

deepin 集结

【深度·春秋】

06 深度集结 为爱而走

【深度人·在说】

22 做好 deepin 与客户的桥梁

【深度·生活】

82 深度户外拓展

【行业·观察】

28 Ubuntu 放弃战斗 Linux 桌面的悲哀

38 勒索病毒提了个醒
自主掌握操作系统有多重要

【深度·社区】

48 这系统 吓到我了

【深度·伙伴】

72 使用深度 deepin 国产操作系统的一点心得与体会!

74 要想完全放弃 Windows 使用 Linux 需要多少勇气?

76 和深度一起度过的时间

80 与众不同的 deepin

06期
2017年06月

【深度·讲坛】

56 Linux 全局事件监听技术

59 deepin-wm 的动态背景模糊实现

42 【特别策划】

深度原生 最美和声



深度操作系统公众号
www.deepin.com
内部资料 免费发放

从建国初期的两弹一星，到后来的京张铁路，再到现在的华为基站，不管是为了国家的安全，还是为了人民的衣食住行，打破技术封锁，阻止因信息的不对等带来的剥削，一直有人在努力着。

而深度系统、深度人也是这其中的一员，长久以来，人们都希望中国有一款国产的、自主可控的操作系统，摆脱微软的垄断，守住信息安全的底线，期间有很多操作系统昙花一现，但深度不同，它不仅靠自己的力量活了下来，还一直不断努力并壮大着。

当一个“码农”从敲代码中获得了无穷的乐趣后，后果便会变得很严重，深度系统就是由这样的一类人聚集在一起慢慢孕育出来的，一开始可能只是想做一个美观易用的Linux桌面环境，后来又觉得光桌面不行啊，只有光杆司令，软件中心也要做一个好用的才行，没好用的截图工具自己做一个好了……从此在造轮子的路上一发不可收拾。

深度系统现在已经有10几款原生应用了，可以说每款原生应用都是应运而生。为了深度操作系统使用起来更顺手，在顺手的基础上还要获得极致的体验，深度操作系统经过不断的打磨和完善，从一开始集成的开源应用中总结经验，发现用户的需求，然后将一些体验不完美的应用替换成原生应用。

如果将深度操作系统比作恒星，那么深度应用就是围绕恒星运转的行星，它们组成了一个完整的星系，按照不变的规律运行，稳定而和谐，又孕育着无穷的生机。

>>>
产品部

deepin 集结

策划 Hosted by
武汉深之度科技有限公司 Wuhan Deepin Technology Co., Ltd.
编辑 Edited by
《deepin集结》杂志编辑部 Editorial Office of DEEPIN JIJIE

总编辑 Editor-in-chief
刘闻欢 Liu Wenhuan
副总编 Deputy Editor
许可 Xu Ke
执行编辑 Executive Editor
郝俊 Hao Jun
编辑 Editor
许峰 Xu Feng Zhang Wenhao
采编 Assistant Editor
李会会 Li Huihui Jiang Wen
美术设计 Art Editor
秦娣 Qin Di

网站 Website
<http://www.deepin.com>
邮箱投稿 Contribution
deepin-magazine@deepin.com
市场推广 Marketing
account-marketing@deepin.com

武汉联络处 Wuhan Office
地址 Address
武汉市光谷大道77号
光谷金融港B18栋6楼
邮编 430223
电话 +86-27-87805607

北京联络处 Beijing Office
地址 Address
北京市西城区新街口外大街28号普天德胜B座603室
邮编 100088
电话 +86-10-62669499

上海联络处 Shanghai Office
地址 Address
上海市长宁区愚园路1258号15A01室
邮编 200050
电话 +86-21-60726030

准印证号 (鄂) 4300107
承印单位 武汉金港彩印有限公司
出版日期 2017年06月
发放对象 公司员工、用户及合作伙伴、Linux爱好者

本刊为非盈利性杂志，用于传播企业文化，交流学习技术经验
版权所有，未经同意不得转载。

02 深度·春秋

- 02 深度录音 V1.0 正式发布——录制你想要的声音
- 04 深度家族又添加了一个新成员——深度录屏正式发布啦!
- 06 深度集结 为爱而走
- 08 深度科技受邀参加“我们正在前进·龙芯 2017 产品发布暨合作伙伴大会”
- 10 深度科技参加第 72 届中国教育装备展示会
- 12 深度科技受邀参加福建省版权局召开省直机关软件正版化工作培训会议
- 14 湖北省召开全省软件正版化工作培训会议
- 17 自主可控 安全可期
- 17 陈一新邀专家等座谈 共谋武汉发展新民营经济之策
- 18 深度服务器操作系统特性与规划

22 深度人·在说

- 22 做好 deepin 与客户的桥梁

28 行业·观察

- 28 Ubuntu 放弃战斗 Linux 桌面的悲哀
- 38 勒索病毒提了个醒 自主掌握操作系统有多重要
- 40 武汉光谷国产 Deepin 操作系统完美防御勒索病毒

48 深度·社区

- 48 这系统 吓到我了
- 51 对于 Deepin15.4, 我想说

54 深度·案例

- 54 重庆市渝中区文化委桌面操作系统国产化试点

58 深度·讲坛

- 56 Linux 全局事件监听技术
- 59 deepin-wm 的动态背景模糊实现
- 61 RPM 包制作与 YUM 仓库基础
- 65 GRUB 与系统引导

72 深度·伙伴

- 72 使用深度 deepin 国产操作系统的一点心得与体会!
- 74 要想完全放弃 Windows 使用 Linux 需要多少勇气?
- 76 和深度一起度过的时间
- 80 与众不同的 deepin

42 深度·策划



深度原生 最美和声

无论是气势恢宏的音乐，还是悠扬婉转的音乐，亦或是空灵震神的音乐，都是热情洋溢的自由艺术，像风、像四季、像海洋，像自然那样无边无际，又像自然生态那样生生不息。

一系列深度原生应用就像一段段和声，与深度操作系统一起，组成一首来自开源生态、来自极客文化，和源自用户心声的音乐。

82 深度·生活

- 82 深度户外拓展

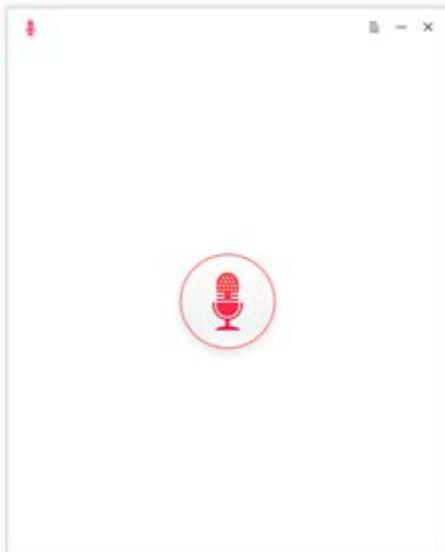


深度录音 V1.0 正式发布 ——录制你想要的声音

● 深度科技 产品部 / 文

深度录音是一款设计简洁、美观易用的录音应用，无论您是想记录自己的歌声还是通话或会议录音，都可以使用深度录音来完成，它还支持可视化录音、录音回放、录音列表管理等功能。

通过更新系统以获取深度录音，或者直接在深度商店搜索下载。



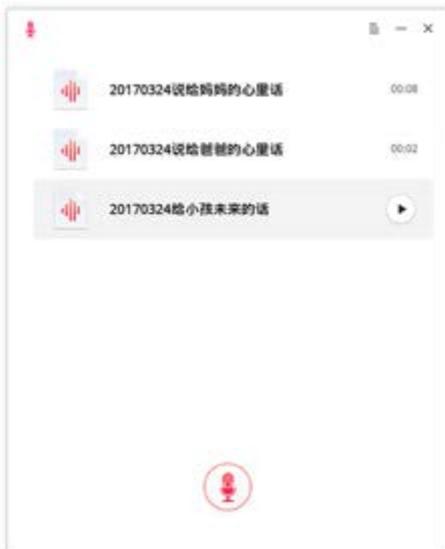
界面设计简洁

统一系统小工具设计界面布局，让焦点聚集在中心显示录音 Logo，只需轻轻点击便可开始。



操作即点即录

点击界面录音按钮就可以开始，说出想记录的话，唱出心中的歌，上下跳动的节奏记录每一次波动。



列表模式显示

录音记录通过列表模式显示，一目了然。录音可以快速回放，可以编辑重命名及删除。



文件快速定位

对于录制好的录音和歌声，可以快速定位到录音文件，第一时间分享给家人和好朋友。d



深度家族又添加了一个新成员 ——深度录屏正式发布啦！

● 深度科技 产品部 / 文

深度录屏是深度科技团队打造的一款屏幕录制小工具，主要将录屏为 GIF 和 MP4 格式，可以自动捕捉窗口和手动定义录制区域，只需点击开始和结束即可，自动保存文件到桌面。

大家可以通过更新系统以获取深度录屏，或者直接在深度商店搜索下载。

随心所欲 自动捕捉

类似深度截图捕捉窗口的方式，自动捕捉和识别窗口，也可以自定义窗口大小。更方便的是还可以通过快捷键 (Ctrl + Shift + R) 快速启动录制。



一键切换 两种格式

深度录屏支持录制 GIF 和 MP4 格式，格式快速自由切换，总有一种方式适合你的表达，同时优化了其性能和录制质量。



录屏倒数 温馨提醒

点击开始录屏后，可以直接操作屏幕，在录制区域中央显示 3 秒钟倒计时提示，给用户录制之前充足的准备时间。



闪烁监控 触发即停

在托盘区域录屏图标以 800ms 的时间间隔闪烁提示录制中，点击托盘图标或者按下深度录屏的快捷键 (Ctrl + Shift + R) 后停止录制。d





深度集结 为爱而走

● 深度科技 市场部 / 文

2017年4月17日是第29个世界血友病日，4月15日第六届“为爱而走”慈善健走大会在北京奥林匹克森林公园举行，近5000名志愿者以行走的力量呼吁全社会关注血友病和其他弱势儿童。

2017年“为爱而走”慈善健走活动，是由中国民主

建国会北京市委员会、政协北京市朝阳区委员会、中华少年儿童慈善救助基金会、中国妇女发展基金会、有爱有未来外企志愿行动、北京血友之家罕见病关爱中心、FESCO北京外企人力资源服务有限公司、BTV《生活面对面》、新浪微公益等单位倡议发起，并得到了80余家知名中外企业和爱心机构的支持。



深度科技组织北京公司部分员工和深度的粉丝共同参与了这次公益慈善活动。呼吁大家关注公益，关爱罕见病，作为一家创业公司，公司上下一心，奉献爱心，也借此机会倡导平日里坐在办公室的工程师们注重运动，关注绿色健康生活。

“为爱而走”慈善健走大会是北京地区开展较早、具有广泛参与度及较高社会关注度和影响力的公益体育活动，由中国民主建国会北京市委员会、政协北京市朝阳区委员会等单位主办，2012年以来已举办六届，超过300家跨国企业和爱心机构，35000余名志愿者现场参与，行走历程近15公里，募集善款超过80万元，直接帮助近93所学校的流动儿童

和包括血友病在内的众多困难家庭。

今年的“为爱而走”活动是第三次以血友病为主题。血友之家理事长关涛表示，由于社会对血友病存在误解以及血友病患者家庭因病致困等原因，很多患者面临教育和工作的诸多困难，“希望通过这个活动，让更多人看到和了解血友病，给予血友病患者以更多关注和支持。”

