**苏州市计量测试院有限公司**

**系统卡慢事项分析**

## 1. 客户反馈信息收集

### 1.1. 基本描述

结合客户反馈的问题描述进行验证，确认卡慢现象表现：

|  |  |
| --- | --- |
| **客户名称** | 苏州市计量测试院有限公司 |
| **上线时间** | 2019年10月 |
| **问题描述** | ☑系统整体卡慢 □个别页面或者功能卡慢 |
| **主要涉及节点** | 因业务数据量非常大且个性化功能较多，系统整体运行相对较慢，主要集中在计量证书/记录、财务管理、统计报表等模块，比如证书记录编制修改、证书记录查询、数据导出、财务数据查询等等。 |

### 1.2. 记录问题发生的时间点

| **发生日期** | **发生时间点** | **持续时间** | **影响范围** | **特点** |
| --- | --- | --- | --- | --- |
| 长期 | 系统使用期间 | 持续 | 整体 | 系统整体运行慢，尤其是下午时间段（实验室人员下午集中出证较多），证书记录模块操作卡慢情况明显，对其他模块操作也有影响。 |
| 2025/8/22 | 15:33左右 | 持续 | 整体 | 证书查询节点数据导出慢。如果数据量多，则导出更慢 |
| 2025/8/25 | 当天首次登录系统 | 10-50s | 整体 | 首次登录系统之后，界面数据加载很慢，节点打不开。 |

### 1.3. 发布记录确认

| **发布日期** | **发布类型** | **发布内容** | **效果观察** |
| --- | --- | --- | --- |
| 2025/8/13 | 部署监控 | 服务器部署监控程序（涉及web服务器131/132、oracle服务器133/137），主要是用来监控服务器CPU/空间使用情况等。 | / |
| 2025/9/4 | 系统更新 | 更新了部分优化内容：  1、登记号类型转换优化 - 优化类型转换避免不走索引；  2、exchangeDao.getInfo 接口优化- 改为需要时查询数据库；  3、所有分页查询count性能优化 - 减少不必要的统计；  4、退件/延期-异常申请优化-改批量；  5、银行到款认领记录查询优化性能。 | 异常申请/银行到款认领记录查询反应时间缩短，目前查询时间在3s左右。 |

| **库名** | **委托单数量** | **登记号数量** | **样品表数量** | **登记号日志表数量** | **原始记录数量** | **原始记录数据表数量** | **原始记录日志表数量** | **证书数量** | **证书数据表数量** | **证书日志表数量** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| szjls | 1557425 | 4041938 | 13864223 | 57923222 | 4648676 | 127598744 | 24812012 | 4646700 | 84963261 | 17523108 |

## 2. 运维层排查

### 2.1. 系统信息核查

#### 2.1.1. 系统运行背景信息收集

系统上线运行时间，近期开机时长，稳定性情况

1. 系统上线时间  
   2019.10 月
2. 系统档位定级  
   M
3. 当前数据库总数据量，增长趋势
   1. 数据库总量：待补充，年增长量：待补充
   2. 核心业务表数据

| **库名** | **委托单数量** | **登记号数量** | **样品表数量** | **登记号日志表数量** | **原始记录数量** | **原始记录数据表数量** | **原始记录日志表数量** | **证书数量** | **证书数据表数量** | **证书日志表数量** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| v3\_fsjl | 118709 | 581737 | 1980627 | 14742764 | 1127644 | 37766842 | 5501248 | 1194985 | 14505088 | 5659608 |

1. 部署架构（单机/集群，负载均衡，缓存等）

Web负载两台，数据盘NFS挂载，Oracle数据库RAC集群，第三方部署维护。

#### 2.1.2. 部署资源评估和建议参照表

* 场景及分档

| **档位** | **典型机构规模** | **预估总用户/峰值在线** | **预估TPS**  **(业务)** | **日增数据**  **（行）** | **日增文件**  **（报告）** | **年度存储增量**  **（含文件）** |
| --- | --- | --- | --- | --- | --- | --- |
| S | 1-3个实验室 | 50 / 20 | 5~10 | 2万~5万 | 2~5 GB | 1~2 TB |
| M | 区/市级检测中心 | 200 / 80 | 15~40 | 10万~30万 | 10~20 GB | 5~10 TB |
| L | 市级平台+多机构 | 800 / 300 | 50~120 | 50万~150万 | 30~80 GB | 20~50 TB |
| XL | 省级平台 | 3000 / 1200 | 200~400 | 200万+ | 100~200 GB | 80~150 TB |

* 数据库推荐配置（按主库）

| **档位** | **Oracle** | **MySQL 8** | **DM8（达梦）** | **存储建议** |
| --- | --- | --- | --- | --- |
| S | 4C/16G，SSD 1×500GB | 4C/16G，SSD 1×500GB | 4C/16G，SSD 1×500GB | 本地SSD/NVMe，Raid10 |
| M | 8C/32G，SSD 1×1TB | 8C/32G，SSD 1×1TB | 8C/32G，SSD 1×1TB | 数据/日志分盘，≥20K IOPS |
| L | 16C/64G，SSD 2×2TB | 16C/64G，SSD 2×2TB | 16C/64G，SSD 2×2TB | 日志独盘，归档独盘 |
| XL | 24~~32C/128G，SSD 4×2~~4TB | 24~~32C/128G，SSD 4×2~~4TB | 24~~32C/128G，SSD 4×2~~4TB | 外接SAN/分布式存储，高带宽万兆+ |

**IOPS目标：M≥20K，L≥40K，XL≥80K；顺序写优先保证日志盘低延迟（<2ms）**

* 应用与网关（Nginx / API 网关）推荐配置

| **档位** | **Nginx/网关** | **应用节点（JVM/Tomcat/SpringBoot）** | **建议JVM参数（示例）** |
| --- | --- | --- | --- |
| S | 1台，2C/4G | 1节点，4C/8G | -Xms3g -Xmx3g -XX:+UseG1GC |
| M | 2台（HA），2C/4G/台 | 2节点，4C/8G/节点 | -Xms6g -Xmx6g G1GC，MaxGCPauseMillis=200 |
| L | 2台（HA），4C/8G/台 | 3~4节点，8C/16G/节点 | -Xms12g -Xmx12g G1GC，I/O线程池独立 |
| XL | 2台（HA），8C/16G/台 | 6~8节点，8~16C/16~32G/节点 | -Xms16g -Xmx16g G1GC，容器化/弹性HPA |

* 中间件（Redis）推荐配置

| **组件** | **S** | **M** | **L** | **XL** | **备注** |
| --- | --- | --- | --- | --- | --- |
| Redis | 单机 1×2C/4G | 主从哨兵 2×2C/4G | 哨兵集群 3×2C/8G | Cluster 分片 6+ 节点 | 用于会话、缓存、队列；RDB+AOF混合 |

#### 2.1.3. 形成《系统环境信息表》为后续排查提供基础数据

| **服务器IP** | **操作系统** | **CPU** | **内存** | **根目录** | **数据盘** |
| --- | --- | --- | --- | --- | --- |
| 10.2.0.131 | RedHat6.5 | 8核 | 32G | 50G | 13T 挂载10.2.0.135存储 |
|
|
|
| 10.2.0.132 | RedHat6.5 | 8核 | 32G | 50G | 13T 挂载10.2.0.135存储 |
|
|
| 10.2.0.135 | RedHat6.5 | 8核 | 8G | 50G | 2T+13T+14T |
| 10.2.0.136 | RedHat6.8 | 16核 | 32G | 50G | 2T |
| 10.2.0.137 | RedHat6.8 | 16核 | 32G | 50G | 2T |
| 10.2.0.138 | RedHat6.5 | 4核 | 8G | 50G | 42G |
|
| 10.2.0.155 | RedHat6.5 | 4核 | 4G | 50G | 200G |
|
|
| 10.2.0.171 | RedHat6.5 | 8核 | 16G | 50G | 238G |
|
| 10.2.0.195 | RedHat6.5 | 8核 | 64G | 50G | 4T |
| 10.2.0.220 | RedHat6.5 | 8核 | 16G | 50G | 500G |
|
|
|

