

CPU 虚拟实验平台的设计与实现研究

雷煊彬 周富肯

(广东东软学院, 广东 佛山 528225)

摘要: 对于高校学生来说, 实验在学生深化理论知识方面作用显著, 可以有效培养学生的实践动手能力。近些年以来, 实验也在各高校得到了重视。远程虚拟实验具有较强的灵活性, 在实验方面可以进行应用。这种方式不但有效缓解了传统实验模式的困境, 同时也促使高校的实验室建设速度更快。本文借助目前的 B/S, 设计了 CPU 虚拟实验平台, 文中从不同方面对系统进行了论述, 以供参考。

关键词: CPU; 虚拟实验平台; 设计; 实现

近些年以来, 高校对于实验教育的重视度越来越高, 这也就促使高校开始对实验进行改革。由此也出现了很多以实验为基础的课程, 这些不但能够强化学生对各专业知识理解, 同时也使学生的实践能力得到提升。高等教育现在已经逐渐被普及, 招生规模也呈现扩大趋势。传统实验模式需要消耗较大的人力、物力, 显然已经无法满足当前的教学需求, 所以这就需要高校教育工作者对此进行探索, 打破传统的实验教学模式, 给学生提供更具创新性的学习平台, 培养他们的自主性和创新意识。文中将会从不同方面对 CPU 虚拟实验平台进行论述, 对其功能等加以阐述。

一、高可用和高并发设计

在分布计算中, 可用性与并发是需要考量的问题。高可用性确保了系统能够随时接入, 而高并发性则确保了多个人在此平台上可以同时开展实验。

(一) 系统拆分

早期的远程实验平台是单体结构, 所有涉及到的代码都被集成到了一个工程之中, 整个体系非常复杂, 一些细微的变化都会对功能的正常应用产生影响。为了保证每一个函数都能正常工作, 我们必须做很多的回归测试, 这从一定程度上会减缓平台开发速度。因此, 在采取了分布式体系结构之后, 就必须按照具体的要求将其拆分开。

1. 拆分需求

(1) 除已完成的远程教学实验系统外, 与之相匹配的数据库管理系统、考核评价系统也已启动。同时, 为降低多个系统及程序间的相关性与耦合性, 以及多个不同应用程序在同一编码库中进行编码(我们已经利用 Gitlab 构建了仓储管理系统), 需要将该系统进行拆分, 以避免多个系统相互影响。

(2) 学校为该项目提供了虚拟服务器, 但是该平台必须要有与实物相连的实验箱。为了保证科研成果的安全性, 必须将这两个部分进行分离。

(3) 提高远程实验平台的使用效率, 尽管单体系统也能够达到高可用性, 但从维护的观点来看, 每次对系统进行修改时, 都需要对各节点进行重新设置, 这反而会给系统埋下很多不可控风险。

2. 拆分实现

原来的远程实验平台是独立的体系结构, 在为这个项目提供域名和虚拟服务器的时候, 出现了问题, 那就是不能与实验板机柜相连, 所以必须把与实验有关的功能分离开来。基于“高聚合、少耦合”的分解原理, 从单个结构中提取与实验有关的部件作为实验服务系统。另外, 该实验服务系统中包括了远程实验平台的主要功能, 因此将该实验服务系统进行拆分, 可以有效提高软件的安全性。

(二) 数据拆分和读写分离

1. 数据拆分: 数据的拆分和数据库的拆分, 可以分成水平拆分与垂直拆分两种。远程实验平台按照业务特性对表格进行纵向分类, 在各个数据库中建立相应得分。通过对数据进行拆分, 不仅实现了对数据库功能的明确划分, 而且还分担了各数据库的压力。用户体系如图 1 中的用户信息、实验配置的表格。同时, 该实验服务系统还提供了一些与实验有关的表格。

