

# 计算机系统会议论文是如何评审的

陈海波

上海交通大学

关键词：论文评审 计算机系统领域

**编者按：**正如人们所公认的，计算机系统领域的论文评非常严格。该领域的学术会议一直由美国麻省理工学院、加州大学伯克利分校、斯坦福大学、卡耐基梅隆大学等顶级高校和谷歌、微软等大公司主导。欧洲、亚洲国家与美国有很大差距。例如，整个亚洲学术界在计算机系统领域 SOSP、OSDI、EuroSys 等几个顶级会议上发表的论文总数近年来只有 10 篇左右。要改变这种情况，除了要提高计算机系统领域研究水平，增强自身科研实力外，了解这个领域论文评审体系也很重要。上海交通大学陈海波教授是我国少数在计算机系统领域国际会议发表论文的作者之一。因此，本刊特别邀请他介绍一下计算机系统学术会议论文评审的经历与体会，以飨读者。

与所有学科一样，一个好的论文评审体系是学术会议生命力与学术声誉的重要保障。理解学术会议论文评审的过程对于开展相关科学研究有一定的帮助。作为一名计算机系统领域的研究者，笔者有幸成为一些重要学术会议的程序委员会委员，参加了 ACM 操作系统原理大会 (ACM Symposium on Operating System Principles, SOSP)、欧洲计算机系统大会 (ACM European Conference on Computer Systems, EuroSys)、Usenix<sup>1</sup> 技术年会 (Usenix Annual Technical Conference, Usenix ATC) 等会议的论文评审工作与程序委员会会议，对计算机系统领域的论文评审流程和

录用标准有了一定的了解。本文将介绍对计算机系统领域论文评审的体会，希望对同行们有所帮助。

## 计算机系统领域会议介绍

计算机系统领域学术会议狭义上是指 ACM SIGOPS (ACM 操作系统专业组) 和 Usenix 发起与支持的一些会议，如 SOSP/OSDI (Usenix Symposium on Operating System Design and Implementation, Usenix 操作系统设计与实现研讨会)，EuroSys, Usenix ATC 等。会议论文录用数量通常较少，一方面由于论文中提到的方法与理论的

可行性和有效性需要通过实际系统来验证，研究工作周期相对较长；另一方面由于计算机系统领域普遍采用单轨 (single-track) 会议议程，以便参会人员专注于会议报告，使得会议的主题报告分组数量有限。因而，每届会议录用的论文数目通常不超过 30 篇。

此外，众多 IT 巨头公司，如谷歌、微软、脸谱、亚马逊、威睿、雅虎等，通常将其核心技术与系统 (如 MapReduce, Dryad, GFS, VMware ESX, Dynamo, BigTable, Spanner 等) 通过这些会议以学术论文的形式公开，因此学术界必须直面来自工业界的竞争，这进一步加大了在这些会议上发表论文

<sup>1</sup> 高等计算机系统协会。

的难度。到目前为止，亚洲学者在 SOSP/OSDI 发表的学术论文数目仍然为个位数，除微软亚洲研究院以外，国内学者尚未在 OSDI 上独立发表过学术论文。

尽管这些会议录用论文数量较少，但是参会人员通常较多，并且参会人员都非常用心地去听论文报告，讨论也非常热烈。每届

根据惯例，各个会议的指导委员会提前一到两年选举出程序委员会主席，然后再由程序委员会主席邀请领域内的学者组成程序委员会。通常来自工业界与学术界的成员比例为 1 : 3 到 1 : 2 左右。与相关领域其他学术会议一样，计算机系统领域的各个学术会议依赖于志愿者的服务。担任程序委员

分配待审的稿件。

由于投稿数量逐年增加，为了保证审稿质量，大部分学术会议采用多轮审稿制度（如图 1）：

- 第一轮，为一篇论文安排三个左右的程序委员会委员进行评审，第一轮结束后根据审稿意见与分数进行淘汰，进入第二轮后，再安排两到三个委员进行评审。

- 第二轮评审完之后，根据审稿意见与分数进行淘汰。第二轮的淘汰标准比第一轮要严格些，一般要求至少有一名委员愿意为一篇平均分数不错的论文争辩。

- 第三轮，一般由两名左右的程序委员会委员对论文进行终审。对于一些特别出色的论文，在第二轮评审后，如果所有的委员都认为论文非常出色并给出了“接收”级别的分，就可以安排一名委员进行审稿或者直接进入程序委员会讨论。