此次参与活动的每位小伙伴都获得了大会组委会准备的纪念首日封和捐赠证书，在这样一个晴空万里的有意义的日子里，大家度过了不平凡的一天，希望来年，我们继续在这里集结一起，为爱而走。d



深度科技受邀参加

“我们正在前进·龙芯 2017 产品发布暨合作伙伴大会”

● 深度科技 市场部 / 文

2017 年 4 月 25 日，“我们正在前进·龙芯 2017 产品发布暨合作伙伴大会”在北京朗丽姿西山花园酒店隆重举行。工业与信息化部彭红兵副司长，国家集成电路产业投资基金有限公司丁文武总经理，中国工程院倪光南院士、李国杰院士等领导专家出席并致辞。

深度科技作为龙芯的合作伙伴受邀参展，总经理刘闻欢出席并参与了高峰对话的讨论。此外，出席本次活动的还有 300 余家龙芯合作伙伴代表以及用户代表、知名媒体等，各行业精英 1000 余人齐聚盛会，

分享国产基础软硬件自主创新成果。

在本次发布会的高峰对话环节中，龙芯还特别邀请了国产操作系统的四家重量级代表厂商“中标软件”、“湖南麒麟”、“普华软件”、“深度科技”的负责人，就“国产操作系统与 CPU 如何合力打造自主产业生态”这个话题进行了对话探讨，各国产操作系统厂商负责人就各自操作系统产品特点与发展路线、如何打造国产软硬件统一生态平台，以及对如何看待国内产业政策等方面发表了各自看法和建议。



刘闻欢总经理表示：深度科技作为一家操作系统厂商，是一家从社区发展而来的商业公司，从2004年由爱好者参与并研发产品开始，到2008年发布社区版本，2011年才开始逐步迈入商业化，并在2014—2015年进入到国产自主可控领域，且参与到了党政军事业中来。

深度科技与龙芯有些概念是一致的，想要做好国产的软硬件，必须要重视生态建设，所以深度科技在生态上做了很多工作。首先，深度科技联合了国内知名企业并发布了在国产操作系统上的一些基础的软件，发布了深度商店。龙芯申威国产CPU厂商

合作，逐步把好的产品方案引用到国产的CPU中来。深度科技坚信，随着国产基础软件与像龙芯这样的核心硬件厂商的合作，中国的自主可控事业会更好地发展。

随着国家对网络与信息安全的高度重视，自主软硬件目前进入快速发展的新阶段。市场和用户才是检验产品和企业真正的“试金石”，建立和完善国产操作系统的生态更是一场持久战。未来，深度科技将携手合作伙伴建立自己的生态体系，从技术、产品、服务等方面不断加强自己的核心能力。d



深度科技参加 第 72 届中国教育装备展示会

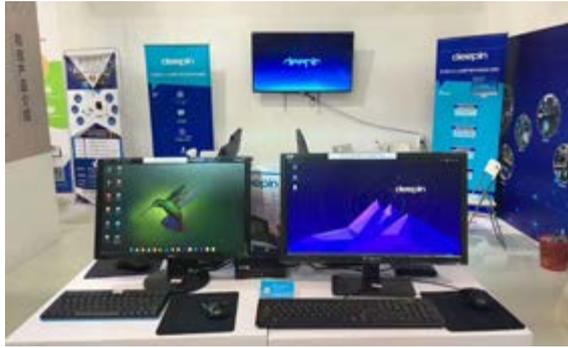
● 深度科技 市场部 / 文

5月7日深度科技参加了由中国教育装备行业协会主办，福建省教育厅承办，各省、自治区、直辖市和计划单列市教育装备行业协会协办的第72届中国教育装备展示会。据了解，本届展会共有1200余家单位参展，吸引了近10万人次参观。

深度科技在此次展示会上，展示了2017年最新版本的深度操作系统桌面版V15.4，该操作系统能够满足用户的日常办公和娱乐，支持x86及龙芯、申威等国产CPU平台，可以替代微软的Windows桌

面操作系统，可提供定制化解决方案，满足特性需求，提供国产化解决方案，满足自主可控的要求。

此前深度操作系统已完成福建省教育厅班班班通教育云平台项目，提供了一整套教育云系统国产化方案，将电子白板、PC终端、高拍仪等设备整合为一体，组成了一个集远程教学、课件同步、电子备课等功能为一体的多媒体教学系统，让老师通过文字、图片、声音、视频等多种方式，和学生之间进行交互。



深度科技作为一家专注基于 Linux 的国产操作系统研发与服务的高科技企业，在众多 Linux 操作系统产品中取得了众多突出的成绩，并在全球获得了广

泛的认可。公司拥有系统研发、软件包维护、内核开发、国际化、交互设计等专业人才，能够满足不同用户对操作系统定制化和研发的需求。d



深度科技受邀参加福建省版权局召开 省直机关软件正版化工作培训会议

为贯彻落实汪洋副总理4月11日主持召开的全国打击侵犯知识产权和制售假冒伪劣商品工作电视电话会议精神和推进使用正版软件工作部际联席会议《2017年推进使用正版软件工作计划》，2017年5月26日，福建省版权局在福州召开2017年福建省省直机关软件正版化工作培训会议，进一步完善推进使用正版软件工作长效机制，巩固和拓展软件正版化成果。

本次会议由福建省版权局版权管理处副处长谢振芳主持，福建省版权局版权管理处处长邱建宏、机关事务管理局国资处处长肖光华、福建省数字福建建设领导小组办公室技术应用处副处长项宗庭、福建

省版权局版权管理处副处长谢振芳、福建省版权局版权管理处副主任科员常桂旭等先后讲话。

福建省版权局版权管理处处长邱建宏在讲话中强调要将正版软件列入预算管理，根据实际需求列入预算，需保证正版化采购资金，加强个人的软件正版化考核，必要时纳入考核指标中，切实可行的加强正版化的重视。

机关事务管理局国资处处长肖光华在讲话中指出，要将软件正版化纳入绩效考评的范畴之中。软件资产配置应符合以下要求：购置软件应符合正版化要求；自行开发软件应拥有完全自主知识产权，开发

过程中应用第三方软件产品应取得合法授权；采购计算机应预装正版操作系统软件；对软件兼容性、授权方式、信息安全、升级等售后服务应有明确约定；向有关部门申请立项开发的软件资产，应严格按照立项开发的要求实施。

福建省数字福建建设领导小组办公室技术应用处副处长项宗庭在讲话中强调，每台电脑需要保证一套正版系统和一套正版办公软件。

福建省版权局版权管理处副处长谢振芳在讲话中强调，正版化软件从 1999 年开始，国务院办公厅明确规定，必须安装经授权的软件。国家在逐步在加强正版化工作的强度，也加强了正版化工作的打击力度，目前正版化已经对国家机关全部覆盖。国家相关要求，正版化工作纳入考评范畴，要求每个机关单位在年底 11 月份报备考评表。购置办公电脑时要求一并购置正版操作系统、办公软件、杀毒软件。



深度科技高级技术经理出席会议并表示：深度科技一致致力于软件正版化工作。不久前，国家版权局和湖北省相关领导亲临公司视察并一致认可深度操作系统产品。目前，深度科技已在湖北省正版化工作中取得优异成果，与此同时，还重点关注上海和福建的软件正版化工作，为政府机关和事业单位提供优秀的产品和优良的服务。

深度桌面操作系统具有简洁易用、美观实用的特性，拥有国内唯一自主研发的桌面环境，集成软件商店，提供更多 Linux、Windows 和安卓的常用应用，更好支持打印机和网银 UKey 应用，可完全满足用户日常办公、学习和生活的应用需求。

截止到 2016 年，深度操作系统下载超过 5000 万次，提供 30 种不同的语言版本，以及遍布六大洲的 70 多个镜像站点的升级服务。在全球开源操作系统排行榜上，深度操作系统已跃居前十，也是排名最高的中国操作系统产品。d





湖北省召开全省软件正版化工作培训会议

5月25日至26日，湖北省使用正版软件工作领导小组办公室和湖北省版权局采取以训代会的形式在武汉举办2017年全省政府机关软件正版化工作培训班。培训班上通报了2016年全省软件正版化工作督查情况，布置了2017年软件正版化工作重点任务，宣传推广了软件正版化工作信息管理平台操作使用办法，正版软件采购相关知识，典型交流，并就软件正版化工作涉及到的信息安全防护（病毒、勒索软件的简单防护）、正版软件识别、国产软件发展等具体问题进行了探索研究，提供了解决方案。

会议强调：2017年是政治年，也是软件正版化工作关键年，国务院部际联席会议将对省级机关进行全覆盖督查，软件正版化信息管理系统将正式运行，国产软件推广运用试点也计划在我省启动。任务艰

巨，责任重大，全省上下各级党政部门、各企事业单位都要把思想统一到国务院决策部署上来，要进一步提高认识，明确任务，抓好落实。要从源头抓起，严把计算机软硬件采购关，要从领导做起，严格落实第一责任人的主体责任，要从制度抓起，严格落实督促检查、目标考核和责任追究制度，建立长效管理运行机制，推进软件正版化工作常态化、规范化、标准化。

会议由湖北省版权局版权处邹喜玲调研员主持，会议强调：“2017年的工作重点是全面推进软件正版化工作规范化、常态化、制度化和信息化，进一步完善长效机制，加大督查力度，加强信息融合，强化技术手段，提高管理水平，推进软件正版化工作再上新台阶。”并着重从以下几个方面做了要求：

1. **推进软件正版化工作规范化。**重点是要完善软件正版化长效机制。推进各地各部门加强软件正版化制度建设，完善职责分工、软件采购、软件管理、督促检查、考核监督等软件正版化工作关键环节相关制度，从制度层面保障软件正版化工作常态化；完善软件资产管理制度，细化软件资产配置计划管理，严格审核各部门通用办公软件资产配置计划，与采购管理相结合，定期实采购数据；规范计算机软硬件采购。推进各地各部门新采购计算机时，落实预装正版操作系统软件的规定，严格审核需要购置的办公软件和杀毒软件购置计划，从源头上防止盗版软件流入政府机关；推广正版软件管理工作指南。指导各地各部门以落实部际联席会议印发的《正版软件管理工作指南》为抓手，规范正版软件使用管理，建立健全正版软件管理台账。

2. **开展软件正版化工作督促检查。**一方面要开展政府机关软件正版化工作督促检查。省领导小组各成员单位要配合国务院督查组对我省省级政府机关软件正版化进行全覆盖检查工作。省级联合督查组全面检查市州级政府机关软件正版化工作。另一方面要发挥第三方专业机构人员参与督查优势，扩大市州级政府机关软件正版化工作督查覆盖面。指导各市州督查本市州及所辖区县政府机关和企业软件正版化工作。同时要深化软件资产管理绩效考评和监



督检查。改进软件资产管理绩效考评指标，对省级国家机关软件资产管理情况进行抽查。

3. **开展国产软件应用试点。**配合国务院在我省部分地区和单位开展国产软件应用试点工作，要以信息安全的高度来理解正版化工作，各政府机关应选购安全、自主可控的国产软件产品。

金山办公软件副总裁刘昌伟做了题为《WPS Office 国产软件创新发展之路》的演讲，介绍了金山办公软件 WPS 的发展情况及当前形势下正版化工作的一些思考和建议。现在 WPS 全平台月报活达到 2 亿，每天使用 WPS 进行创作、编辑、阅读的文档数量超过 3 亿个。WPS 紧随时代发展和技术趋势，不再是以前单一的单机版产品，目前构建了以桌面和移动 WPS 为核心，涉及到秀堂、写得等一系列的围绕办公创作的产品家族，以云文档为协同共享为中心的文档协作中心，以即时消息作为沟通交流和文档更新等通知信息为基础的消息中心，以对文档进行全生命周期管理为核心的文档安全中心。同时 WPS 积极响应国家号召，全力推进自主可控办公软件开发，支持国产全部主流的 CPU、操作系统。怎样彻底解决目前正版化存在的一系列问题，避免陷入以下陷阱，例如：

