

图1 核心平台架构

### 2.3 交易处理器

交易处理器将自身的作业 ID 置入处理组队列中, 由交易分发器读取处理组队列中的 ID 进行作业分配。交易处理器的创建过程见算法 3。

#### 算法 3. 交易处理器创建算法

1. 根据业务组编号读取配置参数作业优先级、运行次数。
2. 调用 CHGJOB 函数修改作业运行优先级。
3. 调用 STRCMTCTL 函数启动事务处理。
4. 循环调用核心交易主程序, 阻塞等待分发器分发的通讯句柄。
5. 判断核心交易主程序的返回码成功时提交确认事务, 失败时回滚事务。
6. 累计循环执行次数, 达到最大运行次数时提交结束作业命令 PJ 作业重启。

### 2.4 交易分组服务算法

交易分组服务算法, 根据交易报文中的交易码所属的业务属性和优先级, 将服务请求分配到具备不同系统资源优先级的队列中的作业, 具体如算法 4 所示。

#### 算法 4. 交易分组服务算法

1. 初始化, 执行算法 1、算法 2、算法 3。
2. 交易服务器接收外围程序通讯接口请求。
3. 交易服务器将 socket 句柄随机分配给交易分发器。
4. 交易分发器接收到交易服务器分配的 socket 句柄, 调用 peek 函数读取报文信息, 获取原交易码。根据原交易码取该交易对应的数据队列 DTAQ 及对应交易处理器组别。
5. 交易分发器调用函数 givedescriptor() 将 socket 句柄分发给对应的交易处理器。
6. 交易处理器调用函数 takedescriptor() 接收 socket 句柄。

7. 交易处理器调用 peek 函数预读取取出交易报文头。调用失败时登记异常日志、关闭通讯句柄、退出本次循环继续等待下次交易服务器分发的通讯链接。
8. 交易服务器读取服务器 IP, 根据报文头交易时间、客户端地址、IP 控制等进行安全性校验。
9. 交易服务器根据报文头中的原交易码, 调用 QUSRTVUI 函数取出用户视图对应的业务组号。赋值对应的处理作业数据队列名=分发器编号+业务组编号。
10. 交易分发器从执行的 DTAQ 中取得空闲的作业 ID, 将通讯句柄发送给交易处理器。
11. 交易处理器程序执行完将作业 ID 发送到对应的分组 DTAQ 中, 阻塞等待交易分发器程序传入的 socket 句柄。

## 3 实验与验证

本文采用 i750 的机器作为实验与验证环境, 物理配置 32 Core 处理器 (激活 12 Core), 主频 3.6 GHZ, 内存 150 GB, 采用 IBM 机器自带的资源监控工具。实验中设置队列数量为 2 个, 一个为高优先级 H (执行账务类交易或者高频交易), 一个为低优先级 L (执行低频的查询交易)。实验通过压力测试和稳定性测试进行比较<sup>[19]</sup>。

### 3.1 压力测试

压力测试比较监听-分发两层结构的核心平台 (以下简称“Core2”) 和具备分发服务的三层结构的核心平台 (Core3) 的性能。

#### (1) 单交易压力测试

单交易压力测试中选择 13 支交易进行单交易压力测试, 验证在 1500 个用户并发下的最大 TPS 情况, 如表 1 所示。

表 1 1500 个用户并发下单交易压力测试情况 (单位: 笔/s)

交易名称	特性	Core2	Core3
网银转账	H	414.962	408.715
根据账号查询活期余额	H	2568.037	4552.689
根据账户查询客户信息	H	4145.079	4321.736
验证账户交易密码	H	1713.243	2950.129
网上支付转账交易	H	613.404	596.241
CDM存款户名查询	H	1747.741	2557.467
ATM取款	H	643.374	653.252
根据卡号查询户名币种	L	2127.563	2044.485
贷款开销户查询	L	559.563	553.179
逾期贷款查询	L	325.556	321.441
贷款挂息明细查询	L	414.962	408.715

从表 1 中可以看出, Core3 通过分组服务, 在高并发时低优先级队列中因为具备较低的资源, 最大

TPS 低于 Core2,但是在高优先级的交易中具备更大的 TPS.