- 第三轮评审完成后根据总体审稿意见，设定一个标准决定进入程序委员会讨论的论文列表。如果一些稿件在这个阶段仍然有较大的争议，程序委员会主席会安排额外的委员或者小领域专家进行审稿。因此，一般论文都有 5~7 个审稿意见，有些甚至达到 9 个。

- 在程序委员会讨论环节，会议通常会为每篇论文安排一位讨论主持人（discussion leader，或称为 advocator），一般由对这篇论文持正面观点并且评审自信度较高的委员担任，或者请一位之前没有审过这篇论文的委员担任。讨论主持人负责仔细阅读论文与审稿意

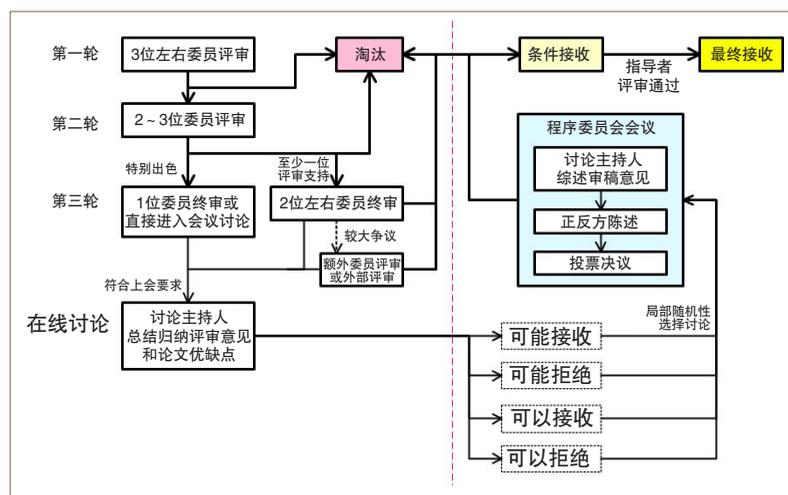


图1 计算机系统领域学术会议论文评审流程示意图

SOSP/OSDI 录用的论文只有二十多篇，但参会人数均在 500 人以上。并且这些会议论文的平均影响因子大都处于各类数据库如 Citeseer(X) 与清华大学 ArnetMiner 等的前列。

## 程序委员会的构成

与会议论文数量少相对应，这些学术会议程序委员会规模也小，规模通常在 30 人左右，SOSP/OSDI 只有 25 人，委员会成员主要是领域内的著名学者或近期研究表现突出的学者。

会的成员通常为义务劳动。

## 论文评审过程

计算机系统领域会议投稿通常不细分主题。每个委员在接收投稿前通常会选择多个感兴趣的审稿主题。投稿结束后，审稿系统（通常采用哈佛大学埃迪·科勒（Eddie Kohler）的 HotCRP 系统）会根据作者提交论文时选择的主题进行自动匹配。每个委员再对每篇论文进行浏览，并对自己感兴趣的论文进行竞拍，最后程序委员会主席根据竞拍数量和平衡性为每个委员

见,总结归纳一篇论文的优缺点以及正反方的意见,根据程序委员会之前的在线讨论,争取达成共识。

总之,计算机系统领域会议论文的大部分审稿意见具有建设性并且较为详尽,一些论文获得的评审意见甚至比论文的篇幅还要长。程序委员会成员试图通过提供建设性建议以帮助投稿者获得有用的反馈。有兴趣的读者可以参考莫西·罗斯科(Timothy Roscoe)撰写的“如何为系统会议写审稿意见”<sup>[1]</sup>。