个人版与专业版（或企业版），需要认识纠正；不可混肴版本授权，个人版软件厂商有严格范围使用限制，纳入不了正版范畴，尽快购买专业版。

捆绑销售，涉嫌违反不正当竞争和反垄断法；操作系统 OEM，不应单独购买；国家工商总局对某国外公司进行搜查，涉嫌违反不正当竞争和反垄断法。



受诱惑购买老版本，并没有节省费用；低价推荐 2010、2013 版本，低版本始终面临产品兼容问题，随时面临再掏一笔费用进行升级到新版本问题

国内生产就是国产，偷换概念是绝对错误的想法，信息安全是国家大计。习主席 2016 年 4.19 讲话再次强调推动信息安全建设，逐步实现自主可控。

刘昌伟副总建议：借鉴其他省市经验，抓紧建立正版化长效机制，通过随“机”授权、场地授权解决全覆盖问题，通过软件资产与计算机硬件同步管理破解软件资产管理难题。从软硬件采购源头管理一劳永逸的解决正版化问题，根据今年国家版权局工作指南，希望金山 WPS 可以全力协助推动企事业单位 2017 年目标任务、开展国产软件应用试点、支持加大督促检查力度等方面工作。

会议期间，金山办公软件公司联合武汉深之度科技有限公司搭建了国产化平台展区，参会人员积极参观并咨询 WPS 在国产平台上的使用情况，并表示在有条件的情况下愿意优先使用国产软硬件产品。

会后，金山 WPS 副总裁刘昌伟、深度科技有关负责

人与湖北经信委相关领导就 WPS 业务和国产化试点如何在湖北省加大拓展力度进行了深入探讨。

另据报道，5 月 3 日下午，省委书记蒋超良，省委副书记、省长王晓东与来湖北考察的小米通讯技术有限公司和金山软件集团董事长雷军一行座谈。蒋超良代表省委、省政府对雷军一行表示欢迎，他说，雷军董事长作为从湖北走出去的优秀企业家，对家乡的发展一直高度关注、大力支持，作出了积极贡献。此次与我们签订合作协议，充分体现了卓越的战略眼光和对家乡发展的大力支持，对此表示衷心感谢。蒋超良说，网络的本质在于互联，信息的价值在于互通。湖北省委、省政府深入学习贯彻习近平总书记网络安全和信息化工作座谈会和全国科技创新大会上的重要讲话精神，结合省情实际，加强信息基础设施建设，强化信息资源深度整合，打通经济社会发展的信息“大动脉”，用“互联网+”带动新老业态竞相发展，引领湖北经济转型升级。当前，湖北网信事业发展基础不断夯实、发展环境日益优化，我们将充分利用科教人才优势和良好基础条件，在互联网核心技术研发上加大投入。希望雷军董事长将湖北、武汉作为投资和创新发展的新基地，共同创造、分享湖北经济转型发展的新成果。雷军介绍了小米公司、金山软件和顺为资本的经营状况和发展战略，表示自己对湖北、对武汉的发展始终非常关注，特别希望能为家乡发展贡献力量。武汉人才密度高，拥有很好的技术研发力量和产业基础，目前正处于爆发式增长的时间节点。希望通过共同努力，推动战略合作项目尽早落地，同时联合一批互联网龙头企业，全力发展物联网、大数据、人工智能等产业，将武汉建成现代信息产业发展的新高地。d

自主可控 安全可期

写在前面：面对这波被称为“史上最大规模”的电脑勒索病毒，为了保护重要信息数据的安全，各大企业及单位纷纷做好了防范新一轮攻击的准备。

深度操作系统作为一款优秀的国产操作系统软件产品，拥有自主可控、美观易用等特点。北京信息安全测评中心作为北京市经信委直属单位，已部署深度操作系统实现了全部办公终端的国产化替换，有效避免了此次电脑勒索病毒的风险。

2017年5月12日开始，号称“史上最大规模”勒索蠕虫肆虐全球，至少有150个国家和地区的20万个受害者中毒，“周一上班电脑怎么开机”也成为刷爆微信朋友圈的最热门话题。

针对本次事件，全国各信息安全主管部门、各大安全企业周末紧急发布各种预警通知、应急处置方案，在一定程度上遏制了这次蠕虫的进一步传播。但与其每次发生安全事件都采取“亡羊补牢”的处理方式，不如让安全更加自主可控起来。

北京信息安全测评中心作为北京市信息安全基础设施，深入领会并积极践行国家有关精神，主动作为、先行先试，率先在今年年初启动并于5月1日前完成了全部办公终端的操作系统国产化工作，用自主可控的深度Linux操作系统替代微软Windows操作系统，避免了“上班第一件事就是拔网线”这类被动的防护手段。

本次安全事件仅仅网络安全问题的冰山一角，信息化的安全保障工作任重而道远。历史经验告诉我们，只有技术上的自主可控，才有可能做到安全风险的可控，才能最终不受制于人。d

陈一新邀专家等座谈 共谋武汉发展新民营经济之策

全市民营经济发展大会将于近期召开。昨日上午，市委邀专家学者、企业家及党外人士座谈，就武汉发展新民营经济听取意见建议。市领导蔡杰、曹裕江与会。省委副书记、市委书记陈一新出席会议并提出，要坚持以习近平同志系列重要讲话精神和治国理政新理念新思想新战略为指导，聚焦富民，聚力创新，深入研究武汉发展新民营经济的思路、任务和举措。

“什么是新民营经济？”“为什么要发展新民营经济？”“武汉发展新民营经济有哪些优势条件？”“要鼓励发展什么样的新民营经济？”“怎么样发展新民营经济？”座谈会上，陈一新抛出问题，请大家畅所欲言，结合实际共同思考。

中南财经政法大学经济学院院长卢现祥说，武汉结合自身优势提炼出新民营经济很有意义，这一导向对武汉发展非常有利。全国政协委员、省民营企业家发展协会会长万安培建议，武汉新民营经济要拿出务实管用的政策，努力培育领军人物和重点企业。

武汉深之度科技有限公司董事长刘闻欢认为，武汉有发展新民营经济的良好条件，可以通过发展新民营经济吸引更多人才来汉创新创业。

陈一新对与会代表们的踊跃建言表示感谢，要求相关部门认真采纳、深入研究。

“新的发展阶段呼唤新民营经济，发展新民营经济，目的是富民，动力是创新。”陈一新认为，新民营经济“财富共享度更高、创新动力更足、捕捉市场能力更强、体制机制更活”，是“富民兴汉的源头活水、创新驱动发展的生力军、深化供给侧结构性改革的重要力量、培育名企名品名家的摇篮”。

新民营经济“新”在哪？陈一新初步概括：创业主体新、发展动力新、商业模式新、融资方式新、治理结构新、市场需求新、产业领域新、集聚格局新。他说，发展新民营经济，党委政府要在政策创新和环境营造上下功夫，做好服务工作，鼓励引导支持企业加强技术创新、产业创新、商业模式创新、管理创新，以创新驱动发展。d



深度操作系统服务器版

率先实现了对x86和国产CPU平台的统一框架支持
金融、运营商、政府、国防、电子政务等多领域应用
通过了工信部相关测试认证

深度服务器操作系统特性与规划

© 北京 研发团队

概述

作为操作系统厂商，深度的核心任务就是打造一款安全可靠、可用性高、兼容性好、性能优异、易用易维护的操作系统。只有拥有了一个优秀的平台，让其它软硬件厂商能在其之上开发出一流的软硬件，进而构建完整的生态圈，才能迎来整个产业的繁荣。

鉴于 Linux 服务器在服务器行业内使用非常广泛，因此在其上优秀的开源软件众多，有不少看成是世界一流水平的软件，例如 Web 服务器领域的 Apache 与 Nginx、数据库领域的 MySQL 与 PostgreSQL、中间件 Tomcat 与 WildFly、虚拟机平

台 KVM 与 OpenStack、容器平台 Docker、监控报警系统 Zabbix、Statsd、Grafana、自动化运维软件 Ansible 与 Saltstack 等等。因此在软件组件的选择上，我们会遴选在各个领域最好的，得到了大量用户认可与广泛使用的软件推荐给用户，只有在无法满足我们的一流要求，而且用户确实需要的情况下，我们才会着手自行开发对应的软件，并承诺对相应软件的长期更新与维护。

非功能性需求

对于深度服务器操作系统，我们认为除了通用于常见的 Linux 系统的功能以外，应该特别对于操

作系统的非功能性特性加以强调。一般而言，对于任何系统的非功能性特性来说，都至少需要考虑下面几个方面（有时可能还会考虑更多方面）：

- **兼容性 (Compatibility)**，它指的是操作系统应该支持目标领域的绝大部分软硬件，包括旧有的设备与系统，以及新研发的设备与系统。这一方面要求操作系统在硬件驱动接口上能继承旧有的规范，另一方面在软件开发接口上能支持最广泛的软件运行环境，在这一点上，使用 Linux 操作系统是一个可靠的保证，当然，我们也需要付出更多的努力，保证最广泛的接口兼容性来提高整体系统的兼容性。

- **可用性 (Availability)**，一般适用于服务器系统，表示系统长时间运行的可能性，可用性为 100% 的系统表示此系统从不宕机，此时可用性最大，但在实际情况中是不可能的。如果系统的可用性为 99.999%，则表示若持续运行一年，平均的宕机时间为 $365 \times 24 \times 3600 \times (1 - 0.99999) = 316$ 秒，即不到 6 分钟。在五个 9 的可用性要求下，需要系统自动化程度与容错设计与实施非常优秀。

- **性能 (Performance)**，同时适用于服务器系统与桌面系统。对于服务器系统一般表示的是在同样的硬件资源（电量、处理器、内存……）约束下，服务器系统能够保证一定的服务质量（例如不出错、每次请求响应的延迟不超过 500 毫秒等）的前提下，能够同时支撑的最大用户数，此指标越高越好。对于桌面系统一般表示的是在同样的硬件资源约束下，运行同样的程序，消耗的系统资源（电量、处理器、内存、磁盘、网络带宽）与请求响应延迟的情况，此指标越低越好。

- **安全 (Security)**，同时适用于服务器系统与桌面系统，一般对服务器系统来说更看重，因为其服务的用户更多。它表示的是系统是否能让所有的参与者得到他/她按照系统预先限定应该得到的信息/数据。一般它牵涉到的有用户鉴权、数据保密、访

问控制、数据完整性、数据备份与恢复、系统审计等。

- **可伸缩性 (Scalability)**，一般适用于服务器系统。它表示的是当系统负荷上升时，系统所需要的资源呈现怎样的上升趋势。在最理想的情况下，我们期望资源增长与用户增长呈线性关系，但是实际上，往往很难做到这一点，主要的问题就在于系统必须有全局性的中控功能，例如各用户之间的交互，以及系统资源的共享。对于单机服务器系统而言，主要指的是多核情况下系统是否可以承担合理的更多的业务负载，包括计算负载或者网络处理等。

- **可维护性 (Maintainability)**，同时适用于服务器系统与桌面系统。它表现在多个方面，对于系统开发与后续版本维护来说，它意味着一旦对系统进行更改（安全补丁或功能升级），系统是否容易修改。对于系统运营来说，表示系统运维人员是否能够方便地对系统进行监控与管理。

