前操作系统的负载状态

```
postgres=# select id,name,value from dbperf.os_runtime;
```

id	name	value
0	NUM_CPUS	16
1	NUM_CPU_CORES	8
2	NUM_CPU_SOCKETS	1
3	IDLE_TIME	2868821548
4	BUSY_TIME	349395171
5	USER_TIME	208848790
6	SYS_TIME	140546381
7	IOWAIT_TIME	3942616
8	NICE_TIME	384
9	AVG_IDLE_TIME	179301346
10	AVG_BUSY_TIME	21837198
11	AVG_USER_TIME	13053049
12	AVG_SYS_TIME	8784148
13	AVG_IOWAIT_TIME	246413
14	AVG_NICE_TIME	24
15	VM_PAGE_IN_BYTES	55390208
16	VM_PAGE_OUT_BYTES	206245888
17	LOAD	1.04
18	PHYSICAL_MEMORY_BYTES	135010942976

系统级-Instance Time

- INSTANCE_TIME
- 系统级的时间消耗细分。判断是否整个系统是否存在负载，网络，IO，CPU上的瓶颈

```
postgres=# select * from dbperf.instance_time;
```

stat_id	stat_name	value
0	DB_TIME	150010772266082
1	CPU_TIME	26047679385779
2	EXECUTION_TIME	101619432865080
3	PARSE_TIME	257221047661
4	PLAN_TIME	886902809918
5	REWRITE_TIME	16416698250
6	PL_EXECUTION_TIME	250884596
7	PL_COMPILATION_TIME	102665637
8	NET_SEND_TIME	1076322517306
9	DATA_IO_TIME	93354767615932

系统级-Memory

- MEMORY_NODE_DETAIL, SHARED_MEMORY_DETAIL
- 1. 实例级内存（全局，动态内存，共享内存，通信等）分配，使用率
- 2. 共享内存上下文分配/使用率，定位定界可细化至特性级别，结合session级别内存上下文分配视图接口，可帮助定位会话级内存使用问题

```
postgres=# select * from dbperf.MEMORY_NODE_DETAIL;
```

nodename	memorytype	memorybytes
dn_6001_6002_6003	max_process_memory	184320
dn_6001_6002_6003	process_used_memory	118211
dn_6001_6002_6003	max_dynamic_memory	69393
dn_6001_6002_6003	dynamic_used_memory	6060
dn_6001_6002_6003	dynamic_peak_memory	29870
dn_6001_6002_6003	dynamic_used_shrctx	342
dn_6001_6002_6003	dynamic_peak_shrctx	400
dn_6001_6002_6003	max_shared_memory	113902
dn_6001_6002_6003	shared_used_memory	111637
dn_6001_6002_6003	max_cstore_memory	1024
dn_6001_6002_6003	cstore_used_memory	0
dn_6001_6002_6003	max_sctpcomm_memory	4000
dn_6001_6002_6003	sctpcomm_used_memory	124
dn_6001_6002_6003	sctpcomm_peak_memory	124
dn_6001_6002_6003	other_used_memory	430
dn_6001_6002_6003	gpu_max_dynamic_memory	0
dn_6001_6002_6003	gpu_dynamic_used_memory	0
dn_6001_6002_6003	gpu_dynamic_peak_memory	0
dn_6001_6002_6003	pooler_conn_memory	0
dn_6001_6002_6003	pooler_freeconn_memory	0
dn_6001_6002_6003	storage_compress_memory	0
dn_6001_6002_6003	udf_reserved_memory	0

系统级-Memory 样例

```
postgres=# select * from db_perf.SHARED_MEMORY_DETAIL where level=2 order by usedsize desc limit 10;
```

contextname	level	parent	totalsize	freesize	usedsize
IncrCheckPointContext	2	StorageTopMemoryContext	59473984	7728	59466256
unique sql hash table	2	UniqueSQLContext	3603008	12976	3590032
wlm session info hash table	2	Workload manager memory context	664672	2704	661968
libcomm socket version lookup hash	2	CommunicatorGlobalMemoryContext	1040384	500080	540304
libcomm nodename & node_idx lookup hash	2	CommunicatorGlobalMemoryContext	1040384	508288	532096
libcomm ip & status lookup hash	2	CommunicatorGlobalMemoryContext	1040384	508288	532096
libcomm socket & node_idx lookup hash	2	CommunicatorGlobalMemoryContext	1040384	508288	532096
libcomm tid lookup hash	2	CommunicatorGlobalMemoryContext	1040384	508416	531968
instance statistics hash table	2	Workload manager memory context	267376	2704	264672
instr user hash table	2	InstrUserContext	57344	31248	26096

系统级-Session

- SESSION_STAT, SESSION_TIME, SESSION_MEMORY, SESSION_MEMORY_DETAIL, SESSION_STAT_ACTIVITY

1. 会话级负载强度，逻辑/物理读写强度，排序活动等，用于甄别出负载强度高，使用资源多的会话
2. 会话级时间细分，用于甄别出不同时间维度上session的消耗
3. 会话级总内存和内存上下文分配/使用，可以定位单个session在特性级上的内存问题
4. 获取实时系统活跃会话列表，包括会话基本信息，会话事务，语句，等待事件，会话状态（active,idle等），用于分析当前系统活跃用户的活动信息，比如从语句的开始时间判断当前会话是否一致阻塞在某一个语句，从等待事件可以判断某一个或者多数会话阻塞在哪个事件或者节点（分布式场景）

系统级-Session 样例

与CachePlan相关的内存使用，总使用量超过max_dynamic_memory之后，新动态内存申请将失败

contextname	sum	?column?	count
SessionCacheMemoryContext	7112.6729736328125000	2554.9589233398437500	1419
CachedPlan	5824.4609375000000000	1869.2473297119140625	254584
CachedPlanQuery	4809.9882812500000000	1899.9498825073242188	254584
CachedPlanSource	3458.3906250000000000	1292.1986312866210938	254584
SessionTopMemoryContext	963.0247802734375000	137.3696060180664063	1419
TopTransactionContext	512.1719360351562500	13507843017578125000	20
CBBTopMemoryContext	452.1148757934570313	128.0665893554687500	1419
OpfusionContext	429.7578125000000000	169.5303115844726563	33798
StorageTopMemoryContext	308.7414550781250000	104.3135833740234375	1419
OpfusionTemporaryContext	264.0468750000000000	262.4997253417968750	33798

contextname	sum	?column?	count
SessionCacheMemoryContext	5023.9603271484375000	1727.8703689575195313	1025
CachedPlan	3912.2207031250000000	1255.0918579101562500	170646
CachedPlanQuery	3228.8427734375000000	1274.5434112548828125	170613
CachedPlanSource	2317.8222656250000000	864.5426101684570313	170646
SessionTopMemoryContext	675.4298095703125000	99.6781845092773438	1025
CBBTopMemoryContext	317.2198715209960938	88.7432098388671875	1025
OpfusionContext	290.7421875000000000	114.7094039916992188	22802
Timezones	218.7454757690429688	2.7683258056640625	1025
StorageTopMemoryContext	218.5113525390625000	72.6146316528320313	1025
OpfusionTemporaryContext	178.1406250000000000	177.0968322753906250	22802