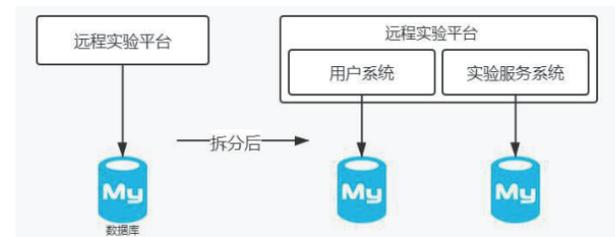


图 1 数据拆分

2. 系统数据库读写拆分: 原有的单体结构下, 客户端拥有很多表格, 而且随着时间的推移, 其数目也在不断增加。同时, 大多数远程实验平台都是基于业务查询的需求, 只有很少的涉及到写, 因此使用读取和写入相结合的方法可以有效提高数据库的负荷。本次我们研发的远程实验平台用户系统运用了“一主一从”的方式, 即由主库进行写, 从库则负责读取信息。当主机写入之后, 会同步传送给从库。

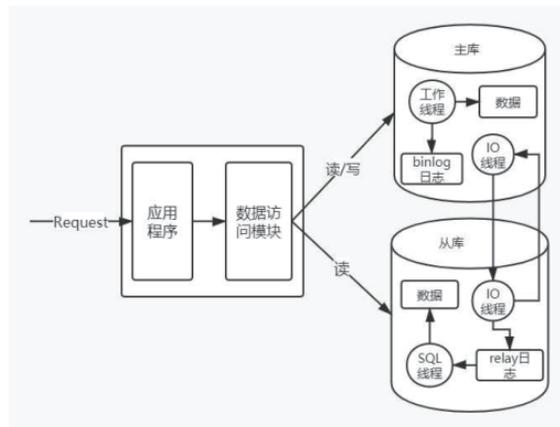


图 2 用户系统读写分离实现原理

(三) 缓存设计和 session 一致性设计

1. 缓存设计

该系统中, 为能够有效提升远程实验平台自身的性能, 研究中把对用户系统信息进行了缓存。缓存存储了两种类型数据, 一种为涉及到频繁访问的热点数据, 该类型数据信息置于缓存当中,

能够减少数据库 qps。另外，由于将缓存数据存储在系统的内存当中，因此能够实现数据的快速查询，提高了整体的反应能力，还增加了并发数量。另外就是一些比较耗费时间的数据，例如在进行虚拟实验的时候，每一次都要进行数据库检索，会导致整个系统反应时间变慢。将信息进行缓存合并之后，再将其存储到缓存中，可以显著加快系统的响应速度。

用户系统整合 redis 实现缓存流程：

(1) 导入了 spring-boot-starter- 预测。

(2) 应用程序 .yaml 来设置 redis 链接地址。

(3) 重新设置 redis 的序列化器，把数据序列化成 JSON、重写 key 生成器。接着，编程人员可以应用 RestTemplate 编写操作 redis，如 redisTemplate.opsForValue。

(4) 配置缓存、CacheManagerCustomizers。

(5) 测试系统缓存性能。

用户系统经过整合和缓存之后，有时缓存与数据库数据无法保证一致，这对用户有一定影响。为确保数据库与缓存双写保持一致，本次系统中采取了缓存 + 数据读写模式。执行读操作时，需要先读缓存。若没有命中就需要读数据库，再把所读取的数据放到缓存当中，然后返回响应数据，反之则直接发挥数据库。进行写操作的过程中，可以直接对数据库进行更新和优化，随后将缓存删除。之所以在写操作时将缓存删除，而不是对缓存进行更新，是因为在用户系统中的部分缓存情境，所要缓存的数据不仅仅是数据库当中直接取得的值，该字段所对应的缓存需要相关人员查询表格信息，对其进行处理之后方可获取最新缓存值。此外，缓存的升级代价很大，若系统对存储的表格信息经常更新，意味着缓存也会经常更新，不过缓存的读取次数有限。若是把缓存删除，那么缓存按照一次计算即可，这将会为相关企业节约成本。

