

# 宕机/慢问题跟踪以及信息采集

(本文档仅供参考)

## 问题现象

一般出现类似于服务器CPU 和内存一直在98%-100%，后台报锁死错误，一直卡住，无法开发报表的情况

## 一、可能性分析

公共设置 用户管理设置 查询设置 灵活分析|组合分析设置 多维分析设置 页面设置 缓存设置 电子表格设置 透视分析设置 分析报告设置 界面设置

自动优化

缓存设置的系统初始值按单节点1.4G内存设置，系统管理员可根据实际内存值进行修改。

自动优化

业务数据缓冲池

最大对象总数: 100 初始值( 50) 恢复初始值

对象最大激活个数: -1 初始值( -1) 恢复初始值

数据集定义对象池

最大对象总数: 600 初始值( 300) 恢复初始值

对象最大激活个数: -1 初始值( -1) 恢复初始值

参数缓存

参数缓存:  是  否 初始值( 是) 恢复初始值

电子表格

报表对象: 200 初始值( 100) 恢复初始值

报表执行结果: 200 初始值( 100) 恢复初始值

报表输出结果: 200 初始值( 100) 恢复初始值

数据集最大行数: 40000 初始值( 20000) 恢复初始值

【系统选项】—>【查询设置】—>【内存数据库最大返回行数】及【内存数据库最大返回单元格数】也不能设置过大，如果设置很大，往内存数据库加载的数据量就很大，会占用服务器的内存，造成内存溢出问题。

其中【内存数据库最大返回单元格数】=内存数据库最大返回行数\*列字段

公共设置 用户管理设置 查询设置 灵活分析|组合分析设置 多维分析设置 页面设置 缓存设置 电子表格设置 透视分析设置 分析报告设置 界面设置 移动端

查询设置

内存数据库最大返回行数: 1000 初始值( 1000) 恢复初始值

内存数据库最大返回单元格数: 200000 初始值( 200000) 恢复初始值

查询缺省返回行数: 10 初始值( 10) 恢复初始值

是否获取总行数:  是  否 初始值( 是) 恢复初始值

自动缓存:  是  否 初始值( 是) 恢复初始值

分页策略:  SQL分页  结果集分页 初始值( SQL分页) 恢复初始值

正常默认值大小就够了，不建议调整过大

## 内存数据库最大返回行数

内存数据库是为了提高查询性能，所存储缓存数据的数据库。

内存数据库最大返回行数是指默认从数据库中返回多少条记录放到缓存中。默认记录是1000条。

内存数据库最大返回行数可以进行修改，此值设置越大，则内存占用越多。

如内存数据库最大返回行数为1000条，查询对应数据集为使用缓存，并且设置每页返回行数为100。

则报表前10页数据直接从内存数据库中取数据，当翻页到11页时，则会从数据库返回第二个1000条数据放到内存数据库中。

报表会继续从缓存数据库中查询数据。

### 1.3 原因3

灵活分析引用的数据集设置了Excel模板，然后进行大数据量导出，详细说明请查看wiki文档：<https://history.wiki.smartbi.com.cn/pages/viewpage.action?pageId=46071847>

### 1.4 原因4

内存溢出，代码逻辑存在问题或者服务器内存设置太小不满足当前系统用户的使用，这个是需要采集信息发回具体分析的，具体可参考现象三采集信息。



为了定位分析具体原因，通常需要采集有用的信息辅助分析问题，因此需要我们根据实际现象来采集信息

## 二、信息采集

### 2.1 现象一：服务器可正常访问，但是后台显示CPU占用过高

分析：请访问listthreads.jsp界面，如：<http://192.168.1.10:17300/smartbi/vision/monitor/listthreads.jsp>

（需要把IP和端口号 修改成现场的服务器IP和端口号），加载此界面完成后Ctrl+S保存网页内容发回分析，可分析是哪个线程占用过高的CPU：

名称	CPU	堆栈
http-apr-17300-exec-18	0.07%	sun.misc.Unsafe.park(Native Method) java.util.concurrent.locks.LockSupport.park(LockSupport.java:186) java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.await(AbstractQueuedSynchronizer.java:2043) java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:442) org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:104)
AWT-Windows	0%	sun.awt.windows.WToolkit.eventLoop(Native Method) sun.awt.windows.WToolkit.run(WToolkit.java:299) java.lang.Thread.run(Thread.java:744)
Abandoned connection cleanup thread	0%	java.lang.Object.wait(Native Method) java.lang.ref.ReferenceQueue.remove(ReferenceQueue.java:135) com.mysql.jdbc.AbandonedConnectionCleanupThread.run(AbandonedConnectionCleanupThread.java:40)
Add Index Version Thread	0%	sun.misc.Unsafe.park(Native Method) java.util.concurrent.locks.LockSupport.park(LockSupport.java:186) java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.await(AbstractQueuedSynchronizer.java:2043) java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:442) smartbi.metadata.cache.RemoteIndexNotifier\$1.run(RemoteIndexNotifier.java:252)
Attach Listener	0%	
ClientConnector.Noop	0%	java.lang.Object.wait(Native Method) java.util.TimerThread.mainLoop(Timer.java:552) java.util.TimerThread.run(Timer.java:505)
ContainerBackgroundProcessor[St...		java.lang.Thread.sleep(Native Method)