系统级-Thread

- THREAD_WAIT_STATUS

- 获取实例工作线程/辅助线程列表列表，判断线程的运行状态，当前正阻塞在哪个事件上，正在等待哪个锁，或被哪个会话阻塞（提供可以判断系统中的会话等待链的接口）

```
postgres=# select * from db_perf.thread_wait_status where wait_event = 'DataFileRead';
```

node_name	db_name	thread_name	wait_query_id	tid	sessionid	lwtid	psessionid	tlevel	smpid	wait_status	wait_event	locktag	lockmode	block_sessionid
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1082552813036033499	140035248957184	104648274	9494		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1154610406729111864	140033105127168	104648833	9653		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	578149654486229541	140023747274496	104651361	10370		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	578149654486229591	140023076022016	104649487	10433		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	650207248344169296	140027699717888	104649149	10078		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	289919278452132798	140037767022336	104652625	9291		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	2163979673082568230	140037194409728	104648865	9333		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	506092060264781704	140034273609472	104651763	9565		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	434034466470418107	140037462890240	104649227	9314		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	938437624816758820	140033820399360	104650743	9594		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1154610406729112598	140035980994304	104650409	9444		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	2380152455294765114	140034873501440	104649814	9522		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	722264842724443636	140036039722752	104647669	9437		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1154610406729112522	140045302363904	104650997	8658		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	217861684356327301	140039286646528	104647395	9169		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	506092060264781954	140028228298496	104650869	10038		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1371346138826442223	140032500999936	104649232	9710		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	361976872638002110	140037278308096	104648518	9324		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	145804090284996820	140037321307904	104649182	9322		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1659576514847568769	140025500866304	104648791	10222		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	650207248344170052	140037220628224	104649515	9332		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	2380152455294765657	140030361483008	104652261	9874		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	145804090284997312	140033623234304	104649337	9610		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1082552813036034538	140044012091136	104650298	8749		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1731634108978714504	140043949700864	104650838	17891		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1226668001020828916	140045843429120	104652266	8616		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1515461326859162432	140037735560960	104650683	9292		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1443403732728636219	140044811106048	104649003	8688		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	794322436441438503	140044400064256	104648421	8719		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	1226668001020828974	140033373628160	104649759	9625		0	0	wait io	DataFileRead			
dn_6001_6002_6003	locationdb	PostgreSQL JDBC Driver	145804090284997879	140028043704064	104649948	10057		0	0	wait io	DataFileRead			

大量的thread 在接口DataFileRead上等待IO完成

系统级-Events

- WAIT_EVENTS
- 各功能模块中IO, LOCK,LWLOCK,STATUS四类事件的等待次数, 等待失败次数, 等待时间等维度的统计信息, 可以帮助定位特性级细粒度时延性能问题

```
postgres=# select * from db_perft.wait_events order by avg_wait_time desc limit 10;
```

nodename	type	event	wait	failed_wait	total_wait_time	avg_wait_time	max_wait_time	min_wait_time
dn_6001_6002_6003	STATUS	Sort	879	0	8362008736	9513092	35063916	17
dn_6001_6002_6003	LWLOCK_EVENT	CBMParseXlogLock	276	0	646956768	2344046	16743111	125
dn_6001_6002_6003	LWLOCK_EVENT	DoubleWriteLock	3076011	0	767018343739	249354	25600222	2
dn_6001_6002_6003	STATUS	analyze	47960	0	6629834360	138236	21988131	74
dn_6001_6002_6003	STATUS	wait cmd	84666095559	0	9342299227691221	110342	1130542550432	1
dn_6001_6002_6003	IO_EVENT	WALInitSync	12739	0	241002878	18918	421099	9850
dn_6001_6002_6003	IO_EVENT	DoubleWriteFileWrite	39345060	0	406890682765	10341	533799	835
dn_6001_6002_6003	IO_EVENT	MPFL_READ	1	0	7353	7353	7353	7353
dn_6001_6002_6003	LWLOCK_EVENT	ControlFileLock	600	0	3898702	6497	50246	23
dn_6001_6002_6003	LOCK_EVENT	transactionid	21005	0	87119754	4147	239511	14

系统中Top10 重events

系统级-Utility

- COMM_DELAY,COMM_RECV_STREAM,COMM_SEND_STREAM,COMM_STATUS
 - REPLICATION_STAT
 - GLOBAL_GET_BGWRITER_STATUS,GLOBAL_PAGEWRITER_STATUS
 - POOLER_STATUS
-
1. 获取通信组件时延信息，接收和发送流状态，用于诊断通信链路容量和时延故障
 2. 获取分布式部署形态下主备同步状态信息，用于诊断主备时延，复制性能故障
 3. 获取后台全量/增量检查点信息，buffer中待落盘脏页信息，后台刷脏工作线程的状态将影响buffer pool的效率，磁盘IO繁忙程度，这些信息可以帮助优化缓存和IO性能
 4. 分布式部署形态下，CN和DN之间的连接池将影响事务的执行效率和成功率，pooler状态可以帮助诊断连接池瓶颈

对象级-Database

- STAT_DATABASE, STAT_DATABASE_CONFLICTS
- 提供数据库级别的活跃连接数，负载强度，块读写性能，行活动，死锁，临时下盘文件等信息。根据这些信息可以帮助建立数据库访存模型（读写比等负载特点），识别热点数据库，诊断数据库级别大颗粒性能瓶颈


```
SQL>=# select * from dbperf.STAT_DATABASE where datname='locationdb';
--[ RECORD 1 ]--
datid      | 16838
datname    | locationdb
numbackends| 5716
xact_commit| 39992292457
xact_rollback| 880
blks_read  | 169569638479
blks_hit   | 1374532771749
tup_returned| 2336377913854
tup_fetched| 2239191044182
tup_inserted| 7856159838
tup_updated| 62097735490
tup_deleted| 1176056006
conflicts  | 0
temp_files | 178
temp_bytes | 181256544256
deadlocks  | 0
blk_read_time| 0
blk_write_time| 0
stats_reset| 2020-09-19 16:48:33.646671+08
```

重IO操作
大量查询下盘

对象级-Table

- STAT_USER_TABLES, STAT_SYS_TABLES, STAT_ALL_TABLES
STATIO_USER_TABLES, STATIO_SYS_TABLES, STATIO_ALL_TABLES
GLOBAL_STAT_HOTKEYS_INFO
- 1. 表上行扫描活动，索引扫描活动，行变更活动，活跃行占比，表维护操作活动（vaccum, analyze），页面读取缓冲命中率等，这些信息可以帮助建立表级别的访存模型（读写比，缓存效率等），识别热点表，预警表维护操作，诊断表级别细粒度性能瓶颈
- 2. 识别表级别热key，定界热点访问造成的网络，IO瓶颈

对象级-Table样例

```
-[ RECORD 1 ]-----+-----+-----+
reloid          | 16874
schemaname      | 
relname         | t_cell_fp_3
seq_scan        | 46
seq_tup_read    | 5182600379
idx_scan        | 4695318663
idx_tup_fetch   | 6221619027
n_tup_ins       | 263613956
n_tup_upd       | 2169453196
n_tup_del       | 0
n_tup_hot_upd   | 2101692435
n_live_tup      | 199836881
n_dead_tup      | 34774109
last_vacuum      | 2021-01-22 10:14:18.086408+08
last_autovacuum | 
last_analyze     | 2021-03-26 09:03:20.616898+08
last_autoanalyze | 
vacuum_count     | 2
autovacuum_count | 0
analyze_count    | 152
autoanalyze count | 0
```


对象级-Index

- STAT_USER_INDEXES, STAT_SYS_INDEXES, STAT_ALL_INDEXES
STATIO_USER_INDEXES, STATIO_SYS_INDEXES, STATIO_ALL_INDEXES
- 索引使用统计: index scan次数, index scan返回的索引项, 通过index scan返回的表行数等, 索引页的缓存效率等, 用以评估索引收益和效率

对象级-File

- FILE_IOSTAT,FILE_REDO_IOSTAT,STAT_BAD_BLOCK
- 1. 数据（数据，索引）文件的IO性能统计指标（读写数目，耗时，时延），可以帮助建立数据文件物理访存的模型，识别文件级别的物理IO强度和瓶颈
- 2. 获取操作Redo 文件的性能，帮助诊断redo日志操作的性能瓶颈

对象级-Lock

- LOCKS
- 对象锁涉及到的对象，事务，会话，锁信息，实时显示当前系统锁等待关系，识别热点锁

对象级-Sequence

- STATIO_USER_SEQUENCES, STATIO_SYS_SEQUENCES, STATIO_ALL_SEQUENCES
- sequence的缓存效率

应用级-Transaction

- TRANSACTIONS_RUNNING_XACTS, TRANSACTIONS_PREPARED_XACTS
- 获取当前正在运行的单节点/两阶段事务列表，结合session和thread视图，诊断运行时间超过预期的事务

应用级-Statement

- STATEMENT_COUNT, STATEMENT
- 1. DDL, DML(select, insert, update, delete), DCL语句的分布比率, 帮助建立负载特征模型, 预警负载异常变动
- 2. 语句级别 (归一化SQL, 模板SQL) 的响应时间, 执行次数, 行活动, 软硬解析比, 时间模型, 网络开销, 排序性能 (时间, 内存, 溢出), 执行器HASH性能 (时间, 内存, 溢出)。据此可以识别热点语句, 定位语句性能瓶颈, 建立语句性能基线, 以低成本预警语句性能变化。

