

# 基于 Linux 的超算系统中 Windows 应用程序运行环境探究\*

徐海坤, 谢一曼, 吴青, 陈军, 邹有

(中南大学信息与网络中心, 湖南长沙 410083)

**摘要:**超算系统大多是基于 Linux 操作系统搭建的, 限制了基于 Windows 操作系统的应用软件使用。此外, 超算系统操作的高门槛使不熟悉 Linux 操作系统的用户望而却步, 造成超算系统用户流失。基于 Linux 超算系统环境, 探索兼顾超算系统运维管理便利性的 Windows 应用程序使用方法。研究通过 X11 转发、Wine 和虚拟化等技术, 为用户提供兼容超算作业调度系统 Windows 应用程序运行环境, 同时提供安全、稳定的用户个人文件访问方法。所采用的配置方法与实例, 可为具有类似需求的超算中心提供解决方案, 从而拓宽用户软件应用范围, 提高用户满意度。

**关键词:**超算系统; Windows 应用; 虚拟化; X11 转发

**中图分类号:** TP393

**文献标志码:** A

**doi:** 10.3969/j.issn.1007-130X.2022.09.005

## Exploring Windows application running environment in Linux-based supercomputing systems

XU Hai-kun, XIE Yi-man, WU Qing, CHEN Jun, ZOU You

(Information and Network Center, Central South University, Changsha 410083, China)

**Abstract:** The operating systems of most supercomputers are based on Linux operating system, which limits the use of applications based on Windows operating system. In addition, the high demand of operating the supercomputing system also discourages users who are not familiar with Linux, resulting in the loss of supercomputing center users. We explore the ways of using Windows applications to maintain the convenience of supercomputing system operation and maintenance management in the Linux environment of supercomputing systems. Through X11 forwarding, Wine, virtualization and other technologies, we provide users with a Windows program running environment, which is compatible with the supercomputer job scheduling system. At the same time, we provide users with secure and stable file access methods. The configuration methods and examples used in this paper can provide solutions for the supercomputing center with similar requirements, so as to broaden the scope of user software application and improve user satisfaction.

**Key words:** supercomputing system; Windows application; virtualization; X11 forwarding

## 1 引言

高性能计算是现代科学研究最重要的技术手段之一, 国内许多高校和研究机构都建立了超算系统, 以满足学科建设和科研工作中日益增长的计算

需求<sup>[1-4]</sup>。目前, 绝大部分超算系统都是基于 Linux 操作系统搭建的, 以 CentOS、Red Hat 和 Ubuntu 为主。近年来, 国内高性能计算蓬勃发展, 集群和用户规模不断扩大, 用户需求也逐渐多样化, 部分用户产生了在超算系统中运行 Windows 应用程序的需求。这种需求在当前的超算领

\* 收稿日期: 2022-01-07; 修回日期: 2022-05-18

通信地址: 410083 湖南省长沙市中南大学信息与网络中心

Address: Information and Network Center, Central South University, Changsha 410083, Hunan, P. R. China

域中虽然不是主流,但却是每个超算系统都需要面对的问题。一方面由于 Linux 系统和 Windows 系统互不兼容,大量 Windows 应用程序没有基于 Linux 系统的版本;另一方面由于 Windows 系统具有良好的图形界面,部分用户习惯使用 Windows 操作系统。在高校环境中,缺乏 Linux 系统使用经验的用户比例大,这一需求显得尤为突出。部分建设了校级超算系统的高校对科研人员自行采购的计算硬件有严格控制,造成了科研人员 Windows 系统使用需求与校级超算系统环境之间的矛盾。此外,随着云计算的普及,各大商业云服务商提供的弹性伸缩计算服务 ECS(Elastic Compute Service)为用户使用 Windows 操作系统提供了便捷的途径,这一趋势也造成了 Windows 系统的超算用户流失。