(2) 混合交易压力测试

混合交易中采用福建农信生产的高频 26 支交易,根据业务量占比设置压力测试数据.表 2 和表 3 是不同并发用户数下两种算法的 TPS 及 CPU 数据.从表 2 和表 3 可看出,Core2 和 Core3 在用户数 1502 时候达到了最大的 TPS 处理能力,分别为 1473.774 笔/s 和 1483.155 笔/s,Core3 在拥有更大的 TPS 同时比 Core2 拥有更低的 CPU 使用率.图 2 和图 3 分别显示 Core2 和 Core3 的机器性能情况,可以看出 Core3 与 Core2 相比,CPU 没有达到满载,也没有出现 CPU 资源等待,体现了分组分配带来的动态调整能力.

表 2 不同用户数下 TPS 比较表 (单位: 笔/s)

用户数	Core2	Core3
252	257.02	257.469
502	512.418	511.666
752	761.513	762.366
1002	1011.222	1011.222
1252	1259.716	1260.635
1352	1368.124	1364.218
1502	1473.774	1483.155

表 3 不同用户数下 CPU 比较表 (单位: %)

虚拟用户数	Core2	Core3
252	18.16	17.59
502	37.45	36.06
752	56.08	53.12
1002	74.09	71.91
1252	92.12	91.72
1352	93.42	90.77
1502	99.11	97.91

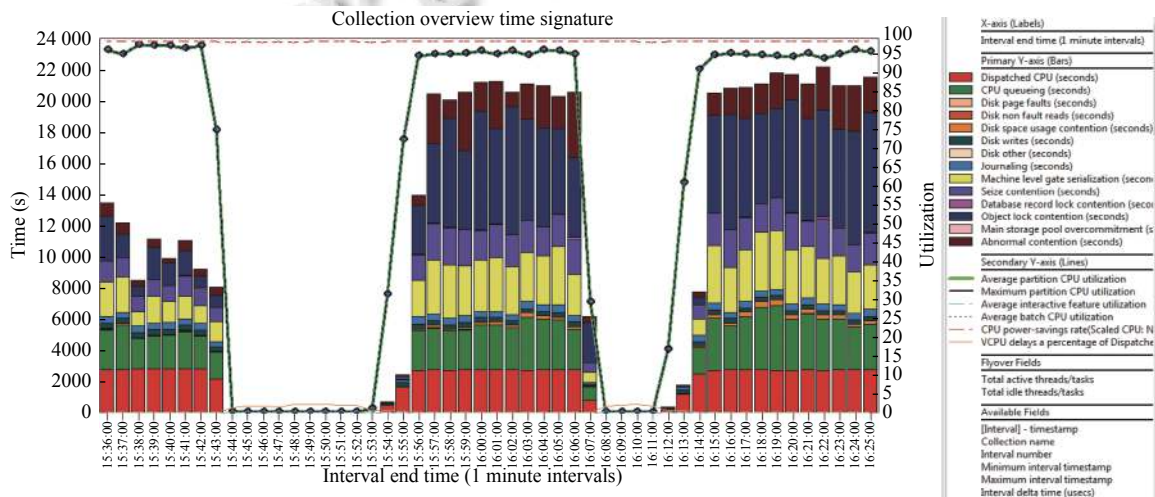


图 2 Core2 机器资源情况

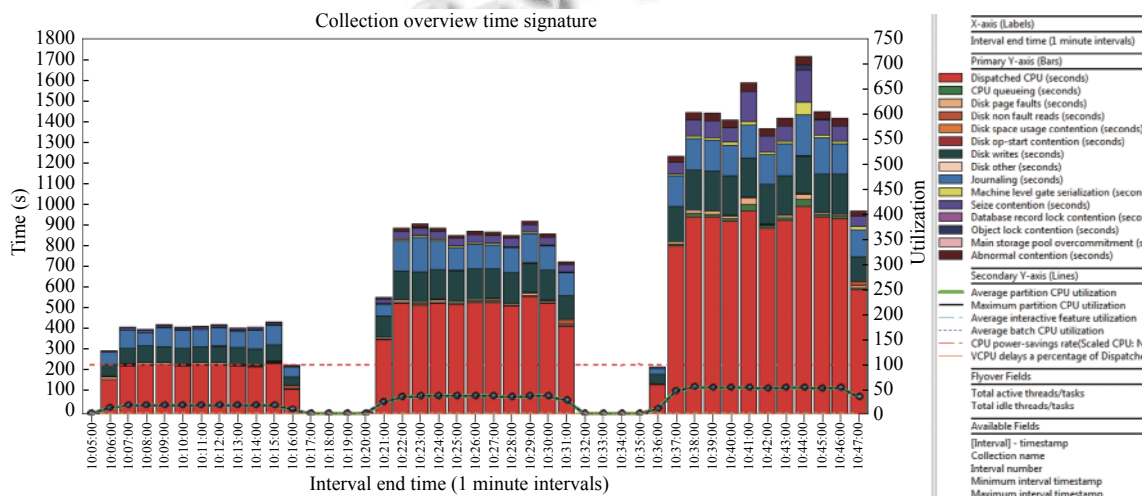


图 3 Core3 机器资源情况

### 3.2 稳定性测试

稳定性场景测试采用 Core3 混合场景测试中最优处理能力的 80% 发起并发压力. 设置并发 1250 个用户混合场景配置负载压力下持续稳定运行 12 小时, 稳定性测试过程共计成功完成了 52 906 925 笔交易, 交