应用级-Active session Profile

- LOCAL_ACTIVE_SESSION, GS_ASP(public schema)
- ASP(Active Session Profile) 活跃会话概要信息，通过采样实例活跃会话的状态信息，低成本复现过去一段时间的系统活动，主要包含会话基本信息，会话事务，语句，等待事件，会话状态(active,idle等)，当前正阻塞在哪个事件上，正在等待哪个锁，或被哪个会话阻塞。可以从中获取如下主要概要信息：
 1. 最近用户session最耗资源的的事件
 2. 最近比较占资源的session/SQL把资源都消耗在哪些event上
 3. 最近执行时间/执行次数最多的是哪些SQL(进而可以找出表，数据库)
 4. 最近钟最耗资源的用户的信息
 5. 最近阻塞其他session最多的session

LOCAL_ACTIVE_SESSION的默认采样频率是1s，内存视图，ASP的默认采样频率是10s，持久化在存储

应用级-Full SQL trace&Slow Query

- STATEMENT_HISTORY
- 记录全量SQL信息，分为L0, L1,L2三个等级，可以获取实例信息，客户端信息，语句概要信息，执行信息，行活动信息，Cache/IO，时间模型，网络统计信息，锁概要信息，锁详细信息等。通过全量SQL，可以得到整个系统所有语句的执行流水以及他们的详细性能数据(持久化的)。除Statement视图提供的能力外，还额外提供了详细加放锁信息，可以诊断到单语句级别的性能波动。
- 达到慢查询阈值设置的语句性能信息，性能要素和全量SQL一致

应用级-Full SQL trace

Level 0

性能影响<1%
默认常开

实例信息

客户端信息

语句概要信息

执行上下文信息

元组访存

Buffer

Time Breakdown

网络开销

Level 1

性能影响<3%
建议常开，规划存储

执行计划

LOCK/LOCK WAIT
计数

LWLOCK/LWLOCK WAIT
计数

Level 2

性能影响<30%
建议短暂开启

LOCK/LOCK Wait
Time

LWLOCK/LWLOCK Wait
Time

LOCK Start

- Timestamp
- Locktag
- lockmode

LOCK Release

- Timestamp

LWLOCK Start

- Timestamp
- Locktag
- lockmode

LWLOCK Release

- Timestamp

- 支持慢SQL与Full SQL
- 支持独立TRACE Level设置
- 系统表存储，存储周期可配置
- 时间范围内的报表生成

核心指标矩阵-1

分类	项目	Instance	Session	Unqiue Query	Query	DB	Table	Index	Redo
时间细分	DB_TIME – 作业在多核下的有效时间花费 CPU_TIME – CPU时间的消耗 EXECUTION_TIME – 执行器内花费的时间 PARSE_TIME – SQL解析的时间花费 PLAN_TIME – 生成Plan的时间花费 REWRITE_TIME – SQL重写的时间消耗 PL_EXECUTION_TIME – 存储过程的执行时间 PL_COMPILATION_TIME – 存储过程编译时间 NET_SEND_TIME – 网络上的时间花销 DATA_IO_TIME – IO时间上的花销	INSTANCE_TIME	SESSION_TIME	STATEMENT	STATEMENT_HISTORY				
内存	全局内存块（全局，动态内存，共享内存，通信）	MEMORY_NODE_DETAIL							
	内存上下文分配，使用	SHARED_MEMORY_DETAIL	SESSION_MEMORY SESSION_MEMORY_DETAIL						
	排序			STATEMENT					
	执行器HASH			STATEMENT					
网络	通信组件性能	COMM_DELAY COMM_RECV_STREAM COMM_SEND_STREAM COMM_STATUS	COMM_DELAY COMM_RECV_STREAM COMM_SEND_STREAM COMM_STATUS						
	SQL网络消耗			STATEMENT	STATEMENT_HISTORY				

核心指标矩阵-2

分类	项目	Instance	Session	Unqiue Query	Query	DB	Table	Index	Redo
IO活动	读/写物理文件次数						FILE_IOSTAT		
	读/写物理文件总时长						FILE_IOSTAT		
	读写物理文件响应时间						FILE_IOSTAT		
	写xlog文件总时长								FILE_REDO_IOSTAT
	写xlog文件响应时间								FILE_REDO_IOSTAT
	数据块（元组，toast）缓冲命中率		SESSION_STAT	STATEMENT	STATEMENT_HISTORY		STATIO_USER_TABLES GLOBAL_STAT_HOTKEYS_INFO		
	索引块缓冲命中率						STATIO_USER_INDEXES		
行活动	表扫描/索引扫描		SESSION_STAT				STAT_USER_TABLES		
	索引扫描次数，索引返回数，索引返回表行数							STAT_USER_INDEXES, STATIO_USER_INDEXES	
	行更新（插入，更新，删除）					STAT_DATABASE			
	查询返回行，随机/顺序扫描行			STATEMENT	STATEMENT_HISTORY	STAT_DATABASE			

核心指标矩阵-3

分类	项目	Instance	Session	Unqiue Query	Query	DB	Table	Index	Redo
负载	提交/回滚事务数	WORKLOAD_TRANSACTION	SESSION_STAT			STAT_DATABASE			
	SQL分布(DDL,DML,DCL,SUID)	WORKLOAD_SQL_COUNT	SESSION_STAT						
	SQL执行次数			STATEMENT					
EVENTS	IO	WAIT_EVENTS	WAIT_EVENTS						
	LOCK	WAIT_EVENTS	WAIT_EVENTS						
	LWLOCK	WAIT_EVENTS	WAIT_EVENTS						
	STATUS	WAIT_EVENTS	WAIT_EVENTS						
LOCK	实时锁信息	LOCKS	LOCKS						
	历史锁信息				STATEMENT_HISTORY				



Microsoft Excel
工作表

WDR-WDR Snapshot

- 定期对DBE_PERF内global/summary视图采集快照
dbperf.summary_statement -> snapshot.snap_summary_statement
dbperf.summary_workload_sql_count -> snapshot.snap_summary_workload_sql_count
- 查看当前WDR snapshot列表:
select * from snapshot.snapshot order by start_ts;
- 查看某次WDR snapshot中相关summary/global快照的时间花费
select * from snapshot.tables_snap_timestamp where snapshot_id = 1 order by start_ts;