担任程序委员会的委员需要亲自完成所有的评审工作,只有当多数委员对论文的评审意见较低时才去寻找外部评审。因此,每个委员通常需要仔细阅读30~50篇左右的论文<sup>2</sup>,并给出详细的审稿意见。微软亚洲研究院的周礼栋研究员曾说过,担任SOSP的程序委员会委员意味着审稿期的几个月内不能接受其他的审稿任务,或占用时间过多的其它工作。

计算机系统领域的会议也较少为论文提供辩驳(rebuttal)的时间<sup>3</sup>。一位资深学者认为,程序委员会委员通常对自己的评审意见是高度自信的,辩驳在增加作者与审稿人的工作量的同时并不大可能改变审稿的结果。

在整个审稿与讨论过程中,对于是采用单盲还是双盲的方式,学术界存在一些争议:单盲可以让审稿人更加系统地评价一个工作,

而双盲可能会更公平些<sup>[2]</sup>。ACM系统会议通常采用双盲方式,而Usenix系统会议则通常采用单盲方式。

## 程序委员会会议

计算机系统领域的学术会议通常都要求所有程序委员会委员参加面对面的会议。有的会议目前允许少部分委员选择不参加面对面会议,即选择成为轻量级程序委员会委员(light PC member),但也必须完成所有的审稿流程。

在程序委员会会议开始前,程序委员会主席通常根据讨论主持人的总结与线下讨论的意见把要讨论的论文分为几类,如可能接收(likely accept)、可能拒绝(likely reject)、可以接收(may accept)、可以拒绝(may reject)等,分别列入列表中。在开会时再对各列表进行逐一讨论。为了讨论的公平性,一种做法是对各类的讨论采取局部随机方式,即对一个列表中的论文讨论了一部分后,切换到另一个列表进行讨论。

程序委员会会议通常需要一到两天的时间,因而需要严格保证讨论的效率和公平性。会议的大致流程是:针对一篇论文,首先由讨论主持人综述审稿意见,包括论文的主要内容、优缺点,以及审稿委员之间是否已经形成一致意见,然后再由对论文持不同意见的

正方与反方分别阐述,其他委员提问,最后由委员会主席进行调查与组织投票,以决定论文是否录用。一篇论文的讨论大都控制在十几分钟以内,对争议较大的论文,讨论时间会长很多。

在程序委员会会议结束后,委员会主席会将委员会讨论的意见和审稿意见一起发给作者。有些会议可能会组织委员将录用论文的审稿意见进行总结并公开(如EuroSys 2012)。

目前,许多会议的论文录用均采用有条件的指导性接收(conditional/shepherd accept)的方式,并且为每篇论文安排一位委员担任这篇论文的指导者。作者在收到通知后,需要按照审稿意见,将在最终版本中计划所作的修改一一列出,并在截止日期前两周将修改稿发给指导者。指导者对论文最终版进行预审,确认相关问题已解决,也可以再提出进一步的修改与调整意见。作者修改完成后再次发给指导者确认,最后指导者向程序委员会主席确认论文已经通过检查可以发表。如果论文的修改没有达到指导者的要求,可能会被淘汰。

## 感受与体会

**撰写逻辑严密的论文** 一个领域通常存在一套大家公认的评价论文的方法。计算机系统领域

<sup>2</sup> 如OSDI 2012年每个委员的审稿量平均为45篇。

<sup>3</sup> EuroSys曾提供过辩驳的机会。

的研究大多属于实证科学研究，因而系统和方法的实用性、有效性与新颖性一样重要。大部分论文的研究成果均需要开发实际的系统来验证方法的可行性与有效性，而非仅仅依靠理论分析。计算机系统领域论文的评价标准是：

- 1 试图解决一个重要的问题；
- 2 提出一个有趣并且强有力的方法；
- 3 展示方法的实用性与有效性；
- 4 对方法进行恰当总结；
- 5 清晰地表明作者完成的工作；
- 6 明确指出工作的创新。

这些评价标准也体现在各个会议的征文中。上述六点要求中前两点者要求比较主观，这需要对领域发展有一定的判断。后四点要求较客观。罗伊 (Roy) 和雷德尔 (Reddel) 撰写的“如何 (不) 写出一篇好的系统论文” (How (and How Not) to Write a Good Systems Paper)<sup>[9]</sup> 以第九届 SOSP 的审稿过程与方法为例，介绍了如何撰写一篇好的计算机系统论文。