### 2.2. 基础资源排查

#### 2.2.1. Linux 命令检查

| **基础资源** | **诊断命令** | **异常指标** | **实例名** | **诊断结论** |
| --- | --- | --- | --- | --- |
| 磁盘 | iostat -x 1 5 查看磁盘 IO 使用率  df -h 查看磁盘空间  sar -d 1 5 查看磁盘吞吐  iotop 实时监控进程 IO | util > 80% 长时间居高不下  I/O 等待时间（await）过高 > 50ms  磁盘空间使用率 > 85%  单进程占用 IO 异常 | 数据库 | 正常 |
| CPU | top/htop 查看整体负载  mpstat -P ALL 1 查看每核利用率  sar -u 1 5 查看 CPU 使用率  `ps -eo pid,ppid,cmd,%cpu --sort=-%cpu | CPU 使用率长期 > 85%  单核 CPU 占用过高（热点线程）  系统负载（load average） > CPU 核数 2 倍  中断或上下文切换过多 | 数据库 | 正常 |
| 内存 | free -m 查看内存使用  vmstat 1 5 查看内存与 swap 使用  sar -r 1 5 查看内存指标  `ps -eo pid,ppid,cmd,%mem --sort=-%mem | 可用内存（available）< 10%  swap 使用频繁，si/so 值持续增加  单进程内存占用过高  内存泄漏导致进程常驻高占用 | 应用服务器 | 正常 |
| 网络 | sar -n DEV 1 5 查看网卡收发流量  iftop实时监控主机网络连接带宽  ss -s/ netstat -s查看连接状态和错误统计  ethtool eth0 查看网卡速率、丢包、错误  ping测试延时和丢包  mtr结合路由跟踪和丢包检测 | 单网卡带宽长期接近满载（>80%）  丢包率 > 1%  网络延迟 RTT > 100ms 且抖动大错误包、重传率持续升高  大量 TIME\_WAIT 或 CLOSE\_WAIT 连接 | - | 正常 |

#### 2.2.2. 监控工具排查

部署prometh监控，因漏洞问题已关闭，暂无

### 2.3. 结论

**服务器资源无问题，需要研发层定位排查**

## 3. 开发层排查

### 3.1. 服务运行参数核对

#### 3.1.1. 数据库

##### 3.1.1.1. Oracle 11g

内存相关

* sga\_target：当前 10G；建议 ≥ 20–40% 热数据量（本项目 20G+） 20G-24G
* sga\_max\_size：等于sga\_target，保证弹性
* pga\_aggregate\_target：当前 8G
* shared\_pool\_size：0
* db\_cache\_size：自动

CPU 相关

* parallel\_max\_servers：当前640,
* parallel\_degree\_policy：MANUAL

I/O 相关

* filesystemio\_options=none
* disk\_asynch\_io=TRUE
* db\_writer\_processes=2
* sessions= 1536 ； 最大连接数设置，根据并发量及核心业务的平均 tps 进行评估设置

监控

-- 查看Oracle最大连接数（sessions）

select value from v$parameter where name='sessions';

-- 查看当前连接数

select count(\*) from v$session;

-- 按用户分组查看连接数（应用连接池用户）

select username, count(\*) from v$session where username is not null group by username;

-- 查看等待连接的请求数

select \* from v$session\_wait where wait\_class != 'Idle';

-- 查看正在执行的慢SQL

SELECT b.sid oracleID,

b.username 用户名,

b.serial#,

paddr,

sql\_text 正在执行的SQL,

b.machine 计算机名称,

b.PREV\_EXEC\_START,

'alter system kill session ''' || b.sid||','||b.serial#||''';'

FROM v$process a, v$session b, v$sqlarea c

WHERE a.addr = b.paddr

AND b.sql\_hash\_value = c.hash\_value

and b.status = 'ACTIVE'

order by b.PREV\_EXEC\_START;

-- 根据sql\_id查看具体SQL

select sql\_text from v$sql where sql\_id='xxx';

-- 查看锁等待（是否有长时间未释放的锁）

select \* from v$lock where block=1;

###### 结论

建议开启异步IO

##### 3.1.1.2. MySql 8

不涉及

##### 3.1.1.3. 达梦( DM 8 )

不涉及

##### 3.1.1.4. Nginx服务

* Worker/连接
  + worker\_processes auto
  + worker\_rlimit\_nofile 200000（配合系统的 ulimit 文件句柄设置）
  + events { use epoll; worker\_connections 65535; multi\_accept on; }
* HTTP 核心
  + sendfile on; tcp\_nopush on; tcp\_nodelay on;
  + keepalive\_timeout 65; keepalive\_requests 10000;
  + client\_max\_body\_size 50m（按上传需求）
  + gzip on; gzip\_types text/\* application/json application/javascript
* 反向代理
  + proxy\_connect\_timeout 5s; proxy\_read\_timeout 60s; proxy\_send\_timeout 60s;
  + proxy\_buffers 16 64k; proxy\_busy\_buffers\_size 128k; proxy\_buffering on;
  + upstream keepalive 100–1000

###### 结论

正常

##### 3.1.1.5. 应用服务（Tomcat）

* JVM 参数设置

ava.vendor = Oracle Corporation

sun.java.launcher = SUN\_STANDARD

catalina.base = /njmd/tomcat-back

sun.management.compiler = HotSpot 64-Bit Tiered Compilers

sun.nio.ch.bugLevel =

catalina.useNaming = true

os.name = Linux

sun.boot.class.path = /usr/java/jdk1.8.0\_181/jre/lib/resources.jar:/usr/java/jdk1.8.0\_181/jre/lib/rt.jar:/usr/java/jdk1.8.0\_181/jre/lib/sunrsasign.jar:/usr/java/jdk1.8.0\_181/jre/lib/jsse.jar:/usr/java/jdk1.8.0\_181/jre/lib/jce.jar:/usr/java/jdk1.8.0\_181/jre/lib/charsets.jar:/usr/java/jdk1.8.0\_181/jre/lib/jfr.jar:/usr/java/jdk1.8.0\_181/jre/classes

java.util.logging.config.file = /njmd/tomcat-back/conf/logging.properties

java.vm.specification.vendor = Oracle Corporation

java.runtime.version = 1.8.0\_181-b13

user.name = root

tomcat.util.scan.StandardJarScanFilter.jarsToScan = log4j-taglib\*.jar,log4j-web\*.jar,log4javascript\*.jar,slf4j-taglib\*.jar

shared.loader =

tomcat.util.buf.StringCache.byte.enabled = true

user.language = en

java.naming.factory.initial = org.apache.naming.java.javaURLContextFactory

sun.boot.library.path = /usr/java/jdk1.8.0\_181/jre/lib/amd64

PID = 14523

jdk.tls.ephemeralDHKeySize = 2048

java.version = 1.8.0\_181

java.util.logging.manager = org.apache.juli.ClassLoaderLogManager

user.timezone = Asia/Shanghai

sun.arch.data.model = 64

java.util.concurrent.ForkJoinPool.common.threadFactory = org.apache.catalina.startup.SafeForkJoinWorkerThreadFactory

java.endorsed.dirs = /usr/java/jdk1.8.0\_181/jre/lib/endorsed

sun.cpu.isalist =

sun.jnu.encoding = UTF-8

file.encoding.pkg = sun.io

package.access = sun.,org.apache.catalina.,org.apache.coyote.,org.apache.jasper.,org.apache.tomcat.

file.separator = /

java.specification.name = Java Platform API Specification

java.class.version = 52.0

user.country = US

LOG\_FILE = /njmd/tomcat-back/logs/njmind-base.log

java.home = /usr/java/jdk1.8.0\_181/jre

java.vm.info = mixed mode

os.version = 2.6.32-431.el6.x86\_64

path.separator = :

java.vm.version = 25.181-b13

java.protocol.handler.pkgs = org.apache.catalina.webresources

java.awt.printerjob = sun.print.PSPrinterJob

sun.io.unicode.encoding = UnicodeLittle

awt.toolkit = sun.awt.X11.XToolkit

package.definition = sun.,java.,org.apache.catalina.,org.apache.coyote.,org.apache.jasper.,org.apache.naming.,org.apache.tomcat.