目前,计算资源比较充裕的商业超算中心针对 Windows 应用程序运行需求,大多采用“裸金属”计算资源租用,或搭建专门的云平台提供 Windows 实例来解决。这 2 种方法的优点是:计算环境搭建较简单,用户使用便捷;缺点是:需要划分专门的计算资源用于提供 Windows 服务,资源利用不够灵活,需要额外的运维管理成本和独立的计费系统。此外,对于同时需要 Linux 系统和 Windows 系统的用户,实现 2 种系统的文件共享配置很复杂,用户对 Windows 实例的管理员权限也会带来一定的安全隐患。为解决这些问题,本文选择基于 Linux 操作系统运行 Windows 应用的思路,探寻对用户和运维人员都更便捷的解决方案。

## 2 Linux 操作系统下运行 Windows 应用程序

目前在超算 Linux 系统上运行 Windows 应用程序的主流技术方案有 2 种<sup>[5]</sup>:一种是基于兼容层的 API 转换技术;另一种是基于软件层的虚拟机 VM(Virtual Machine)技术。两者的本质区别是兼容层模拟的是 Windows 应用程序,而虚拟机技术模拟的是整个 Windows 操作系统。

### 2.1 基于兼容层的 API 转换技术

兼容层 API 转换技术是将 Windows API 调用翻译成动态的 POSIX(Portable Operating System Interface)调用,以运行 Windows 应用程序。最早使用该方法的软件是 Wine。Wine 从体系结构上体现为一个 Wine Server 进程和一组动态链接库的集合,其图形界面通过 X11 转发。实际运

行时系统至少会启动 3 个进程以运行 Windows 应用程序:应用进程本身、Wine 服务进程和 X11 服务进程。Wine 的体系结构如图 1<sup>[6]</sup>所示。

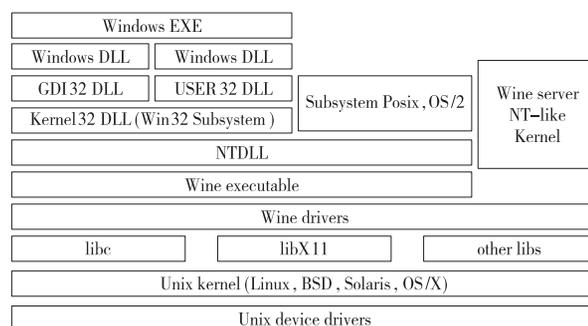


Figure 1 Architecture of Wine

图 1 Wine 体系结构

Wine 不像虚拟机或者模拟器一样模仿内部的 Windows 逻辑,而是运用 API 转换技术开发出 Linux 到 Windows 相对应的函数来调用动态链接库(DLL),以降低性能损耗和资源占用。但是,Windows DLL 库都是封闭源码,Wine 开发人员需要耗费大量的时间和精力从底层开始重新设计、开发和测试,因此目前版本的 Wine 功能还不够完善,主要表现为支持的应用不多,已有的应用运行不够稳定等。具有类似功能的软件还有 CrossOver。CrossOver 是基于 Wine 开发的商业软件,为主流的 Windows 应用软件提供商业支持,相较于 Wine 更稳定。

### 2.2 虚拟机技术

虚拟机 VM 是一种特殊的软件,可以在计算机平台和终端用户之间创建一种环境,终端用户基于该软件所创建的环境来操作其他软件。

从虚拟化的实现方式来看,常见的虚拟化架构有 2 种:裸机型和托管型<sup>[7,8]</sup>。裸机型是指虚拟机监控程序直接运行在主机的物理硬件上,不必预先加载底层操作系统,如图 2a<sup>[7,8]</sup>所示,直接访问底层硬件而无需其他软件(如操作系统和设备驱动程序),代表性软件有 VMware ESXI。托管型是指虚拟机监控程序安装在现有操作系统之上,依赖于主机预先安装的操作系统的管理对 CPU、内存、存储和网络资源的调用,如图 2b<sup>[7,8]</sup>所示,代表性软件有 Oracle VM VirtualBox、VMware Workstation 和 KVM(Kernel-based Virtual Machine)。

