CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

2024A1-服务器数据库系统TPC-C性能优化

A1-星期六不上发条

- 队长：李义
- 队员：李义、谢雯馨

- 2024年12月12日

CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

目录

content

CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

1 团队介绍

团队介绍

李义	中山大学	计算机技术
谢雯馨	华南理工大学	管理科学与工程

CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

2 赛题回顾

赛题回顾

- 选手需基于海光3号7380 CPU服务器作为计算节点，从操作系统、数据库、等方面完成国产数据库系统调优工作，实现TPCCRunner测试得到的TPMC测试得分提升，需在远程环境进行实测，并提交相应的调优方案PPT。
- 参赛团队调优方案同时进行2个架构的国产平台
  - 1.海光3号7380 CPU服务器操作系统（KOS）、数据库（MySQL）、测试工具（TPCCRunner）
  - 2.飞腾S5000cCPU服务器操作系统（KOS）、数据库（瀚高数据库）、测试工具（TPCCRunner）上进行调优；
- 调优条件
  - TPCCRunner工具**不限定库和并发**，数据测试模型使用工具默认配置，配置文件和运行脚本不允许修改，测试TPMC最高值。

调优结果综合得分=（海光平台TPMC得分*50%）+（飞腾平台TPMC得分*50%）。

CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

3 主要工作

海光环境搭建

海光环境	整机环境
管理网IP	10.51.52.243
CPU	Hygon C86 7380
核心+线程数	128核心，256线程
操作系统	KOS 5.8 SP1
硬盘（挂载/home）	512 GB
内存	512 GB
搭载软件	MySQL Ver 8.0.32
测试环境	TPCC-Runner

调优前TPMC得分

TPCCRunner工具限定warehouse 数量为10，slave 10并发

数据测试模型使用工具默认配置

```
<Parameters>
    [server]: 10.51.52.243
    [port]: 3306
    [DBname]: tpcc
      [user]: root
      [pass]: 123456
 [warehouse]: 10
[connection]: 10
    [rampup]: 300 (sec.)
   [measure]: 300 (sec.)
```

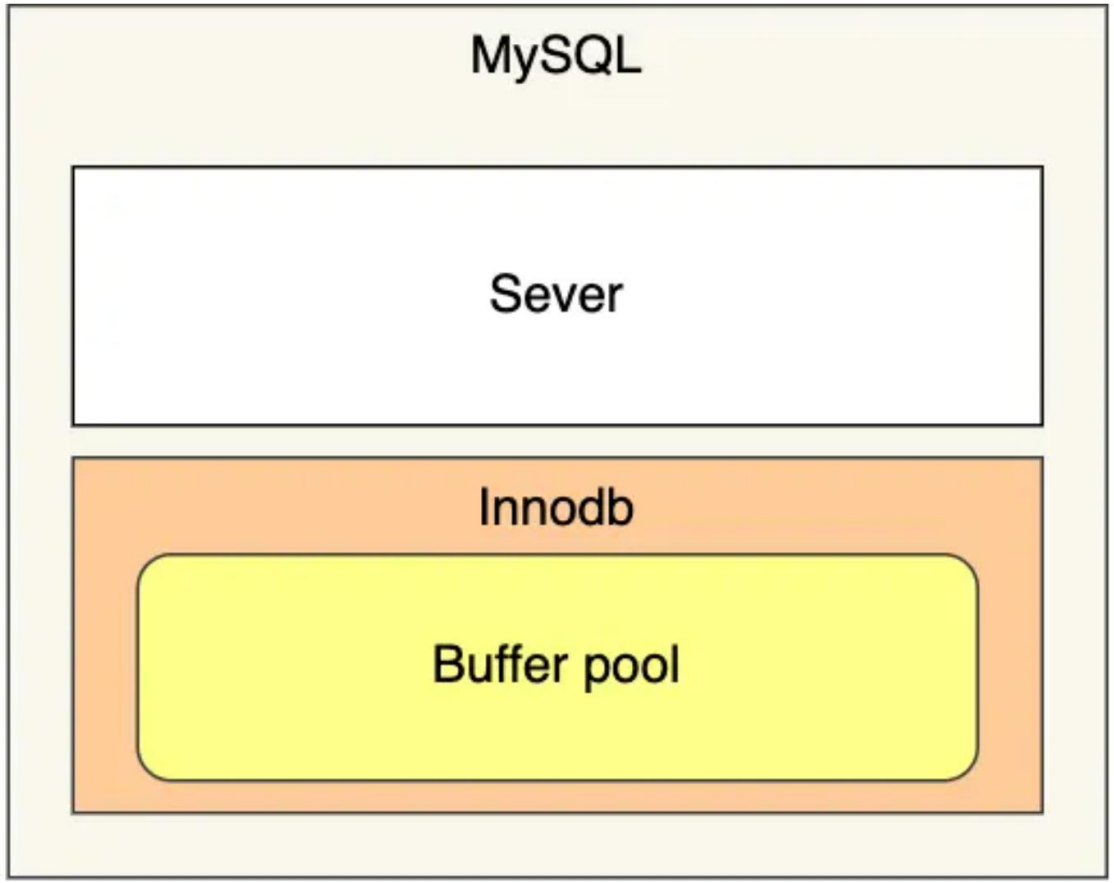
```
<Constraint Check> (all must be [OK])
 [transaction percentage]
        Payment: 43.44% (>=43.0%) [OK]
   Order-Status: 4.37% (>= 4.0%) [OK]
       Delivery: 4.35% (>= 4.0%) [OK]
    Stock-Level: 4.31% (>= 4.0%) [OK]
 [response time (at least 90% passed)]
      New-Order: 0.00%  [NG] *
        Payment: 0.00%  [NG] *
   Order-Status: 3.04%  [NG] *
       Delivery: 0.00%  [NG] *
    Stock-Level: 0.00%  [NG] *
```