### 2.2 现象二：如果服务器没有宕机，只是报表响应时间长。

信息收集：打开CPU采样，录制报表打开的过程，可参考文档：[系统监控-性能（CPU采样）](#)

### 2.3 现象三：所有用户无法访问服务器，即服务器宕机了。

#### 2.3.1 服务器已经重启的情况下，先拿到以下信息初步定位分析问题。

1、导出系统日志，以管理员登录系统后，在搜索栏上输入【导出系统日志】进行导出：



2、进入【会话管理】->【会话分析】->打开【最大会话分析】界面，选择宕机前后时间段，查询数据，导出Excel文件，可以通过【空闲内存数】大概推测是否是内存溢出导致的，如果【空闲内存数】很少，说明宕机原因就是服务器内存溢出导致的：

注意：时间间隔设置 1分钟

III 首页 定制管理 会话管理 最大会话数分析 x

← → ↺ ★ 图形 字段 ? 参数 打印 定位 透视

最大会话数分析

开始时间\* 2019-10-17 05:46:00 结束时间\* 2019-10-17 09:46 时间间隔\* 1分钟 查询报表(Q)

[首页][上页][下页][尾页] 第1 页, 共 1页 每页 1000 行, 共 49行

日期时间	最大会话数	总内存数	空闲内存数
2019-10-17 08:57	1	4011	331
2019-10-17 08:58	1	4011	237
2019-10-17 08:59	1	4011	161
2019-10-17 09:00	1	4011	211
2019-10-17 09:01	2	4011	259
2019-10-17 09:02	1	4011	227
2019-10-17 09:03	1	4012	406
2019-10-17 09:04	1	4012	282

导航 | 系统选项 | 会话管理 | 最大会话数分析 x

← → ↺ ★ 图形 字段 ? 参数 打印 导出 定位 透视

开始时间\* 2021-10-18 00:00:00 结束时间\* 2021-10-19 23:59:59 时间间隔\* 10分钟 查询报表(Q)

共 235 行 每页 10 行 << 1 / 24 >>

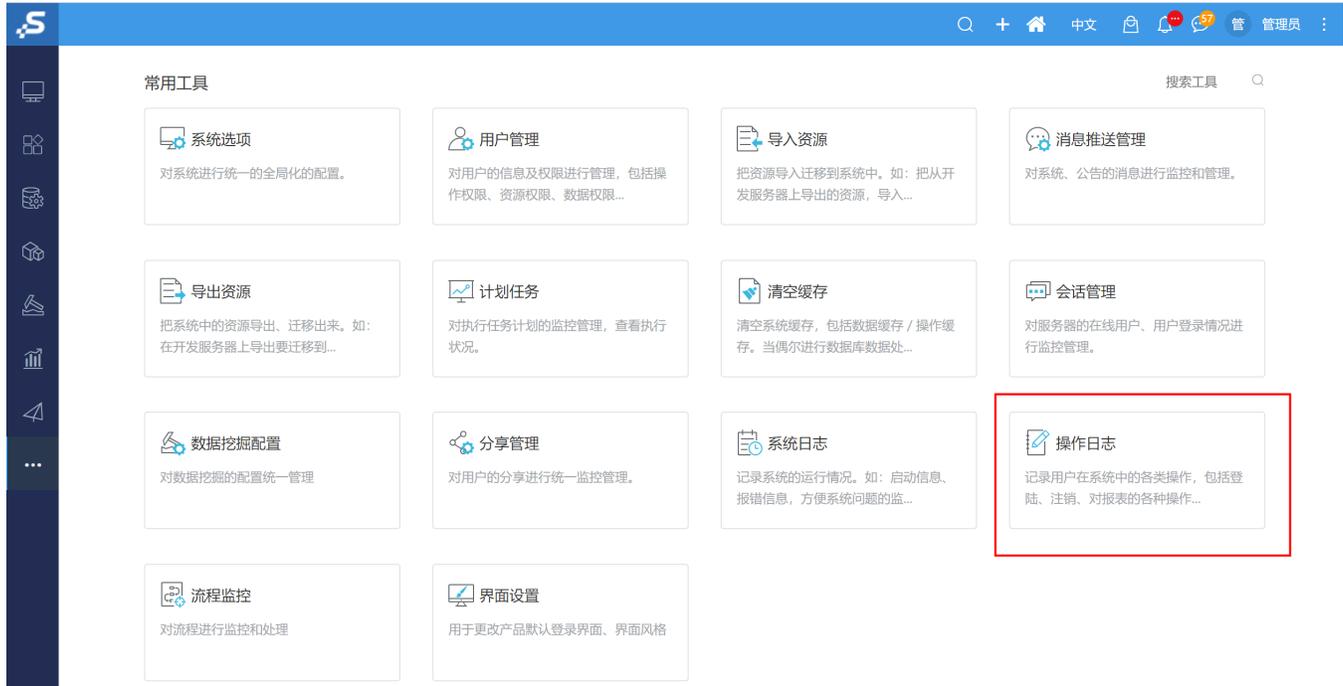
日期时间	最大会话数
2021-10-18 00:00	5
2021-10-18 00:10	5
2021-10-18 00:20	5
2021-10-18 00:30	5
2021-10-18 00:40	5
2021-10-18 00:50	5
2021-10-18 01:00	5
2021-10-18 01:10	0
2021-10-18 01:20	0
2021-10-18 01:30	0