java.naming.factory.url.pkgs = org.apache.naming

user.home = /root

org.apache.catalina.security.SecurityListener.UMASK = 0027

java.specification.vendor = Oracle Corporation

tomcat.util.scan.StandardJarScanFilter.jarsToSkip = annotations-api.jar,ant-junit\*.jar,ant-launcher.jar,ant.jar,asm-\*.jar,aspectj\*.jar,bootstrap.jar,catalina-ant.jar,catalina-ha.jar,catalina-jmx-remote.jar,catalina-storeconfig.jar,catalina-tribes.jar,catalina-ws.jar,catalina.jar,cglib-\*.jar,cobertura-\*.jar,commons-beanutils\*.jar,commons-codec\*.jar,commons-collections\*.jar,commons-daemon.jar,commons-dbcp\*.jar,commons-digester\*.jar,commons-fileupload\*.jar,commons-httpclient\*.jar,commons-io\*.jar,commons-lang\*.jar,commons-logging\*.jar,commons-math\*.jar,commons-pool\*.jar,dom4j-\*.jar,easymock-\*.jar,ecj-\*.jar,el-api.jar,geronimo-spec-jaxrpc\*.jar,h2\*.jar,hamcrest-\*.jar,hibernate\*.jar,httpclient\*.jar,icu4j-\*.jar,jasper-el.jar,jasper.jar,jaspic-api.jar,jaxb-\*.jar,jaxen-\*.jar,jdom-\*.jar,jetty-\*.jar,jmx-tools.jar,jmx.jar,jsp-api.jar,jstl.jar,jta\*.jar,junit-\*.jar,junit.jar,log4j\*.jar,mail\*.jar,objenesis-\*.jar,oraclepki.jar,oro-\*.jar,servlet-api-\*.jar,servlet-api.jar,slf4j\*.jar,taglibs-standard-spec-\*.jar,tagsoup-\*.jar,tomcat-api.jar,tomcat-coyote.jar,tomcat-dbcp.jar,tomcat-i18n-\*.jar,tomcat-jdbc.jar,tomcat-jni.jar,tomcat-juli-adapters.jar,tomcat-juli.jar,tomcat-util-scan.jar,tomcat-util.jar,tomcat-websocket.jar,tools.jar,websocket-api.jar,wsdl4j\*.jar,xercesImpl.jar,xml-apis.jar,xmlParserAPIs-\*.jar,xmlParserAPIs.jar,xom-\*.jar

java.library.path = /usr/java/packages/lib/amd64:/usr/lib64:/lib64:/lib:/usr/lib

java.vendor.url = http://java.oracle.com/

spring.beaninfo.ignore = true

java.vm.vendor = Oracle Corporation

common.loader = "${catalina.base}/lib","${catalina.base}/lib/\*.jar","${catalina.home}/lib","${catalina.home}/lib/\*.jar"

java.runtime.name = Java(TM) SE Runtime Environment

sun.java.command = org.apache.catalina.startup.Bootstrap start

java.class.path = /njmd/tomcat-back/bin/bootstrap.jar:/njmd/tomcat-back/bin/tomcat-juli.jar

java.vm.specification.name = Java Virtual Machine Specification

java.vm.specification.version = 1.8

catalina.home = /njmd/tomcat-back

sun.cpu.endian = little

sun.os.patch.level = unknown

java.awt.headless = true

java.io.tmpdir = /njmd/tomcat-back/temp

java.vendor.url.bug = http://bugreport.sun.com/bugreport/

server.loader =

os.arch = amd64

java.awt.graphicsenv = sun.awt.X11GraphicsEnvironment

java.ext.dirs = /usr/java/jdk1.8.0\_181/jre/lib/ext:/usr/java/packages/lib/ext

user.dir = /njmd/tomcat-back/bin

line.separator =

java.vm.name = Java HotSpot(TM) 64-Bit Server VM

ignore.endorsed.dirs =

file.encoding = UTF-8

java.specification.version = 1.8

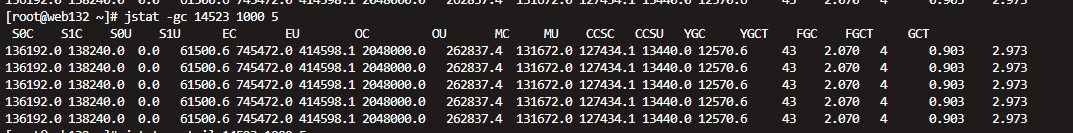
VM Flags:

Non-default VM flags: -XX:CICompilerCount=4 -XX:InitialHeapSize=3145728000 -XX:MaxHeapSize=3145728000 -XX:MaxNewSize=1048576000 -XX:MinHeapDeltaBytes=524288 -XX:NewSize=1048576000 -XX:OldSize=2097152000 -XX:+UseCompressedClassPointers -XX:+UseCompressedOops -XX:+UseFastUnorderedTimeStamps -XX:+UseParallelGC

Command line: -Djava.util.logging.config.file=/njmd/tomcat-back/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djdk.tls.ephemeralDHKeySize=2048 -Djava.protocol.handler.pkgs=org.apache.catalina.webresources -Dorg.apache.catalina.security.SecurityListener.UMASK=0027 -Dfile.encoding=UTF-8 -Xms3000m -Xmx3000m -Xmn1000m -Dignore.endorsed.dirs= -Dcatalina.base=/njmd/tomcat-back -Dcatalina.home=/njmd/tomcat-back -Djava.io.tmpdir=/njmd/tomcat-back/temp

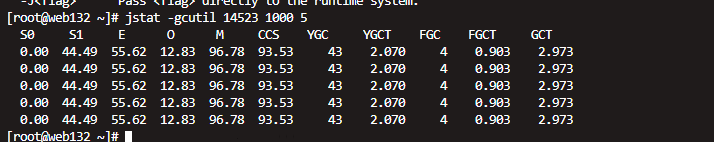
* JVM 内存监控

[root@web132 ~]# jstat -gc 14523 1000 5

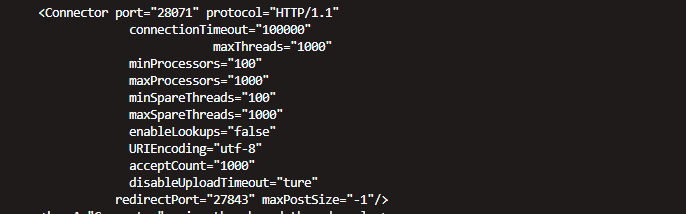


# 查看GC使用率

jstat -gcutil <pid> 1000 5

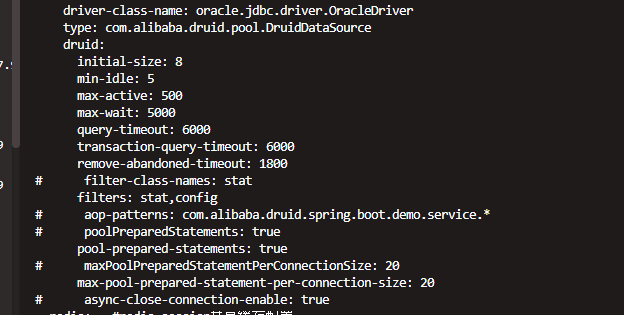


* Tomcat Connector



* JDBC 连接池

不同的连接池中间件对应的参数略有差异，自行调整



###### 结论

**数据库连接、jvm参数、gc均正常**

##### 3.1.1.6. 缓存服务器（Redis）

基本配置

#

# works better.

repl-diskless-sync no

repl-disable-tcp-nodelay no

# By default the priority is 100.

replica-priority 100

# requirepass foobared

requirepass B!W#^x3V&jCR

lazyfree-lazy-eviction no

lazyfree-lazy-expire no

lazyfree-lazy-server-del no

replica-lazy-flush no

appendonly no

# The name of the append only file (default: "appendonly.aof")

appendfilename "appendonly.aof"