- 手工创建WDR snapshot
select create_wdr_snapshot();
CCN + monadmin

- 默认:
1小时自动运行一次
数据保留8天

postgres=#	select * from snapshot.tables_snap_timestamp where snapshot_id = 1 order by start_ts;	snapshot_id	db_name	tablename	start_ts	end_ts
1	postgres	1	postgres	snap_global_stat_all_indexes	2020-11-30 11:24:28.256662+08	2020-11-30 11:24:28.717085+08
1	postgres	1	postgres	snap_summary_stat_all_indexes	2020-11-30 11:24:28.719939+08	2020-11-30 11:24:28.909906+08
1	postgres	1	postgres	snap_global_stat_all_sequences	2020-11-30 11:24:28.911715+08	2020-11-30 11:24:29.044961+08
1	postgres	1	postgres	snap_summary_stat_all_sequences	2020-11-30 11:24:29.046286+08	2020-11-30 11:24:29.127136+08
1	postgres	1	postgres	snap_global_stat_all_tables	2020-11-30 11:24:29.129442+08	2020-11-30 11:24:29.255711+08
1	postgres	1	postgres	snap_summary_stat_all_tables	2020-11-30 11:24:29.256273+08	2020-11-30 11:24:29.314828+08
1	postgres	1	postgres	snap_global_stat_all_indexes	2020-11-30 11:24:29.317396+08	2020-11-30 11:24:30.206004+08
1	postgres	1	postgres	snap_summary_stat_all_indexes	2020-11-30 11:24:30.208444+08	2020-11-30 11:24:30.443964+08
1	postgres	1	postgres	snap_global_stat_user_functions	2020-11-30 11:24:30.446404+08	2020-11-30 11:24:30.536203+08
1	postgres	1	postgres	snap_summary_stat_user_functions	2020-11-30 11:24:30.538770+08	2020-11-30 11:24:30.619492+08
1	postgres	1	postgres	snap_global_stat_all_tables	2020-11-30 11:24:30.621862+08	2020-11-30 11:24:31.548082+08
1	postgres	1	postgres	snap_summary_stat_all_tables	2020-11-30 11:24:31.550761+08	2020-11-30 11:24:31.734140+08
1	postgres	1	postgres	snap_global_os_runtime	2020-11-30 11:24:32.234764+08	2020-11-30 11:24:32.338688+08
1	postgres	1	postgres	snap_global_os_threads	2020-11-30 11:24:32.350761+08	2020-11-30 11:24:32.493409+08
1	postgres	1	postgres	snap_summary_workload_sql_count	2020-11-30 11:24:32.494029+08	2020-11-30 11:24:32.512369+08
1	postgres	1	postgres	snap_summary_workload_sql_elapsed_time	2020-11-30 11:24:32.512853+08	2020-11-30 11:24:32.526598+08
1	postgres	1	postgres	snap_global_workload_transaction	2020-11-30 11:24:32.526935+08	2020-11-30 11:24:32.539464+08
1	postgres	1	postgres	snap_summary_workload_transaction	2020-11-30 11:24:32.539874+08	2020-11-30 11:24:32.553744+08
1	postgres	1	postgres	snap_global_thread_wait_status	2020-11-30 11:24:32.55615+08	2020-11-30 11:24:32.571551+08
1	postgres	1	postgres	snap_global_memory_node_detail	2020-11-30 11:24:32.572168	2020-11-30 11:24:32.700539+08
1	postgres	1	postgres	snap_global_memory_detail	2020-11-30 11:24:32.701131+08	2020-11-30 11:24:32.730009+08
1	postgres	1	postgres	snap_global_comm_send_stream	2020-11-30 11:24:32.731014+08	2020-11-30 11:24:32.825342+08
1	postgres	1	postgres	snap_global_comm_recv_stream	2020-11-30 11:24:32.825939+08	2020-11-30 11:24:32.832711+08
1	postgres	1	postgres	snap_global_comm_status	2020-11-30 11:24:32.83312+08	2020-11-30 11:24:32.879921+08
1	postgres	1	postgres	snap_global_pooler_status	2020-11-30 11:24:32.880449+08	2020-11-30 11:24:32.925063+08
1	postgres	1	postgres	snap_global_stat_db_cu	2020-11-30 11:24:32.92581+08	2020-11-30 11:24:32.938821+08
1	postgres	1	postgres	snap_global_database	2020-11-30 11:24:32.939244+08	2020-11-30 11:24:33.051677+08
1	postgres	1	postgres	snap_summary_stat_database	2020-11-30 11:24:33.052063+08	2020-11-30 11:24:33.062456+08
1	postgres	1	postgres	snap_global_stat_database_conflicts	2020-11-30 11:24:33.070291+08	2020-11-30 11:24:33.104298+08
1	postgres	1	postgres	snap_summary_stat_database_conflicts	2020-11-30 11:24:33.104758+08	2020-11-30 11:24:33.119559+08
1	postgres	1	postgres	snap_global_stat_bad_block	2020-11-30 11:24:33.120352+08	2020-11-30 11:24:33.126588+08
1	postgres	1	postgres	snap_global_file_redo_iostat	2020-11-30 11:24:33.126948+08	2020-11-30 11:24:33.133075+08
1	postgres	1	postgres	snap_global_file_redo_block	2020-11-30 11:24:33.133447+08	2020-11-30 11:24:33.135277+08
1	postgres	1	postgres	snap_global_file_redo_iostat	2020-11-30 11:24:33.138622+08	2020-11-30 11:24:33.146852+08
1	postgres	1	postgres	snap_summary_rel_iostat	2020-11-30 11:24:33.147246+08	2020-11-30 11:24:33.151248+08
1	postgres	1	postgres	snap_global_file_iostat	2020-11-30 11:24:33.153594+08	2020-11-30 11:24:33.161411+08
1	postgres	1	postgres	snap_summary_file_iostat	2020-11-30 11:24:33.161799+08	2020-11-30 11:24:33.167208+08
1	postgres	1	postgres	snap_global_file_iostat	2020-11-30 11:24:33.167585+08	2020-11-30 11:24:33.284888+08
1	postgres	1	postgres	snap_global_replication_slots	2020-11-30 11:24:33.285464+08	2020-11-30 11:24:33.321919+08
1	postgres	1	postgres	snap_global_replication_status	2020-11-30 11:24:33.322385+08	2020-11-30 11:24:33.331576+08
1	postgres	1	postgres	snap_global_replication_running_xacts	2020-11-30 11:24:33.341915+08	2020-11-30 11:24:33.377484+08
1	postgres	1	postgres	snap_summary_transactions_running_xacts	2020-11-30 11:24:33.377644+08	2020-11-30 11:24:33.383483+08
1	postgres	1	postgres	snap_global_transactions_prepared_xacts	2020-11-30 11:24:33.433858+08	2020-11-30 11:24:33.462431+08
1	postgres	1	postgres	snap_summary_transactions_prepared_xacts	2020-11-30 11:24:33.462674+08	2020-11-30 11:24:33.473235+08
1	postgres	1	postgres	snap_global_statement_count	2020-11-30 11:24:33.473619+08	2020-11-30 11:24:33.478979+08
1	postgres	1	postgres	snap_summary_statement_count	2020-11-30 11:24:33.479368+08	2020-11-30 11:24:33.571399+08
1	postgres	1	postgres	snap_global_config_settings	2020-11-30 11:24:33.591062+08	2020-11-30 11:24:33.60516+08
1	postgres	1	postgres	snap_summary_config_settings	2020-11-30 11:24:33.605629+08	2020-11-30 11:24:33.623924+08
1	postgres	1	postgres	snap_global_wait_events	2020-11-30 11:24:35.624593+08	2020-11-30 11:24:36.264418+08
1	postgres	1	postgres	snap_summary_user_login	2020-11-30 11:24:36.265219+08	2020-11-30 11:24:36.271457+08
1	postgres	1	postgres	snap_global_checkpoint_status	2020-11-30 11:24:36.271678+08	2020-11-30 11:24:36.276476+08
1	postgres	1	postgres	snap_global_double_write_status	2020-11-30 11:24:36.276955+08	2020-11-30 11:24:36.283985+08
1	postgres	1	postgres	snap_global_pagewriter_status	2020-11-30 11:24:36.284466+08	2020-11-30 11:24:36.291092+08
1	postgres	1	postgres	snap_global_rto_status	2020-11-30 11:24:36.291522+08	2020-11-30 11:24:36.293989+08
1	postgres	1	postgres	snap_global_recovery_status	2020-11-30 11:24:36.30029+08	2020-11-30 11:24:36.305407+08
1	postgres	1	postgres	snap_global_readpool_status	2020-11-30 11:24:36.311656+08	2020-11-30 11:24:36.324559+08
1	postgres	1	postgres	snap_statement_responsetime_percentile	2020-11-30 11:24:36.324094+08	2020-11-30 11:24:36.328502+08
1	postgres	1	postgres	snap_global_respond_time	2020-11-30 11:24:36.328779+08	2020-11-30 11:24:36.333072+08
1	postgres	1	postgres	snap_class_vital_info	2020-11-30 11:24:36.337235+08	2020-11-30 11:24:36.343857+08

WDR-生成WDR Report

- `select generate_wdr_report(begin_snap_id Oid, end_snap_id Oid, int report_type, int report_scope, int node_name);`
 - `report_type` - summary/ detail/ all
 - `report_scope` - cluster/ node
 - `node_name` – 结点级别report时候, 指定node name; cluster级别省略或者指定为 NULL
 - 要在postgres库内执行
- 以下情况WDR Report不可生成:
 - 两次Snapshot之间结点(CN/DN)重启
 - 两次Snapshot之间Drop DB

