

数据库&应用问题分析

说明

oracle系统相关

数据库

安装路径

日志文件路径

系统相关日志分析

客户端监听日志

启动数据库

监听

服务配置路径

监听检测命令

系统优化概述

oracle执行异常分析

锁排查

SQL语句慢分析

IO分析

IO 查看工具(任务管理:ctrl+alt+del; 底部栏, 右键)

查询高IO语句(点击跳转)

低效语句分析

高IO底层原理说明

高频语句分析

高频查询语句(点击跳转)

通过程序调试功能排查

网络排查

网络连接概念

数据库连接排查

流量异常

问题收集

lsnrctl 操作非常慢

解决方式

框架无法访问数据库

案例

特殊订单高频率读取手机导致取号慢

数据面板低效语句拖卡系统

关键字:数据库排查, 性能分析, 功能慢分析

说明

oracle相较于mssql, 出现的问题范围相关较广, 从监听, 到各种奇怪的启动问题, 都需要具备一定的排查能力, 为此梳理一份针对oracle的排查说明. 本次以oracle 为例进行问题排查说明. mssql 同样可以找到相似工具

oracle系统相关

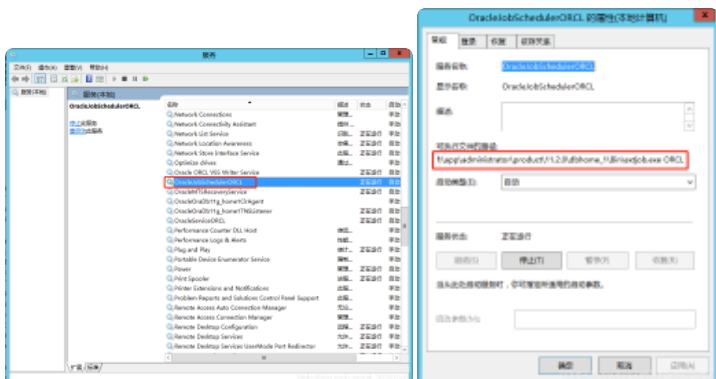
数据库

安装路径

需要明确oracle安装路径, 通常格式如下(下面统一用oracle_home表示根路径). 所有监听日志, 以及配置, 都是在该路径下进行处理

xxxxx\product\11.2.0\dbhome_1

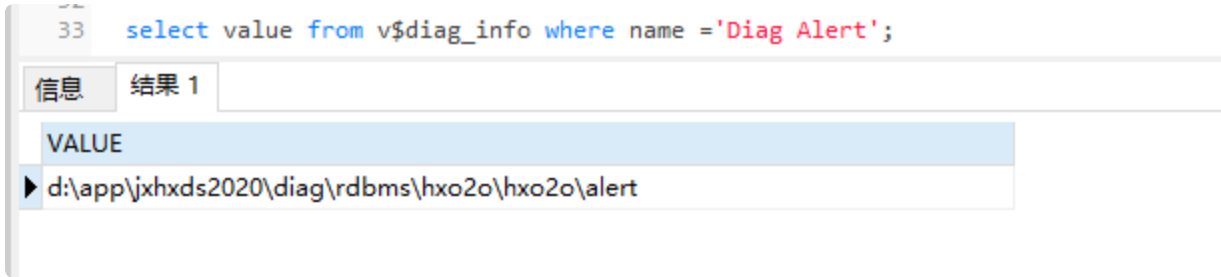