# appendfsync always

appendfsync everysec

# appendfsync no

no-appendfsync-on-rewrite no

auto-aof-rewrite-percentage 100

auto-aof-rewrite-min-size 64mb

aof-load-truncated yes

aof-use-rdb-preamble yes

lua-time-limit 5000

slowlog-log-slower-than 10000

slowlog-max-len 128

latency-monitor-threshold 0

notify-keyspace-events ""

hash-max-ziplist-entries 512

hash-max-ziplist-value 64

list-max-ziplist-size -2

list-compress-depth 0

set-max-intset-entries 512

zset-max-ziplist-entries 128

zset-max-ziplist-value 64

hll-sparse-max-bytes 3000

stream-node-max-bytes 4096

stream-node-max-entries 100

activerehashing yes

# Both the hard or the soft limit can be disabled by setting them to zero.

client-output-buffer-limit normal 0 0 0

client-output-buffer-limit replica 256mb 64mb 60

client-output-buffer-limit pubsub 32mb 8mb 60

# proto-max-bulk-len 512mb

hz 10

dynamic-hz yes

aof-rewrite-incremental-fsync yes

rdb-save-incremental-fsync yes

* 内存与策略

maxmemory=8c–32g

maxmemory-policy=allkeys-lru（常用）

hash-max-ziplist-entries 512

hash-max-ziplist-value 64

* 持久化

未开启持久化

* 性能

io-threads 4

###### 结论

**未配置持久化，但是与系统卡慢无关**

### 3.2. 监控及日志分析

#### 3.2.1. Nginx 日志分析

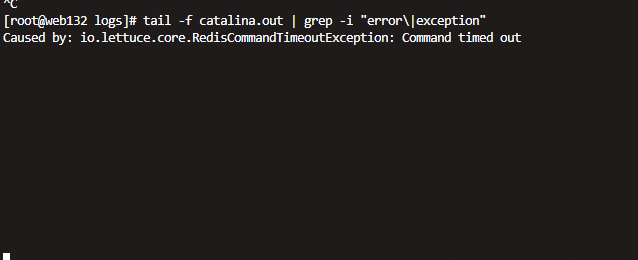
* 不涉及

#### 3.2.2. 应用日志分析

* Tomcat 日志

# 查看应用错误日志（是否有频繁的超时、异常）

tail -f $CATALINA\_HOME/logs/catalina.out | grep -i "error\|exception"