- **稳定性 (Stability)**，同时适用于服务器系统与桌面系统。表示的是在同样的操作序列下，系统是否有一致的表现。例如在不同负载下，系统表现出来的延迟是否一致，系统的吞吐量是否一致，系统的行为是否一致等，后者的例子如在鼠标移到桌面的同一个图标上的时候，是否总是会有同样窗口的缩略图展示在一致的位置上。

- **易用性 (Usability)**，同时适用于服务器系统与桌面系统，但是主要适用于桌面系统。表示的是系统使用方式是否方便快捷、美观大方、不伤害使用者（比如亮度能自动调节）、便于残障人士操作、适用于定制（例如设置字体、时区、主题、安装卸载升级等）。

在上面几个特性中，某些特性是有关联性的。例如性能与易用性有时是正相关的，如用户一般期望系统运行与操作快捷，这实际上既是易用性方面的要求，也是性能方面的要求。可维护性与易用性在很多时候也是正相关的。但是性能一般和安全是



负相关的，很多时候，更高的安全性会带来性能的下降，很多时候也会带来易用性的下降。

以上几个方面，将会是深度服务器操作系统长期跟踪，持续优化的方面。

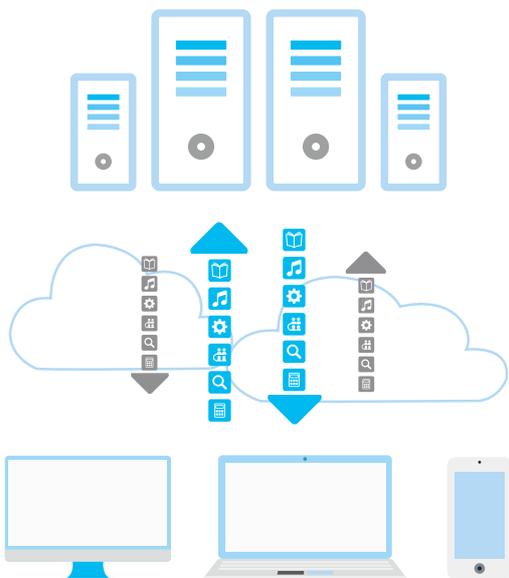
版本计划

我们将服务器版本划分为大版本与小版本 (Update):

- 大版本为整数，例如 V15，或称 15.0
- 更新为小数点后的数字，例如 V15.1，或称 15.1

大版本维护周期为 7 年，小版本跟随相应大版本的维护周期。大版本发布周期为 3 年，小版本发布周期在开始 3 年为半年，后来 4 年为 1 年。

例如 V15 发布于 2016 年 5 月，则其维护周期到 2023 年 5 月结束。V15.1 发布时间为 2016 年 11 月，V16 发布时间为 2019 年 5 月。当然，实际发布周期可能会随着当时的情况略有不同，但是不会误差很大。



小版本实际上就是上个版本积累到现在的更新的集合而已，为了方便用户，特意将所有的更新加上原来的系统安装镜像生成一个新的系统安装镜像(ISO)。

对于深度服务器操作系统的第一个正式发布版本，我们将其版本号命名为 15 (出于 2015 的考虑，而且与深度桌面版尽量保持一致)，这也是我们的第一个大版本。下一个大版本号将是 16。

除了数字版本以外，我们也将每个大版本用中国传说中神兽的名字来命名。其中第一个版本的名字将是夔(kui)，第一个版本用夔(kui)为名是借用了夔独角独足吼声大的传说，独即为一，吼声大既是指代初生婴儿的呱呱之啼，也希望能吼醒国内的 Linux 操作系统圈。

发布产品

深度操作系统服务器版软件支持 x86_64 和国产 CPU 架构平台，发布产品包括：

- 深度操作系统服务器版软件 (x86_64 平台)
- 深度操作系统龙芯服务器版软件 (龙芯平台，3B1500/3B2000 等)
- 深度操作系统申威服务器版软件 (申威平台，1600/1610 等)

此外，基于深度操作系统服务器版软件，发布解决方案产品包括：

- 深度虚拟终端管理系统
- 深度高可用集群软件
- 深度日志分析软件

每次产品发布将包括如下内容：

- 系统安装镜像 (ISO 文件)
- 发布说明、产品介绍资料
- 产品技术白皮书、用户安装手册、用户使用手册

产品优势

深度操作系统服务器版软件具有如下产品优势:

更安全、更可靠

- 基于开源的 Linux 内核构建, 保证了软硬件兼容性。
- 通过定制裁剪进一步提高了系统的稳定性、兼容性、性能与安全性。
- 为企业 IT 基础架构或数据中心的快速部署提供全方位的解决方案。

更稳定、更高效

- 依托多年的 UNIX/Linux 研发经验, 为政府、行业等大客户提供定制服务。
- 提供深度定制和裁剪服务, 可满足用户的特定需求。
- 提供定制化的解决方案、技术支持、产品运维等服务。

广泛支撑平台

- 大数据: 集成和快速部署 Hadoop, 协助用户搭建大数据应用的基础服务平台。
- 云计算: 为虚拟机应用进行裁剪优化与模块内置, 在更小体积达到更好性能。
- 容器应用: 支持构建 Docker 服务平台, 集成集群管理、性能监控等管理功能。
- 高可用: 集成深度高可用集群软件, 为用户打造更稳定更高性的应用支撑平台。
- 可信计算: 支持 TPM、TCM 和 TPCM 标准, 提供可信免疫框架和高可信保障。
- 自主可控: 支持基于国产 CPU 平台, 构建全国产化应用开发和运行环境。

技术特色

深度操作系统服务器版软件具有如下技术特色:

具有自主选择权和控制权

深度科技掌控深度操作系统的所有源代码, 通过深入分析操作系统产品的功能定位和用户需求, 在深度操作系统基础版本之上, 自主定义深度操作系统系列产品, 可以满足不同领域不同应用场景的需求。

全面支持主流 CPU 架构

深度操作系统服务器版软件全面支持包括 x86_64、龙芯、申威、ARM、兆芯等主流 CPU 硬件平台, 为用户提供更多的平台选择。

稳定可靠和优越性能

深度科技基于多年的 Linux 操作系统产品研发经验, 基于稳定的 Linux 内核, 结合服务器产品的功能需求和特性, 对内核与系统软件进行合理的定制和优化, 在保障系统稳定性和安全性的同时, 尽可能地提高系统的整体性能, 并可根据用户场景进行性能调优服务。

更强兼容和易维护性

深度操作系统服务器版软件兼容当前主流服务器设备, 提供对主流中间件和数据库软件产品的适配支持, 为用户构建软硬件系统架构提供更多的组合选择。同时, 提供简单易用的系统维护工具, 有助于提高运维效率。d



>>>

做好 deepin 与客户的 桥梁

深度科技 资深工程师 李鹏

深度科技的用户除了惊叹深度操作系统让人惊艳的桌面环境，更是对深度科技的售后服务大加赞赏，他们爱深度的系统，更爱深度的服务。

之所以被大家成为“让放心的操作系统”，深度科技的售后服务团队功不可没……



这样的我，这样的深度

记得那是 2008 年的春天，告别了校园生活，经历了某通信公司七轮的面试，我开始了由学生到工程师的转变。记得入职时，培训讲师说从这里离职的人都很感谢在这个公司工作的经历。现在想想还真是，当时周围很多人在公司军事化管理的环境熏陶下，可以说痴迷于技术，醉心于工作，导致加班成了主流氛围，入职导师第一天就带我去买午休用的床垫，后来才知道那是晚上加班用的。后来到了第二家公司，也是做 linux 嵌入式开发的，当时的一个想法是奔着为国家 3G、4G 通信事业添砖加瓦去的，然而时间一长，国企工作氛围的种种固有特点实在不适合一颗躁动的心。就这样，我来到了深度，一个充满活力，讲究情怀的团队，恩，当时介绍我来的现任老大就是这么说的。

事实证明，的确如此。深度很有活力，大部分都是年轻人，个人认为，年轻就会创造更多的可能，年轻就意味着更多的激情，激情是成就一切的必要条件。记得路遥说过一句话：只有初恋般的热情和宗教般的意志，人才有可能成就某种事业。初恋般的热情不就是激情么……那谁，你想啥呢！

来到深度给我印象最深的另一件事也是介绍我来的现任老大说的，那就是扁平化的管理，畅通的沟通。举一个例子：刚来公司一周左右，有天中午三个人一起去吃饭，其中一个和我互不认识，吃完饭他买了两个饮料，一看我也是一起的，又跑去买了一瓶，还说不好意思，不知道你也是一起的，我说没事。边吃边聊我才知道这就是 boss 刘，当时真有点冒汗。时间长了，管理者们在工作上一丝不苟，生活上平易近人的作风越发明显。





扎根深度北京运营中心

在我来到深度两年的时间里，公司虽说换了两次办公地点（第一次我在公司附近租了个房子，心想这下上班可近了，结果公司搬家了，但是这并没有让我觉得飘摇不定，反而有一种不断向前走的感觉。之所以有这种感觉，我认为很大程度上是由于工作方向和目标的一贯坚持。

我所在的部门是技术部，主要承担售前、实施和售后工作，以及服务器研发和测试工作。平时在工作中与用户和其他厂商的人接触较多，做过用户需求调研工作，如知道创宇桌面迁移项目；做过适配测试和实施工作，如工商总局云平台项目；做过服务器系统灌装实施工作，如浪潮服务器出厂灌装。在这些工作中，和相关方的合作沟通是否顺畅往往直接影响到项目的进展，甚至一言一行在别人眼里都是公司形象的反映。

操作系统作为最复杂的软件系统，涉及到的内容很多很多，做到面面俱到是个很高的要求。这一点从技术支持工作中就有明显的体会，有时候用户会问某个工具在你们系统下怎么使用，查看某个信息在你们系统下用什么命令，修改某个配置在你们系统下如何修改，诸如此类的问题，要能够给用户准确、专业的支持，就需要平时多积累、多实践。当然，遇到拿不准的问题，首先要有一个正确的态度，告诉用户打算如何处理，能在什么时间给出答复，正确的态度往往会让用户更有安全感。

几个案例分享

商标局 oracle rac 实施案例

在商标局某项目的实施中，涉及到 oracle11g

rac 的安装，安装 oracle11g rac 相对来说是比较复杂的实施工作，一般是由用户请专门的数据库工程师来实施。安装之前，我们先把深度系统针对 oracle11g rac 的依赖包全部打好，然而在数据工程师安装时，仍然卡在关键的一步进行不下去。此时用户说法是，我们就等着这个数据库服务器的部署了，其他业务系统都等着呢，整体进度受到了影响，务必全力尽快解决！

那一天，我和同事搞到晚上十一点，死活就是过不去，然后请求了援兵，组长驱车穿过半个北京在夜里十一点多把已经钻进被窝的研发同事拉出来，然后我们四个熬了个通宵，然而并没有进展，问题依然是问题。第二天早上说是吃口饭再接着搞，刚出门用户来电话说马上过来开会，一进会议室，气氛森严，可以看出来用户也很忧心。用户的领导说我只问你两个问题：第一个，这问题能不能解决，第二个，最快什么时间能解决。我咬着牙说，这个问题能解决，乐观估计一到两天内能解决。然后，公司大神去了现场，为了排除干扰准确定位问题，每次都需要干净环境来测试，重新安装系统，重新安装 oracle，跟踪问题，重复了将近 10 次。可以感受到每个人内心都是崩溃的，但是大家的目标是明确的，必须坚持完成这个目标，然后我们五个人又熬了一个通宵。终于，问题解决了，oracle 11g rac 顺利安装上了，第二天早上进用户会议室时，气氛又变得祥和了。