服务管理器查看安装路径



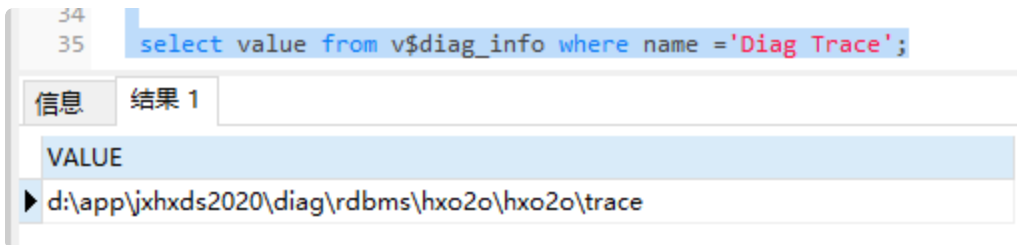
日志文件路径

系统相关日志分析

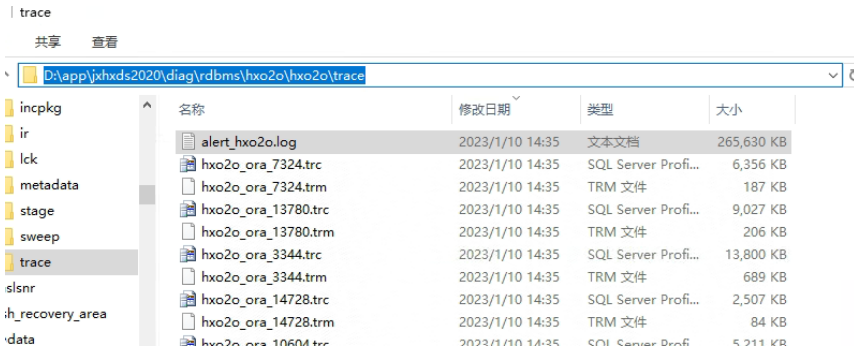
```
select value from v$diag_info where name ='Diag Alert';
```



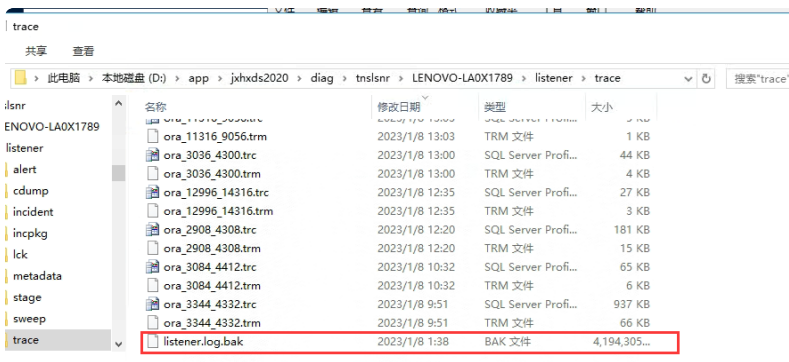
```
select value from v$diag_info where name ='Diag Trace';
```



D:\app\jxhxd2020\diag\rdbms\hxo2o\hxo2o\trace



客户端监听日志



启动数据库

如果数据库启动异常, 通常可以看到相关原因. 如果系统是正常运行过, 后突然出现问题. 需要根据异常搜索对应的处理方式.

进入控制台运行

```
sqlplus /as sysdba
```

启动

```
startup
```

关闭

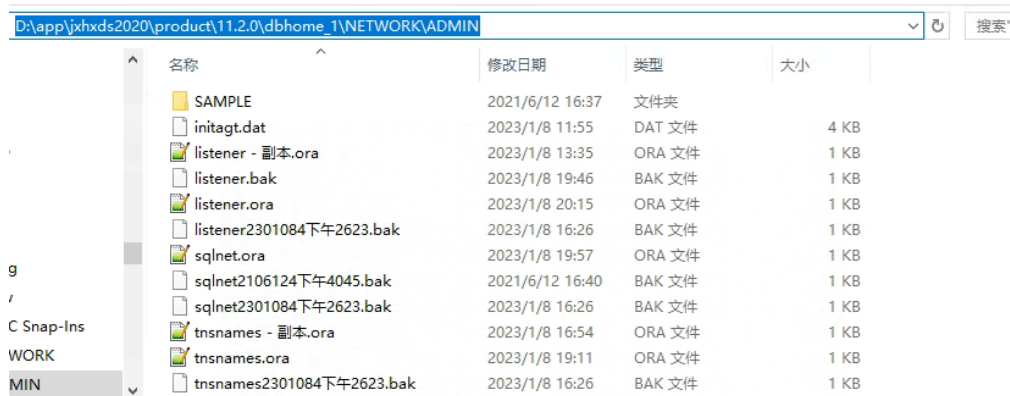
```
shutdown immediately
```

监听

oracle监听是一个独立服务, 不同于mssql, 是一个独立管理程序. 如果监听出现异常, 会导致连接不正常. 需要根据具体情况进行分析. **判断问题时需要先明确好是数据库异常, 还是监听异常, 不能混为一谈.**