<TpmC> 11882.000 TpmC

MySQL性能优化

- 缓冲池是主内存中的一个区域，InnoDB在访问表和索引数据时会在该区域进行缓存。缓冲池允许直接从内存访问频繁使用的数据，这加快了处理速度。
- 在专用服务器上，**通常会将高达80%的物理内存分配给缓冲池。**

- 缓冲池被划分为可能容纳多行的页面。为了提高缓存管理的效率，缓冲池被实现为页面的链接列表；
- 如何利用缓冲池将频繁访问的数据保存在内存中是**MySQL调优**的一个重要方面。



海光环境调优—InnoDB层

2.1 InnoDB缓冲池优化

原始配置

```
innodb_buffer_pool_size=128M
innodb_buffer_pool_instances=8
```

调优后配置

```
innodb_buffer_pool_size=400G
innodb_buffer_pool_instances=64
```

优化原理

- 缓冲池大小设置为总内存的75-8%左右（512GB * 0.75 ≈ 400GB）
- 每个缓冲池实例管理约6.25GB内存，提高并发访问效率
- 多个缓冲池实例减少内部竞争，提高并行处理能力

- 可以将InnoDB的缓存区分成几个部分，这样可以提高系统的**并行处理能力**
- 因为可以允许多个进程同时处理不同部分的**缓存区**。这样就可以同时有多个进程进行数据操作，CPU的效率就高多了。
- 最佳实践：**每个缓冲区大于1G**即可。分配400G缓存，理论上可支持400个单独的缓存区。
- 我们对此进行尝试修改，但对于社区版MySQL Ver 8.0.32，上限为**64**。专业版不受此限制。

海光环境调优-InnoDB层

2.2 InnoDB日志优化

原始配置

```
innodb_log_file_size=48M
innodb_log_buffer_size=16M
```

调优后配置

```
innodb_log_file_size=4G
innodb_log_buffer_size=64M
```

- 更大的日志文件可以**减少磁盘I/O次数**
- 但同时mysql重启时需要更多时间来初始化。
- 设置4G大小是性能与启动时间的平衡

优化原理

- 更大的日志文件减少检查点频率，提高写入性能
- 增大日志缓冲区减少磁盘I/O次数
- 适应高并发OLTP场景的写入需求

海光环境调优-InnoDB层

2.3 InnoDB并发优化

原始配置

```
innodb_write_io_threads=4
innodb_read_io_threads=4
innodb_purge_threads=4
```