Linux 系统使用虚拟机完整模拟 Windows 操作系统,各种 Windows 应用软件基本上都能在虚拟上安装并稳定运行。虽然使用虚拟机技术在资源优化和性能上有较大的缺陷,但是对于需

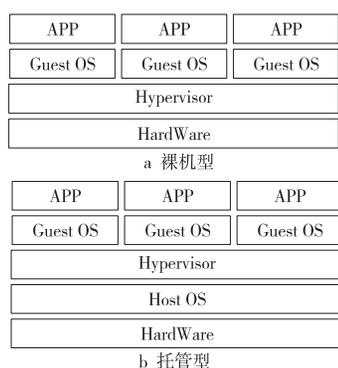


Figure 2 Types of virtualization

图 2 虚拟化实现方式

要使用不同操作系统或是有特别需求的应用软件,是一个不错的选择。它可以有效地降低超算集群的运维成本,提升管理效率。

表 1 对使用 Wine 和虚拟机 2 种解决方案的优缺点进行了对比。本文的解决方案将基于 Wine 和托管型虚拟化开展。

Table 1 Comparison of Wine and virtual machine

表 1 Wine 与虚拟机优缺点比较

解决方案	优点	缺点
Wine	开源免费,运行便捷,占用资源少,启动快;与 Linux 数据直接互通	稳定性较低,兼容性较差
虚拟机	集中化管理,硬件资源动态配置,硬件利用率高;运行环境可靠	配置较复杂;托管型虚拟机计算性能受损较大

### 3 基于超算调度系统启用 Windows 应用程序运行环境

常规的超算系统通常包括硬件系统和软件系统 2 部分。硬件系统由计算节点、高速计算网络、存储和登录管理节点组成;软件系统由底层操作系统、作业调度系统、文件系统和计算环境(编译环境、应用软件等)组成。用户使用超算系统开展计算时,首先需要使用终端登录工具访问登录节点,通过作业调度系统申请计算资源,然后使用系统分配的计算资源开展计算。登录节点和计算节点通过共享文件系统保持数据一致性。

使用超算系统开展计算相较于直接使用 Linux 主机计算有 2 个主要差别:一是需要使用远程登录终端登录;二是需要通过作业调度系统申请资源。因此,需要进行一些必要的配置,使之满足 Windows 应用程序的运行条件。

#### 3.1 X11 转发配置

用户在超算系统中使用 Windows 应用程序的

需求中,绝大部分都需要图形界面,因此需要借助 X11 转发技术在用户本地启动远程超算系统上的图形界面<sup>[9]</sup>。首先,用户需要选用支持 X11 转发的客户端工具,如 Xming、XShell 和 MobaXterm 等。其次,超算系统的登录和计算节点需要启用 X11 转发配置。最后,使用调度系统申请计算资源,并通过 X11 参数使得计算节点上的图形界面可以转发到用户终端。具体步骤如下:

第 1 步:超算服务器修改文件/etc/ssh/sshd\_config 启用 X11 转发。配置“X11Forwarding yes”。

第 2 步:使用调度系统申请计算资源,如 node01,则可以通过“ssh -X node01”访问计算资源来转发计算节点的图形界面。如果作业调度系统支持图形转发,则可直接申请计算资源。以 SLURM 作业调度系统为例,直接运行“srun -n 1 --exclusive --x11 commnad”即可完成资源申请和图形程序启动。

#### 3.2 基于 Wine 的运行配置

基于 Wine 的 Windows 应用程序运行环境配置中,需要安装与操作系统版本兼容的 Wine 软件,并启动对应的 Windows 应用程序,对于需要安装的 Windows 应用程序,也可以基于 Wine 启动安装过程,Wine 将模拟 Windows 目录结构部署应用软件。下面是在 SLURM 调度环境下,使用 Wine 运行 Windows 版本 notepad 程序的示例:

```
srun -n 1 --exclusive --x11 wine notepad.exe
```