服务配置路径

D:\app\jxhxd2020\product\11.2.0\dbhome_1\NETWORK\ADMIN\network\admin



名称	修改日期	类型	大小
SAMPLE	2021/6/12 16:37	文件夹	
initagt.dat	2023/1/8 11:55	DAT 文件	4 KB
listener - 副本.ora	2023/1/8 13:35	ORA 文件	1 KB
listener.bak	2023/1/8 19:46	BAK 文件	1 KB
listener.ora	2023/1/8 20:15	ORA 文件	1 KB
listener2301084下午2623.bak	2023/1/8 16:26	BAK 文件	1 KB
sqlnet.ora	2023/1/8 19:57	ORA 文件	1 KB
sqlnet2106124下午4045.bak	2021/6/12 16:40	BAK 文件	1 KB
sqlnet2301084下午2623.bak	2023/1/8 16:26	BAK 文件	1 KB
tnsnames - 副本.ora	2023/1/8 16:54	ORA 文件	1 KB
tnsnames.ora	2023/1/8 19:11	ORA 文件	1 KB
tnsnames2301084下午2623.bak	2023/1/8 16:26	BAK 文件	1 KB

监听检测命令

查看监听状态:lsnrctl status // 缩写说明 listener control 监听控制

```
管理员: C:\Windows\system32\cmd.exe
C:\Users\Administrator\Desktop>lsnrctl status

LSNRCTL for 64-bit Windows: Version 11.2.0.1.0 - Production on 10-1月 -2023 14:46:46

Copyright (c) 1991, 2010, Oracle. All rights reserved.

正在连接到 (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=LENOVO-LA0X1789)(PORT=1521)))
LISTENER 的 STATUS
-----
别名                LISTENER
版本                TNSLSNR for 64-bit Windows: Version 11.2.0.1.0 - Production
启动日期            10-1月 -2023 08:50:06
正常运行时间        0 天 5 小时 56 分 39 秒
跟踪级别            off
安全性              ON: Local OS Authentication
SNMP                OFF
监听程序参数文件    D:\app\jhxds2020\product\11.2.0\dbhome_1\network\admin\listener.ora
监听端点概要
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=LENOVO-LA0X1789)(PORT=1521)))
服务摘要
服务 "CLRExtProc" 包含 1 个实例。
实例 "CLRExtProc", 状态 UNKNOWN, 包含此服务的 1 个处理程序...
服务 "hxo2o" 包含 2 个实例。
实例 "hxo2o", 状态 UNKNOWN, 包含此服务的 1 个处理程序...
实例 "hxo2o", 状态 READY, 包含此服务的 1 个处理程序...
命令执行成功
```

关闭监听:lsnrctl stop

启动监听:lsnrctl start

通常监听程序异常, 通过上面的简单查看和启动, 就可以看到具体异常, 再根据异常进行处理

系统优化概述

需要明确一个性能分析的底层逻辑. 操作系统的各功能都可以理解成一条水管. 水管的大小就是内存, IO, 这类硬件的尺寸. 同一时刻. 出水量是固定的. 如果其它功能用了. 剩下的可流水的量就小了. 固态硬盘或者机械硬盘, 以及更好的内存, CPU, 只是提升了水流速度. **优化原则本身不变. 在实现相同的功能的情况下, 尽可能减少不必要的水流. 同时性能分析的核心, 就是确认谁在用这条水管, 用了多少. 当系统某个指标不正常时, 都会导致整个系统异常. 特别是数据库排查过程中, 需要耐心像剥洋葱一样, 一层一层深入目标问题点.**

oracle执行异常分析

锁排查

通常sql执行慢, 需要先检查是否出现阻塞(**阻塞排查**), 如果是出现阻塞, 先记录好阻塞的语句. 然后尝试把语句杀死. 先恢复作业. 如果阻塞过多. 为持续阻塞导致. 只能通过重启恢复. 而导致锁的原因很多. 需要具体分析并根据实际情况处理.

排查方式

先确认好是谁的锁导致异常. 是外部系统锁导致异常, 需要查看是哪个机器发起, 并联系对应人员进行优化处理. 如果影响较大, 且无法杀死, 从而导致本系统异常. 甚至可以先禁止访问, 等处理后再恢复访问权限.

SQL语句慢分析

通常SQL如果执行慢, 需要进行优化, 慢会导致整个系统操作不流畅甚至卡死. 大都是功能开发没优化到位, 或者数据库操作相关没做好优化. 导致SQL执行低效, 如果是核心表, 操作频繁和多功能操作的表. 则会出现阻塞的情况. 即所有操作, 在原子操作阶段, 都进入排队的情况. 需要分析好具体什么表, 以及相关操作语句有哪些. 再依次处理.