整个过程像打仗一样，敌人就在眼前，碉堡必须炸掉，虽然过程很艰苦，但是满足了用户的需求，这就够了。

某国家机关信息系统项目实施案例

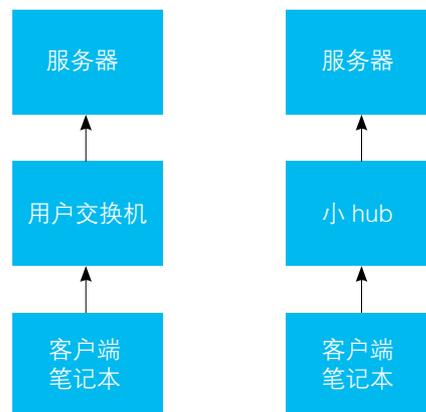
在国家某机关的信息系统部署项目中，同样是



面临着时间紧任务重的情况。三天后就要召开的全国性重要会议就要使用深度操作系统，然而一台服务器的操作系统出现死机情况，而且刚开机就死机，毫无征兆。怎么办？从哪儿下手？刚敲了几条命令就死机了，刚看了几行日志又死机了。第一感觉是这台服务器硬件有问题，然后插入 u 盘从 live 系统启动，打算看看日志里有什么信息，喜人的是 live 系统也死了，这基本就确定了是硬件问题。跟集成商和用户沟通后，叫来了硬件厂商的人，硬件厂商来人对着键盘一顿噼里啪啦后，说要把服务器拉回去测试一下，有可能需要更换配件。为了明确硬件故障，当晚我跟着去了硬件供货商那里，经过各种测试，更换了 cpu 后，不再死机了，果然是硬件问题。就在我松了一口气时，第二天碰到了更诡异的问题。

第二天，业务系统部署完进行了本地测试，一

切 OK，大家都很开心。在大家准备拉拉家常然后撤退时，发现接入用户的交换机后远程无法访问服务器，无法远程访问，等于没有部署。大家都蒙圈了，画图说明下当时现象：



这种情况无法访问

这种情况访问正常



如下可能的原因都尝试了：

- 网线有问题？同样的网线接在小 hub 上就没问题
- 交换机网口有问题？交换机网口接其他设备都能通
- 服务器网口有问题？网口和小 hub 连接就没问题

这时候，第一感觉又来了，硬件问题就是网卡问题，服务器的网卡和用户交换机的网卡不兼容，网络协商不成功，服务器网卡是万兆网卡，交换机网卡是百兆网卡，并且服务器的万兆网卡无法自动协商。建议用户和集成商让硬件厂商更换网卡进行测试，果然，换了网卡后一切 OK。至此，该项目算是上线完成。

这个项目让我明白了一点，判断清楚问题的方向很重要，否则费了很大劲解决不了问题，而项目中留给我们解决问题的时间往往是有限的。

工作中的一些体会

在实际的项目工作中，会遇到各种各样的问题，也会和不同角色、不同性格的人打交道，所有的一切都是不断学习，不断提高自己的机会。

用户的需求很重要

“这个必须要能支持！！”。有时候我们觉得用户的需求不合理，完全没有必要这样做，但是仔细想想，需求合不合理，其实不能单单从技术角度来衡量，需求是一个需要综合考量的事情，客观上方案合理、技术先进、可靠，但又要考虑用户主观上的接受度和思维模式，两方面都不可废弛。

举个最近的例子，在一个桌面项目的测试中，用户提出他们很多人有老花眼，需要桌面字体足够大，一开始大家觉得这个需求匪夷所思，为啥不带老花镜呢？但这就是用户需求，用户就是有这样的特点，这就是他们特有的应用场景，我们不能讲事



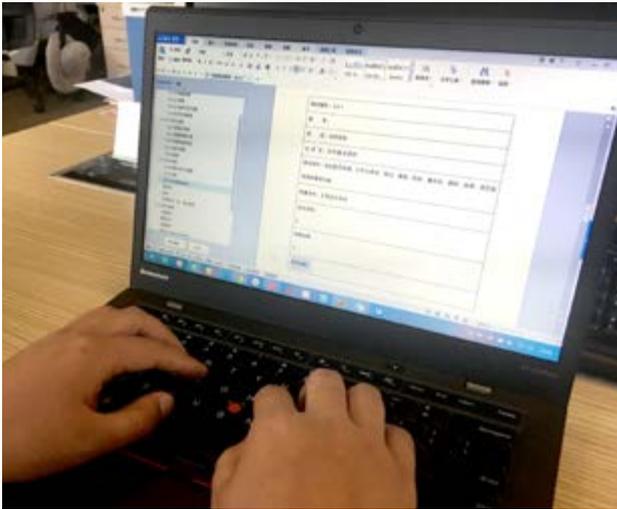
实摆道理地说你这个需求不合理，因为没有这样的特性用户用起来就会不方便，甚至没法使用。

用户的问题很尖锐

“XXX 系统就没这个问题……”。遇到这种情况是比较难处理的，因为眼见为实，任你说的天花乱坠，用户看到的现象摆在那里。首先我们要正视问题，不要回避，万不可给用户推诿的感觉，把问题说清楚，把解决方案拿出来，因为用户的目的是解决自己的问题，并不是要为难谁，只要在现有条件下最大程度满足需求，基本上用户还是认可的。

用户的时间很紧急

“下周一之前必须要完成”。这往往是周五得到的消息，这时候就是考验执行力的时候了，当然



要对事情的可行性有个基本的评估，对于客观原因无法完成的目标，要充分沟通，大家达成一致的意见，剩下的就是甩开膀子干活了。

也是最近的一个例子，端午节我们团队组织去草原春游，本来已经沉浸在一片激动雀跃的氛围中了，而就在出发前一天晚上，接到用户运维工程师电话，一台服务器万兆网卡查看驱动正常，就是无法正常工作，端午节后就要使用，节前必须解决该问题。没话说，先把心从草原收回来，去客户现场解决问题先。

对团队合作的看法

有一句话说的好：“聚是一团火，散是满天星”，个人认为用这句话来描述一个成功的团队再合适不过。以团队目标为行为导向，成员之间高效协作，团队才能达到“一团火”的作用，同时团队成员之间取长补短，持续成长，才能达到单兵作战时的“一颗星”效果。

团队合作平时我们能涉及到的主要是部门内部

同事之间的协作以及跨部门同事之间的合作。部门内部协作主要是互相支撑以及补位，比如某个兄弟负责的项目有任务，恰恰这一天他有事去不了，如何让其他同事能够及时有效地补位，满足项目需要，这必须有平时充分的沟通，互相了解彼此负责项目的内容及各个环节。跨部门同事之间的合作主要是需要得到其他部门同事的支持，如获取研发或设计部门的支持，这种情况必须要有足够的沟通能力，首先要让对方愿意支持你，请求支援的流程要合理，请求支援的态度要诚恳，然后要让对方明白怎么支持你，尽可能用对方的语言描述问题，让对方充分理解需要做什么，怎么做才能满足要求，这样方能事半功倍。

和有梦想的人同行

“因理想而出生，因责任而成长”，深度是一个有理想的团队，每个人也应该有自己的理想，相信和一群有梦想的人一起前行，一定是一件很幸运的事。d



Ubuntu 放弃战斗 Linux 桌面的悲哀

© 深度科技 王勇

这几天看到 Ubuntu 放弃 Unity 和 Mir 开发，转向 Gnome 作为默认桌面环境的新闻，作为一个专究 Linux 十几年的老兵和 Linux 桌面的开发者，内心感触良多。

Ubuntu 做为全球 Linux 界的桌面先驱者和创新者，突然宣布放弃自己多年开发的 Unity，相当于在桌面领域直接放弃了战斗向微软投降，这不仅仅是 Linux 桌面的悲哀，更是对于全球 Linux 黑客坚信 Linux 必胜信心的极大打击。

曾经的 Ubuntu 粉丝

我还记得 2006~2007 年，那时候我还在成都写手机游戏，当时年少轻狂的我，一直幻想自己要做最牛掰的开发者，就像科幻电影里面的黑客一样，无所不能。那时候虽然 Windows 玩的很溜，但是一直都在核心技术外徘徊，不知道如何达到个人目标。

上学的时候玩过 RedHat 6.0，也装过 KDE/Gnome 的桌面环境，但是系统中的各种问题，比如无法使用输入法、中文字体配置很差，甚至因为显卡驱动的原因都无法正常开机，所以 Linux 对于当年的我来说，就像神话一样，只有顶级黑客才能玩的转的系统。

直到后面用了 Ubuntu 6.10 和 Ubuntu 7.04，当时的 Ubuntu 可以说是非常惊艳，原来 RedHat 和 Suse 在桌面端的各种细节问题被一扫而光，装上系统以后就可以直接用，而且还可以非常容易的安装应用软件来深入学习 Linux。可以说是 Ubuntu 带领我真正的入门了 Linux，并经过很多年的深入学习，在 Linux 上开发 Emacs 和 Haskell 相关的项目，直

至后面创业做 deepin 操作系统。

从我个人来说，对 Ubuntu 系统以及背后的开发者都是怀着无比的敬畏和崇拜。

我看这么多年 Ubuntu 的发展

Ubuntu 从当年最佳的 Linux 桌面操作系统到今天宣布放弃自己研发的 Unity 桌面，已经有 13 年历史了，在我看来主要有以下几个阶段：

最初的惊艳

Ubuntu 最初的发展目标就是把原来 Linux 系统需要手动操作的基本配置，比如字体、输入法和显卡驱动等问题内置到操作系统中，使用户不用安装系统后还需要跟乱码、中文输入以及显示等问题斗智斗勇，可以说那个时代的 Ubuntu 是解决了当时 Linux 普及的几个重要问题，把 Linux 系统从当时只有开发者才能玩转的高手系统普及到普通的电脑爱好者就可以折腾使用的系统。

酷炫的 Compiz 管理器

我相信很多 Linux 爱好者都惊叹于当年 Compiz 这个 3D 窗口管理器能够达到的酷炫效果，比如 3D 旋转桌面、拖动窗口的果冻效果、一把火烧掉窗口的效果……特别是同时代的 Windows 系统都还是非常原始的 2D 效果，甚至 XP 的窗口连窗口阴影都没有的时候。当时的 Linux 系统的，特别是 Compiz 吸引大部分用户去尝试 Linux 系统，即使最后大家仅仅只是因为好奇或是玩玩，这大大增强了 Linux 系统的曝光率。

从另外一方面，Linux 在当年 Compiz 发展十年以后，反而是那些简单特效甚至没有特效的 Linux 系统得到最高的用户量，原因我觉得主要有两个：

视觉特效：PC 这种生产力的工具最重要的是高效，非常酷炫的特效长时间用，反而会极大干扰用户视觉，进而影响用户对内容的专注力，而且时间长了也很容易引起用户的视觉疲劳，反而是 Mac 那种恰到好处的轻微动画让用户感到优雅和舒服，这方面 Linux 发行版 Elementary OS 做的要好很多。

交互设计与应用：任何操作系统需要长久留住用户，还是需要依靠优秀的人机交互设计和丰富的应用，操作系统只有给用户带来真实的价值，让用户工作更高效和生活更快乐，用户才会长期坚持下去，光靠酷炫的动画吸引，用户一旦视觉疲劳以后，最终还是放弃 Linux，因为一个操作系统不能解决用户日常遇到的各种问题，在用户心中最多就是一个好玩的玩具而已。