根据笔者的评审经验和与其他资深学者的交流，我国的很多投稿论文存在研究话题的选择与论文写作两个问题。有些论文的工作仅仅是解决了好多年前流行的话题中的一些小问题，这让评审人认为论文没有满足第一点“试图解决一个重要的问题”或第二点“提出了一个有趣并且强有力的方法”。当然，这并不是说我们一定要去追踪当前流行的话题，提出一个新问题或者对一个老问题提出新颖、突破性的解决方法更重要并且更容易被接收。

论文写作上的问题包括英文语法的问题。但如果论文可读性强，只是存在部分语法或拼写错误，审稿人通常可以接受，并且会在论文修改环节帮助作者改进。论文写作中存在的较多问题还是论文的严谨性。一些论文没有清晰地描述出作者到底完成了哪些工作、没有完成哪些工作，给审稿人的感觉是作者在试图隐藏一些信息，从而导致论文被拒。此外，论文的逻辑性与严密性也很重要，从立论到方法的提出、设计的实现与试验、以及总结均须环环相扣，相互呼应。另一个常见的问题是，作者在论文开头提出了一个问题，结果到结束都没有很好地解决这个问题，这无异于“挖坑自埋”。还有一个问题是，论文提出了一个问题，并找到了解决问题的理论方法，但是略过了中间具体设计与实现细节，直接给出了实验验证结果。这种方式会让审稿人怀疑这个理论是无法实现并应用到真实系统中的。

#### 突出表现工作的关键特色

一篇研究论文应该至少传递一个关键信息，例如有一篇论文提出“许多大规模数据并行应用可以通过 MapReduce 进行有效抽象与执行”<sup>[6]</sup>，整篇论文就应该围绕该论点展开讨论与佐证。计算机各领域都非常关注论文中的研究工作能否带动本领域的发展。评审人对一个工作的简单评价是：“我从这个论文中学到了什么？这篇论文中提到的工作如何带动领域发展？”一篇好的论文应该能突出表现研究工作所带来的新的东西。

**鉴定与欣赏他人工作** 一般而言，大家都认为自己的文章比投稿的平均水平要高<sup>[4]</sup>，这种现象称为“乌比冈湖效应”<sup>[5]</sup>。在担任一些会议的程序委员会委员之前，我也有过类似的想法。每次论文被拒后，我总会将自己被拒的论文与其他研究者被录用的论文进行对比，找出后者的不足，用阿 Q 精神激励自己：“这次只是运气不好，下次再投结果会更好”，导致不能客观地评价他人工作。事实上，一个较好的程序委员会通常能从大量投稿中选出相对较好的论文。当然，任何会议都可能存在一定的随机性。例如汤姆·安德森 (Tom Anderson) 将会议评审过程形容为点对点 (peer-to-peer) 系统，指出论文评审可能存在随机性<sup>[3]</sup>。这是因为不同审稿人的研究背景、对一个工作的评价品味各不相同。总体而言，论文的审稿过程都会尽可能地消除随机性。例如，通过对存在分歧的论文安排更多的委员进行评审以及讨论主持人组织的线上讨论与总结归纳等来消除随机性。

**充分完善自己的工作** 由于会议的拒稿率较高，因此投稿中存在一定数量的上次会议被拒后经过修改再投的论文。而一个会议的程序委员会中总会有一部分委员也在其他会议的程序委员会中。因此，虽然各个会议的运行相对独立，但一些论文的审稿历史也会被带到另外一个会议上。我参加过几次程序委员会的会议讨论，均听到过类似的声音：“我在 XXX 会议上评审过这篇论文，当时程序委

员会的意见是,论文提出的方法很有意思,但是实现与实验不能充分支持这个方法。现在的版本已经基本上解决了之前的问题,因此建议接收”,或者“我在XXX会议论文评审中曾指出这篇论文存在的一二三等问题,现在这个版本基本没有处理那些问题,因此建议拒稿”。在此情况下,其他程序委员会往往会听取这样的评价与建议。