# 查看访问日志（记录响应时间>5秒的请求）

grep -E " [5-9][0-9]{2,} " $CATALINA\_HOME/logs/access.log # 响应时间（毫秒）

不涉及

##### 结论

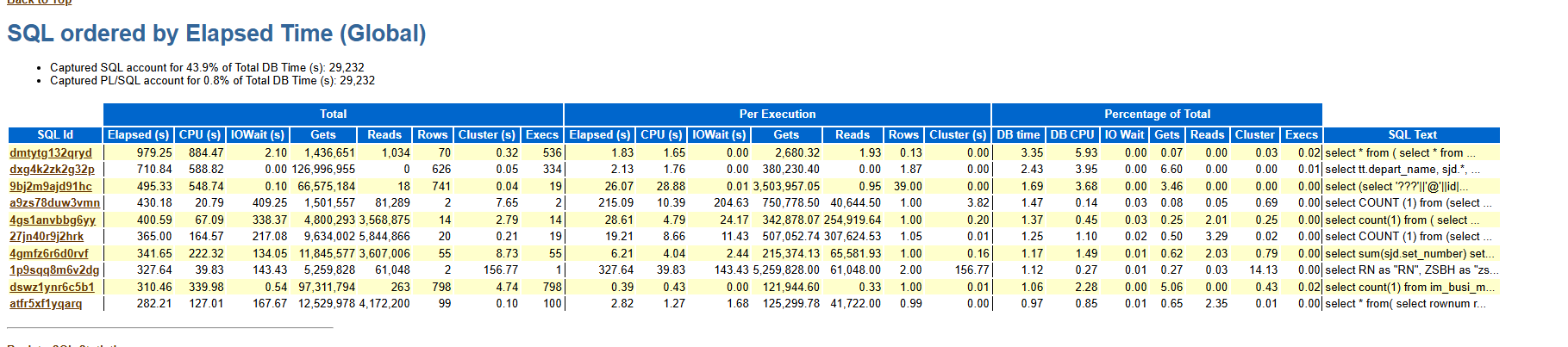
**表现正常**

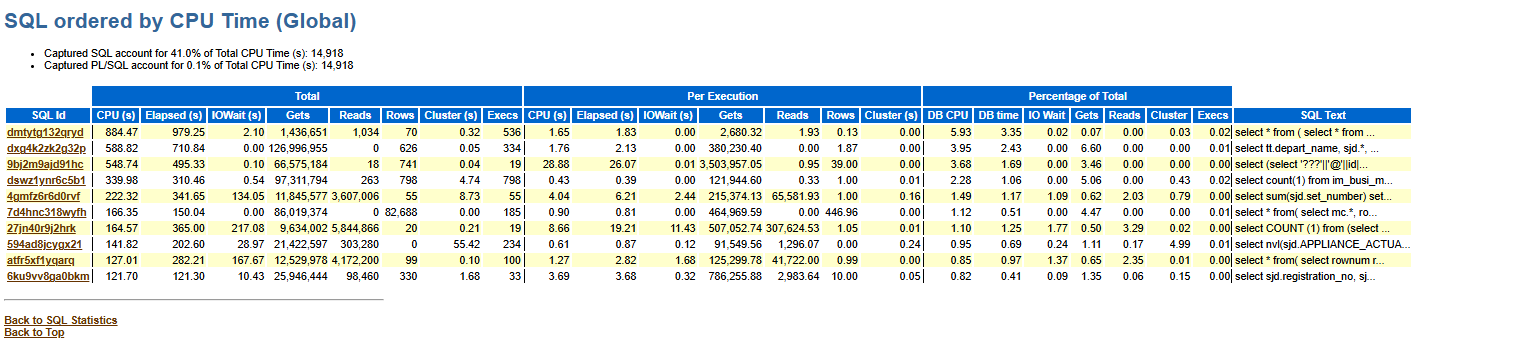
#### 3.2.3. 数据库慢日志分析

* Oracle
  + AWR报告分析

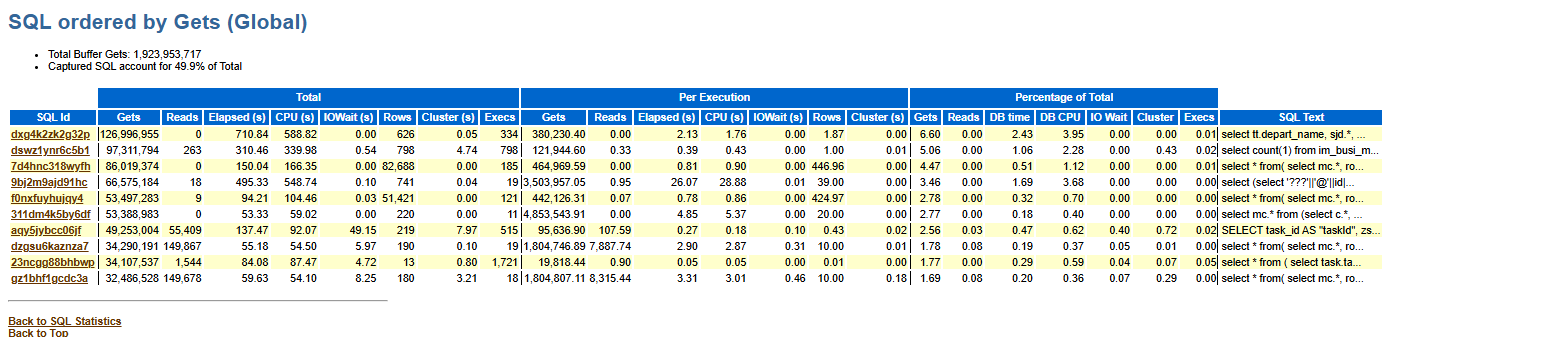
@?/rdbms/admin/awrrpt.sql

* + 输出报告内容里关注：
    - SQL ordered by Elapsed Time（按耗时排序的 SQL）。

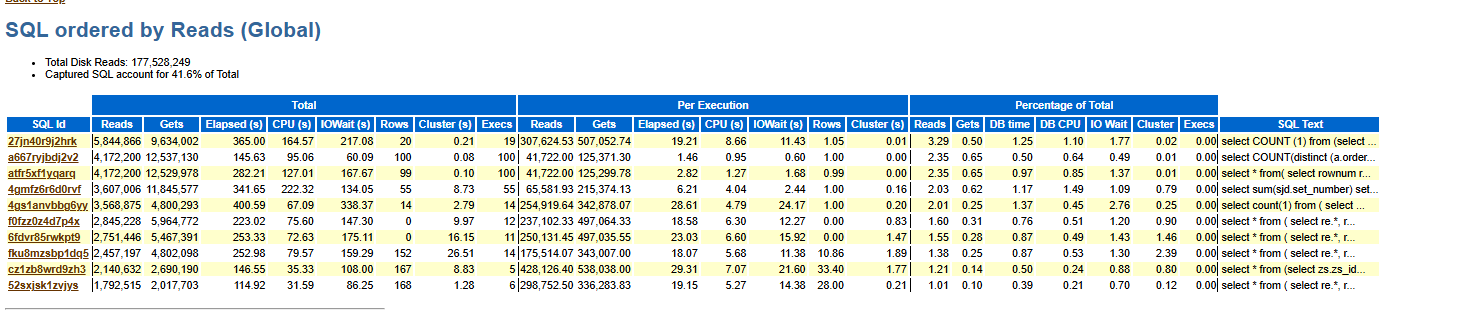




* + - SQL ordered by Buffer Gets（逻辑读最多的 SQL）。



* + - SQL ordered by Physical Reads（物理读最多的 SQL）。



* + ASH 实时会话分析

-- 前卡慢 SQL 的执行情况

SELECT sample\_time,

sql\_id,

session\_id,

session\_state,

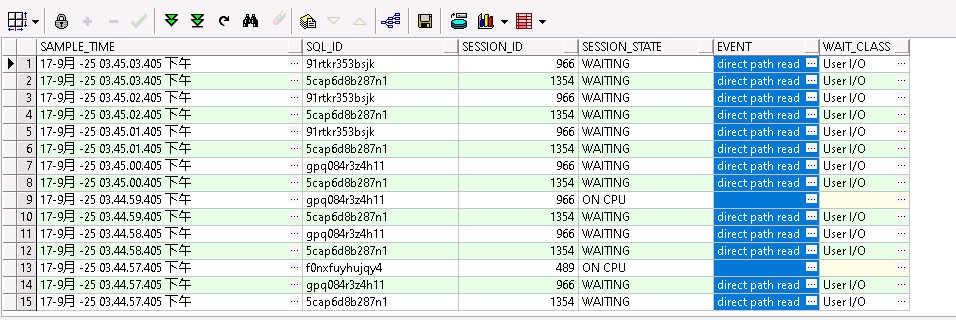
event,

wait\_class

FROM v$active\_session\_history

WHERE sample\_time > SYSDATE - 1/24/60\*5 -- 最近5分钟

ORDER BY sample\_time DESC;

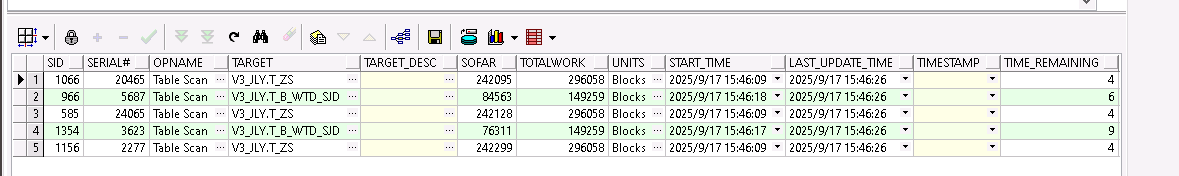


-- 查看长时间执行的操作（大表扫描、索引重建等）。

SELECT opname, start\_time, elapsed\_seconds, time\_remaining, sofar, totalwork

FROM v$session\_longops

WHERE totalwork != 0 AND sofar != totalwork;



##### 结论

**根据awr分析，可看出主要的长时间操作集中在大表（证书表和委托单表）、以及其他**

**（IM数据量增大、ecqs公用数据库查询证书表）有关**

#### 3.2.4. Redis 日志分析

##### 结论

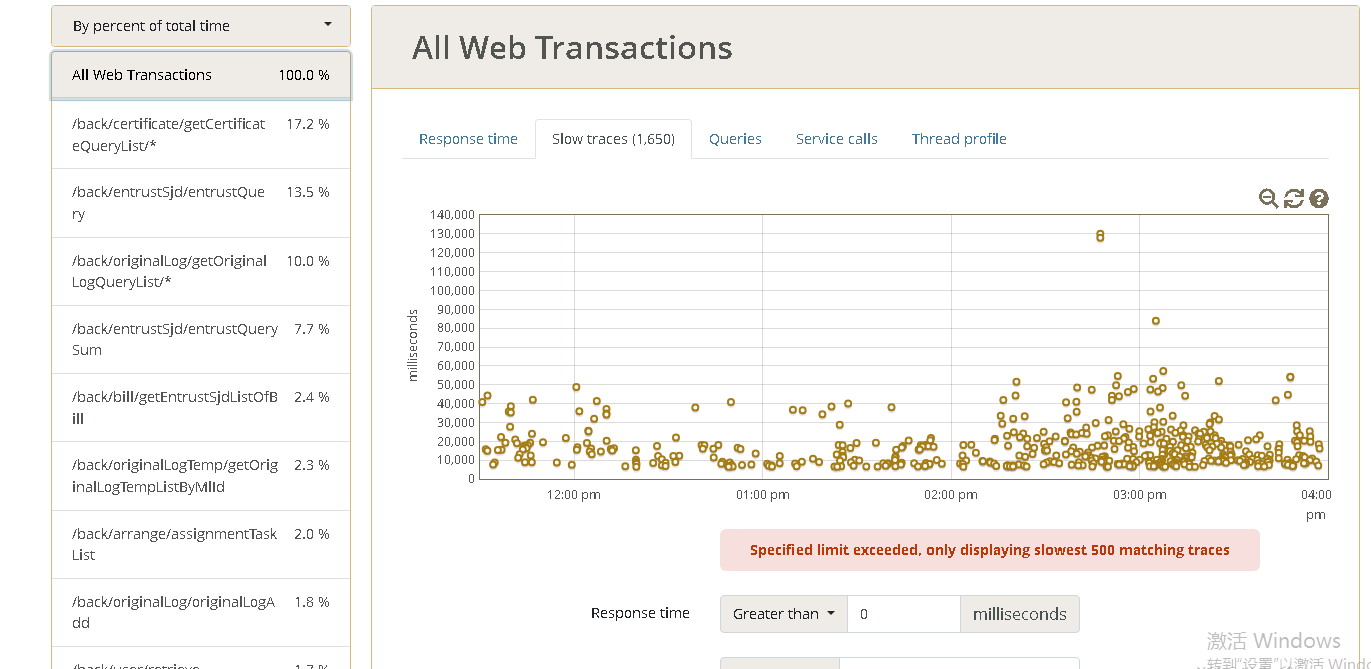
未涉及慢日志，无需关注

#### 3.2.5. 探针监控

* 探针工具

1. Arthas（阿里开源）
   1. 命令行诊断工具。
   2. 支持查看类加载、JVM、方法调用栈、热更新、trace 方法耗时。
   3. 非常适合线上问题定位。
2. SkyWalking 探针
   1. APM 工具，支持 Java 探针自动植入。
   2. 无需改代码，采集链路追踪、性能指标。
   3. 常用于分布式系统监控。
3. Glowroot
   1. 开源 APM，支持 JVM 指标、SQL 性能监控。
   2. 探针安装简单，适合中小规模系统