Gnome3 vs Unity

在 2011 年底，Gnome 发布了它最新的 Gnome 3 系统，当时 Gnome3 以其酷炫的特效加上 JavaScript 的插件体系而吸引了开源社区开发者的兴趣，特别是 Gnome3 内置 GJS 引擎，可以直接在桌面环境中编写 JavaScript 插件后，直接 reload 即可更新桌面环境的功能和视觉效果，还有吊炸天的 inspector 特性，可以直接通过鼠标点击界面后定位到对应的代码位置，再加上 GJS 可以从屏幕顶部打开一个类似 quake terminal 的下拉调试环境，在调试环境中直接像脚本语言那样实时列出任何对象的属性和方法，立即改立即生效。

上面这些开发者特性，在那时候 Google 满天吹嘘 Html5/JavaScript 的美好未来的大环境下，对于开发者有极大的诱惑力，很多开发者都奔着 Gnome3 这些类似 Web 的开发方式而去，并贡献了

大量好玩的插件。

在 Gnome3 之前，Ubuntu 一直都在用 Gnome2 桌面环境，其实 Gnome3 在 2011 年发布之前已经规划了 2 年，从当时的宣传来说是下一代桌面环境，2009-2010 年 Gnome3 还在社区发布了大量吊炸天的预览视频。作为当时桌面系统先锋的 Ubuntu 也非常期待能用上 Gnome3，而无奈 Gnome3 一次又一次跳票，最后 Ubuntu 实在等不及了，就在 Gnome2 桌面环境后台服务的基础之上，开发了自己的 Unity UI。

当时社区也因为 Unity 和 Gnome3 的很多设计理念类似，而谴责 Ubuntu 在进行分裂行为。

Unity 产品的特点主要定位在几点：

- HUD 的左上角搜索设计，快速搜索应用和很多插件提供的搜索结果，这一点和 Gnome3 的 Launcher 插件体系非常像，通过插件可以搜索除应用外的更多搜索结果（比如天气、在线视频、计算器结果等）；

- 任务栏固定在左边，对宽屏更加优化，相对于程序员有更多的纵向空间；

- 全局菜单，通过合并顶部栏和窗口标题栏，进一步节省纵向空间。

我个人并不喜欢这些面向开发人员（而不是面向普通用户）的设计，但是撇开个人的主观设计喜好外，我认为 Unity 是相对于 Gnome3 更成熟的产品，Unity 的很多改动都是针对 PC 桌面系统的真实痛点来改进的，特别是一些程序员的用户很喜欢 Unity 诸多设计。而不是像 Gnome3 那样一味的追求平板用户体验，极大降低了 PC 桌面用户的习惯和易用性，Gnome3 的槽点我后面详细说。

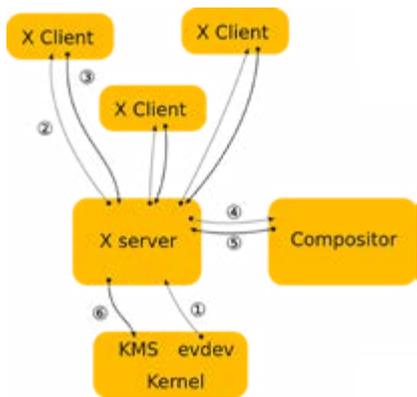
不论从商业公司的独立性发展考虑还是产品质量负责的角度，我都是非常支持 Ubuntu 当时独立开发自己 Unity 桌面环境的行为，Unity 确实在随后的几年里证明了自己的产品质量和设计细节都比

Gnome3 要好很多。

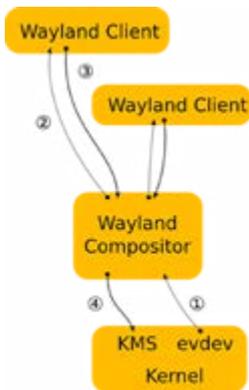
Wayland vs Mir

在 Gnome3 和 Unity 发展的同时，Intel 的雇员 Kristian Høgsberg 正在领导开发新的显示服务器。Wayland 相对于古老的 X11 来说最大的提升是，Gtk/Qt 这些图形库进行图形绘制时，不用像 X11 那样发送绘制消息到 XServer 来进行绘制，而是由 Client 自己进行图形绘制，Wayland 只用担任图层混合器的作用。这样不但减少了 X Client 和 X Server 之间不必要的通讯，而且因为由 Client 自己进行渲染，所以很多画面撕裂和闪屏的现象从原理上就避免了。

大家可以看一下下面的两张架构图来理解两者的差别：



X-architecture.png



Wayland-architecture.png

Wayland 因为要彻底从技术架构上颠覆 Linux 几十年 X11 的渲染方式，不论从架构设计还是代码实现上都会非常复杂，不但要开发协议本身，还需要开发适合 Wayland 的混合器和窗口管理器，最后导致 Wayland 1.0 稳定版一再跳票。

而 Ubuntu 在独立开发 Unity 桌面环境的同时，也仿造了 Wayland 的架构开发了自己的 Mir 显示服务器，除了等不及 Wayland 稳定之外，更重要的是 Ubuntu 要为了它的下一步宏伟计划——Ubuntu Touch，按照 Mark 的设想，Mir 不仅仅要像 Wayland 那样从原理上提升 Linux 图形渲染效率，而且 Mir 还得担负起手机和电脑融合的使命，可以让 Ubuntu Touch 的手机在插上显示器底座时，手机的应用通过 Mir 的支持，可以直接在外接显示器上显示手机应用窗口，最终达到“当你手机放到底座时就是电脑，拿走就是手机”的设想。

不论 Wayland 还是 Mir，虽然底层架构都非常先进，但是为了兼容现有的 X11 程序，它们分别开发了 XWayland 和 XMir 用于在新的显示服务器协议上支持现有的大多数 X11 程序（主要是 Gtk2/Qt3/Qt4 开发的大多数应用）。

具有讽刺意味的是，Wayland 和 Mir 本来就是要解决 X11 那种不适合现代 PC 场景繁琐的通讯协议，甚至很多开发者为了技术的洁癖都在大力安利 Wayland/Mir，但是最后 XWayland 和 XMir 本身的兼容实现却比 X11 的实现更加“恶心”和繁琐，包括 Mir 的开发者最后都放弃 XMir 的开发。

正是因为 Wayland/Mir 这样的技术无法彻底解决和大部分原本就基于 X11 协议而开发的应用的兼容性问题，最后导致基于 Wayland/Mir 开发的桌面系统从“解决渲染性能问题”转变到“无法运行很多现有 Linux 应用”这一个更加让用户难以接受的结果。这也是后面基于 Mir 开发的新版 Unity 难产的重要原因。

Ubuntu Touch

就像每一个程序员都有一个黑客梦一样，每一个 Linux 开发者都希望能用上纯粹的 Linux 手机操作系统，特别是 Linux 技术圈领域，大家更希望有一个面向开发者可玩性更强的 Linux 手机操作系统。

Mark 从 2011 年宣布开发 Ubuntu Touch 到 2015 年第一只搭载 Ubuntu 的智能手机在欧洲上市，Ubuntu 把大多数主力开发都投入到 Ubuntu Touch 手机操作系统的开发中，整整 4 年，世界上最好的 Linux 桌面开发团队把 4 年的时间花费在开发一个在 PC、手机、平板和电视上拥有强迫症般统一界面的 Unity 上，而且希望 Ubuntu Touch 能够占领低端的智能手机操作系统市场，或者像 Mark 说的那样在 Android 外给世界另外一个选择。

这 4 年的时间，因为主力开发 Ubuntu Touch 的原因，Ubuntu 桌面操作系统从原来的高速创新到鲜有改动，往往版本的发布就是软件仓库更新加上壁纸更新就发布新版了，不再像原来一样，每个版本都有易用性的提升和功能增强。

而且大家都知道，2011 年~2015 年这几年正是安卓系统从起步到发展成全球移动操作系统霸主的 4 年，Ubuntu 闭门造车了 4 年，到 2015 年发布第一台 Ubuntu Touch 手机以来，Ubuntu Touch 手机在全球销售惨淡，甚至无人问津。

Ubuntu 今天选择放弃开发 Unity 的重要原因，就是因为 Ubuntu 耗费了公司所有资源，在开发一个用户根本就不会买的手机操作系统，Ubuntu Touch 可以说是 Ubuntu 最愚蠢的决策。

今天的手机和手机操作系统的发展就像当年 Wintel 的 PC 时代一样，芯片和硬件越来越快，硬件的快速迭代更新速度都快于软件层面的优化速度，就算最低端的手机性能甚至都比十几年前电脑更快，而手机作为现代消费品来说，用户更关心的是手机是不是新款的，能否打游戏，美颜拍照功能是否更好，

智能手机时代，大家不再对低端智能手机感兴趣，所以“低端智能手机操作系统”本身就是一个伪命题。

智能手机作为最方便的通讯和娱乐工具，能否打游戏和一些基本应用是使用者最最最基本的要求了，可 Ubuntu Touch 不能玩游戏、不能上 QQ 和微信，只能打开终端和一些网页应用，不要说普通用户，我想就连那些天天嚷嚷“我们需要一款 Linux 手机操作系统”的开发者都不会用它。

强迫症般的统一设计，在 5 寸大屏的手机上，整个任务栏切换全部在左侧，大家可以试一下，在正常拿手机的情况下，能否方便的用大拇指去操作最左侧的 Launcher，甚至还要在手机左侧的顶部和底部上下移动，把手指移动到手机顶部左右移动查看状态栏（你的手指够的到顶部吗？然后再做左右运动？），做为一个创新设计的宣传，这些反人类的设计是我作为一个产品经理非常无法忍受的，我经常吐槽：Ubuntu Touch 的交互设计师平常要不就是把手机放到桌子上设计的（而不是单手体验一下），要不就是 Ubuntu Touch 手机是屏幕厂商开的，Ubuntu Touch 的很多交互设计都非常容易让手机从手中滑落。

Ubuntu Touch 不论从公司战略、产品时机、应用生态还是交互设计等方面来看都是非常愚蠢和失败的产品，而这一切都要怪罪于 Mark 本人开发者的产品思维和一意孤行，白白浪费了世界顶尖开发者的精力和理想。

FlatPak vs Snap

Linux 桌面发展到今天依然无法和微软以及苹果系统本身竞争的一个重要原因就是，Linux 系统软件包之间的依赖问题，系统和应用之间的紧耦合，应用和应用之间的依赖库紧耦合，只要底层库升级了，系统升级有可能会把应用升级挂，一个应用升级很有可能会导致另外一个应用无法使用。



如果把系统所有的软件包和应用的依赖打印出来，你会发现整个桌面 Linux 操作系统就像一张巨大的错综复杂的蜘蛛网，蜘蛛网的任何一环坏掉都会引起系统和应用不稳定甚至无法使用的问题，同时应用厂商为了开源社区的诸多底层库升级疲于奔命，而最后的结果是，Linux 操作系统在底层 5 万个软件包、2000 多个应用、700 个精品应用的时候，就已经非常非常的不稳定了，如果有一天 Linux 平台有上有十万个应用的时候，系统和应用一定会崩溃的，因为任何底层的技术革新都会带来应用厂商的应用失效或者因为 API 不兼容最后无法运行。