内存使用情况

—○— 总内存数 —○— 空闲内存数



通过会话管理确认内存消耗比较多的时间段，然后打开【操作日志报表】查看当前时间段用户做了什么操作：



检查当前服务器的内存配置大小，可以通过【系统监控】-->【概述】或者【导出系统日志】的【version.txt】文件查看：

<b>服务器：</b>	<b>设置</b>	<b>概述</b>
当前服务器		<b>PID:</b> 22324
导出所有		<b>Host:</b> DESKTOP-6BH8LVR
		<b>Mac IP:</b> 005056C00008[192.168.48.1]; 005056C00001[192.168.91.1]; 049226BE5C54[10.10.202.17]
		<b>Main Class:</b> org.apache.catalina.startup.Bootstrap
<b>概述</b>		<b>JVM:</b> Java HotSpot(TM) 64-Bit Server VM (25.131-b11, mixed mode)
监视		<b>Java:</b> version 1.8.0_131, vendor Oracle Corporation
线程		<b>Java Home:</b> D:\jdk\jre
性能		<b>JVM StartTime:</b> 2019-10-16 17:56:58
网络		<b>JVM 参数</b>
日志		<b>-Xms128m</b>
会话		<b>-Xmx4096m</b>
缓存		<b>-XX:MaxNewSize=256m</b>
堆打印		<b>-XX:MaxPermSize=256m</b>
对象池		<b>-Xrunjdw:transport=dt_socket,server=y,suspend=n,address=4560</b>
类查找		<b>-Xdebug</b>
扩展包		<b>-Xnoagent</b>
所有字体		<b>-Dmail.mime.splitlongparameters=false</b>
wSDL地址		<b>-XX:+HeapDumpOnOutOfMemoryError</b>
log4j信息		<b>-Dlog4j.configuration=log4j.properties</b>
		<b>-Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager</b>
		<b>-Djava.endorsed.dirs=D:\Tomcat_V8.5.6_18600\endorsed</b>
		<b>-Dcatalina.base=D:\Tomcat_V8.5.6_18600</b>
		<b>-Dcatalina.home=D:\Tomcat_V8.5.6_18600</b>
		<b>-Djava.io.tmpdir=D:\Tomcat_V8.5.6_18600\temp</b>

从smartbi系统导出的【系统日志】，用压缩工具打开，找到version.txt查看：

名称	大小	压缩后大小	类型
..			本地磁盘
smartbix			文件夹
sessionsInfo.xml	13,716	1,587	XML 文档
pools.txt	3,623	1,313	文本文档
trace.xml	143	122	XML 文档
extensionsInfo.xml	5,411	1,242	XML 文档
smartbi.log	1,350,135	80,843	文本文档
versions.txt	2,444	1,229	文本文档

```

versions.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
【CPU与内存信息】
Processors:      12
MaxMemory:      4,205,838,336
FreeMemory:      3,432,216,208
TotalMemory:     4,195,352,576

【JVM参数】
InputArguments:
"-Xms128m" "-Xmx4096m" "-XX:MaxNewSize=256m" "-XX:MaxPermSize=256m" "-Xrunjdw:transport=dt_socket,server=y,suspe

