

《开源软件设计与开发》课程总结

姓名：王清帅

学号：52194501004

1、开源理解——借助开源工具改善生活

从本科开始，我一直受益于 Github 提供的各种开源工具，Github 的开发者们提供了各种免费的工具，帮助我快速的开发我的项目。我在大三中第一次使用了 WPF 框架的 MaterialDesignInXamlToolkit 工具包，开发了自己的第一个游戏——21 点，这之后我还在我的日常中使用了越来越多的开源工具。

从本学期开始，我开始尝试使用更多的开源工具来改善生活，体验开源工具带来的美好。我本身是一名果粉，平时使用 MBP 学习，对 Windows 的需求不大，因此我尝试借助开源工具将我的台式机改造成一个可以帮助我提升学习和工作质量的设备，下面我描述一下我当前运行的开源工具。

- 1) Ubuntu, 开源的操作系统, 借助该系统, 我搭建了各种服务
- 2) Supervisor, python 开发的进程管理工具, 使用该工具, 我实现了各种脚本和软件服务的自动化管理
- 3) Ecnu.py 校园网登陆脚本, 因为台式机连接的校园网需要登陆验证才可以连接外网, 因此我使用了一个极客在 Github 上分享的登陆脚本进行登陆验证, 并对脚本做了优化, 使其可以自动登陆
- 4) Netatalk, 开源的 afp 实现, 可以实现 mac 上的远程备份功能, 借助该工具, 我可以实时的将自己的数据快照到服务器中, 保证电脑的数据安全性
- 5) Docker, Linux 容器的一种封装, 使用该工具, 我部署了 6 和 7 的服务。
- 6) Overleaf 开源版, 由于 overleaf 官网写论文很慢, 我在自己的服务器上部署了他们的开源版本, 方便我编写论文
- 7) pptp-client+ ipsec-vpn-server 链接集群, 因为 mac 到 10.15 之后不再支持 pptp 的远程 vpn 协议, 因此我考虑使用我的 Ubuntu 服务器当做 vpn 跳板机, 我使用 ubuntu 上的 pptp-client 连接集群, 然后使用 ipsec-vpn-server 的 docker 容器提供 l2tp 的 vpn 对外服务接口供我的 mac 连接, 从而实现我连接实验室集群的目的

以上这些是我已经运行在我的服务器上的工具, 我最近仍在准备部署运行一些开源的工具, 进一步为我的生活服务, 包括 Nextcloud, 一个开源的网盘, 用于替代 onedrive ; vscode online, 提升我编码的体验。另外有两个基础的开源工具包为我的开发提供了坚实的基础, 他们分别是 python 和 python 的各种开源库, java 的运行时 open-jdk。不经意间, 我的生活中已经用上了如此多的开源工具, 我也将会使用更多的开源工具来改善我的生活。

开源让生活更美好！

2、开源贡献

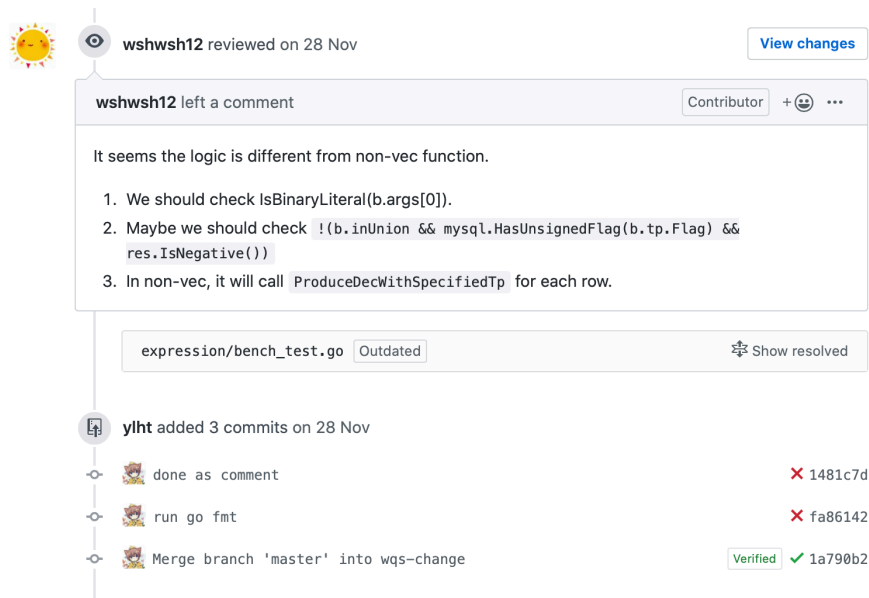
在本次课程中, 我参与了 TiDB 开源项目的开发工作, 并成功提交了 2 个 PR。为 TiDB 向

量化执行引擎的实现贡献了自己的力量。我提交的两个函数实现分别为 builtinCastStringAsDecimalSig 和 builtinLeastTimeSig。下面具体描述一下我的具体工作内容。

1) builtinCastStringAsDecimalSig

链接：<https://github.com/pingcap/tidb/pull/13778>

这个函数将 String 类型的数据转换为 Decimal 类型的数据，在我实现的第一个版本中，我参照了 builtinCastStringAs*（cast string 为其他类型）的实现思路，然后运行通过了测试样例，提交之后，reviewer 反馈告诉我需要考虑更多的情况，我仔细阅读了他的建议，然后按照他的建议进行了修改，最终完成了 PR 的提交，这是我第一次和社区中的开发者进行交流，感觉很开心。