WDR-集群级报表

- 集群级Report
- Summary
 - Database Stat
 - Load Profile
 - Instance Efficiency Percentages
 - IO Profile
- SQL Statistics
- Cache IO Stats
- Object Stats
- SQL Detail



by_cluster.html

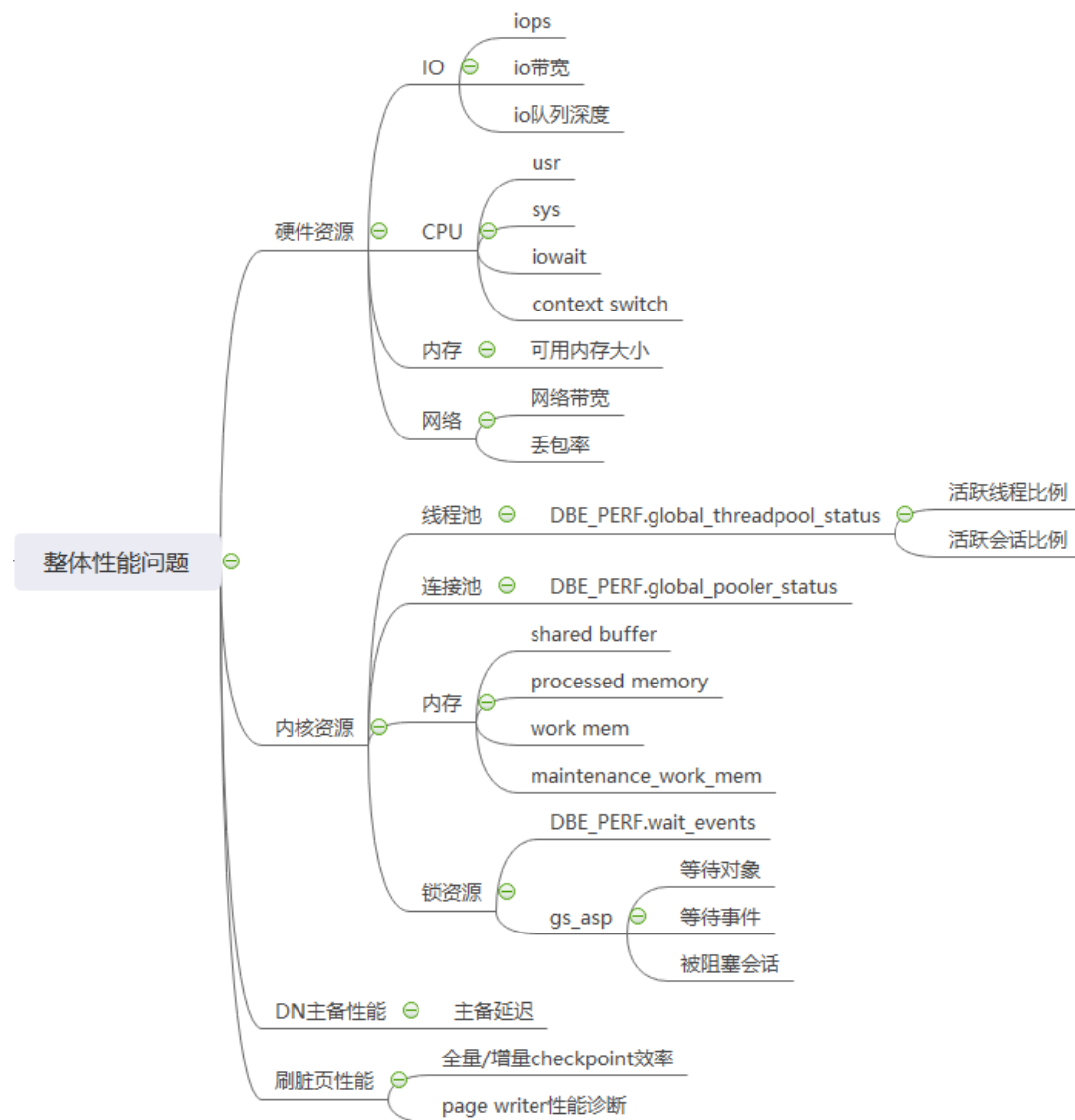
WDR-节点级报表

- Summary
 - Instance Efficiency Percentages
 - Top 10 Events by Total Wait Time
 - Wait Classes by Total Wait Time
 - IO Profile
 - Memory Statistics
- Time Model
- SQL Statistics
- **Wait Events**
- Cache IO Stats
- Utility Status
- Object stats
- Configuration Settings