调优后配置

```
innodb_write_io_threads=64
innodb_read_io_threads=64
innodb_purge_threads=16
```

优化原理

- IO线程数匹配物理CPU核心数，充分利用多核优势
- 增加清理线程数提高后台清理效率
- 提高并行处理能力，改善高并发性能

海光环境调优-操作系统

3.1 内存管理优化

原始配置

```
vm.swappiness=60
vm.dirty_ratio=20
vm.dirty_background_ratio=10
```

调优后配置

```
vm.swappiness=1
vm.dirty_ratio=40
vm.dirty_background_ratio=3
vm.dirty_expire_centisecs=500
vm.dirty_writeback_centisecs=100
```

优化原理

- 降低swap使用倾向，减少磁盘交换
- 优化脏页刷新机制，提高I/O效率
- 平衡内存使用和磁盘写入

海光环境调优-操作系统

3.2 网络优化

原始配置

```
net.core.somaxconn=128
net.ipv4.tcp_max_syn_backlog=1024
net.core.netdev_max_backlog=1000
```

调优后配置

```
net.core.somaxconn=65535
net.ipv4.tcp_max_syn_backlog=65535
net.core.netdev_max_backlog=65535
net.ipv4.tcp_fin_timeout=10
net.ipv4.tcp_tw_reuse=1
```

优化原理

- 增加连接队列长度，应对高并发连接
- 优化TCP连接状态处理，提高连接效率
- 加快端口复用，减少等待时间

海光环境调优-操作系统

3.3 文件系统优化

原始配置

```
fs.file-max=65535
fs.aio-max-nr=65536
```

调优后配置

```
fs.file-max=6815744
fs.aio-max-nr=1048576
```

优化原理

- 提高系统最大文件描述符数量
- 增加异步I/O请求上限
- 支持更多并发文件操作

海光环境调优

4.1 内存管理极限优化

优化后配置

```
large_pages=ON
innodb_numa_interleave=1
vm.swappiness=0
vm.dirty_ratio=50
vm.dirty_background_ratio=5
```

极限优化原理

- 启用大页内存，减少TLB miss，提高内存访问效率
- 启用NUMA内存交错，优化多CPU架构下的内存访问
- 完全禁用swap，避免任何磁盘交换
- 更激进的脏页比率，允许更多数据在内存中修改
- 优化NUMA节点间的内存访问平衡

- **大页（HugePages）**将内存页的大小从常规的 4KB 增加到 2MB 或 1GB，从而：
  - 减少页表的条目数量。
  - 提高 CPU 的TLB命中率，减少缺页引起的内存访问延迟。
  - **大页方式需手动管理内存，**但实践中比透明大页效果更佳
- **启用 NUMA 内存交错**，优化多 CPU 架构下的内存访问。
- 更高的脏页比率意味着更多数据可以在内存中修改，而无需频繁写回磁盘。

海光环境调优

4.2 系统调度优化

优化后配置

```
net.core.somaxconn=131070
net.ipv4.tcp_max_syn_backlog=131070
net.core.netdev_max_backlog=131070
CPUSchedulingPolicy=rr
CPUSchedulingPriority=99
IOSchedulingClass=realtime
```

极限优化原理

- 网络队列翻倍，应对超高并发连接
- 采用实时CPU调度策略，保证数据库进程获得最高优先级
- 采用实时IO调度，优化磁盘访问
- 更激进的TCP参数优化，提高网络吞吐量
- 优化CPU调度策略，减少进程切换开销

海光环境调优

4.3 并发处理极限优化

优化后配置

```
thread_cache_size=512
max_connections=20000
table_open_cache=81920
table_open_cache_instances=128
```