需要确认以下几种情况:

是否有循环没写好, 导致频繁访问数据库.

是否某个语句低效导致执行慢.

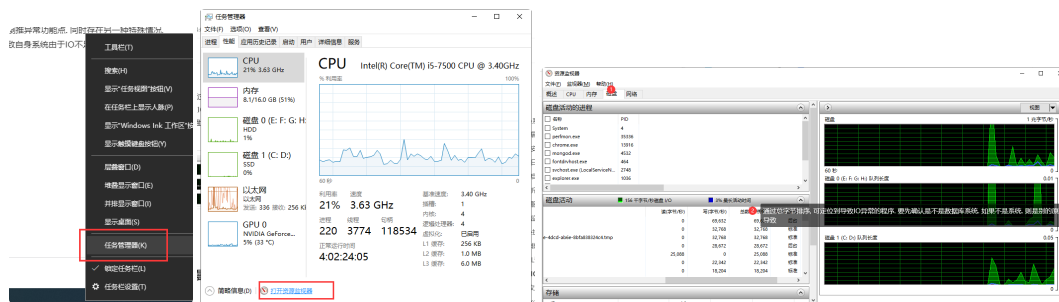
是否表出现了锁或阻塞, 导致执行慢

是否出现别的问题导致访问慢, 如网络, 内存, 磁盘等.

IO分析

如果SQL类语句低效. 可以通过IO分析快速定位到目标语句. 低效SQL通常伴随大量的IO请求, 进而导致内存系统方方面面的特殊异常. 通常通过IO可快速定位到问题语句, 然后根据语句倒推异常功能点. 同时存在另一种特殊情况, 由于数据库系统只有一个磁盘. 如果有其它应用大量消耗IO. 同样会导致自身系统由于IO不足而操作异常. (注:需要明确是数据库IO异常, 才针对数据库进行分析处理. 如果是别的异常, 则需要优先排查好. 不能乱套处理)

IO 查看工具(任务管理:ctrl+alt+del; 底部栏, 右键)



查询高IO语句(点击跳转)

低效语句分析

通常是大表的索引没处理好, 这种可通过语句分析工具快速判断性能点, 通常通过查询条件看是否相关查询条件有加索引. 如果是表过大, 则通过历史迁移或表拆分的情况进行压力分摊.

注:任何数据库都会提供相关分析工具, 通常数据库工具也都会提供相关快捷功能.

对象 | 阻塞源头 @HXO2O (洪兴_新... | 日志查询 @HXO2O (洪兴_... | * 无标题 - 查询

保存 | 查询创建工具 | 美化 SQL () 代码段

HXO2O | 运行已选择的 | 停止 | 解释已选择的

1 select * from JKPT_GOODS_SYNC_GL 1 选中需要性能分析的句子

2 运行语句分析

3 根据结果分析语句性能点

Operation	Object	Optimizer	Cost	Cardinality	Bytes	Partition Start	Partition ID	Access Predicates
▶ SELECT STATEMENT ()	(Null)	ALL_ROWS	35142	11447557195937459	(Null)	(Null)	(Null)	(Null)
TABLE ACCESS (FULL)	JKPT_GOODS_SYNC_GL	ANALYZED	35142	11447557195937459	(Null)	(Null)	(Null)	(Null)

详细说明

针对消耗资源过大的点, 选择合适的优化处理

索引原理

高IO底层原理说明

磁盘IO在数据写入的瞬间是有固定上限的. 即瞬间写入的数据大小, 以及数据节点的切换频率, 都会影响到IO的写入速度. 如果语句低效. 没加索引, 会导致整个表文件检索一次才能找到要的数据. 导致功能执行慢以及IO高.