参照该示例,可以运行其他 Windows 应用程序。

#### 3.3 基于虚拟机的运行配置

基于托管型虚拟机运行 Windows 应用程序的方案需要在计算节点安装虚拟机环境,并为用户准备 Windows 操作系统镜像和访问 Windows 计算环境的 VNC(Virtual Network Console)客户端。由于虚拟机中运行的是独立的操作系统,默认情况下,用户无法通过虚拟机访问其在超算平台的数据文件。为解决这一问题,本文在 Windows 操作系统镜像中预置了 SSH(Secure SHell)文件系统 SSHFS(Secure SHell File System)<sup>[10]</sup>,用于通过 SSH 连接来挂载用户在宿主机上的文件目录。这种方式配置简洁,且安全性高。用户通过 NAT(Network Address Translation)模式访问宿主机即可挂载数据目录,避免了用户滥用虚拟机的管理员权限配置虚拟网络带来的安全风险。本文在

SLURM 作业调度器中,采用 libvirt 开源虚拟化套件运行 Windows 虚拟机,详细步骤如下所示:

**第 1 步** 确认计算节点启用的支持虚拟化的设置,下述命令有输出即代表支持虚拟化:

```
grep -E "(vmx|svm)" /proc/cpuinfo
```

**第 2 步** 在计算节点安装 libvirt 相关虚拟化环境,以 Centos 7 操作系统为例,执行:

```
yum install -y libvirt virt-install
```

**第 3 步** 准备 Windows 操作系统镜像,从微软官方获取 Windows 10 操作系统的 VMware 镜像,然后转换为 KVM 镜像格式。Windows 10 官方 VMware 镜像下载地址为: [https://az792536.vo.msecnd.net/vms/VMBuild\\_20190311/VMware/MSEdge/MSEdge.Win10.VMware.zip](https://az792536.vo.msecnd.net/vms/VMBuild_20190311/VMware/MSEdge/MSEdge.Win10.VMware.zip)。

通过以下 qemu-img 工具完成镜像格式转换:

```
qemu-img convert -f vmdk -O qcow2 MSEdge-Win10-VMware-disk1.vmdk win20g.qcow2
```

**第 4 步** 启动镜像,安装 SSHFS 软件和 WinFSP 软件用于挂载用户位于宿主机的文件系统。以下批处理文件可供自动挂载:

```
@echo off
echo 请输入平台用户名和密码加载数据到 X 盘
echo 注意:关闭此窗口 X 盘将断开!
del C:\Users\winhpc\.ssh\known_hosts 1>nul 2>nul
set /p user=username:
start /B cmd /C " set "PATH=C:\Program Files\SSHFS-Win\bin;% PATH%" &&. " C:\Program Files\SSHFS-Win\bin\sshfs.exe"-o idmap=user-StrictHostKeyChecking=no %user%@10.0.2.2:X:"
```