2.session 一致性

将传统的单体架构转换成分布式架构之后，首要处理的问题就是 session 的一致性。单个服务端程序仅需将会话保存在服务端，而分布式程序则可能无法获得会话。因此，本次远程实验平台采用了后端统一集中存储的方案。该平台采用单独的缓存服务器对 session 进行保存，用户发出请求后，该系统按照用户名字将会话信息保存在 redis 中，从而保证了站点的稳定性和并行性。系统中 session 被独到的频率较高，选择数据库对数据进行存储，会增加数据库负担，所以我们使用 redis 来保存 session。

二、CPU 实验功能设计

(一) CPU 虚拟实验面板文件标准设计

CPU 实验的目的在于检验 CPU 设计是否正确，了解指令的运行流程，以及系统与硬件之间的联系，从而进一步了解计算机结构与工作原理。CPU 实验的程序是，对有关实验虚拟实验面板进行加载，按照 CPU 种类对主存储内容、控制存储器内容 / 指令存储器内容等进行编辑。结束实验主板测试之后，对 FPGA 进行程序设计，把实验线路文档下载到实验板上。FPGA 编程工作环节完成后，就能进行微单步、指令单步等操作，对相关信息记录、信息流以及控件状态等进行更新。另外，可以对虚拟元件进行操作，根据实际需求更新控件的状态。

CPU 实验虚拟面板除了 CPU 的名称和版本号，还包括了七个模块，其中大部分都是关于数据流的实时展示，以及数据的更新。

1) CAVas: CPU 实验虚拟面板与逻辑线路虚拟实验面板内容是完全相同的。这部分中重点记载了虚拟控制板中所采用的各种虚拟元件信息，如开关、按键，数码管等。此外，还包括了一些

其他信息，如背景图片、虚拟元件尺寸等。在对逻辑电路实验以及 CPU 实验中涉及到的虚拟元素的操作浏览器向服务器传送了一个虚元素信息，然后由服务器对信息进行加工，并将其作为扫描链信息传送到实验面板上，然后再将该新的数据传送到实验面板上，对所得到的数据进行分析，随后把所得到的数据转化成虚拟元件信息，由前端对其进行更新，例如按下按钮时，LED 灯就被点亮，数码管数值也会有所变动。

2) ctrlInfo: 这部分中记录和保存了 cpu 的全部控制信号数据信息，各含有七种特性。

3) busInfo: 若此 CPU 信型号具有总线，那么此部分就会记下总线在虚拟面板上的线路总数及拐点信息。假如 CPU 幸好在数据传输中应用的为专用通道，那么这部分就为空。

4) 数据通道信息: 数字化系统中传输的通道被称作数据通路，本次研究中，将数据通路分割成了若干个小段，通过该通道的即时展示确定哪个通道被加到该通道中。Key 是这个片段在数据通路中的标识，而 linePath 则是数据通路片段起点坐标讯息。

5) cpuInfo: 这部分主要记录了 CPU 实验数据，其中包含了主内存控制存信息，当它是 RISC 的 CPU 时，可以看作是数据存储器信息和指令存储器信息、支持的功能信息 cpuFunction 和显示的窗口信息 cpuWindow。

6) 微型结构: 该系统向使用者展示了基本性能，使其无需人工编写代码，只需要根据微指令编码器机器的功能，就可以实现编码。具体来说，需要首先按照微指令形式将其划分成几个区域，每个区域都有一些数据信息，其中包含了 key、bitWidth、fieldNumber 和 microcommand。

7) 注册信息部件: 寄存器数据主要显示在运行记录和虚拟面板框图中相应寄存器图形中，其主要是在所述运行的记录和所述虚面板结构中对应的寄存器图案中，虚拟面板当中这部分主要记录的是寄存器位宽 regWidth、寄存器名称以及寄存器 key。装载 CPU 的虚拟实验板时，系统就会根据预先的指令，对面板数据进行分析解析。解析的流程为先获取 cpu 名，主内存起始地址、寄存器组所有名称、窗口显示信息、微指令编码器信息、canvas 部分、主内存增量传送到前端，即可显示出相关信息。随后把控制信号信息等信息储存在缓存当中，便于发挥其其他功能。例如，在产生信息流的时候，控制信号、数据路径等信息都是从缓存中获得的。