高频语句分析

通常某个功能慢, 如果不是低效语句, 且基本排查外部干扰. 则需要进一步确认是否存在一些循环或高频读取的功能没处理好. **特别是一些循环读取的功能.** 数据库每次操作, 都需要发起一次连接, 发送语句给数据库系统. 如果是高频语句, 则会导致连接频繁发起, 拖慢功能执行速度. 即使表很小. 频繁的建立连接, 也会导致功能执行非常慢, 特别是数据不在同一台电脑的时间, 特殊情况会更多. **如果表更新频率不高. 直接整表读或者叠加缓存处理. 减少数据库的不必要交互, 提升速度.**

高频查询语句(点击跳转)

通过程序调试功能排查

```
15 "http_port_desc": "指定程序启动端口，默认可以不加。不加的情况下，系统会根据is_te:
16 "http_port": 8000,
17
18 "db": {
19   "dialect_desc": "mssql|oracledb:oracledb对node有版本要求，node12和node10两种
20   "dialect": "oracledb",
21   "debug_desc": "数据库调试模式，开启后，数据库的所有语句都会进行输出",
22   "debug": true,
23   "connection": {
24     "host_desc": "如果是oracledb，需要将端口直接配置到host，不加，默认使用1521，
25     "host": "192.168.1.52:60010",
26     "port": 60010,
27     "user": "JKPT RPA".
```

启用数据库调试模式，功能执行过程会输出到控制台。如果控制台复现某个功能时，出现高频率的交互情况。必定是出现循环中反复读取数据库甚至有功能没处理好，出现死循环的场景。需要分析代码情况，再进行处理。

读取频率高的语句，通过整表拉取或缓存的方式，减少数据库交互。

网络排查

网络连接概念

https://stjy.yuque.com/jkpt/customer/network_check

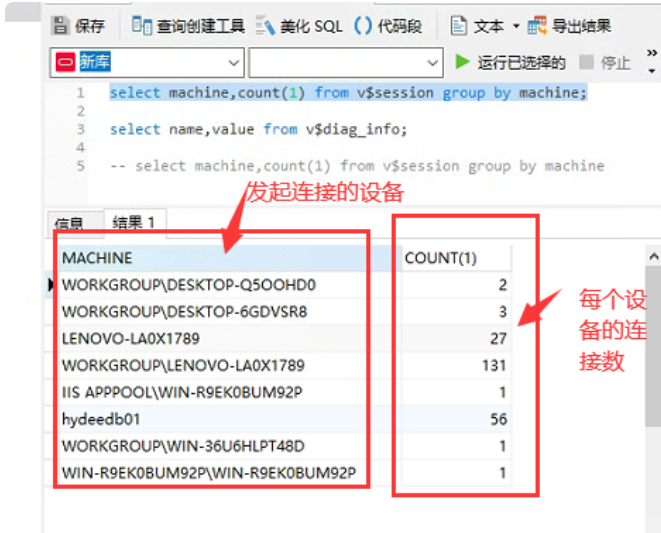
如果是网络类异常，需要先确认具体网络因素。是应用服务器到数据库服务器的延迟太大，还是连接无法建立，甚至别的原因。如果是本地，基本排除路由，防火墙相关网络因素，集中精力判断监听相关服务是否正常。特殊问题说明，有的应用和数据库可能是同电脑，但是访问设置成了路由提供的IP(如:192.168.x.x)。即数据实际是经过路由再回到电脑本地。即使应用和数据库同台电脑。一样存在外部因素。如果出问题，先调整成127.0.0.1再进行网络排查。减少干扰因素。