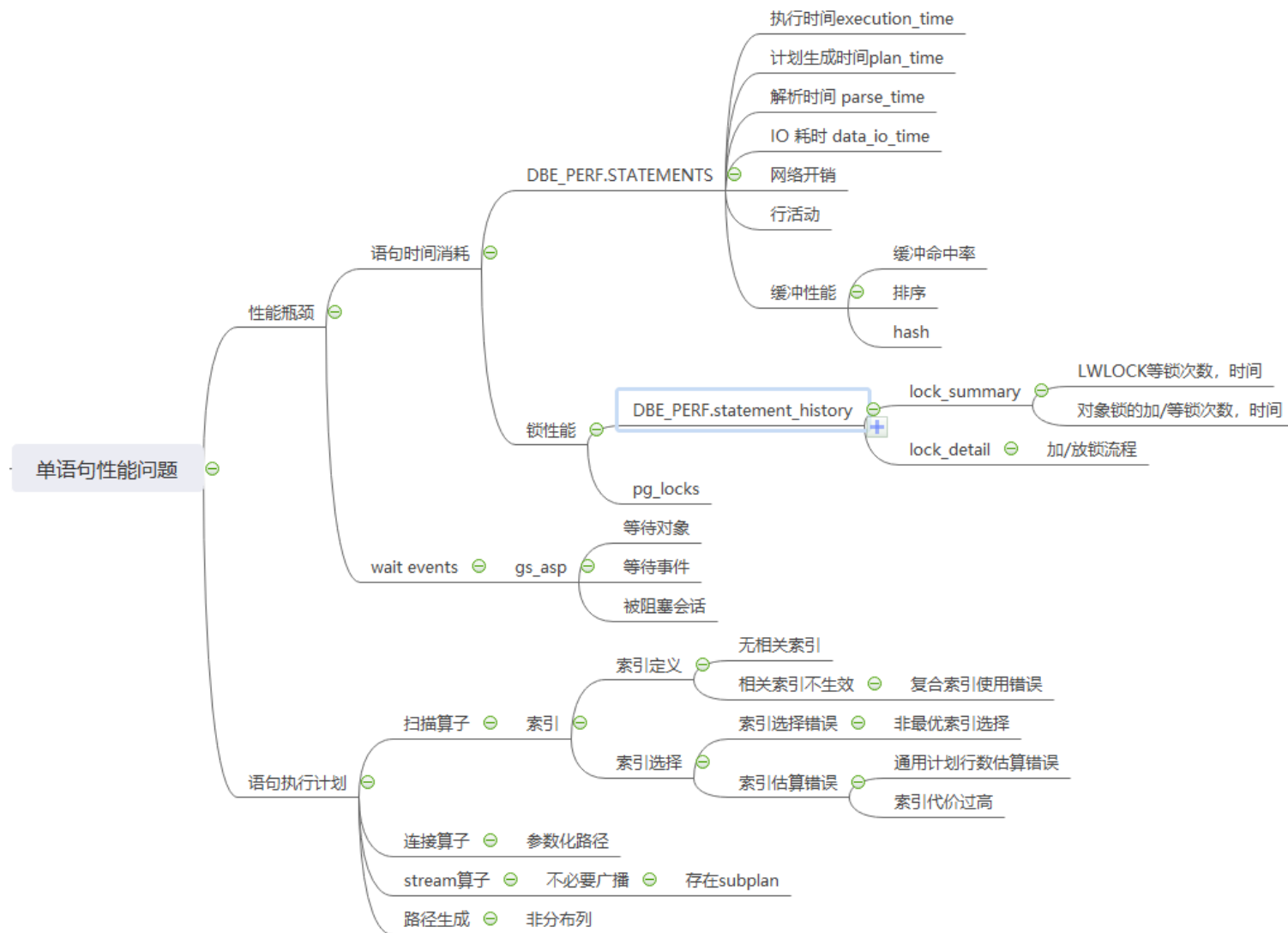


by_node.html

性能调优案例分析-整体性能问题



性能调优案例分析-单语句性能问题



性能诊断案例1-不正确的索引1

现象：业务侧监控发现某query时延达到1.2s，不满足性能要求

```
query | update t_tcp_damiao_online set likescount=$1 where damiaokey=$2 and damiaoid=$3
userconnect=# select * from db_perf.statement where unique_sql_id= 211127385;
-[ RECORD 1 ]-----+-----
node_name           | cn_5001
node_id             | 1120683504
user_name           | [REDACTED]
user_id             | 16844
unique_sql_id       | 211127385
query               | select
                        |
                        | [REDACTED]
                        |
                        | from [REDACTED] where [REDACTED]=$1
                        | order by createTime desc
                        | limit $2,$3
n_calls             | 677
min_elapse_time     | 1164992
max_elapse_time     | 1373288
total_elapse_time   | 851558721
n_returned_rows     | 3083
n_tuples_fetched    | 7817
n_tuples_returned   | 3398884220
n_tuples_inserted   | 0
n_tuples_updated    | 0
n_tuples_deleted    | 0
n_blocks_fetched    | 200909247
n_blocks_hit        | 200909247
n_soft_parse        | 611
n_hard_parse        | 67
db_time             | 1701704365
cpu_time            | 850855209
execution_time      | 849777189
parse_time          | 3546
plan_time           | 22291
rewrite_time        | 158
pl_execution_time   | 0
pl_compilation_time | 0
net_send_time       | 18550
data_io_time        | 0
```

性能诊断案例1-不正确的索引2

```

userconnect=# select reltuples ,relpages  FROM pg_class where relname ='t_ online';
 reltuples | relpages 
-----+-----
  29915242 |  1766350
(1 row)

```

```

userconnect=# \d+ [redacted]
Table [redacted]

```

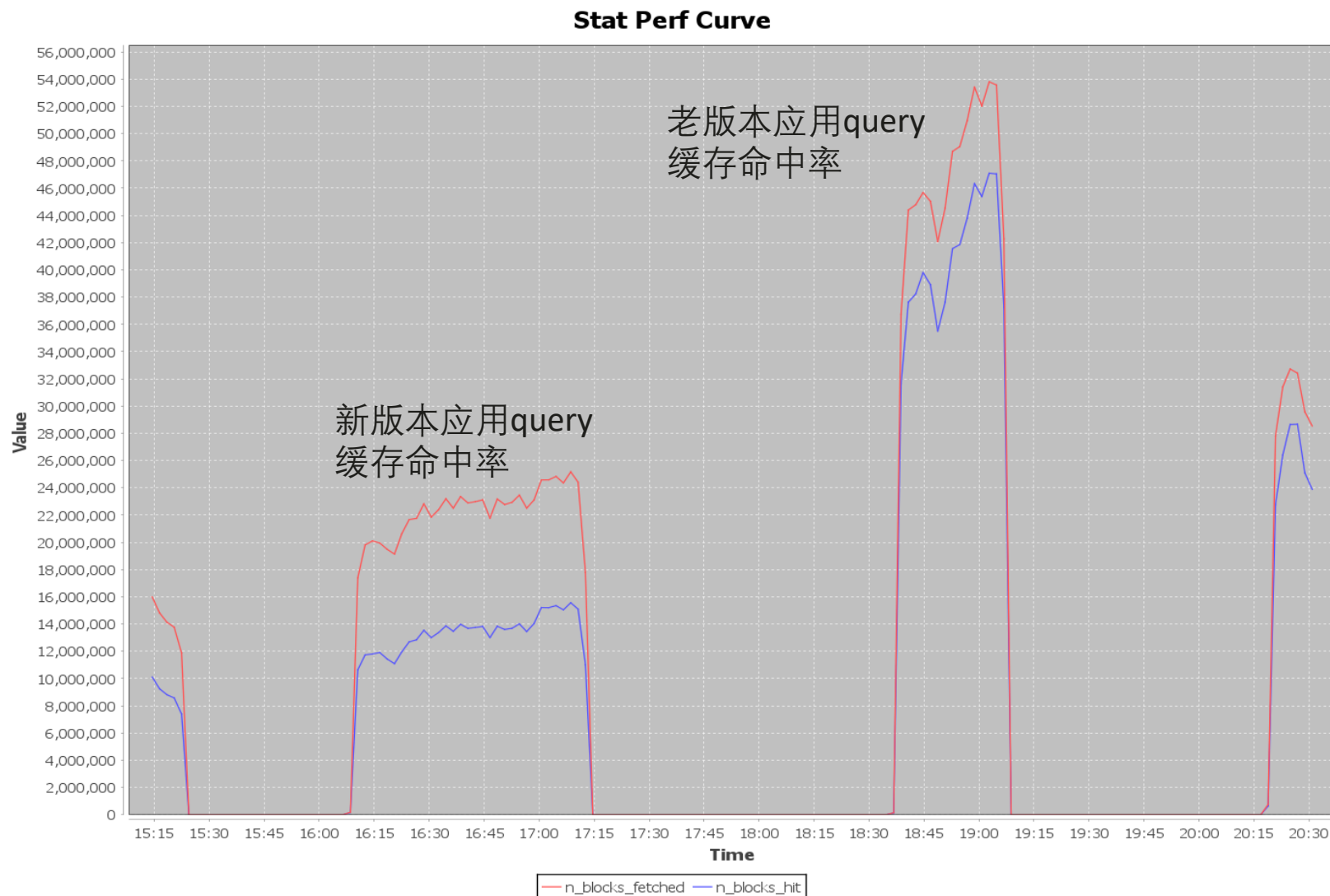
Column	Type	Modifiers	Storage	Stats target	Description
[redacted]	character varying	not null	extended		
appid	character varying	not null	extended		
hc	character varying	not null	extended		
projectid	character varying	not null	extended		
scenes	character varying	not null	extended		
contentid	character varying	not null	extended		
[redacted]	character varying	not null	extended		
[redacted]	character varying		extended		
[redacted]	character varying	not null	extended		
[redacted]	character varying		extended		
encuserid	character varying		extended		
tag	character varying	not null	extended		
timeoffset	bigint	default 0	plain		
createtime	bigint	default 0	plain		
likescount	integer	default 0	plain		
reportscount	integer	default 0	plain		
extensions	character varying		extended		
shauserid	character varying	not null	extended		

```

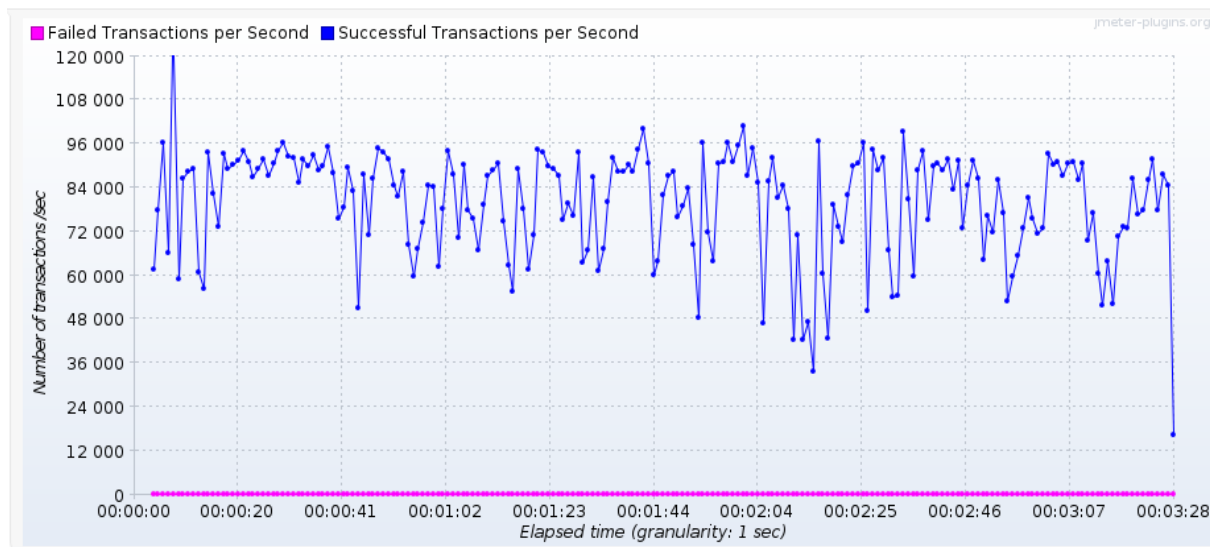
Indexes:
    "t_[redacted]_key" PRIMARY KEY, btree (d_[redacted]_id, d_[redacted]_key) TABLESPACE pg_default
    "idx_d_[redacted]_online_createtime" btree (createtime) TABLESPACE pg_default
    "idx_d_[redacted]_online_shauserid" btree (shauserid) TABLESPACE pg_default
Has OIDs: no
Distribute By: HASH(d_[redacted]_key)
Location Nodes: ALL DATANODES
Options: orientation=row, compression=no