所以，为了迎接将来 Linux 操作系统应用生态爆发的那一天，Linux 操作系统必须像微软和苹果那样，在系统和应用之间构建一条抽象 API 层，这有点类似微软的 Win32 API 和 .Net 的意思，只要应用遵守中间层运行时 API 和接口以及应用之间做到互相隔离，应用升级就不会影响到其他应用，系统底层做再多规模的底层库革新，只要革新完以后保证中间层调用接口不变，系统的升级就完全不会影响到应用的稳定。

举一个最简单的例子，微软从 Windows XP 开始，操作系统底层技术一直在变，但是你会发现十几年前应用厂商为 Windows XP 开发的应用依然可以在 Windows 10 最新的操作系统上运行，这种类型的兼容性才是桌面应用生态赖以生存的重要技术保障。

而 Gnome 开发的 FlatPak 和 Ubuntu 开发的 Snap 就是为了解决应用之间的隔离而生的技术，这两个技术都通过 Linux 内核的 cgroups 技术，FlatPak 是通过 OsTree 来解决应用多版本的问题。通过把应用之间隔离在沙盒之内，保证不同应用之间或者应用不同版本之间完全在文件系统上进行隔离，这样就彻底解决应用之间升级互相影响的问题，而且这两种技术通过引入应用程序运行时的 runtime 接口，保证了系统和应用之间有一个稳定的接口，

从侧面也保证了系统和应用之间的依赖分离。

FlatPak 和 Snap 都能解决 Linux 系统依赖的问题，FlatPak 的目标是解决 Linux 桌面现有应用的问题，Snap 则是为了解决 Ubuntu Touch 而生的。

听说 Mark 希望 Snap 不但能够运行在 Linux 上，还能运行到别的操作系统上。在我看来这也是一个错误的选择，微软和苹果早就有相关的成熟技术，根本不需要 snap，反而是 Linux 才需要这样的技术，希望 Ubuntu 能够把 snap 的技术发展壮大，有利于 Linux 操作系统的应用生态。

Gnome 是一个比 Unity 更差的选择

从 Ubuntu 放弃 Unity 的那一刻起，Ubuntu 已经放弃桌面了。很多不了解桌面环境开发的用户会觉得 Ubuntu 不是回归 Gnome 了吗？应该是开源社区的好事啊？

桌面操作系统的用户体验和交互细节息息相关的就是桌面环境，而 Gnome3 桌面环境从各方面来说都不是面向普通用户的高质量产品。

核心开发者更关心技术折腾，不再关心产品质量和 API 兼容性。

Gnome 好的地方是开发了封装大量基于 GObject 的库，外部开发者能够快速基于这些 GObject 库来开发程序。

但是从开发者的技术选型来说，比如最开始做 vala，gnome shell 的时候做 GJS/JavaScript，到最近讨论用 rust，这些都可以看出 Gnome3 的开发者更关注的是实现桌面环境后台的编程语言和技术是否是最酷最新的，追求这个世界上最先进的技术是每个开发者的源动力，但是 Linux 桌面这十几年之所以发展停止不前的重要原因是：用户喜欢一个开箱即用、美观和稳定的桌面系统，用户不关心实现这些的后台技术是否是最酷的。对于真正理想的研发团队，或者真正的顶尖开发者来说，编程语言和

最新的开发库只是工具和程序表达方式而已，只要编程能力高，什么程序都是可以开发的。

而 Linux 社区的开发者，大多数开发者本身都是喜欢折腾最新技术的，因为最新的技术对于开发者来说可以学到很多新的知识，他们在新的编程语言和新的技术基础之上能够做出自己的东西，让自己更有价值和快乐，而这些都是非常自然的地方。但是我们应该考虑在先进技术的同时，面向真正的桌面用户做出稳定可靠的产品。

Gnome3 即使面向开发者都是极度不友好的，整个桌面环境就完全在一个 JavaScript 的环境中，中间没有任何整体的 API 设计和插件接口，Gnome3 的插件作者都是通过 Gnome Shell 调试器追踪到一个 JavaScript 的 Object 对象，通过属性链的方式魔改很多对象的属性方式，最后来开发出 Gnome3 的插件。不通过插件 API 接口，而是通过对象的属性来编写的插件，导致只要 Gnome3 的核心代码一改变，或者属性链中的任一个变量发生了变化，Gnome3 插件就失效了，目前 Gnome3 在线插件的页面也只有 20% 的插件能够在最新版的 Gnome3 环境中安装运行。

Gtk+ 已经无力和 Qt 抗衡

我个人作为 Gtk+ 骨灰粉，也是 Gtk2hs 的核心开发者，对 Gtk+ 2000 多个 API 以及很多 GObject 库（比如 vte, webkit, poppler 等）都非常熟悉。但是自从 Gnome3 弄伤心以后（我曾经写了 12 个 Gnome3 插件，1 万多行代码都因为 Gnome3 各种乱改 API 后升级失效），我对 Gtk+ 的信心也随之大大降低。

特别是 Gtk+3 这个被 Gtk+/Gnome 开发者盛赞的版本，除了添加了一下 Overlay 的控件和 CSS 属性功能外，再没有大的功能改进。反观 Qt 从 Qt4 时代的各有千秋到 Qt5 的大幅度提升质量和功能，包

括最新的 QML 大大提升了图形应用程序的开发（虽然 QML 对显卡兼容性还是有点问题）。

Gtk+ 的很多 API 至今多有很多隐藏的小技巧，需要多年的踏坑经验才能完全掌握，而 Qt 本身的 API 设计要优秀的多。

Gnome3 不是一个产品质量的桌面环境

Gnome3 并不是一个产品质量的桌面环境：

JavaScript 本身不适合作为桌面环境这种常驻程序的开发语言，不论在执行速度，特别是在内存的控制上非常容易导致长时间运行不关机，内存越用越多的情况。

Gnome3 除了窗口管理器 mutter 是单独一个进程外，所有的桌面模块（任务栏、顶部条、启动器、工作区）都是在同一个进程中运行，这样就导致只要有任何插件发生错误，整个桌面环境的所有模块都会连带崩溃而无法使用。

Gnome3 的整个设计理念都是面向平板的，比如托盘区域默认隐藏，任务栏要戳一下热点才能显示，启动器图标过于巨大等，这些设计都会严重的影响 PC 日常使用的习惯和易用性。

所以 Gnome3 与其说是不如 Unity 桌面环境（我相信 Ubuntu 的 Mark 也知道），还不如说 Ubuntu 彻底放弃在桌面上的创新和主力研发，彻底向微软投降了。

开源社区 vs 商业产品

从最初的 RedHat 放弃桌面系统，Suse 和微软深入合作，2016 年 Ubuntu 移植 Bash 到 Win10，到如今 Ubuntu 放弃 Unity 和 Mir 后，这么多年，大家在开源事业上努力奋斗，但是从最开始的斗志昂然到如今的放弃，为什么开源的桌面系统做不好呢？

我个人也在带领团队开发 deepin 操作系统，这么多年开发 Linux 桌面操作系统的经历，我个人对



于开源社区和商业产品的理解有几点：

Geek 是这个世界的革命先锋，但不能以 Geek 理念做产品

电脑极客在以飞快的速度在改变世界，正是有这些非常聪明的极客和开源黑客，全世界的科技，特别是开源技术在最近十年获得了极大的发展。

电脑极客在所有人的类型是非常稀少的，这些全世界 1% 的人用各自的聪明才智推动开源运动的发展，包括 Ubuntu 的 CEO Mark，本人也是极客中的极客，曾经的 Debian 开发者，在推动 Ubuntu 初期解决了很多 Linux 桌面易用性的问题。

电脑极客与其说是一种职业，还不如说是一种生活方式，电脑极客希望全世界都像电脑运行那样：刚正不阿、完美、统一的美，没有一丝一毫的例外情况，所有的结果都应该在预计和期望范围。

但是不能用极客的思维去做产品，因为这个世界本质就是混沌的、有各种各样的缺陷、更强调社会习惯和协作，而普通用户组成的非电脑社会，更多人只在乎这个社会都在怎么使用电脑，大家在流行使用什么，他们更在于电影明星在用什么，模仿各种潮流，而不在于一个程序员内心在想什么。

Ubuntu 最开始推崇 Unity，是从产品质量和公司的掌控力出发的，开源社区指责的分裂我觉得是不客观的。但是后面为了统一 PC、手机、平板、电视的整个操作界面，所有界面都基于左侧任务栏 + 顶部任务栏 + 搜索导向的 Unity 启动器来设计的，从极客爱好者来看，这是一种从程序代码到总体实现的一种统一，一种认为世界会赞赏的方式来构建 Ubuntu 系统。

反观我们自己作为一个普通人，我们会怎么对待这几样设备？

PC：作为生产力工具，比如要写一个文档，写段代码，做一个复杂的表格和图像时，PC 绝对是生

产力的代表，但是因为 PC 大多数都需要一个人在固定的地方长时间的工作，所以现在 PC 不是沦落了，而是专注于工作环境的计算平台，所以 PC 感觉给人很严肃很累，而人做为一个天性懒惰的动物，大多数人是尽量干完工作就离开 PC 的，但是无论如何，PC 作为整个社会的生产力工具，确实必不可少。

手机：作为大家最顺手的工具，玩游戏、聊天、浏览页面等，可以说是现代人的第三只手，大部分时间都离不开手机这种移动计算设备。

平板：一种非常轻量的电脑，不需要过多的知识，不需要看着密密麻麻的的键盘，只用直觉化的操作即可。

电视：我能想到的最佳状态就是，躺在沙发上，拿着遥控板，看电影，上下左右确定，这就是最常用的操作。

我们认真的总结，即可发现这几个设备的常用交互方式：

PC：鼠标 + 键盘

手机：大拇指

平板：食指

电视：遥控板

所以电脑极客经常问的一句话“为什么不做手机操作系统？”，更多的时候是在问“为什么不能像 PC 那样把别的计算平台做的更加赋予生产力？”，所以当初 Ubuntu Touch 追求的“同一套 Unity 界面，让不同的计算平台用同一套界面和交互设计来满足”，这本身就是一个伪命题，或者说是电脑极客自己的一厢情愿。

这几个计算平台无论从使用场景、用户交互习惯还是用户对这些计算设备的期望都完全不同，当初微软融合电脑和平板本身，都已经让喜欢 PC 的用户大大不爽，最后迫使微软停止融合的设计，改回最初的 PC 操作设计。而 Ubuntu 所有计算平台的操作融合更是错的非常离谱，当第一代 Ubuntu Touch

手机推出市场后，先不论应用的稀缺，光是操作系统就和主流的“看到图标就点，返回就按 Home 键”的主流手机操作系统截然不同，最后惨淡收场。

所以，极客本身是世界革命的科技先锋，但是不能用极客思维去做产品，因为极客毕竟还是这个世界上的非常少数的人，极客思维本身无法代表更为广泛的普通用户。

底层技术的革新需要采用温水煮青蛙的方式，而不是激进的踏进沙漠去构建大海

经常有人问我，“deepin 什么时候用 wayland”，我首先想回答的是“合适的时候就会用”，但是我更想回答的是“deepin 只会推动稳定的技术和产品给用户，而不是拿用户当小白鼠”。

Wayland 和 Mir 本身都是非常革新的显示技术，我相信也是 Linux 未来显示技术的主流，但是作为现阶段来说，Wayland 依然不是一个稳定的产品级方案：

1. 显卡厂商原来都是基于 X11 的 libGL 进行驱动编写的，虽然现在很多显卡厂商也在往 Wayland 迁移，但是整体的显卡驱动兼容性还不如 X11，直接的问题就是一旦用了 Wayland 后，桌面系统会相对于 X11 更容易遇到花屏和无法登入图形界面的情况。