数据库连接排查

<https://stjy.yuque.com/jkpt/inside/mp76dd#Jvl7w>

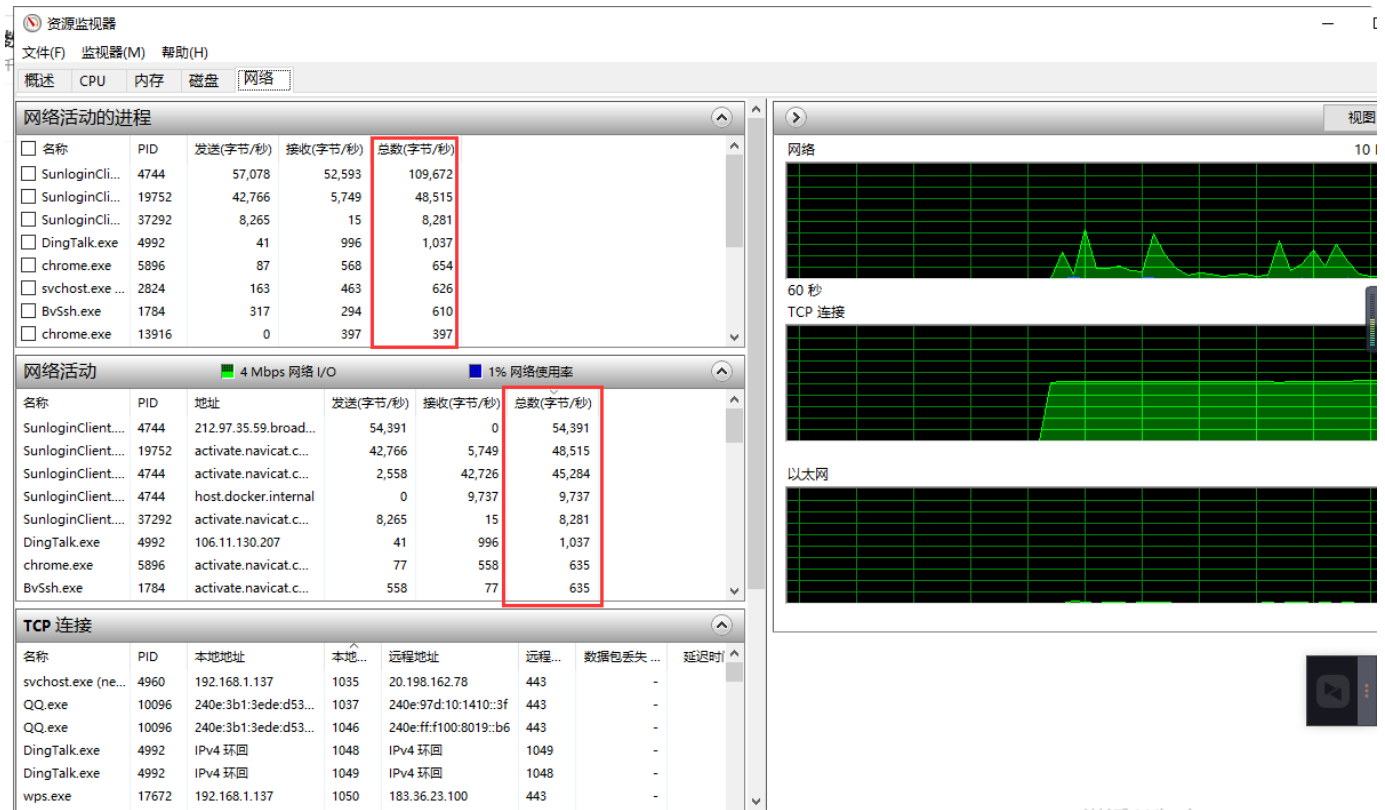
排查是否存在连接数过高的问题。如果是，需要分析是具体什么设备发起。确认发起点或者发起点在执行的功能后，进行针对性出来。如果是外部的设备发起。定位目标后，通知目标调整。一个端口的连接数是有最大限制的。同时应用端口的承载功能也是有限的，如果过多，会导致新发起的连接异常无法进入数据库的情况，或者导致原有数据库连接不稳定的问题。应用通常不需要那么多的连接数就可以完成功能。如果连接数

过多. 需要按实际情况进行排查. 像千手是4核, 且每个核有5个连接的池. 即启动阶段, 最低也消耗了20个连接数. 运行过程还会增加. 需要具体问题具体分析



如果是系统自身连接数过高, 需要判断是否存在阻塞或锁表类原因, 由于前面的连接未执行完成. 所以新功能只能不断发起新连接, 导致更多的连接, 更多的阻塞.

流量异常



如果自身系统没有大流量的功能, 但是系统分析任务管理器后存在大流量情况. 需要让服务器管理人员排查谁在发起大量请求. 整个路由器的带宽是优先的. 如果某个应用过多的占用带宽, 即水管流量被占用, 剩下的给这边系统的就不会太多, 表象为访问慢.