通过对虚拟实验面板的设计，可以让平台实验具有很大的灵活性，不仅可以提前在后台设置多个用于教学实验的虚拟实验面板，让使用者直接对其进行检验，同时还可以将其用于创新实验。使用者可以依据自身实际硬件需求，生成相应的虚拟实验面板。虚拟实验面板是一种新型的实验平台，其既能有效增强实验的有效性，又能给使用者留下了较大的创造余地，使其能进行更多的实验活动，激发使用者的兴趣和动力。

(二) 实验数据交互设计

该系统的关键是实现了实验数据的实时交互。其中包括两种类型的接口，第一种是按照指令生成 JTAG 边界扫描链路的数据，并将其输入到实验板中，获得寄存器及相应的控制信号，然后通过一般的流程进行实时渲染，在虚拟的屏幕上修改进度纪录和流程。二是根据按键、开关互动类的虚拟元件状态，生成 JTAG 边界扫描链数据，并将其传送到实验面板，取得 LED 灯、数码管灯等显示器类虚部件的状态资料。进行逻辑回路实验时，只要将第 2 种要求发出，CPU 实验都会进行发送，因此系统的这个功能说明了 CPU 实验中数据交换的原则和步骤。

（三）主存控存读写

1. 控存读写设计

在 CPU 实验过程中，利用控存窗口与 CPU 实验控制存储器进行相互作用，并向控存窗口中写出相应的微指令，或将从所述控存窗口中读取的所述微指令更新到所述控存窗中。

把向控存窗口中写的微指令代码分成两种情形，分别是在 FPGA 程序设计之前和程序之后。FPGA 程序设计之前，CPU 实验并未把电路文件下载到实验板，因此不能写出微指令，在编写单格微指令的时候，仅限于对其进行形式的检测。关于格式方面的检查主要是针对微指令长度和字符方面。如伪指令的格式出现有误，则会传回“输入控存格式错误”的指令，并会在控制视窗中清除该错误的微指令。如果数据信息是正确的，那么数据就会被暂时保存到控存窗口中，直到 FPGA 的程序被完成之后，才会被集中写出来。编写单格微指令后，FPGA 对其进行修改，将其直接写到控存窗口中。如果写控制操作错误，将会返回一个“写入控存失败”。

2. 主存读写设计

主存窗口用来与 CPU 实验中主存存储器进行交互：将主存窗口中显示的内容写入到主存储器；或者在主存窗口中刷新显示从实验板中读出的内容。

（四）通用微指令编码器和主存汇编功能

1. 通用微指令编码器

CPU 中控制器是最重要的构成部分，其主要任务是按照命令的内容解码生成控制信号，并对其进行有效控制，使其有序运行，以完成命令所要求的各项功能。其实现的途径主要有组合逻辑和微程序两种。本课题的通用微指令编码器就是针对使用微程序产生控制信号的 CPU，否则，该系统将无法工作。为程序控制就是把编程概念融入到硬件逻辑控制当中，对微指令进行编码和存贮，把一个命令的运行分解成若干条命令的读取与执行。通过对微指令编码器的研究，减少了手工编写代码，缩减了烦琐的运算过程，可以明显提高实验效率。

2. 主存汇编功能

机器指令是用二进制编码来指示就安吉执行某些特定运算的指令。汇编语言是一种运算代码，以辅助记忆的方式取代了机械指令。在各种装置上，与各种计算机语言的命令系统相对应。主存汇编功能是将编译程序代码转换为相应的计算机代码，现在只有在 JUC-II 处理器上才能被使用。

三、管理功能设计

（一）实验板管理设计

远程虚拟实验平台的实验板数量有限，为了提高实验板的使用效率，就需要设计出高效的实验板分配和回收策略。其次实验板有概率发生崩溃，平台要确保分配给用户的实验板时可用的。

