# **论高可靠性系统中软件容错技术的应用**

摘要

  2019年3月，我单位联合某高校研发了《程序在线评测比赛考试系统》。系统以程序代码在线提交自动评测功能为核心，分为题库模块、评测机模块、实验作业模块、考试模块、比赛模块、抄袭判定模块、用户管理模块等，支持对接教务平台。在项目中我担任系统架构师，负责架构设计工作。

  本文以该平台的评测功能、数据库功能、自动出卷功能为例，主要论述了软件容错技术和方法在项目中的具体应用。系统通过采用集群化的形式进行应用部署，通过主备形式的数据库部署进行软件容错，通过程序设计方面进行软件的容错与避错。以上措施对提高系统的可用性、安全性和可扩展性方面起到了很好的效果，满足了系统的性能需求，保证了系统的稳定运行，获得了用户的一致好评。

正文

  笔者在一个专为高校建设计算机专业智能教学一体化平台的单位任职，过往成果有《计算机组成原理仿真实验系统》等。2019年3月，我单位联合某大学研发了《程序在线评测比赛考试系统》项目（以下简称为“OJ系统”），以取代原有传统的编程上机考试平台。

  系统以程序代码的在线提交自动评测功能为核心，主要分为题库模块、评测机模块、实验作业模块、考试模块、比赛模块、抄袭判定模块、用户管理模块等。题库模块主要负责试题和测试用例的管理，用户根据试题要求编写程序代码提交到系统，系统将测试用例与程序代码发送给评测机模块，由评测机自动编译、执行、判分，并将结果发送给其他相关模块进行统计；实验作业模块用于在线布置作业，从题库中选取试题，设置截止日期等要求；考试模块用于学生在线考试，按教师预先设置的参数自动从题库随机抽题生成试卷，以及向教务平台上传考试成绩；比赛模块主要用于ACM竞赛的培训；抄袭判定模块用于鉴定代码与他人代码雷同率；用户管理模块负责用户信息的管理。在这个项目中，我担任了系统架构师的职务，主要负责系统的架构设计相关工作。

  OJ平台以微服务架构作为基本架构风格，基于Spring Cloud框架，采用Java语言开发，将平台服务划分为三类，分别为核心服务、平台Web服务、平台保障服务。其中核心服务主要分为程序评测服务、试卷判分服务、比赛判分服务、代码相似度比对服务等。平台Web服务主要提供给用户使用的界面。平台保障服务主要包括WebAPI框架、平台报警服务、MQ消息中间件服务等。对于核心服务、平台Web相关的服务必须要能保证在考试、竞赛期间的稳定运行，对软件的可靠性要求非常高，所以在OJ平台中的这些核心服务就必须具备一定的容错能力，在某个服务运行时出错的情况下不能影响到整个系统的正常运行，这就要求在软件架构设计中必须考虑到软件容错技术的应用。

  提高软件系统可靠性的技术主要分为：容错技术、避错技术。容错技术主要通过冗余实现，分为结构冗余、时间冗余、信息冗余、冗余附加。结构冗余又分为静态冗余、动态冗余和混合冗余。软件避错技术主要有N版本程序设计、恢复块方法和防卫式程序设计。结合Web软件的性质，我主要采用了集群技术、数据库主从方式和程序设计方面来进行软件的容错与避错处理。下面就从以上三方面详细讨论我所采用的容错技术和方法。

1. 集群技术

  平台中各服务如果在运行时部署在一台服务器上，那么当服务器故障，整个平台将不能再提供任何服务，所以一般规模的应用需要采取集群的部署方式。以OJ系统中的评测机模块为例，程序评测属于耗时操作，用户提交的程序需要统一存入MQ队列中排队，然后由评测机依次进行读取和判分。如果评测机只部署了单个应用，那么当评测机因执行用户提交的不安全代码导致宕机时，后续所有待评测程序都将被阻塞。通过多机同时部署相同功能的评测机集群，再结合服务管理中心的软负载均衡功能，在系统正常运转时，多个评测机并行工作，可提高评测效率，在部分评测机宕机时，评测工作交由其他正常评测机分担，以留出供故障评测机重置恢复的时间，防止了因单台评测机宕机致整个平台不可用的问题出现，保证了业务的连续性，提高了系统的可靠性。

2. 数据库主从部署

  对于OJ平台来说，后端数据库存储的稳定性是极其重要的，所有的程序提交记录、考试答题记录、成绩统计记录都将存储到数据库中。如果数据库在运行过程中频繁宕机，那么带来的问题是不可容忍的。这里就要求数据库存储有非常高的可靠性，同时有很强的容错性，在这里主要采用主从式方法部署数据库结构，实现读写分离的架构。在不出问题的情况下，对于一些时效性要求不高的场合，从库可以分担一部分读流量，当主库发生读写问题时，可快速由其他的从库升级为主库，继续服务，达到容错的效果。此外，OJ系统还增加了数据库宕机报警的功能，防止宕机的数据库实例过多以至于并发高的情况下没有可用的从库可升级为主库，提高了平台的可靠性。数据库所依赖的硬件存储采用了基于RAID6的磁盘阵列，最高可容许两块硬盘同时发生故障，降低数据文件丢失的可能性。

3. 程序设计方面

  根据以往的设计经验，系统的不可靠大部分是由于程序内部的设计、网络请求参数的配置或者连接池参数的配置不当所导致的。所以通过程序设计方面进行软件的容错是非常重要的。最普遍的就是防卫式程序设计，例如平台中的考试服务，需要根据预设的试卷规格，调用题库服务中的题目信息来自动生成试卷。当大量考生同时进入考试功能时，题目服务在被调用的一刻如果出现网络拥塞或者丢包，这时候考试服务必然会收到抛出的错误信息，如果没有通过恰当的容错处理，那么一定会给考生显示进入考试失败的错误。这里我采用了try-catch机制加10次重试的容错处理机制，就解决了进入考试时因网络原因导致进入考试失败的问题。

总结

  通过采用了以上容错技术的方法和措施后，评测机模块达到了六余度，数据库达到了四余度，考试出卷功能等待时间控制在了5秒内，使整个系统达到了可靠性和实时性的要求。

  系统自2019年10月正式上线已运行一年有余，在学校的日常教学考试和竞赛培训中投入使用，截至目前已有3000以上的学生用户、评测了70000条以上的程序代码，获得了单位同事领导和学校教师们的一致好评。

  然而在使用过程中，系统还是出现了可靠性方面的故障。比如评测机虽然使用了集群部署，但由于提交作业的学生初次学习编程、对试题理解有误，短时间内提交了大量不安全代码，导致所有评测机实例全部宕机。我们对问题分析和试验后，将原部署在物理机上的评测机改为部署在虚拟化平台，结合心跳机制，自动回滚快照重置失效实例，省去了物理机系统重启的时间，将单个评测实例的恢复时间缩短到了10秒内。另外，我们使用了机器学习技术，对代码进行预判断，有宕机风险的代码将会被分流和延后处理。最终这个问题得到了解决。

  实践证明，OJ系统项目能够顺利上线，并且稳定运行，与系统的可靠性设计密不可分。经过这次软件容错技术的方法和实施的效果后，我也看到了自己身上的不足之处，在未来还会不断地更新知识，完善本系统在可靠性方面的设计，使整个系统能够更加好用，更有效地服务于高校师生。