

200127

上海市浦东新区杨高南路288号19-21层
上海金融期货信息技术有限公司
李悦萌 女士

关于：申请号为**202210049847.3**的专利申请案
申 请 人：上海金融期货信息技术有限公司
发明名称：一种快速结算系统
我方编号：CNJRQH-0136.21A564

李悦萌 女士

您好！

关于本案的第1次审查意见通知书，已按贵方指令于期限内答复国知局，附件是答复文本的副本，请查收。

如有问题请随时联系我们。谢谢！

顺颂 业祺



专利代理师 施浩
2024年12月31日

Encls.

hdc.doc

意见陈述书

① 专 利 或 申 请 利	申请号或专利号 2022100498473
	发明创造名称 一种快速结算系统
	申请人或专利权人（第一署名人）上海金融期货信息技术有限公司
② 陈述事项：关于费用的意见陈述请使用意见陈述书(关于费用) 以下选项只能选择一项 <input checked="" type="checkbox"/> 针对国家知识产权局于 <u>2024</u> 年 <u>09</u> 月 <u>10</u> 日发出的 <u>第 1 次审查意见通知书</u> （发文序号 <u>2024091002258250</u> ）陈述意见。 <input type="checkbox"/> 针对国家知识产权局于__年__月__日发出的__（发文序号__）补充陈述意见。 <input type="checkbox"/> 针对国家知识产权局于__年__月__日发出的药品专利权期限补偿审查意见通知书（发文序号__）陈述意见。 <input type="checkbox"/> 主动提出修改（根据专利法实施细则第 57 条第 1 款、第 2 款的规定） <input type="checkbox"/> 公布公告事项 <input type="checkbox"/> 其他事宜	
③ 陈述的意见： 意见陈述请见附件	
④ 附件清单 【附件名称】权利要求书 【附件名称】修改对照页 【附件名称】其他证明文件 已备案的证明文件备案编号：__	

意 见 陈 述 书

⑤ 当事人或专利代理机构

上海专利商标事务所有限公司

意见陈述书正文

尊敬的审查员，您好！

针对本案的审查意见，申请人的意见陈述如下。

一、针对权利要求1至5不具备创造性的审查意见

权利要求的修改：

将“其中，产品组是各个期货交易所产品划分而成，客户组是由所有投资者划分而成；

其中，结算进程模块、结算总控模块和结算管理端采用C++全内存、多线程、组件化设计，结算进程模块、结算总控模块和结算管理端都封装成动态库”的技术特征加入到原独立权利要求1中。

上述修改没有超出申请文件原始记载的范围，符合专利法第33条的规定。

意见陈述理由：

申请人认为，修改后的独立权利要求1是具备创造性的，理由如下。

本案所要解决的技术问题是：现有期货公司结算系统由于结算一次耗时较长，如果期间一旦业务有遗漏或出错需要重新结算，系统执行时间就增加更多，在结算的时间窗口有限的情况下，期货公司业务人员为了保障公司客户数据准确、完整压力巨大。

为解决上述技术问题，本案的关键技术特征是：本发明具有以下几个创新点：1、用C++全内存、多线程、组件化设计，将结算系统划分为多个子模块、各个模块都封装成动态库。2、多个结算进程并发结算，支持平行扩展，提升系统的性能和容量。3、业务上支持分交易所、分客户组、分产品组结算，如果出现问题，只需要重结出问题的客户组或者产品组。

对比文件1的技术摘要为：一种期权保证金预清算方法、装置及终端设备，

所述方法包括：当满足预置触发条件时，从交易所获取客户当日已完成的交易数据，所述预置触发条件包括预置的清算周期、预置的清算时间节点和接收到操作员的清算指令中的一种或多种；根据客户的现金结算金额、前一日的衍生品保证金账户数据文件、前一日的期权资金净额交收文件以及所述当日已完成的交易数据进行计算，得到衍生品保证金账户余额。对比文件1可以解决现有技术中保证金预清算功能只支持盘后保证金预清算，易使客户无法及时划入资金造成损失，同时，保证金预清算的过程中并非考虑现金结算的情况，结果不正确，不利于证券期货公司的风险管理的问题。

对比文件2的技术摘要为：一种CDVS多进程驱动方法、装置、电子设备及存储介质，该方法包括：接收应用程序的调用请求信息；为应用程序分配文件数据结构和空闲的功能数据结构；根据调用请求信息、空闲的功能数据结构和文件数据结构，通过空闲的功能数据结构对应的进程调用CDVS驱动程序，驱动CDVS芯片进行数据处理；根据文件数据结构，读取处理结果。对比文件2为每个进程分配了文件数据结构和功能数据结构，通过空闲进程队列实现多个进程的调度，实现CDVS驱动程序的多进程调用。当进程请求不到CDVS芯片的服务时，就会休眠等待，有空闲资源时会被自动唤醒进行工作，从而避免人工干预，能够充分提高CDVS芯片的硬件资源使用率。