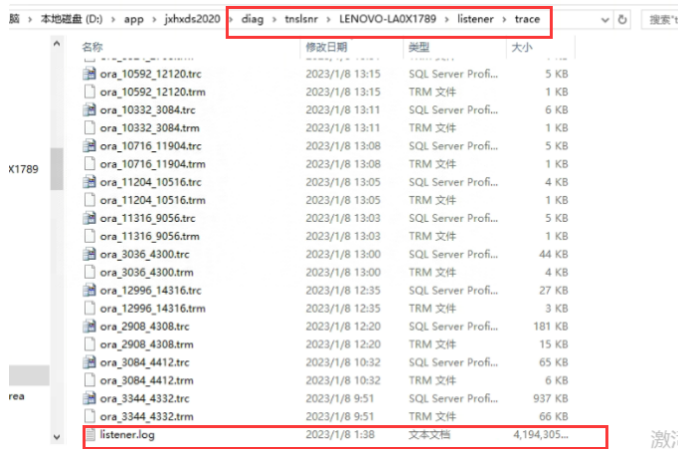
问题收集

Isnrctl 操作非常慢

说明

如果Isnrctl操作非常慢,可能是日志过大导致.日志过大原因可能是频繁出现异常连接,导致日志增速过大.操作过程中,会出现非常久的无法操作的情况.同时会出现oracle连接非常不稳定的问题.刚访问可以上,但是过阵子就无法访问的情况.同时监听相关命令操作非常慢,但是并不是报错,且没有错误提示.

判定方式



解决方式

方式一:关闭Isnrctl stop, 然后重命名文件, 再启动即可;

方式二:永久关闭

```
D:\app\jhxds2020\product\11.2.0\dbhome_1\NETWORK\ADMIN\listener.ora - Notepad++ [Administrator]
文件(F) 编辑(E) 搜索(S) 视图(V) 编码(N) 语言(L) 设置(T) 工具(O) 宏(M) 运行(R) 插件(P) 窗口(W) ?
listener.ora tnsnames.ora sqlnet.ora alert_hxo2o.log config_json config_json
7 (SID_NAME = CLRExtProc)
8 (ORACLE_HOME = D:\app\jhxds2020\product\11.2.0\dbhome_1)
9 (PROGRAM = extproc)
10 (ENVS = "EXTPROC_DLLS=ONLY:D:\app\jhxds2020\product\11.2.0\dbhc
11 )
12 (SID_DESC =
13 (GLOBAL_DBNAME = hxo2o)
14 (ORACLE_HOME = D:\app\jhxds2020\product\11.2.0\dbhome_1)
15 (SID_NAME = hxo2o)
16 )
17 )
18
19 LISTENER =
20 (DESCRIPTION_LIST =
21 (DESCRIPTION =
22 (ADDRESS = (PROTOCOL = TCP)(HOST = LENOVO-LA0X1789)(PORT = 1521)
23 )
24 )
25 )
26 ADR_BASE_LISTENER = D:\app\jhxds2020
27
28
29 #---ADDED BY TNSLSNR 08-1月 -2023 20:15:45---
30 LOGGING_LISTENER = OFF
31 #
```

框架无法访问数据库

<https://stjy.yuque.com/jkpt/inside/zsagy5h6cc3ky8m6>

需要了解后应用间, 建立连接的一些基础概念. 通过调试模式分析启动过程具体出错因素. 并根据具体出错内容排查.

案例

特殊订单高频率读取手机导致取号慢

场景

读取特殊订单里的黑名单手机号码列表时, 没有一次读取, 也没有做缓存, 每个订单都建立N次连接去查询手机号码, 导致取号操作非常慢.

解决方式

1. 所有黑名单号码一次读取, 并缓存到系统一段时间. 该类数据实际只是偶尔更新, 并不需要一直从数据库直接获取数据.
2. 极端情况, 甚至可以将数据走永久缓存, 然后在特殊订单出现修改和删除时, 清理缓存就好.

数据面板低效语句拖卡系统

场景

新上的数据面板功能, 每次访问以及页面的定时器触发, 都会执行大量统计语句, 读的都是订单表等关键大表且没有做合理条件处理. 导致频繁的全表跑. 阻塞其它功能.

解决方式

1. 统计功能并不需要实时, 可先叠加一层缓存, 一小时更新一次. 避免多人访问以及定时器等频繁访问, 拖垮系统.
2. 统计功能, 补充一个扩展表, 将已经不需要统计的语句, 排除掉, 只统计关键部分. 提升系统性能.
3. 补充按天统计功能, 像订单类, 默认只需要查看最近7天甚至当天数据, 并不需要全表跑.
4. null查询优化, 像判断是否取号, 是通过判断取号字段是否为null进行判断. 可以多补充是否为商家待发货状态, 甚至再叠加一个扩展字段. 通过"NONE"代替null查询出未取号内容. 再补充索引.(注:索引对null无效)