* Glowroot



涉及SQL

证书查询

select ZS\_ID as "zsId",RN as "RN",ZSBH as "zsbh",ZSQJMC as "zsqjmc",ZSDWMC as "zsdwmc",CONTACTER as "contacter",APPLIANCE\_ACTUAL\_NAME as "applianceActualName",COMPANY\_NAME as "companyName",XHGG as "xhgg",CCBH as "ccbh",SBBH as "sbbh",ZZCS as "zzcs",ENTRUST\_DATE as "entrustDate",ZSJDRQ as "zsjdrq",ZSYXRQ as "zsyxrq",ZS\_TYPE as "zsType",JDJL as "jdjl",DEPART\_ID as "departId",DEPART\_NAME as "departName",POST\_NAME as "postName",ADDUSERID as "adduserid",ADDUSER as "adduser",ADD\_TIME as "addTime",PRINTUSER as "printuser",cma\_flag as "cmaFlag",PRINT\_TIME as "printTime",XLHG\_FLAG as "xlhgFlag",INTEGRATED\_FLAG as "integratedFlag",CNAS\_FLAG as "cnasFlag",REGISTRATION\_NO as "registrationNo",ORDER\_NO as "orderNo",ORDER\_SINGLE\_NO as "orderSingleNo",QJ\_FLAG as "qjFlag",QJYT as "qjyt",JDYNAME as "jdyname",HYYNAME as "hyyname",PZR\_NAME as "pzrName",JDF as "jdf",FJF as "fjf",XLF as "xlf",JJF as "jjf",FWF as "fwf",QTF as "qtf",WORKLOAD as "workload",intensiveWorkload as "intensiveWorkload",JSGF as "jsgf",JSGC as "jsgc",JDWD as "JDWD",JDSD as "jdsd",XZQH as "xzqh",CHARGE\_ADD\_TIME as "chargeAddTime",MB\_ID as "mbId",IS\_SETTLEMENT as "isSettlement",ZSQFRQ as "zsqfrq",IS\_SETTLE\_ACCOUNT as "isSettleAccount",IS\_AUTHORISE as "isAuthorise",APPLIANCE\_NAME as "applianceName",FEE\_REMARK as "feeRemark",COMPANY\_TYPE as "companyType",IS\_COMPULSORY\_VERIFY as "isCompulsoryVerify",APPLIANCE\_USE as "applianceUse",APPLIANCE\_WAREHOUSE\_FLAG as "applianceWarehouseFlag",FEE\_STATUS as "feeStatus",ISYSJL as "isYsjl",CERTIFICATE\_WAREHOUSE\_FLAG as "certificateWarehouseFlag",TASK\_SOURCE as "taskSource",JDDD as "jddd" from ( select zs.zs\_id, /\*证书ID\*/

replace(province.DIVISION\_NAME||'\'||city.DIVISION\_NAME||'\'||area.DIVISION\_NAME,'\\','') as xzqh,

to\_char(charge.add\_time, 'yyyy-mm-dd') charge\_add\_time,

charge.FEE\_REMARK,

zs.zsbh, /\*证书编号\*/

zs.zsdwmc, /\*证书单位\*/

zs.jddd2 || ' ' || zs.jddd as jddd,/\*证书检定地点\*/

to\_char(wtd.ENTRUST\_DATE, 'yyyy-mm-dd') ENTRUST\_DATE, /\*委托日期\*/

zs.xhgg, /\*型号规格\*/

zs.ccbh, /\*出厂编号\*/

zs.sbbh, /\*设备编号\*/

zs.zzcs, /\*制造厂商\*/

wtd.ORDER\_NO, /\*委托单号\*/

wtd.task\_source,/\*任务来源\*/

wtd.contacter,/\*联系人\*/

sjd.ORDER\_SINGLE\_NO, /\*流转单号\*/

to\_char(zs.ZSJDRQ, 'yyyy-mm-dd') ZSJDRQ, /\*检定日期\*/

to\_char(zs.zsyxrq, 'yyyy-mm-dd') zsyxrq, /\*有效日期\*/

to\_char(zs.zsqfrq, 'yyyy-mm-dd') zsqfrq, /\*签发日期\*/

zs.jdjl, /\*检定结论\*/

zs.pzr\_name, /\*批准人\*/

depart.id depart\_id, /\*检测部门ID\*/

depart.depart\_name, /\*检测部门\*/

post.post\_name, /\*岗位\*/

adduser.user\_name adduser, /\*添加人\*/

printuser.user\_name printuser, /\*打印人\*/

zs.add\_user as addUserId ,/\*添加人ID\*/

to\_char(zs.print\_time, 'yyyy-mm-dd') print\_time, /\*打印日期\*/

zs.jdwd, /\*温度\*/

zs.jdsd, /\*湿度\*/

(select WMSYS.WM\_CONCAT(data.data\_value1) jdyname

from t\_zs\_data data

where data.zs\_id = zs.zs\_id and data.data\_type = 'jdy') jdyname, /\*检定员\*/

(select WMSYS.WM\_CONCAT(data.data\_value1) hyyname

from t\_zs\_data data

where data.zs\_id = zs.zs\_id and data.data\_type = 'hyy') hyyname, /\*核验员\*/

wtd.company\_name, /\*委托单位\*/

zs.zsqjmc, /\*报告器具名称\*/

/\*sjd.fee\_status,\*/ /\*是否收费\*/

decode(sjd.fee\_status,1,'已录费',0,'未录费') as fee\_status, /\*是否收费\*/

/\*sjd.is\_settlement,\*/ /\*是否结算\*/

decode(sjd.is\_settlement,1,'已结算',0,'未结算') as is\_settlement, /\*是否结算\*/

/\*sjd.is\_settle\_account,\*/ /\*是否到账\*/

decode(sjd.is\_settle\_account,1,'已结账',0,'未结账') as is\_settle\_account, /\*是否到账\*/

/\*zs.is\_authorise,\*/ /\*SC0/SC1\*/

decode(zs.is\_authorise,0,'授权',1,'未授权') as is\_authorise, /\*SC0/SC1\*/

/\*zs.Cnas\_Flag ,\*/ /\*是否CNAS\*/

decode(zs.Cnas\_Flag,0,'否',1,'是') as Cnas\_Flag, /\*是否CNAS\*/

sjd.appliance\_name, /\*委托单器具名称\*/

sjd.appliance\_actual\_name, /\*委托单器具实际名称\*/

/\*sjd.Is\_Compulsory\_Verify,\*/ /\*委托单是否强检\*/

decode(sjd.Is\_Compulsory\_Verify,0,'否',1,'是') as Is\_Compulsory\_Verify, /\*委托单是否强检\*/

sjd.Appliance\_Use, /\*委托单器具用途\*/

/\*zs.Qj\_Flag,\*//\*证书是否强检\*/

decode(zs.Qj\_Flag,0,'否',1,'是') as Qj\_Flag, /\*证书是否强检\*/

zs.Qjyt, /\*证书器具用途\*/

/\*sjd.Appliance\_Warehouse\_Flag,\*/ /\*器具库房状态\*/

decode(sjd.Appliance\_Warehouse\_Flag,0,'未入库',3,'已入库',4,'已出库') as Appliance\_Warehouse\_Flag, /\*器具库房状态\*/

/\*sjd.Certificate\_Warehouse\_Flag,\*/ /\*证书库房状态\*/

decode(sjd.Certificate\_Warehouse\_Flag,0,'未入库',3,'已入库',4,'已出库') as Certificate\_Warehouse\_Flag, /\*证书库房状态\*/