将本发明的修改后的独立权利要求1的技术方案与对比文件1-2的技术方案相比，区别如下。

1、本发明配置为选择需要结算的交易所、客户组、产品组，产品组是各个期货交易所产品划分而成，客户组是由所有投资者划分而成。

对比文件1-2中并没有详细揭示上述的分组方式。

上述区别技术特征并不是本领域的惯用技术手段。若审查意见认为是惯用手段，应不难通过检索提出对比文件，仅在未提出任何书面证据的情况下就简单断言区别技术特征是惯用手段，并不符合审查指南第二部份第八章第4.10.2.2节的规定：「审查员在审查意见通知书中引用的本领域的公知常识应当是确凿的，如果申请人对审查员引用的公知常识提出异议，审查员应当能够说明理由或提供相应的证据予以证明」。

且，上述区别技术特征可以为本发明带来如下的技术效果：

业务上支持分交易所、分客户组、分产品组结算，如果出现问题，只需要重结出问题的客户组或者产品组。

提高结算效率：通过将产品组按照期货交易所进行划分，系统可以针对每个交易所的具体规则和产品特性进行更高效的结算处理。这避免了不同交易所间产品结算规则的混淆，使得每个结算进程在处理同一交易所的产品时能够更加专注和高效。

增强结算准确性：产品组和客户组的明确划分，确保了结算过程中使用的数据（如产品价格、交易量、客户持仓等）的准确性和一致性。每个产品组对应特定的交易所规则，而客户组的划分则有助于区分不同投资者的交易和持仓情况，从而避免结算过程中的数据混淆和错误。

灵活性和可扩展性：这种分组方式使得系统能够灵活地适应不同交易所的新产品上线或投资者群体的变化。当新的交易所产品被引入或投资者群体发生变化时，系统可以轻松地调整产品组和客户组的划分，而无需对整个结算流程进行大规模的重构。

优化资源分配：通过结算总控模块对结算指令的排队和分配，系统可以更有效地利用空闲的结算进程资源。产品组和客户组的划分使得结算指令可以被更精确地分配到具有相应处理能力的结算进程上，从而优化了整个系统的资源利用率。

便于管理和监控：分组后的结算数据在结算数据库中更加有序和清晰。这使得系统管理员和监控人员可以更方便地追踪和分析每个交易所、产品组和客户组的结算情况，及时发现潜在的问题并进行处理。

综上所述，新增的分组技术特征在上述技术方案中起到了提高结算效率、增强结算准确性、增加系统灵活性和可扩展性、优化资源分配以及便于管理和监控等技术效果。

2、本发明中，结算进程模块、结算总控模块和结算管理端采用C++全内存、多线程、组件化设计，结算进程模块、结算总控模块和结算管理端都封装成动态库。

对比文件1-2中并没有详细揭示上述的分组方式。

上述区别技术特征并不是本领域的惯用技术手段。若审查意见认为是惯用手段，应不难通过检索提出对比文件，仅在未提出任何书面证据的情况下就简单断言区别技术特征是惯用手段，并不符合审查指南第二部份第八章第4.10.2.2节的规定：「审查员在审查意见通知书中引用的本领域的公知常识应当是确凿的，如果申请人对审查员引用的公知常识提出异议，审查员应当能够说明理由或提供相应的证据予以证明」。

且，上述区别技术特征可以为本发明带来如下的技术效果：

提升系统性能：C++作为一种高效的编程语言，结合全内存设计，可以显著减少磁盘I/O操作，提高数据处理速度。多线程设计使得系统能够同时处理多个结算指令，充分利用多核CPU的计算资源，从而大幅提升系统的整体性能。

增强系统可扩展性：组件化设计使得系统的各个模块（如结算进程模块、结算总控模块、结算管理端）具有高度的独立性和可重用性。当系统需要扩展新功能或优化现有功能时，可以方便地添加或替换相应的组件，而无需对整个系统进行大规模的修改。此外，将模块封装成动态库（如DLL或.so文件）进一步增强了系统的模块化程度，便于在不同平台或系统间进行部署和迁移。

提高系统稳定性和可靠性：多线程设计虽然可以提升性能，但也带来了线程同步和互斥等复杂问题。通过合理的线程管理和同步机制，可以确保系统在高并发环境下的稳定性和可靠性。同时，动态库的使用也减少了系统间的耦合度，降低了因单个模块故障而导致整个系统崩溃的风险。

便于维护和升级：组件化和动态库的设计使得系统的维护和升级变得更加容易。当某个模块出现问题时，可以单独对该模块进行修复或升级，而无需影响其他模块的正常运行。此外，动态库的更新也更为灵活，可以在不重启整个系统的情况下替换或升级库文件。

支持灵活的部署和配置：通过将系统划分为多个独立的组件并封装成动态库，可以支持更加灵活的部署和配置方式。例如，可以根据不同的业务需求或系统环境，选择性地加载或卸载某些组件，以达到最佳的系统性能和资源利用率。