1. 实验板分配和回收

实验板分配：当使用者启动实验时，若使用者已使用实验板，就需要返回到“成功获取实验板”。如果用户未占用实验板，就会询问资料库中是否有一片已指派给使用者的实验板。使用者在使用该测试版时，会先判定有无闲置的实验板。如果此时有闲置的实验板，系统会直接分配给用户，反之则会开始排队。当用户处于队列时，需要将请求的线程、会话等信息存储到自己的数据库中，然后把请求对象放到请求的队列中。此后，排队线程池分配一个线程给当前请求，并运行此线程，然后将请求线程挂起。然后就会启动闲置的实验板，如果没有闲置的话，正在排队的线程就会被封锁，直至闲置的实验板显示出来。若存在闲置的实验板，

就会从要求列中取出请求对象，并把实验板分配给该用户。

2. 实验板工作状态检查

该系统中的实验结果全部来源于实际的实验板，与传统的模拟方法比较，在保证实验结果可供进一步研究的前提下，也存在一定不足。那就是真正的实验班可能出现坍塌，造成实验板不能用。基于此，为进一步确保用户能够宣导能用的实验板，我们在系统设计中进行了多种设置，以此确保用户实验可以正常进行。

（1）使用者启动实验开始申请实验板时，系统获得指定使用者的实验板的工作代码，若无此面板，就会提醒使用者再次提交新的实验板，并设定修改实验板的状况为不可使用。

（2）采取定时任务的方式，对实验板进行自动化管理。每个实验板都会在五分钟内进行一次进行自动检测。轮询的作用包括：第一，将已损坏的实验板从资料库中删除；第二，将已修复的实验板设定为可以使用。另外，轮询还给实验板提供了额外优势，那就是在电源恢复后，实验板的情况不需要人为改变，只需要进行一遍轮询，就可以恢复实验板的正常使用。

（二）实验数据管理设计

1. 实验项目配置管理

为了让使用者能够更快地启动实验，我们还设计并实现了“实验课题”的组态管理系统。在使用者按一下课程的时候，客户机就会发出六条要求。

对于使用者所选的实验计划，将由“/api/index/start”来确定。当选定一个实验时，将弹出来的是一个逻辑线路实验或一个 CPU 实验，由 CPU 来确定。通过对“课程”和“创新性”的实验，来确定是否载入“虚拟实验”或“空白”的“实验”。设计式和确认式将确定实验线路档案是否载入资料库或使用手工上载。

2. 实验记录管理

通过对实验数据进行分析和对实验结果进行分析，并对实验结果进行了分析。在使用者执行实验时，系统会将使用者之实验资料，例如：实验种类，开始时间，实验名称，实验板 IP 等资讯进行纪录。通过 Spring 架构中的 AOP 技术，提取和包装了企业过程中的“记录”和“数据”，把“数据”和“数据”分开，并在适当的地方对“数据”进行“横向切割”。

在远程实验平台上，存在着许多实验运算的运算，若在各个运算中都要编写相应的实验运算程序，不仅会导致程序重复，而且很难进行系统的维修。很明显，在更改实验纪录函数时，要对所有的处理方法进行修正是很不明智的。因此，本项目将实验记录的产生与更新提取为切片，并嵌入实验作业关联的过程，从而提升了程序的重用与可维护性。

四、结束语

本次系统设计最初时，我们查阅了大量的资料，借鉴了先进的信息技术，才设计了该系统，并且分析了国内外的情况。我们希望通过本次研究，可以为高校提供可靠性十足的数据库，确保其安全性与完整性，提高高校实验教学的效率。

参考文献：

- [1] 王京. 计算机网络虚拟实验教学平台的设计与实现探索 [J]. 数码世界, 2018 (10).
- [2] 张海波. 计算机网络虚拟实验教学平台的设计与实现探索 [J]. 现代交际, 2018 (21): 2.