/\*comp.Company\_Type,\*/ /\*单位类别\*/

decode(comp.Company\_Type,1,'第一类别',2,'第二类别',3,'第三类别',4,'第四类别',5,'第五类别',6,'第六类别') as Company\_Type, /\*单位类别\*/

decode(ZS.YSJL\_ID, null, 0, 1) as isYsjl, /\*是否原始记录生成\*/

zs.ZS\_TYPE, /\*证书类型\*/

to\_char(zs.ADD\_TIME, 'yyyy-mm-dd') ADD\_TIME, /\*添加日期\*/

zs.registration\_no /\*登记号\*/,

zs.mb\_id, /\*模板id 历史证书用\*/

decode(sjd.integrated\_flag, 0, '否', 1, '是') integrated\_flag, /\*是否综合类仪器\*/

decode(zs.XLHG\_FLAG, 0, '否', 1, '是') XLHG\_FLAG, /\*是否调修后合格\*/

round(jdf\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as JDF, /\* 检测费 \*/

round(fjf\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as FJF, /\* 附加费 \*/

round(xlf\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as XLF, /\* 修理费 \*/

round(jjf\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as JJF, /\* 加急费 \*/

round(fwf\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as FWF, /\* 交通费 \*/

round(qtf\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as QTF, /\* 其他费用 \*/

round(WORKLOAD\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as WORKLOAD , /\* 工作量 \*/

round(charge.intensive\_Workload\_actual/decode(sjd.CERTIFICATE\_NUMBER,0,1,sjd.CERTIFICATE\_NUMBER),2) as intensiveWorkload, /\* 强检工作量 \*/

(select WM\_CONCAT(data\_value4) from t\_zs\_data data where data.zs\_id = zs.zs\_id and data.data\_type='jsgc') JSGC, /\*方法规范\*/

(select WM\_CONCAT(data\_value4) from t\_zs\_data data where data.zs\_id = zs.zs\_id and data.data\_type='jsgf') JSGF, /\*技术规范\*/

nvl(charge.fee\_Remark,' ') feeRemark, /\*收费标准\*/

decode(zs.cma\_flag, 0, '否', 1, '是') cma\_flag, /\*是否cma\*/

rown rn

from (

/\*内部表开始\*/

select t1.\*

from (select mc.\*, rownum rown

from (select zs.zs\_id ,zs.ADD\_TIME

from t\_zs zs

left join t\_b\_wtd\_sjd sjd on sjd.registration\_no = zs.registration\_no

left join t\_b\_wtd wtd on wtd.wtd\_id = sjd.wtd\_id

left join t\_s\_depart depart on depart.id = sjd.depart\_id

left join t\_s\_post post on sjd.post\_id = post.id

left join t\_s\_user adduser on adduser.id = zs.add\_user

left join t\_s\_user printuser on printuser.id = zs.print\_user

left join T\_COMPANY comp on comp.id = wtd.company\_id

left join T\_CHARGE charge on zs.registration\_no = charge.registration\_no

WHERE 1=1

AND( INSTR(XHGG, 'GT-B60') > 0 ) and zs.add\_time > add\_months(sysdate,-12) and (sjd.Depart\_Id= 280 or sjd.integrated\_flag = 1)

order by ZS\_ID desc ,zs.zs\_id desc

) mc where rownum <= 50

) t1 where rown >= 1

/\*内部表结束\*/

) zs1

left join t\_zs zs on zs1.zs\_id = zs.zs\_id

left join t\_b\_wtd\_sjd sjd on sjd.registration\_no = zs.registration\_no

left join t\_b\_wtd wtd on wtd.wtd\_id = sjd.wtd\_id

left join t\_s\_depart depart on depart.id = sjd.depart\_id

left join t\_s\_post post on sjd.post\_id = post.id

left join t\_s\_user adduser on adduser.id = zs.add\_user

left join t\_s\_user printuser on printuser.id = zs.print\_user

left join T\_COMPANY comp on comp.id = wtd.company\_id

left join T\_CHARGE charge on zs.registration\_no = charge.registration\_no

left join t\_division province on province.division\_num = comp.PROVINCE\_ID and province.IS\_DELETE = 0

left join t\_division city on city.division\_num = comp.city\_id and city.IS\_DELETE = 0

left join t\_division area on area.division\_num = comp.area\_id and area.IS\_DELETE = 0

order by ZS\_ID desc ,zs1.zs\_id desc

) a

受理查询分页

SELECT count (1) total\_count

FROM t\_b\_wtd\_sjd sjd

LEFT JOIN t\_b\_wtd wtd ON sjd.wtd\_id = wtd.wtd\_id

WHERE 1 = 1

AND sjd.ADD\_TIME >= sysdate - 730

AND (sjd.depart\_id = 360

OR wtd.task\_source LIKE '%周吴同%')

select RN as "RN",YSJL\_ID as "ysjlId",ZSBH as "zsbh",ZSQJMC as "zsqjmc",ZSDWMC as "zsdwmc",APPLIANCE\_ACTUAL\_NAME as "applianceActualName",COMPANY\_NAME as "companyName",XHGG as "xhgg",CCBH as "ccbh",SBBH as "sbbh",ZZCS as "zzcs",ENTRUST\_DATE as "entrustDate",ZSJDRQ as "zsjdrq",JD\_TYPE as "jdType",JDJL as "jdjl",DEPART\_ID as "departId",POST\_NAME as "postName",DEPART\_NAME as "departName",ADDUSERID as "adduserid",ADDUSER as "adduser",ADD\_TIME as "addTime",INTEGRATED\_FLAG as "integratedFlag",CNAS\_FLAG as "cnasFlag",XLHG\_FLAG as "xlhgFlag",REGISTRATION\_NO as "registrationNo",ORDER\_NO as "orderNo",ORDER\_SINGLE\_NO as "orderSingleNo",QJ\_FLAG as "qjFlag",QJYT as "qjyt",JDYNAME as "jdyname",HYYNAME as "hyyname",COMPANY\_TYPE as "companyType",EXISTZS as "existzs",szxxmName as "szxxmName",szxxmName as "szxxmName",IS\_AUTHORISE as "isAuthorise",IS\_SETTLE\_ACCOUNT as "isSettleAccount",FEE\_STATUS as "feeStatus",IS\_SETTLEMENT as "isSettlement",APPLIANCE\_USE as "applianceUse",IS\_COMPULSORY\_VERIFY as "isCompulsoryVerify",MB\_ID as "mbId",APPLIANCE\_WAREHOUSE\_FLAG as "applianceWarehouseFlag",CERTIFICATE\_WAREHOUSE\_FLAG as "certificateWarehouseFlag",APPLIANCE\_NAME as "applianceName",ADMINISTRATIVE\_AREA as "administrativeArea" from ( select ysjl.ysjl\_id, /\*记录ID\*/

ysjl.zsbh, /\*记录编号\*/

ysjl.zsdwmc, /\*证书单位\*/

decode(ysjl.SZXXM\_FLAG,0,'本院',1,'洁净中心项目',2,'平板中心项目',3,'国家空气净化中心') szxxmName,

to\_char(wtd.ENTRUST\_DATE, 'yyyy-mm-dd') ENTRUST\_DATE, /\*委托日期\*/

ysjl.xhgg, /\*型号规格\*/

ysjl.ccbh, /\*出厂编号\*/

ysjl.sbbh, /\*设备编号\*/

ysjl.zzcs, /\*制造厂商\*/

wtd.ORDER\_NO, /\*委托单号\*/

sjd.ORDER\_SINGLE\_NO, /\*流转单号\*/

to\_char(ysjl.ZSJDRQ, 'yyyy-mm-dd') ZSJDRQ, /\*检定日期\*/

ysjl.jdjl, /\*检定结论\*/

depart.id depart\_id, /\*检测部门ID\*/

depart.depart\_name, /\*检测部门\*/

post.post\_name, /\*检测岗位\*/

adduser.user\_name adduser, /\*添加人\*/

ysjl.add\_user as addUserId ,/\*添加人ID\*/

ysjl.jdwd, /\*温度\*/

ysjl.jdsd, /\*湿度\*/

(select WMSYS.WM\_CONCAT(data.data\_value1) jdyname

from t\_ysjl\_data data

where data.ysjl\_id = ysjl.ysjl\_id and data.data\_type = 'jdy') jdyname, /\*检定员\*/