```

性能诊断案例2-应用升级后性能劣化



性能诊断案例3-无辜的内核1



Benchmark表现抖动严重

```
postgres=# select * from db_perf.local_threadpool_status;
```

node_name	group_id	bind_numa_id	bind_cpu_number	listener	worker_info	session_info
dn_6001_6002_6003	0		0	1	default: 12 new: 0 expect: 12 actual: 12 idle: 11 pending: 0	total: 102 waiting: 0 running: 1 idle: 101
dn_6001_6002_6003	1		0	1	default: 12 new: 0 expect: 12 actual: 12 idle: 12 pending: 0	total: 100 waiting: 0 running: 0 idle: 100
dn_6001_6002_6003	2		0	1	default: 12 new: 0 expect: 12 actual: 12 idle: 12 pending: 0	total: 99 waiting: 0 running: 0 idle: 99
dn_6001_6002_6003	3		0	1	default: 12 new: 0 expect: 12 actual: 12 idle: 12 pending: 0	total: 100 waiting: 0 running: 0 idle: 100

(4 rows)

线程池表现

性能诊断案例2-无辜的内核2

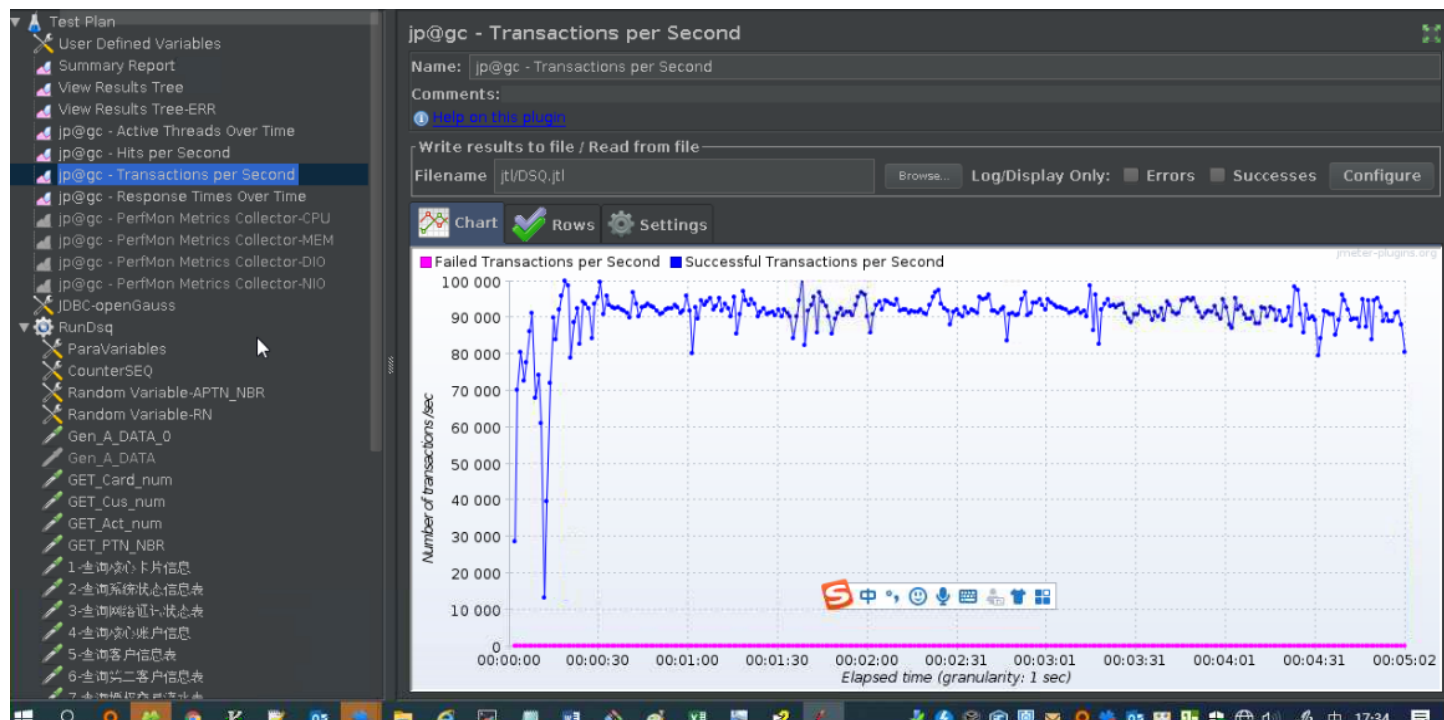
```
postgres=# select * from dbperf.instance_time
stat_id | stat_name | value
-----+-----+-----
0 | DB_TIME | 29250559668
1 | CPU_TIME | 0
2 | EXECUTION_TIME | 6565306626
3 | PARSE_TIME | 1320743
4 | PLAN_TIME | 10969750
5 | REWRITE_TIME | 442004
6 | PL_EXECUTION_TIME | 1960443
7 | PL_COMPILATION_TIME | 22866
8 | NET_SEND_TIME | 4940524962
9 | DATA_IO_TIME | 999727
(10 rows)
```

实例时间细分

```
Expanded display is on.
postgres=# select * from dbperf.statement where query like '%CSI_COR_PLT_T%'
-[ RECORD 1 ]-----+-----
node_name | dn_6001_6002_6003
node_id | 0
user_name | dsq
user_id | 16385
unique_sql_id | 170974637
query | SELECT rn FROM CSI_COR_PLT_T T where rn = $1
n_calls | 250628761
min_elapse_time | 29
max_elapse_time | 168511
total_elapse_time | 12456318451
n_returned_rows | 250628761
n_tuples_fetched | 490000
n_tuples_returned | 250873760
n_tuples_inserted | 0
n_tuples_updated | 0
n_tuples_deleted | 0
n_blocks_fetched | 754381816
n_blocks_hit | 754323904
n_soft_parse | 250623766
n_hard_parse | 5000
db_time | 19087188948
cpu_time | 0
execution_time | 4371727071
parse_time | 77649
plan_time | 2074401
rewrite_time | 12385
pl_execution_time | 0
pl_compilation_time | 0
net_send_time | 3094196779
data_io_time | 987500
```

关键SQL表现

性能诊断案例3-无辜的内核3



更新benchmark版本

性能诊断案例4-集群整体性能不达标1

```
postgres=# select wait_status, count(*) from pg_thread_wait_status group by wait_status;
wait_status | count
-----
HashAgg - build hash | 1
Sort | 1
wait cmd | 478
flush data | 1
wait node: dn_6004_6005_6006, total 1 | 29
wait node: dn_6007_6008_6009, total 1 | 414
wait node: dn_6001_6002_6003, total 1 | 29
none | 662
wait pooler get conn | 1
wait node: dn_6010_6011_6012, total 1 | 34
(10 rows)

postgres=#
```