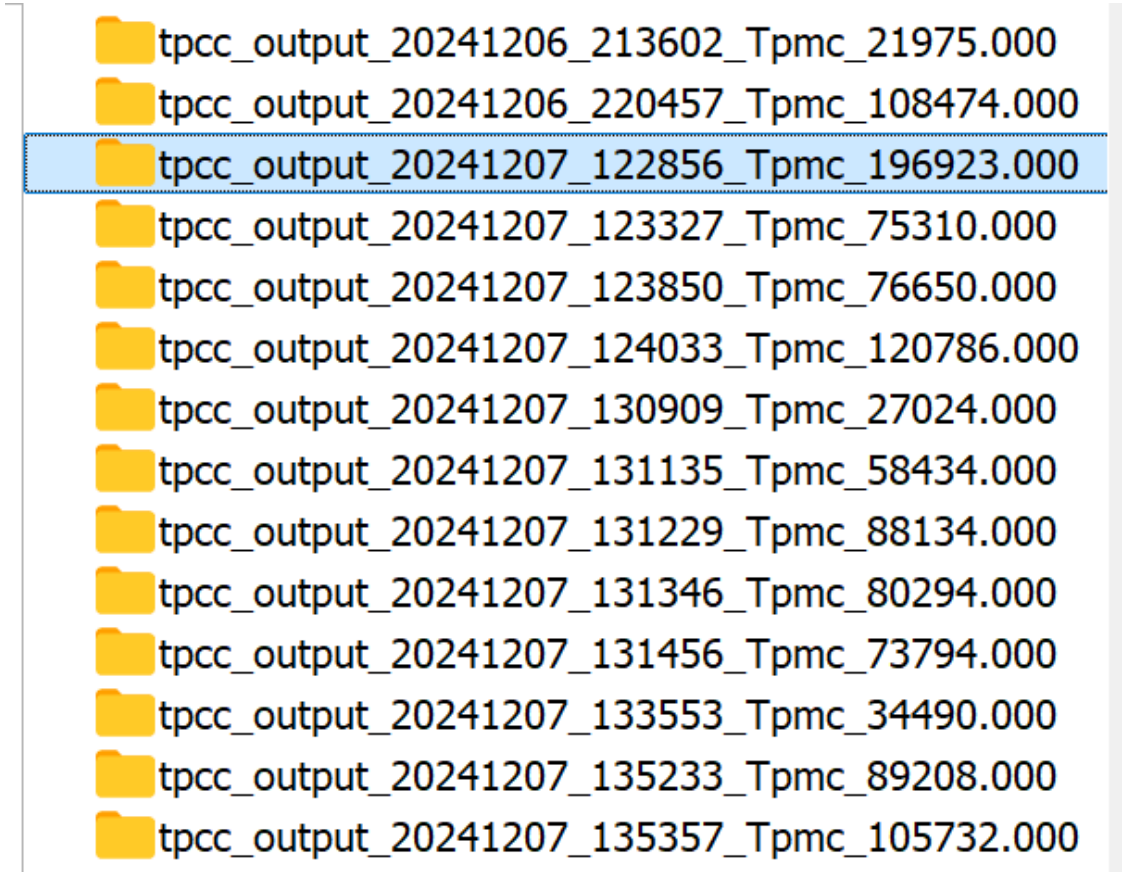
极限优化原理

- 线程缓存翻倍，减少线程创建和销毁开销
- 最大连接数翻倍，支持更高并发
- 表缓存翻倍，提高表访问效率
- 缓存实例数增加到物理核心数，减少锁竞争

海光调优-最佳实践

TPCCRunner工具限定warehouse 数量为1000，slave 500并发

参数	值
server	10.51.52.243
port	3306
DBname	tpcc
user	root
password	123456
warehouse	1000
connection	500
rampup	600 (sec.)
measure	600 (sec.)