(select WMSYS.WM\_CONCAT(data.data\_value1) hyyname

from t\_ysjl\_data data

where data.ysjl\_id = ysjl.ysjl\_id and data.data\_type = 'hyy') hyyname, /\*核验员\*/

wtd.company\_name, /\*委托单位\*/

wtd.ADMINISTRATIVE\_AREA,

ysjl.zsqjmc, /\*报告器具名称\*/

/\*sjd.fee\_status,\*/ /\*是否收费\*/

decode(sjd.fee\_status,1,'已录费',0,'未录费') as fee\_status, /\*是否收费\*/

/\*sjd.is\_settlement,\*/ /\*是否结算\*/

decode(sjd.is\_settlement,1,'已结算',0,'未结算') as is\_settlement, /\*是否结算\*/

/\*sjd.is\_settle\_account,\*/ /\*是否到账\*/

decode(sjd.is\_settle\_account,1,'已结账',0,'未结账') as is\_settle\_account, /\*是否到账\*/

/\*ysjl.is\_authorise,\*/ /\*SC0/SC1\*/

decode(ysjl.is\_authorise,0,'授权',1,'未授权') as is\_authorise, /\*SC0/SC1\*/

/\*ysjl.Cnas\_Flag ,\*/ /\*是否CNAS\*/

decode(ysjl.Cnas\_Flag,0,'否',1,'是') as Cnas\_Flag, /\*是否CNAS\*/

sjd.appliance\_name, /\*委托单器具名称\*/

sjd.appliance\_actual\_name, /\*委托单器具实际名称\*/

/\*sjd.Is\_Compulsory\_Verify,\*/ /\*委托单是否强检\*/

decode(sjd.Is\_Compulsory\_Verify,0,'否',1,'是') as Is\_Compulsory\_Verify, /\*委托单是否强检\*/

sjd.Appliance\_Use, /\*委托单器具用途\*/

/\*ysjl.Qj\_Flag,\*/ /\*证书是否强检\*/

decode(ysjl.Qj\_Flag,0,'否',1,'是') as Qj\_Flag, /\*记录是否强检\*/

ysjl.Qjyt, /\*记录器具用途\*/

/\*sjd.Appliance\_Warehouse\_Flag,\*/ /\*器具库房状态\*/

decode(sjd.Appliance\_Warehouse\_Flag,0,'未入库',3,'已入库',4,'已出库') as Appliance\_Warehouse\_Flag, /\*器具库房状态\*/

/\*sjd.Certificate\_Warehouse\_Flag,\*/ /\*证书库房状态\*/

decode(sjd.Certificate\_Warehouse\_Flag,0,'未入库',3,'已入库',4,'已出库') as Certificate\_Warehouse\_Flag, /\*证书库房状态\*/

/\*comp.Company\_Type,\*/ /\*单位类别\*/

decode(comp.Company\_Type,1,'第一类别',2,'第二类别',3,'第三类别',4,'第四类别',5,'第五类别',6,'第六类别') as Company\_Type, /\*单位类别\*/

ysjl.JD\_TYPE, /\*检定类型\*/

to\_char(ysjl.ADD\_TIME, 'yyyy-mm-dd') ADD\_TIME, /\*添加日期\*/

ysjl.registration\_no, /\*登记号\*/

ysjl.mb\_id, /\*模板id 历史证书用\*/

decode(ysjl.ZS\_ID\_FLAG,0,'否',1,'是') existZs, /\*是否生成证书\*/

decode(sjd.integrated\_flag,0,'否',1,'是') integrated\_flag, /\*是否综合类仪器\*/

decode(ysjl.XLHG\_FLAG,0,'否',1,'是') XLHG\_FLAG, /\*是否调修后合格\*/

rown rn

from (

/\*内部表开始\*/

select t1.\*

from (select mc.\*, rownum rown

from (select ysjl.ysjl\_id ,ysjl.ADD\_TIME

from t\_ysjl ysjl

left join t\_b\_wtd\_sjd sjd on sjd.registration\_no = ysjl.registration\_no

left join t\_b\_wtd wtd on wtd.wtd\_id = sjd.wtd\_id

left join t\_s\_depart depart on depart.id = sjd.depart\_id

left join t\_s\_post post on sjd.post\_id = post.id

left join t\_s\_user adduser on adduser.id = ysjl.add\_user

left join T\_COMPANY comp on comp.id = wtd.company\_id

WHERE 1=1

AND( INSTR(ysjl.ZSQJMC, '水准仪') > 0 ) and (( 1=1 and sjd.Depart\_Id= 6 ) or sjd.integrated\_flag = 1 )

order by YSJL\_ID desc ,ysjl.ysjl\_id desc

) mc where rownum <= 10

) t1 where rown >= 1

/\*内部表结束\*/

) ysjl1

left join t\_ysjl ysjl on ysjl1.ysjl\_id = ysjl.ysjl\_id

left join t\_b\_wtd\_sjd sjd on sjd.registration\_no = ysjl.registration\_no

left join t\_b\_wtd wtd on wtd.wtd\_id = sjd.wtd\_id

left join t\_s\_depart depart on depart.id = sjd.depart\_id

left join t\_s\_post post on sjd.post\_id = post.id

left join t\_s\_user adduser on adduser.id = ysjl.add\_user

left join T\_COMPANY comp on comp.id = wtd.company\_id

order by YSJL\_ID desc ,ysjl1.ysjl\_id desc

) a

##### 结论

**根据排名靠前的请求（证书查询、受理查询、受理查询统计、原始记录查询）来分，主要是相关表数据量大，业务查询关联复杂导致查询缓慢。需优化索引，做分区表以及业务拆表（或者数据分离冷数据，保障正常业务优先）的方式优化查询性能。**

## 4. 卡慢排查问题结论

1. 当前系统瓶颈主要集中在复杂 SQL 查询 + 大表 + 其他（IM数据量增大、ecqs上报证书）。
2. 建议分阶段处理：

* 短期（2-3 周）：快速止痛，点状治理。

1. 优化SQL: 减少复杂sql中的子查询（有数据库层处理的子查询改为应用层处理）和非必要的关联等。
2. 索引优化：通过增加&合并索引&函数索引&强制索引等方式优化索引。
3. 针对一些非实时的统计（比如首页统计）采用预处理方式提前计算（比如凌晨预执行计算）并缓存，减少日间的数据库访问。

* 中期（1-3 月）：进行数据分区，持续跟进并逻辑优化。

1. 优化并调整业务逻辑：增加冗余字段减少数据库层的关联查询，进而减少数据层资源的消耗。
2. 通过数据分区将不同范围的数据物理分离（减少数据库层检索），适应不同场景的业务查询（比如按照时间分区）

* 长期（3-6 月）：拆表&冷热数据分离&缓存&归档，重构查询逻辑。

数据模型层：支持报表域的数据模型层，将业务域和报表域区分， 业务域满足常规日常检索，增删改查；报表域层满足大范围的搜索统计。

应用层：支持预处理大数据的业务逻辑查询统计，通过定时任务定期同步&增（全）量&修正业务数据，提前生成报表数据便于用于快速检索；支持业务归档，将业务闭环，用户可自行选择查询不同归档的数据。

增加缓存： 对于不频繁变动的数据做缓存处理，定期更新缓存减少数据库的读写

数据库层：通过业务分表减少非必要检索，将冷热数据数据库层分离开，减少冷数据库操作对整体系统的影响。

中间件层：考虑增加新的NOSQL中间件（比如ES），提供支持超大数据的实时全文检索