运行上述批处理文件,并输入正确的用户名和密码后,用户可以在系统中通过盘符 X 访问自己在宿主机上的文件。

**第 5 步** 为用户准备 VNC 桌面查看工具,在登录节点上安装 vncviewer:yum install -y tigervnc

**第 6 步** 为用户准备作业提交模板,以 SLURM 作业调度系统为例,用户提交作业并成功分配计算资源后,即可在作业输出文件中获取访问 Windows 实例的命令和口令。SLURM 作业模板如下所示:

```
#!/bin/bash
# SBATCH --job-name=win
# SBATCH --exclusive
# SBATCH --nodes=1
# SBATCH -p winPartition
```

```
if [ ! -f "$HOME/win20g.qcow2" ];then
echo "Preparing image file.. "
cp/public/sourcecode/vm/win20g.qcow2 $HOME/win20g.qcow2
echo "Image file is ready!"
fi;
PASSWD=$RANDOM
echo
echo "Please use the following command and passwd to visit your windows instance. "
echo "vncviewer $HOSTNAME:5 $UID"
echo "Password: $PASSWD"
echo
echo
ssh $HOSTNAME virsh undefine $USER-winvm >&./dev/null
ssh $HOSTNAME virt-install--name $USER-winvm --ram 168000 --vcpus=sockets=2,cores=24,threads=1 --import --disk path = $HOME/win20g.qcow2 --graphics vnc,password= $PASSWD,listen=0.0.0.0,port=5 $UID --os-variant win10 --network default --check path_in_use=off
```

## 4 方案实践与测试

为检验所述 2 种解决方案的效果,本文在中南大学校级超算平台上进行了实际部署与测试。与实验相关的主要软硬件配置如下所示:操作系统:CentOS Linux release 7.5.1804;作业调度系统:SLURM 20.02.3;并行文件系统:Lustre 2.12.3;管理节点和登录节点硬件:2×Intel Xeon Gold 6248R CPU,192 GB 内存,240 GB SSD 硬盘;Wine 版本:6.3;Libvirt 版本:3.9.0。

### 4.1 应用启动测试

根据上述步骤描述,2 种方案中用户都可以成功启用并运行 Windows 应用程序,效果如图 3 和图 4 所示。

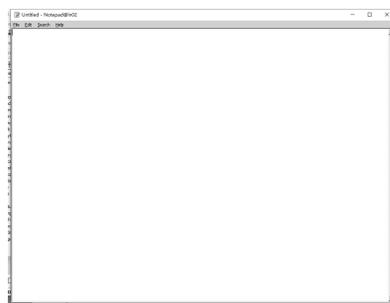


Figure 3 Starting the Windows application based on Wine  
图 3 基于 Wine 启动 Windows 应用程序

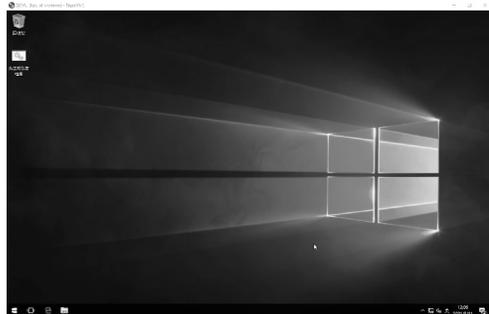


Figure 4 Starting the Windows VM based on the virtualization solution

图4 基于虚拟化方案启动 Windows 虚拟机

## 4.2 应用运行性能测试

为检验上述 2 种运行 Windows 应用程序方案的计算性能,本文选择了基于 OpenMP 的多线程矩阵乘法算例进行测试。采用相同版本的编译器和参数将代码分别编译为 Linux 版本和 Windows 版本,编译软件均采用 GCC 8.1.0,测试矩阵规模  $3600 \times 3600$ ,填充随机值。其中,Windows 版本分别在 Wine 环境和 KVM 虚拟机环境下运行。测试结果如表 2 所示。

Table 2 Efficiency of OpenMP multi-threaded matrix multiplication running in different environments

表 2 不同环境下 OpenMP 多线程矩阵乘运行效率

运行环境	不同线程数运行时间/s					
	1	2	4	8	16	32
Linux	58.25	34.46	19.95	9.79	4.51	2.12
Wine	57.85	33.87	19.70	8.71	4.33	2.16
KVM	214.84	119.11	60.13	30.75	15.63	8.03