易成功率 100%, 整个过程系统运行平稳, 最优 TPS 约为 1400 笔/s, 如图 4 所示. 图 5 显示了 i750 的机器资源使用变化, CPU、内存及 IO 读写资源使用情况也比较平稳, 未发现交易性能下降、内存泄露、异常退出等现象.

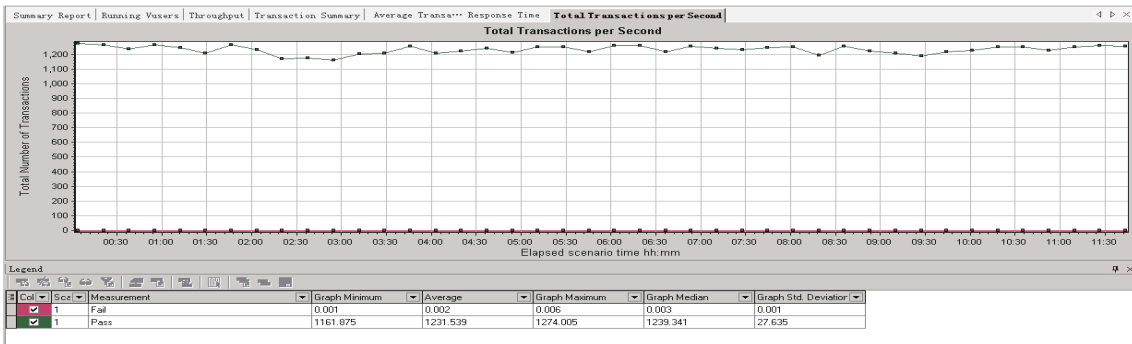


图 4 12 小时 TPS 变化

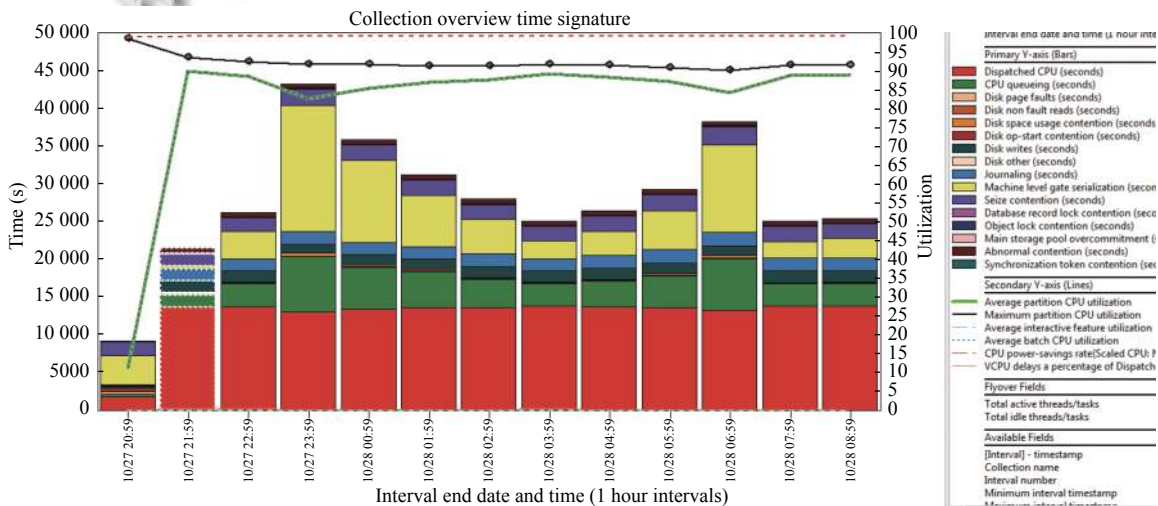


图 5 系统资源使用变化趋势图

### 3.3 实际应用效果

本文设计的“集中式分组交易核心平台”被应用在福建省农村信用社联合社(简称“福建农信”)的 AS400 系统中(i780, 64Core). 经过升级的核心平台处理能从升级前 755 笔/s 上升到 2500 笔/s, 系统资源占有情况下降了 1/3, 满足了福建农信的核心服务需求.