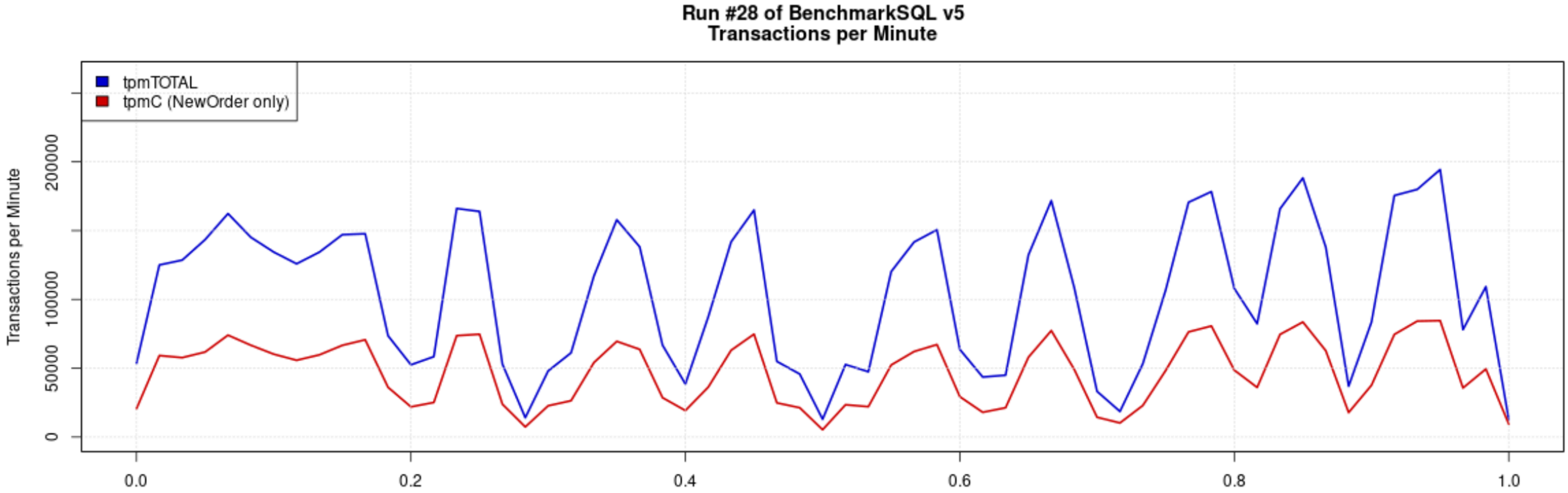


<TpmC>　196923.000 TpmC

飞腾环境

海光环境	整机环境
管理网IP	10.51.52.247
CPU	飞腾S5000cCPU
核心+线程数	128核心，256线程
操作系统	KOS 5.8 SP1
硬盘（挂载/home）	512 GB
内存	512 GB
搭载软件	瀚高数据库V9.0
测试环境	BenchmarkSQL

调优前TPMC得分



仓库数warehouses:	1000	Overall tpmC:	47647.00
并发终端terminals:	256	Overall tpmTotal:	105938.00

飞腾环境调优

1. 块设备优化

预读优化

- 操作系统会预先读取设备上的一定量的数据，以提高顺序读操作的性能
- 当应用程序请求某个数据块时，磁盘不仅返回该块的数据，还会提前读取下一个或一系列数据块到缓存中。
- 果后续请求的数据块恰好在预读的数据范围内，就能提高访问速度，减少 I/O 延迟。

```
# 设置/dev/mapper/keyarchos-root设备的预读值
sudo blockdev --setra 16384 /dev/mapper/keyarchos-root

# 设置/dev/sda2设备的预读值
sudo blockdev --setra 16384 /dev/sda2

# 设置/dev/mapper/keyarchos-home设备的预读值
sudo blockdev --setra 16384 /dev/mapper/keyarchos-home

# 设置/dev/sda1设备的预读值
sudo blockdev --setra 16384 /dev/sda1
```

飞腾环境调优

2. 数据库调优

- **max_connections**：根据系统能力，调整最大连接数。
- **shared_buffers**：分配大量内存用于缓存，提高I/O性能。
- **work_mem 和 maintenance_work_mem**：适当增大内存，提高排序和索引创建效率。
- **wal_buffers 和 wal_writer_delay**：加大 WAL 缓存，并减少 WAL 刷新的等待时间，提高写入性能。
- **effective_io_concurrency**：增加并发I/O读块数，适合多核CPU架构。
- **autovacuum_max_workers**：增加自动垃圾回收工作进程，提高数据库的维护效率。

```
# 连接设置
max_connections = 1000                  # 调整最大连接数以支持更多并发连接
unix_socket_directories = '.'           # 指定Unix套接字目录

# 内存配置
shared_buffers = 256GB                  # 使用更大的内存以提高缓存命中率
huge_pages = on                         # 使用大页提高内存效率
work_mem = 2GB                          # 为每个操作分配更多内存
maintenance_work_mem = 8GB              # 加速建立索引和维护操作
autovacuum_work_mem = 8GB               # 加速垃圾回收
dynamic_shared_memory_type = mmap       # 使用mmap提高共享内存性能