从表 2 可以看出,应用程序基于 Wine 运行效率较高,与纯 Linux 环境下几乎没有差别;而基于 KVM 虚拟机的运行性能损失较大,效率约为纯 Linux 环境和 Wine 的 1/4。因此,对于运行环境复杂度不高的 Windows 应用,可以优先采用基于 Wine 的方案;对于复杂的 Windows 应用,在综合考量运行稳定性和效率的情况下,可以采用基于虚拟机的方案。

## 5 结束语

本文就 Linux 超算系统中构建 Windows 应用程序运行环境的方案进行了探究,针对主流超算系统中无法便捷管理和运行 Windows 应用程序这一问题,探索了基于 Wine 和虚拟机技术 2 种解决方案,并对比了各自方案的优缺点。这 2 种解决方案

相较于独立部署 Windows 计算节点或提供 ECS 云服务,均具有部署简单、管理方便和应用安全便捷等特点,同时兼容现有作业调度系统和计费系统。2 种解决方案也在中南大学校级超算平台上得到了应用与实践,可供其他超算中心参考。在未来的工作中,如何提高基于 Wine 方案的稳定性以及降低基于虚拟机方案的性能损失,均有待进一步研究。

## 参考文献:

- [1] Yuan Guo-xing, Zhang Yun-quan, Yuan Liang. State of the art analysis of China HPC 2019[J]. Computer Engineering & Science, 2019, 41(12): 2095-2100. (in Chinese)
- [2] Zhang Yun-quan, Yuan Liang, Yuan Guo-xing, et al. State-of-the-art analysis and perspectives of China HPC development in 2019 [J]. Frontiers of Data & Computing, 2020, 2(1): 18-26. (in Chinese)
- [3] Yao Ge. Requirements and development of high performance computing application[J]. Electronic Technology & Software Engineering, 2019(20): 136-137. (in Chinese)
- [4] Sun Ning-hui, Tan Guang-ming. Development and policy of high performance computer [J]. Bulletin of the Chinese Academy of Sciences, 2019, 34(6): 609-616. (in Chinese)
- [5] Wang Ya-jun, Liu Jin-gang. Techniques of Windows programs running in Linux system[J]. Journal of Computer Applications, 2008, 29(8): 2128-2131. (in Chinese)
- [6] Hu Yu-jie, Li Shan-ping. Methods for running Windows applications under Linux [J]. Computer Engineering, 2003, 29(11): 169-191. (in Chinese)
- [7] Chen Jun-xiang Li Gui-jie. On application of virtualization and Docker in data management of university libraries[J]. Library Science Research & Work, 2016(5): 52-56. (in Chinese)
- [8] Obasuyi G C, Sari A. Security challenges of virtualization hypervisors in virtualized hardware environment [J]. International Journal of Communications, Network and System Sciences, 2015, 8(7): 260-273.
- [9] Scheifler R W, Gettys J. X window system—The complete reference to Xlib, X protocol, ICCCM, XLFDF[M]. Newton: Digital Press, 1992.
- [10] Szeredi M. SSHFS; SSH filesystem [EB/OL]. [2021-10-11]. <http://fuse.sourceforge.net/sshfs.html>.

## 附中文参考文献:

- [1] 袁国兴,张云泉,袁良. 2019 年中国高性能计算机发展现状分析[J]. 计算机工程与科学, 2019, 41(12): 2095-2100.
- [2] 张云泉,袁良,袁国兴,等. 2019 年中国高性能计算机发展现状分析与展望[J]. 数据与计算发展前沿, 2020, 2(1): 18-26.
- [3] 姚舸. 高性能计算应用的需求与发展[J]. 电子技术与软件工程, 2019(20): 136-137.
- [4] 孙凝晖,谭光明. 高性能计算机发展与政策[J]. 中国科学院院刊, 2019, 34(6): 609-616.

- [5] 王亚军,刘金刚. Windows 程序运行于 Linux 系统的技术[J]. 计算机应用,2008,29(8):2128-2131.
- [6] 胡玉杰,李善平. Windows 程序在 Linux 上的运行[J]. 计算机工程,2003,29(11):169-191.
- [7] 陈军相,李桂杰. 虚拟化及 Docker 轻量容器技术在高校图书馆中的应用[J]. 图书馆研究与工作,2016(5):52-56.