查CN结点等待状态，
等某个DN

nodename	type	event	wait	failed_wait	total_wait_time	avg_wait_time	max_wait_time	min_wait_time
dn_6007_6008_6009	STATUS	wait cmd	579158759	0	414905071679455	716392	77970003912	1
dn_6007_6008_6009	LWLOCK_EVENT	CBMParseXlogLock	11	0	3746816	340619	1366722	11
dn_6007_6008_6009	STATUS	Sort	328	0	44349602	135212	578803	34
dn_6007_6008_6009	LWLOCK_EVENT	transactionid	561270	0	75063707447	16816	1884114	4
dn_6007_6008_6009	STATUS	wait wal sync	139124194	0	5073297545729	36465	871253	1
dn_6007_6008_6009	LWLOCK_EVENT	SinvalWriteLock	904	0	19879431	20725	105320	2
dn_6007_6008_6009	STATUS	analyze	1646	0	25748388	15643	245361	102
dn_6007_6008_6009	IO_EVENT	DoubleWriteFileWrite	240192	0	2637891372	10982	691605	1317
dn_6007_6008_6009	LWLOCK_EVENT	SyncRepLock	18649497	0	77055081774	4131	102810	1
dn_6007_6008_6009	LWLOCK_EVENT	UniqueSQLMappingLock	23	0	89138	3875	6146	735
dn_6007_6008_6009	IO_EVENT	SLRUFlushSync	148906	0	442072427	2968	531384	551
dn_6007_6008_6009	LWLOCK_EVENT	ControlFileLock	7	0	17706	2529	7107	17
dn_6007_6008_6009	IO_EVENT	ControlFileSyncUpdate	5928	0	14779673	2493	129940	1434
dn_6007_6008_6009	IO_EVENT	ReplicationsSlotSync	11844	0	28691991	2422	246934	1228

异常节点wait_events

nodename	type	event	wait	failed_wait	total_wait_time	avg_wait_time	max_wait_time	min_wait_time
dn_6001_6002_6003	STATUS	wait cmd	619224219	0	417458841590058	674164	77970003795	1
dn_6001_6002_6003	LWLOCK_EVENT	CBMParseXlogLock	15	0	4055761	270384	1366274	8
dn_6001_6002_6003	STATUS	Sort	334	0	44989622	134459	594878	37
dn_6001_6002_6003	STATUS	analyze	1640	0	22877710	14130	253008	109
dn_6001_6002_6003	IO_EVENT	DoubleWriteFileWrite	241964	0	2611152811	10791	624402	1356
dn_6001_6002_6003	LWLOCK_EVENT	InstanceTimeLock	9	0	59623	6624	50992	4
dn_6001_6002_6003	LWLOCK_EVENT	transactionid	237461	0	1555692884	6551	477742	4
dn_6001_6002_6003	LWLOCK_EVENT	ControlFileLock	11	0	61379	5579	10669	3094
dn_6001_6002_6003	LWLOCK_EVENT	SinvalWriteLock	1015	0	3945563	3887	76016	8
dn_6001_6002_6003	IO_EVENT	SLRUFlushSync	148863	0	438602746	2946	574863	535
dn_6001_6002_6003	LWLOCK_EVENT	DoubleWriteLock	7	0	10567	2795	9535	183
dn_6001_6002_6003	IO_EVENT	ControlFileSyncUpdate	5943	0	15300000	2574	434100	1260
dn_6001_6002_6003	STATUS	wait wal sync	142325202	0	357012774498	2508	453425	1
dn_6001_6002_6003	IO_EVENT	ReplicationsSlotSync	11072	0	28435005	2564	422444	1322
dn_6001_6002_6003	IO_EVENT	DisableConnectFileSync	2	0	4399	2199	2301	2098
dn_6001_6002_6003	IO_EVENT	DoubleWriteFileRead	2	0	4332	2166	3755	577
dn_6001_6002_6003	LWLOCK_EVENT	WALWriteLock	276165362	0	530341741763	1920	666160	1
dn_6001_6002_6003	LWLOCK_EVENT	SyncRepLock	18713272	0	23334904254	1246	68375	1
dn_6001_6002_6003	LWLOCK_EVENT	BufferIOLock	368	0	446662	1213	5957	10
dn_6001_6002_6003	LWLOCK_EVENT	SinvalReadLock	1224	0	1474333	1204	56655	1

正常节点wait_events

性能诊断案例4-集群整体性能不达标2

SELECT nodename AS

node_name,pg_xlog_location_diff(sender_flush_location,receiver_replay_location)

FROM global_wal_sender_status;

发现主备复制LSN此分片差异较大

avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle								
	8.56	0.00	6.39	13.10	0.00	71.95								
Device:	rrqm/s	wrqm/s	r/s	w/s	rMB/s	wMB/s	avgrq-sz	avgqu-sz	await	r_await	w_await	svctm	%util	
da	0.00	38.50	0.00	29.50	0.00	0.27	18.44	0.03	1.05	0.00	1.05	1.05	3.10	
m-0	0.00	0.00	0.00	9.00	0.00	0.04	8.00	0.02	1.78	0.00	1.78	0.22	0.20	
m-1	0.00	0.00	0.00	52.00	0.00	0.23	9.08	0.08	1.47	0.00	1.47	0.56	2.90	
m-2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
db	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
dc	62.50	1679.00	1.50	1987.50	0.50	24.95	26.21	23.61	11.86	3.33	11.87	0.48	94.70	
dd	62.50	1678.50	1.50	2036.00	0.50	25.07	25.70	5.59	2.53	2.00	2.53	0.37	76.35	
de	62.50	1675.50	1.50	2082.50	0.50	25.65	25.69	5.63	2.50	2.33	2.50	0.38	78.90	
df	62.50	1673.00	1.50	2017.50	0.50	25.37	26.24	3.13	1.55	1.67	1.55	0.08	15.15	
dg	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
dh	0.00	4.00	0.00	3.50	0.00	0.03	16.00	0.00	0.86	0.00	0.86	0.57	0.20	
m-3	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
m-4	0.00	0.00	0.00	7.00	0.00	0.03	8.00	0.01	1.00	0.00	1.00	0.29	0.20	
m-5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
m-6	0.00	0.00	256.00	13649.50	2.00	102.01	15.32	336.56	20.29	2.51	20.62	0.07	99.40	

异常分片备机IO

性能诊断案例5-ASP诊断慢查询

1. WDR 显示insert MAIL 性能差，但是执行次数不多

```
805242525 cn_5001 dams 1200258110 执行 1 执行时间 1200258110 1200258110 1200258110 0 0 1 18 0 241008 0 insert into DAMS_MAIL_TOK
```

2. cn_5001的ASP，查找unique_query_id 为805242525的session，看这个语句的开始和结束时间，等待状态

```
81070 | 2020-12-01 09:01:22.164839+08 | t | 16845 | 281459560901984 | 177135 | 2020-12-01 08:25:05.05042+08 | wait node 等待状态
81080 | 2020-12-01 09:01:37.690322+08 | t | 16845 | 281459560901984 | 177135 | 2020-12-01 08:25:05.05042+08 | wait node
81090 | 2020-12-01 09:01:54.171891+08 | t | 16845 | 281459560901984 | 177135 | 2020-12-01 08:25:05.05042+08 | wait node

81920 | 2020-12-01 09:20:44.119893+08 | t | 16845 | 281459560901984 | 177135 | 2020-12-01 08:25:05.05042+08 | wait node
81930 | 2020-12-01 09:20:57.362714+08 | t | 16845 | 281459560901984 | 177135 | 2020-12-01 08:25:05.05042+08 | wait node 等待状态
81940 | 2020-12-01 09:21:10.348382+08 | t | 16845 | 281459560901984 | 177135 | 2020-12-01 08:25:05.05042+08 | wait node
```

3. dn_6013_6014_6015_6016的ASP，查找unique_query_id 为805242525的session，看这个session的等待状态，blocking session

```
805242525 | 16846 | 1120683504 | | db68ea4:0:0:0:0:6 | ShareLock | 219203 | transactionid
805242525 | 16846 | 1120683504 | | db68ea4:0:0:0:0:6 | ShareLock lock blocking session 219203 | transactionid
805242525 | 16846 | 1120683504 | | db68ea4:0:0:0:0:6 | ShareLock | 219203 | transactionid
```

4. dn_6013_6014_6015_6016的ASP，查找session_id为219203 的session，看这个session的状态

```
76842668659766128 | 805242525 | 16846 | 1120683504 | | db68ea4:0:0:0:0:6 | ShareLock | 219203 | transactionid
148900262697948781 | 827661677 | 16846 | -1736975100 | | db68ea4:0:0:0:0:6 | ShareLock | 219203 | transactionid
0 | | | | | | | 谁也没有阻塞 | wait cmd
```

5. Query 805242525 被session 219203阻塞20分钟，在transactionid wait event 上，是死锁的表现，最后胜出的是805242525这个query，表现为这个语句的执行时间为20分钟

Thank you.

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and
organization for a fully connected,
intelligent world.

**Copyright©2018 Huawei Technologies Co., Ltd.
All Rights Reserved.**

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