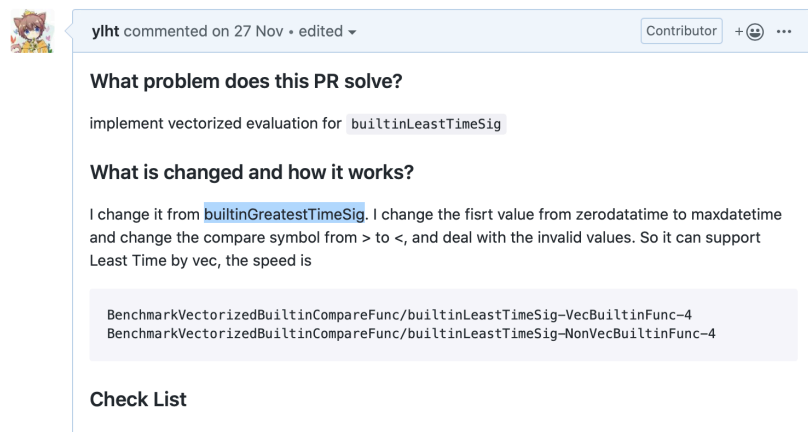


The screenshot shows a GitHub pull request interface. At the top, a comment from user 'wshwsh12' is displayed, dated '28 Nov'. The comment text reads: 'It seems the logic is different from non-vec function.' followed by a numbered list of three points: 1. 'We should check IsBinaryLiteral(b.args[0]).', 2. 'Maybe we should check !(b.inUnion && mysql.HasUnsignedFlag(b.tp.Flag) && res.IsNegative())', and 3. 'In non-vec, it will call ProduceDecWithSpecifiedTp for each row.' Below the comment, a file 'expression/bench_test.go' is listed with a status of 'Outdated'. To the right of the comment is a 'View changes' button. Below the comment, the commit history is shown, with three commits by user 'ylht' on '28 Nov': 'done as comment' (hash 1481c7d), 'run go fmt' (hash fa86142), and 'Merge branch 'master' into wqs-change' (hash 1a790b2, marked as 'Verified').

2) builtinLeastTimeSig

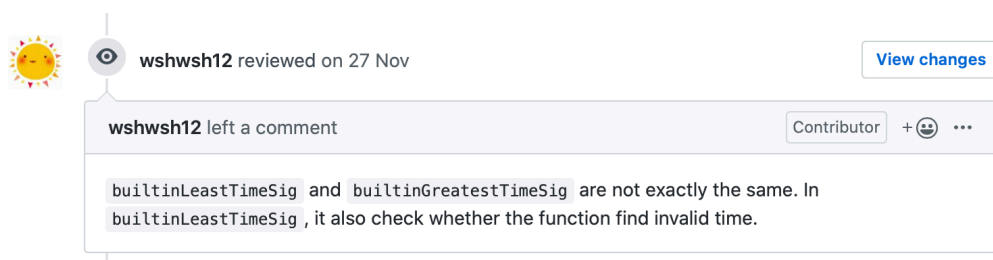
链接：<https://github.com/pingcap/tidb/pull/13781>

这个函数在一组时间中找到最小值，我发现这是 Cmp 文件中最后一个没有被实现的函数，因此可以参考的方法很多，恰好 builtinGreatestTimeSig 函数也已经被实现了，因此我简单的修改了 builtinGreatestTimeSig 的实现方式，就进行了提交，提交如下：



The screenshot shows a GitHub pull request comment from user 'ylht' dated '27 Nov'. The comment is titled 'What problem does this PR solve?' and contains the text: 'implement vectorized evaluation for builtinLeastTimeSig'. Below this, it is titled 'What is changed and how it works?' and contains the text: 'I change it from builtinGreatestTimeSig. I change the first value from zerodatatime to maxdatatime and change the compare symbol from > to <, and deal with the invalid values. So it can support Least Time by vec, the speed is'. Below the text, there are two benchmark names: 'BenchmarkVectorizedBuiltinCompareFunc/builtinLeastTimeSig-VecBuiltinFunc-4' and 'BenchmarkVectorizedBuiltinCompareFunc/builtinLeastTimeSig-NonVecBuiltinFunc-4'. At the bottom, there is a section titled 'Check List'.

但是，reviewer 告诉我，这两个函数并不相同，我在这个函数中需要考虑更多的情况，应该检测是否在 least 中发现了无效的时间。



根据 reviewer 的建议，我重新修改了代码。修改时我发现在这个文件中没有可以参考的函数，因此我把重点放在了原函数的改写上，我重新阅读了原函数，然后提出了一种向量化实现的方案，最后做了提交。但是这次又遇到了新的问题。



Reviewer 给我反馈了一种新的情况，我的代码并不能 cover 中这种情况，因此我尝试把我使用的数据结构从 queue 改成 array 以解决该问题。



经过和 reviewer 的多次交互，我终于完成了此次提交，实现了这个复杂的功能。

在本次的实践中，我最中完成了 2 个 PR，体验到了为开源社区贡献力量的开心，感觉开源从遥不可及变得很轻松很愉快，很高兴课程给我一个这样的机会让我成长。

3、课程反馈

课程工作量适中，但是希望开办一些交流分享会，让大家各抒己见，聊聊各自对开源的看法，头脑风暴一下，或许能产生更多有新意的 idea。