```

飞腾环境调优

```
# 垃圾回收设置
vacuum_cost_delay = 0                    # 垃圾回收不妥协
autovacuum = on                          # 启用自动垃圾回收
autovacuum_max_workers = 16              # 提高垃圾回收并发
autovacuum_naptime = 3s                  # 缩短自动垃圾回收探测间隔
autovacuum_vacuum_cost_delay = 0         # 不延迟垃圾回收

# 后台写入器设置
bgwriter_delay = 5ms                     # 提高脏页刷写频率
bgwriter_lru_maxpages = 2000             # 一次最多刷写更多脏页
bgwriter_lru_multiplier = 10.0           # 提高写入性能

# I/O和WAL配置
effective_io_concurrency = 2000          # 提高并发I/O读性能
wal_level = archive                      # 启用归档模式
synchronous_commit = off                 # 异步提交以提升写入性能
wal_sync_method = open_sync              # 使用O_DIRECT的fsync方法
full_page_writes = off                   # 关闭以提高性能
wal_buffers = 16MB                       # 提高wal缓存区大小
wal_writer_delay = 5ms                   # 提高wal写入速度

# 检查点和提交设置
commit_delay = 10                        # 分组提交等待时间
commit_siblings = 50                     # 触发分组提交的事务数
checkpoint_timeout = 30min               # 检查点间隔时间
max_wal_size = 512GB                     # 检查点之间的最大WAL大小
checkpoint_completion_target = 0.9       # 平滑检查点操作

# 查询优化器设置
random_page_cost = 1.0                   # 适合SSD的成本因子
effective_cache_size = 450GB             # 设置较大的操作系统缓存大小

# 日志设置
log_destination = 'csvlog'               # 日志输出格式
logging_collector = on                   # 启用日志收集
log_truncate_on_rotation = on            # 日志轮转时截断
update_process_title = off               # 禁用进程标题更新
track_activities = off                   # 禁用活动跟踪
```

飞腾环境调优

3. 内核启动参数优化

```
kernel /vmlinuz-3.18.24 numa=off elevator=deadline intel_idle.max_cstate=0 scsi_mod.scan=sy
```

4. 用户和进程资源限制

编辑 `/etc/security/limits.conf` :

```
# 文件描述符限制
* soft    nofile  1048576   # 增加最大打开文件数
* hard    nofile  1048576

# 进程数限制
* soft    nproc   65536     # 增加每个用户的最大进程数
* hard    nproc   65536

# 核心转储限制
* soft    core    unlimited # 用于调试分析
* hard    core    unlimited

# 内存锁定限制
* soft    memlock 1000000000 # 增加可锁定内存量
* hard    memlock 1000000000
```