```

当前服务器内存配置大小

此时若是服务器内存配置的比较小，需要参照基本的服务器要求：[系统环境要求](#) 调整服务器内存大小 [应用服务器JVM参数配置](#)。

### 2.3.2 服务器尚未重启的情况下，配置采集以下信息后再反馈发回分析。

建议现场在服务器宕机发生的时候同时打印堆dump信息发回分析。

- 1、拿回服务器日志信息，包括所有节点smartbi.log以及服务器自身的logs。（对于服务器日志的获取可参考wiki文档：<https://history.wiki.smartbi.com.cn/pages/viewpage.action?pageId=27000922>，根据现场的服务器拿到对应的服务器日志）
- 2、进行打印线程信息，可参考wiki文档（根据具体的服务器进行线程信息和堆dump文件的打印）：<https://history.wiki.smartbi.com.cn/pages/viewpage.action?pageId=29854061>

## 服务器停止响应问题诊断

- WebSphere打印线程信息和堆Dump文件
- 系统出现问题时跟踪
- weblogic打印线程信息和堆Dump文件
- Tomcat打印线程信息和堆Dump文件

注：堆dump文件较大，也可以上传到网盘上，提供下载链接。

### 2.3.3 特殊情况

1、环境突然宕机，且java进程不在了，服务器是部署到linux环境。

需要看是否有加jvm参数-Djava.awt.headless=true，若没有，需添加此参数，wiki：[使用Linux系统部署的Smatbi服务突然无法启动或进程消失](#)

2、查询星环数据库导致java进程消失。

添加jvm参数-DANTLR\_DO\_NOT\_EXIT=true, wiki: [大数据数据库驱动包引起服务器宕机、或系统退出问题解决方案 \(-DANTLR\\_DO\\_NOT\\_EXIT=true\)](#)

3、java进程消失, 且进程消失当天在应用服务器路径下 (如Tomcat的bin目录) 生成了hs\_err\_pid进程号.log文件, 如下图:

名称	大小	类型	修改时间	属性	所有者
catalina-tasks.xml	2KB	XML 文...	2020/7/18, 10:21	-rw-r--r--	root
catalina.bat	15KB	Windo...	2020/7/18, 10:21	-rw-r--r--	root
catalina.sh	22KB	Shell S...	2021/12/27, 16:45	-rwxr-x...	root
commons-daemon-native.tar.gz	202KB	WinRA...	2020/7/18, 10:21	-rw-r--r--	root
commons-daemon.jar	25KB	Execut...	2020/7/18, 10:21	-rw-r--r--	root
configtest.bat	2KB	Windo...	2020/7/18, 10:21	-rw-r--r--	root
configtest.sh	2KB	Shell S...	2020/7/18, 10:21	-rwxr-x...	root
daemon.sh	8KB	Shell S...	2020/7/18, 10:21	-rwxr-x...	root
datagenerate.ext	775KB	EXT 文件	2021/8/18, 20:19	-rw-r--r--	root
digest.bat	2KB	Windo...	2020/7/18, 10:21	-rw-r--r--	root
digest.sh	2KB	Shell S...	2020/7/18, 10:21	-rwxr-x...	root
file-leak-detector-1.13-jar-with-dependencies.jar	930KB	Execut...	2021/1/18, 10:09	-rw-r--r--	root
hs_err_pid6386.log	157KB	文本文档	2022/4/8, 5:04	-rw-r--r--	root
hs_err_pid24966.log	202KB	文本文档	2022/4/20, 1:11	-rw-r--r--	root
lzpTest	0 Bytes	文件	2021/5/12, 10:28	-rw-r--r--	root
MaskRule.log	1.25MB	文本文档	2022/4/20, 1:10	-rw-r--r--	root
MaskRule.log.1	10.00...	1 文件	2022/4/11, 15:37	-rw-r--r--	root
mondrian.properties	74 Bytes	PROPE...	2021/9/24, 14:48	-rw-r--r--	root
nohup.out	2KB	OUT 文件	2021/9/7, 9:49	-rw-----	root
setclasspath.bat	3KB	Windo...	2020/7/18, 10:21	-rw-r--r--	root
setclasspath.sh	4KB	Shell S...	2020/7/18, 10:21	-rwxr-x...	root

此时需要获取当天的hs\_err\_pid进程号.log文件进行分析。

## 2.4 现象四：登录界面一直在转圈，或者操作系统会出现卡顿的现象。

信息采集：如果一直在转圈，先尝试浏览器隐身模式测试，如果仍然一直转圈，可以查看浏览器F12是否有脚本错误，如果没有脚本错误或者操作系统很卡，则需要是在出现问题的时候录制charles发回分析：[Charles](#)