因此，修改后的独立权利要求 1 具有突出的实质性特点和显著的进步，具备创造性。依附其的全部从属权利要求均依法具备创造性。

以上陈述，如有不妥，请指正并给与再次修改的机会，也可通过 021-34183200-2409 和代理人联系。申请人愿意积极配合以使本申请能够早日授权。谢谢！

权 利 要 求 书

1、一种快速结算系统，其特征在于，系统包括结算数据库、结算进程模块、结算总控模块和结算管理端，其中：

5 结算管理端，配置为选择需要结算的交易所、客户组、产品组，将结算指令发送到结算总控模块；

 结算总控模块，配置为对接收到的结算指令进行排队，检查是否有空闲的结算进程，如果有空闲的结算进程就发送结算指令给该空闲的结算进程；

 结算进程模块，设有多个结算进程，结算进程配置为收到结算总控模块发来的
10 的结算指令后，对结算管理端中所选择的交易所、客户组、产品组进行结算，并将结算结果写入到结算数据库；

 结算数据库，用于保存结算进程模块的结算结果；

 其中，产品组是各个期货交易所产品划分而成，客户组是由所有投资者划分而成；

15 其中，结算进程模块、结算总控模块和结算管理端采用 C++全内存、多线程、组件化设计，结算进程模块、结算总控模块和结算管理端都封装成动态库。

2、根据权利要求 1 所述的快速结算系统，其特征在于，结算进程模块的结算进程是并发的，支持平行扩展。

20

3、根据权利要求 1 所述的快速结算系统，其特征在于，结算进程模块包括资金子模块、数据库子模块、清算核心子模块、文件处理子模块，其中结算进程模块配置为执行以下的处理：

 在快速结算系统启动时，资金子模块、数据库子模块、清算核心子模块、文件处理子模块进行初始化；
25

 在初始化完成后进行数据加载：文件处理子模块加载交易所的结算文件，数据库子模块加载结算需要的数据，其中结算需要的数据包括费率、上日持仓、上日资金，其中数据库子模块用于对结算数据库的读写操作；

数据加载完成后进行数据过滤：资金子模块根据结算文件输入、上日持仓数据，对上一步骤加载的包括费率、账户在内的结算数据进行筛选，过滤不需要处理的数据；

- 基于过滤后的结算数据进行资金持仓计算：资金子模块将筛选后的结算数据
- 5 传递给清算核心子模块，由清算核心子模块基于结算数据进行资金持仓计算；

对资金持仓的计算结果进行汇总和保存：资金子模块对资金持仓计算的计算结果进行汇总，并调用数据库子模块对汇总结果加以存储。

权 利 要 求 书

1、一种快速结算系统，其特征在于，系统包括结算数据库、结算进程模块、结算总控模块和结算管理端，其中：

5 结算管理端，配置为选择需要结算的交易所、客户组、产品组，将结算指令发送到结算总控模块；

 结算总控模块，配置为对接收到的结算指令进行排队，检查是否有空闲的结算进程，如果有空闲的结算进程就发送结算指令给该空闲的结算进程；

 结算进程模块，设有多个结算进程，结算进程配置为收到结算总控模块发来的
10 结算指令后，对结算管理端中所选择的交易所、客户组、产品组进行结算，并将结算结果写入到结算数据库；

 结算数据库，用于保存结算进程模块的结算结果；

 其中，产品组是各个期货交易所产品划分而成，客户组是由所有投资者划分而成；

15 其中，结算进程模块、结算总控模块和结算管理端采用 C++全内存、多线程、组件化设计，结算进程模块、结算总控模块和结算管理端都封装成动态库。

~~2、根据权利要求 1 所述的快速结算系统，其特征在于，产品组是各个期货交易所产品划分而成，客户组是由所有投资者划分而成。~~

20

~~32、根据权利要求 1 所述的快速结算系统，其特征在于，结算进程模块的结算进程是并发的，支持平行扩展。~~

~~43、根据权利要求 1 所述的快速结算系统，其特征在于，结算进程模块包括
25 资金子模块、数据库子模块、清算核心子模块、文件处理子模块，其中结算进程模块配置为执行以下的处理：~~

~~在快速结算系统启动时，资金子模块、数据库子模块、清算核心子模块、文件处理子模块进行初始化；~~

在初始化完成后进行数据加载：文件处理子模块加载交易所的结算文件，数据库子模块加载结算需要的数据，其中结算需要的数据包括费率、上日持仓、上日资金，其中数据库子模块用于对结算数据库的读写操作；

5 数据加载完成后进行数据过滤：资金子模块根据结算文件输入、上日持仓数据，对上一步骤加载的包括费率、账户在内的结算数据进行筛选，过滤不需要处理的数据；

基于过滤后的结算数据进行资金持仓计算：资金子模块将筛选后的结算数据传递给清算核心子模块，由清算核心子模块基于结算数据进行资金持仓计算；

10 对资金持仓的计算结果进行汇总和保存：资金子模块对资金持仓计算的计算结果进行汇总，并调用数据库子模块对汇总结果加以存储。

~~5、根据权利要求1所述的快速结算系统，其特征在于，结算进程模块、结算总控模块和结算管理端采用C++全内存、多线程、组件化设计，结算进程模块、结算总控模块和结算管理端都封装成动态库。~~