因此,建议大家在论文被拒收后,根据评审意见进行充分的修改后再投稿。尽管著名的学术会议通常会尽可能给出较详细的、建设性的评审意见,但由于时间仓促,审稿人可能只会给出他认为可以足够拒绝一篇论文的评审意见。因此,在修改论文时,要全面深入地考虑方法与系统的各个方面,避免出现“现在被拒的版本如果投之前的会议应该没问题”的想法。

**投稿时机** 科学研究中一个重要的问题是应该在什么时候将所做的工作进行总结、写成文章并投稿。其中一个重要的影响因素是会议征文的截止日期,这需要进行取舍:针对当前完成的工作,投稿到这个会议是否合适,还是应该继续完善以等待下一个会议?

我的观点是:如果一个工作取得的阶段性成果可以通过实验来验证支撑相关结论,并能给其他研究者带来一些新的思路就可以写成论文并投稿。当然,在论文中需要清晰地描述系统实现中的一些不足,如“当前的系统只能运行单线程应用”等,而非模糊化甚至隐藏系统中的一些不足。因为审

稿人大多为小领域专家,试图隐藏问题的行为很容易被发现,并且容易引起审稿人的反感。另一个要注意的是不要以“碰运气”的心态将极不成熟的工作写成论文提交,或者期望审稿人给出较多建设性的意见以进一步完善工作。这样一方面会增加审稿人的负担,另一方面也可能因为工作太不成熟而仅得到初步的建议。此外,对于一个成熟的团队而言,这样做可能会影响到团队的学术声誉。克劳克罗夫特(Crowcroft)等人针对这个问题,提出使用虚拟货币的方式来惩罚一些作者提交不成熟稿件的行为<sup>[7]</sup>。尽管这种方法目前还没有被各个会议采用,但也说明了学术界对此问题的重视。

**积累评审经验** 对于研究生而言,评审经验对开展研究工会有很大帮助。我的建议是:研究生可以通过以下两种方法来积累评审经验。第一,目前计算机系统领域的部分学术会议如SOSP与EuroSys开始提供影子程序委员会委员(shadow PC),以供研究生和刚刚开始从事科研的研究人员审稿。影子程序委员会委员可以看到(部分)投稿的论文,并且也会召开程序委员会会议进行讨论,形成评审意见。通过与正式的程序委员会审稿意见的结果进行对比,可以培养研究生评价研究工作的能力。第二,如果有师长担任程序委员会委员,研究生可以在遵守相关保密协议与学术行为规范的前提下,申请与师长一起独立审稿,通过与师长的审稿意见进行对比,也可以很

快地积累审稿经验与相关能力。

虽然担任学术会议程序委员会委员需要付出较多的时间、精力与金钱,但正如张晓东教授在一次报告中指出的:“对社会与他人的贡献是检验成功的重要标准”<sup>[8]</sup>。通过担任一些会议的程序委员会委员,可以了解一个研究领域对研究工作的评判流程与理念,可以得到学术界的认同。因此,建议目前正在开展计算机系统研究工作的研究人员多担任相关程序委员会委员的工作。期待我国的研究人员通过对计算机系统领域会议审稿流程的了解,通过更加深入扎实的工作,能在这一基础性并且具有活力的领域发出更多的声音。■

**致谢:**感谢微软亚洲研究院周礼栋研究员和清华大学陈文光教授对本文提出的意见与建议。感谢上海交通大学并行与分布式系统研究所的陈榕博士和夏虞斌博士对本文进行的审阅与校正。



陈海波

CCF高级会员、2009年CCF优秀博士学位论文奖获得者。上海交通大学教授。主要研究方向为系统软件与系统结构。haibo.chen@sjtu.edu.cn

## 参考文献

- [1] Roscoe, Timothy. Writing reviews for systems conferences. <https://cminusminus.org/cs261/background/review-writing.pdf>, 2007.

更多参考文献: [www.ccf.org.cn/cccf](http://www.ccf.org.cn/cccf)