### 4 结语

随着金融科技的发展, 对于传统的核心系统的能力提出了更高要求. 不同银行对集中式、分布式架构

体系中的核心都做了大量的研究和工作. 本文基于 AS400 的集中式架构的银行核心技术体系, 设计了一种具备交易分组能力的核心平台, 设计涵盖交易服务器、交易分发器、交易处理器的三层架构, 结合报文中的业务性质, 实现处理作业按交易分组, 保障交易高峰期时关键业务的处理效率. 实验结果表明, 相较于传统的监听-分发两层结构的核心交易平台, 本文设计的具备交易分组能力的三层结构的核心平台, 能够实现不同交易级别的交易的不同资源调配, 在模拟仿真交易高峰期场景中, 具备较高的 TPS 处理能力, CPU、内

存及 IO 读写资源使用情况也比较平稳,同时在福建农信应用中也获得较好的效果.而本文的不足之处在于未对核心平台的批量处理进行专门的研究,在后续研究工作中,将进一步探索同时调度联机交易和批量交易的解决方案.

### 参考文献

- 1 李法杰,张睿.农村商业银行核心业务系统核算引擎现状分析与改进探讨.数字通信世界,2018,(3):86-87.[doi:10.3969/J.ISSN.1672-7274.2018.03.056]
- 2 成俊.银行核心系统“瘦身”后重新定义.银行家,2008,(5):124-127.[doi:10.3969/j.issn.1671-1238.2008.05.034]
- 3 马胜蓝.蝴蝶蜕变,打造新一代核心系统——福建农信核心账务系统架构整合实践.中国金融电脑,2017,(6):45-50.[doi:10.3969/j.issn.1001-0734.2017.06.011]
- 4 张相广.商业银行大型主机平台运维智能化转型探索与实践.中国金融电脑,2019,(6):52-56.[doi:10.3969/j.issn.1001-0734.2019.06.011]
- 5 薛勇.中国银行分布式架构的创新研究与实践.中国金融电脑,2016,(5):15-21.[doi:10.3969/j.issn.1001-0734.2016.05.003]
- 6 叶又升.商业银行如何“去 IOE”.金融电子化,2014,(8):26.
- 7 刘春霖.云计算在银行业应用趋势研究.数字通信世界,2019,(8):169.[doi:10.3969/J.ISSN.1672-7274.2019.08.128]
- 8 老木.银行核心业务系统采用互联网分布式架构替代主机集中式架构的研究.金融电子化,2015,(6):58-60.
- 9 孙明明.人民银行分布式架构应用研究.金融科技时代,2019,(11):32-37.[doi:10.3969/j.issn.2095-0799.2019.11.007]
- 10 金磐石.分布式架构在银行核心业务系统的应用.计算机系统应用,2017,26(6):46-52.[doi:10.15888/j.cnki.csa.005852]
- 11 李伟.金融科技发展与监管.中国金融,2017,(8):14-16.
- 12 刘晨东,侯晓强,翟宾,等. FinTech 时代商业银行分布式核心系统研究与实践.中国金融电脑,2018,(7):36-41.[doi:10.3969/j.issn.1001-0734.2018.07.008]
- 13 张帝,汤勇.基于开放式平台的银行核心业务系统服务器部署规划浅析.中国金融电脑,2018,(12):47-49.[doi:10.3969/j.issn.1001-0734.2018.12.011]
- 14 吕远阳.热点数据处理模型在商业银行核心系统中的应用.数字通信世界,2019,(2):200,284.
- 15 乔克.银行核心业务系统发展指引——“新一代核心系统”建设的重要意义及思路.银行家,2012,(9):121-124.
- 16 冯华彬.银行核心系统支持 7×24 小时不间断服务分析.中国金融电脑,2013,(5):62-65.[doi:10.3969/j.issn.1001-0734.2013.05.015]
- 17 Myers SE. RPG IV Programming on the AS/400. Upper Saddle River: Prentice Hall, 1998.
- 18 周双娥,欧中红,袁由光.网络化的在线诊断、故障隔离、动态重组技术.计算机应用研究,2003,20(1):92-93.[doi:10.3969/j.issn.1001-3695.2003.01.029]
- 19 孙婷,芦英明,陈中伟.软件系统性能测试方法及流程规划.自动化与仪器仪表,2018,(1):166-169.