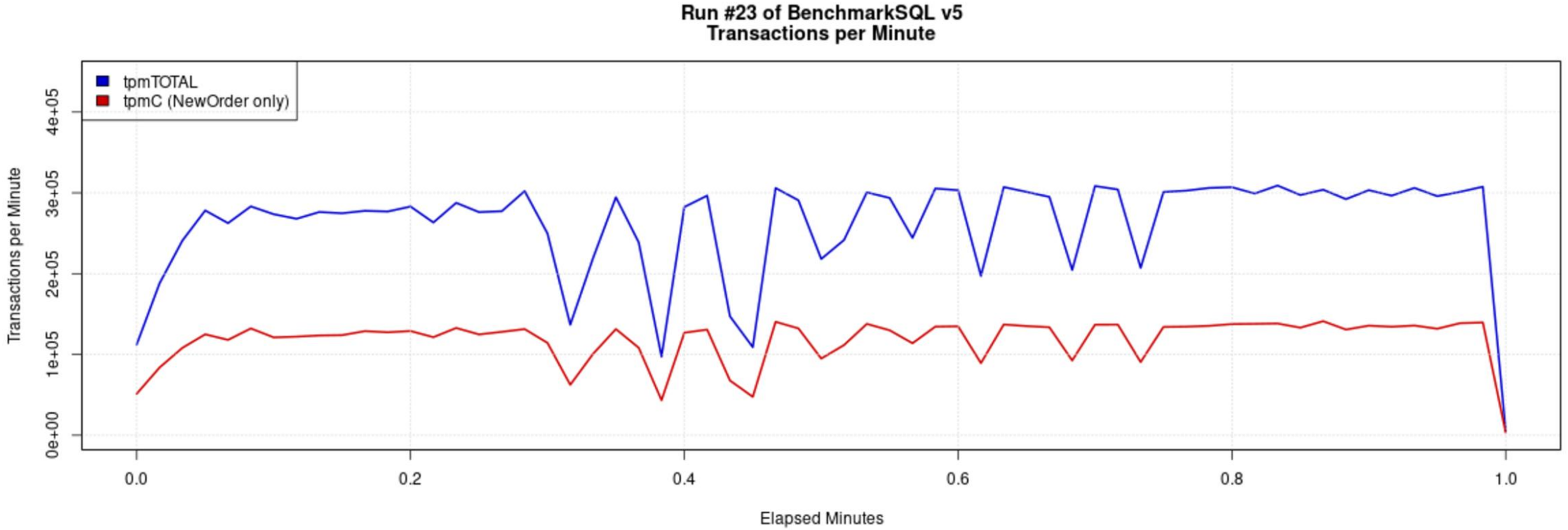
飞腾环境调优

5. 内核参数优化

```
# 共享内存设置
kernel.shmmax = 135497418752          # 最大共享内存段大小
kernel.shmmni = 4096                  # 最大共享内存段数

# 文件系统设置
fs.file-max = 10485760                # 系统最大可打开文件数
fs.aio-max-nr = 1048576               # 最大异步I/O请求数

# 内存管理设置
vm.zone_reclaim_mode = 0              # 禁用NUMA策略
vm.swappiness = 0                     # 关闭交换分区
vm.dirty_background_bytes = 204800000  # 后台刷脏页阈值
vm.dirty_expire_centisecs = 6000      # 脏页过期时间
vm.dirty_writeback_centisecs = 50     # 刷脏页进程唤醒间隔
vm.dirty_ratio = 90                   # 脏页占用内存比例
vm.nr_hugepages = 204800              # 大页内存数量
vm.overcommit_memory = 2              # 内存过度提交模式
vm.overcommit_ratio = 100             # 内存分配最大比率
```

```
# 网络设置
net.core.rmem_max = 16777216       # 最大接收缓冲区
net.core.wmem_max = 16777216       # 最大发送缓冲区
net.core.rmem_default = 1048576    # 默认接收缓冲区
net.core.wmem_default = 1048576    # 默认发送缓冲区
net.ipv4.ip_local_port_range = 9000 65535  # TCP/UDP端口范围

# 信号量设置
kernel.sem = 50100 64128000 50100 1280  # 系统信号量参数
```

调优前TPMC得分



仓库数warehouses:	1000	Overall tpmC:	269846.00
并发终端terminals:	512	Overall tpmTotal:	465599.00

CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

4 总结

调优方式总结

层面	优化内容
操作系统	- 文件系统优化：使用快速存储，减少磁盘 I/O
	- 网络优化：调整网络缓冲区，减少延迟
	- 内核参数优化：调整内存、I/O、NUMA设置
	- 用户资源限制：配置文件句柄和进程限制
	- 操作系统调度：优化进程调度，减少上下文切换
数据库	- 缓冲池：增加缓存池大小，减少磁盘访问
	- 日志优化：调整日志刷新、减少 I/O 操作
	- 并发处理：优化线程数和并行查询
	- 内存管理：合理分配内存，优化查询性能
	- 查询优化：优化索引、使用查询缓存
	- 垃圾回收：优化自动回收和后台写入

CAICT 中国信通院

ACTIC 全国先进计算技术创新大赛

感谢观看！

星期六不上发条

- 队长：李义
- 队员：李义、谢雯馨
- 2024年12月15日