### 作者简介:



**徐海坤**(1992-),男,江西鄱阳人,硕士,实验师,研究方向为高性能计算和大规模并行计算与应用。**E-mail:** xuhaikun@csu.edu.cn

**XU Hai-kun**, born in 1992, MS, experimentalist, his research interests include high performance computing and large-scale parallel computing & applications.



**谢一曼**(1996-),女,湖南娄底人,硕士,助理实验师,研究方向为网络安全和异常检测。**E-mail:** yimanxie@csu.edu.cn

**XIE Yi-man**, born in 1996, MS, assistant experimentalist, her research interests include network security and anomaly detection.



**吴青**(1983-),女,湖南长沙人,硕士,讲师,研究方向为高性能计算与可视化。**E-mail:** wuqing@csu.edu.cn

**WU Qing**, born in 1983, MS, lecturer, her research interests include high performance computing and visualization.



**陈军**(1964-),男,湖南长沙人,博士,高级工程师,研究方向为计算机网络、高性能计算和网络安全与应用。**E-mail:** chenjun@csu.edu.cn

**CHEN Jun**, born in 1964, PhD, senior engineer, his research interests include computer network, high performance computing and cyber security & applications.



**邹有**(1986-),男,湖南湘潭人,硕士,实验师,研究方向为高性能计算与生物信息。**E-mail:** zouyou@csu.edu.cn

**ZOU You**, born in 1986, MS, experimentalist, his research interests include high performance computing and bioinformatics.

## 《计算机工程与科学》征文通知

《计算机工程与科学》是由国防科技大学计算机学院主办的中国计算机学会会刊,是国内外公开发行的计算机类综合性学术刊物,现为月刊。

本刊欢迎关于计算机科学理论、计算机组织与系统结构、计算机软件、计算机网络、计算机器件设备与工艺等学科领域方面的来稿。

本刊常年设有高性能计算专栏。

来稿论文必须未发表、未投到其他会议或期刊。

来稿要求和注意事项:

- (1) 主题明确、文字精练、语句通顺、数据可靠。
- (2) 标题、作者单位、摘要、关键词采用中英文间隔行文;请注明是否基金资助项目论文(注明项目名称和编号),并注明文章中图法分类号。务必附上所有作者中英文简历(姓名、性别、出生年月、籍贯、学位、职称、研究方向)、1寸证件照片(军人请用便服照)、中英文通信地址、联系电话和 Email。
- (3) 作者在投稿时须注明是否是 CCF 会员(高级会员、普通会员、学生会员),若是会员,请注明会员号。
- (4) 来稿请用 WORD 软件编辑,格式为 A4,40 行×40 列,通栏排版,正文为 5 号宋体,论文长度不得低于 5 个标准版面,并请自留底稿。
- (5) 来稿中图形绘制要求工整、清晰、紧凑,尺寸要适当,图中文字用 6 号宋体,线为 0.5 磅。
- (6) 每篇论文格式要求:1 引言;……;最后是结束语。引言和结束语中尽量不用图和表。附录应放参考文献之后。参考文献限已公开发表的。
- (7) 来稿文责自负,要遵守职业道德,如摘引他人作品,务请在参考文献中予以著录。署名的作者应为参与创作,对内容负责的人。文章发表后,如不同意其他报、刊、数据库等转载、摘编其作品,请在来稿时声明。

(8) 本刊不收取作者任何费用(免审稿费、版面费等所有费用)。

联系地址:410073 湖南省长沙市国防科技大学《计算机工程与科学》编辑部

联系电话:0731-87002567

电子邮箱:jsjgcykx@vip.163.com

投稿主页: <http://joces.nudt.edu.cn>