2. Wayland 本身就是对于 X11 协议的一种颠覆，大量的现有 Linux 程序依然是用 Gtk+2/Qt4 开发的，这些应用程序依然无法在 Wayland 上运行，而新的 Gtk+3/Qt5 虽然都已经对 Wayland 进行了支持，但是这类应用还非常少，导致移植到 Wayland 会有大量的应用程序无法使用，虽然有 XWayland 项目，但应用兼容性和重写的问题短期之内还不会有很大的改善。

3. 在 X86 这种 CPU 性能过剩的平台，Wayland 的性能提升并不明显，最起码普通用户无法察觉这种性能提升，我认为 Wayland 现阶段更适合那种能

够完全控制显卡和应用本身，并且对渲染性能特别敏感的计算平台，不适合 PC 这种开放式的高性能计算平台。

在显卡厂商逐渐支持 Wayland 改善显卡驱动的情况下，在 Linux 桌面，我觉得应该大力把桌面环境已经应用迁移到 Gtk+3/Qt5 的图形库上，等有一天显卡厂商驱动兼容性足够好以后，通过修改窗口管理器和移植应用的方式，直接把上层的桌面环境和应用软件整体迁移到 Wayland 上。

当迁移完成后，用户在显卡驱动以及应用方面完全感受不到差异的时候，才能说是负责任和可持续的迁移方案。而不是很多开源社区的用户说的先迁移到 Wayland 上去再说，这样会导致使用 Wayland 的时候，让用户同时面临显卡驱动和更加匮乏的应用生态问题。

打个比方，整个开源社区是全世界，革新技术是革命领袖的话。任何一个 Linux 桌面系统从 X11 迁移到 Wayland 更应该考虑策略和渐进的替换方式，而不是一脚踏进沙漠以后要构建整个海洋。任何一项革新的技术除了技术先进性一定要考虑革命时期的群众基础，只有在革新的时候解决用户的问题，得到用户支持，才能完成底层核心技术的切换，否则就和当年 Intel/HP 大力推崇的安腾 CPU，虽然各种技术先进，但是因为不兼容现有操作系统和应用，最后不得已而抛弃。

安卓生态时代，不要做 Linux 手机操作系统，这不仅是一个技术问题

Linux 桌面面对的是微软的生态帝国，如果拼尽全力，可能还有 1% 的成功可能性。而 Linux 手机操作系统面对安卓的时候，会死的一点都不剩的。

为什么会这么说？因为微软花了几十年构建的生态帝国，是靠当时微软在 PC 行业的优势，当时的时势加上微软自身的努力（只有做过操作系统生态



的团队才能理解微软当年有多苦逼)做到今天的地步的。做操作系统这种承上启下+白菜钱白粉心的企业一般都是前期是孙子后期是爷的命,前期遇到硬件应用和各种各样奇怪的问题,操作系统厂商各种背锅(为啥别的操作系统没有这个问题?),干各种不是操作系统应该干的活,以一己之力推动整个生态的发展。而现在PC的所有生态和需求都已经定型了,微软不会鸟你,也没有各种不懂的媒体跟风干扰你(他们最多就说PC是夕阳产业,哈哈)。而你和你的团队只需要一点点的努力,用十倍于微软苦逼的努力,只要在某些行业和开源社区做出成绩,就会有用户认同你,PC再夕阳大家也不可能离开它,因为生产力是整个社会最基本的动力,大家都用手机和平板办公,整个社会的生产力就会成倍的降低,所以大家离不开PC。我经常讽刺那些说PC没用,平板才是未来的朋友:“你用平板写代码或者你们公司的财务用平板给你算工资,我就愿意相信你”。

移动终端本身就是面向普通老百姓的新一代计算设备,在这个行业所有的人、所有的企业都在围绕安卓构建新的应用,面临一个日新月异更新换代的行业,对于操作系统厂商来说,这些不是机会,而是你无法挑战安卓系统的巨大阻碍,而且每天都会更大,因为有更多的开发者和开发厂商自己掏钱投入安卓阵营。人类社会是一个以跟风和模仿为基调的动物群体,当你做的和大多数人都不一样,各种人就会干扰你,导致你无法专心做好操作系统,而且面对一个不断发展的生态,整个行业的切换成本是你现在无法撼动安卓的重要原因,因为安卓构建的不光是操作系统本身,构建更多的是日益越大的行业投入和切换成本。

反观PC行业,在真正有下一代生产力计算平台出现之前(安卓只是娱乐平台),PC依然是整个社会生产力的核心,大家不可能离开PC,PC也不会

像很多没有思考能力的人期望的那样消失殆尽。任何PC操作系统厂商只要一步一步的花费数十年的时间,终有一天会构建庞大的桌面应用生态,在生产力和开放源代码上给这个社会带来更多的价值,其实Mac系统本身就是一个很好的例子。

PC操作系统不是未来,但是PC操作系统是现代还不可或缺的计算平台,而且PC操作系统是全世界对软件工程要求最高的挑战。当一个团队具备实力能够完整构建像微软那样的PC操作系统能力和知识积累后,这个团队的实力能够在下一代计算平台掌握绝对的控制力。

大家熟知的励志故事就是:苹果在PC被微软打爆以后,用iPhone让微软再也爬不起来。

包格式只对系统核心有用,对于应用厂商的无意义

我一直认为包格式是服务器操作系统和桌面操作系统系统核心才需要的东西,因为具有依赖关系的包特别适合构建相互关联的操作系统底层核心库,只要你升级一下核心库,整个操作系统的核心技术就升级了,不用担心冗余和更新问题。

但是包格式对于应用厂商无意义,操作系统站在应用厂商的出发点最应该考虑的是提供一套接口抽象层和应用隔离机制,抽象接口层的目的是分离系统和应用,应用隔离机制的目的是隔离应用之间在文件系统上的依赖。

这样,才能真正构建一个Linux桌面的生态:

- 抽象层做好以后,操作系统可以在保持接口不变的同时,随意革新操作系统核心技术,而不用担心操作系统升级以后,应用程序无法运行的问题。

- 应用隔离机制做好以后,应用厂商的不同版本和不同应用厂商的应用之间不会互相影响。

- 开发者不用再写完软件还要考虑构建不同发行版的打包格式,这也是阻碍软件无法在所有Linux发行版运行的重要原因。

这样做的目的是让彻底解放操作系统厂商，操作系统厂商的开发人员可以把所有精力放在核心技术的开发和应用开发接口的维护上，而不用每天除了开发操作系统还需要打包各种应用软件包。

应用的质量和功能应该由应用厂商自己维护，而不是丢锅给操作系统开发人员，应用的好坏本质上应该通过用户评价和经济刺激让应用厂商自己维护，如果应用长期没有作者维护，用户自然会抛弃应用的，进行自然淘汰，就像 Windows 上的应用，哪个作者更努力做的更好，用户就会越多。

而不是像现在 Linux 发行版的打包人员，固定的几十个人面对着成千上万的应用软件打包工作，不论从工作量还是持续发展来说，这种分担工作量的组织方式本来就是不合理的，如果每个应用都需要操作系统开发人员来维护，固定人数和固定时间面对无穷的应用时，势必会导致精力不够，甚至核心的系统开发者不再给开源社区做贡献时，他维护的应用的更新和质量就会下降，最后损害的还是用户。

所以应用隔离的技术不仅仅只是技术上模仿微软和苹果，而是从社会的生产力上彻底解决工作分担组织的问题，只有把应用的维护工作丢给应用开发者这些直接利益关系者才能持续的推动整个生态的前进，而不仅仅通过开源精神和信仰把所有的维护负担都丢给 Linux 发行版的开发者。

Linux 桌面的未来在哪里？

说了这么多，Linux 桌面的未来在哪里？

实话实说，我不知道，我要是知道的话，deepin 早已经打败微软了，哈哈哈。

但是从我这么多年开发桌面系统的精力，我总结了几条经验：

- PC 不需要革命性创新，只需满足桌面用户长期的习惯 + 微创新，大多数 Linux 发行版自嗨式的折腾后，用户告诉我们他们最需要的 PC 操作系统就

是那些尽量满足他们 PC 操作习惯的系统，如果有一些局部的微创新能够形成自己的特色就很好了。

- 大力发展应用生态才是王道：Linux 精品应用还在千这个量级，不论自己研发，还是和应用厂商合作开发，抑或通过商业来刺激应用厂商自己开发，才会有越来越多的用户和行业去使用 Linux 操作系统，Linux 桌面要成功，这是最最重要的一点，而不仅仅是拿开源作者或学生业余写的软件当做产品去忽悠现代这些越来越挑剔的桌面用户。

- Linux 桌面需要更深入的去解决普通用户的问题：需要针对 Linux 桌面用户的痛点进行更深入的解决，除了上面说的应用外，比如显卡驱动、打印机驱动以及行业应用都是非常必要和重要的。

- 更多的用户基数才能引爆 Linux 桌面生态：Ubuntu 聚集了大多数 Linux 开发者，真正的 Linux 桌面成熟，需要面对更为广阔的普通用户需求，只有解决了 10% 以上用户的需求（苹果 Mac 系统也就这个占有率），Linux 桌面才能自豪的说终于成功了。

Linux 桌面是世界上最大的软件工程挑战，也是所有 Linux 爱好者的希望

Linux 桌面以及桌面应用生态的构建是世界上最大的软件工程挑战，几十年各种伟大的开源公司和开源社区在为这个目标努力，我相信只要持续努力，最终可以从希望变成理想成真的。

Ubuntu 今天放弃战斗了，非常可惜一面 Linux 桌面的旗帜倒下了。

做为依然奋斗在 Linux 桌面事业的开发者，想跟未来的开源开发者说一句话：

累了就休息一下，但千万不要放弃！



勒索病毒提了个醒 自主掌握操作系统有多重要

摘要：如类似漏洞在战时被利用，破坏力无疑相当于在敌国首都扔一颗核弹。

巴菲特不久前在伯克希尔哈撒韦股东大会上说：“我对大规模杀伤武器是很悲观的，但我认为发生核战争的可能性要低于生化武器与网络攻击。”没想到，巴菲特一语成谶。

关于当前全球肆虐的比特币勒索病毒，正有扩大化的趋势，目前受到了“勒索”软件的攻击的机构遍布全球，包括美国、英国、中国、俄罗斯、西班牙、意大利、越南等地。

当前中英两国受害最为严重，据说英国的 NHS 服务受到了大规模的网络攻击，至少 40 家医疗机构内网被黑客攻陷，电脑被勒索软件锁定，黑客则通过锁定电脑文件来勒索用户交赎金，而且只收比特币。

在国内，除了教育网、校园网以外，这种病毒的影响范围疑似在逐渐扩大。微博上有用户反馈，北京、上海、江苏、天津等多地的出入境、派出所等公安网也疑似遭遇了病毒袭击。但总的来说，中国众多高校中招比较严重，很多即将毕业的大学生成为了受害者。

据英国金融时报报道，该病毒的发行者利用了去年被盗的美国国家安全局（NSA）自主设计的 Windows 系统黑客工具 Eternal Blue，把今年 2 月的一款勒索病毒进行升级之后就成为了 WannaCry。

这个病毒会扫描开放 445 文件共享端口的 Windows 设备，只要用户的设备处于开机上网状态，黑客就能在电脑和服务器中植入勒索软件、远程控制木马、虚拟货币挖矿机等恶意程序。

当前多数病毒需要诱导用户主动点击附有病毒攻击代码的附件或者相关木马链接才能中招，WannaCry 病毒的可怕之处是，它无需用户做任何操作，可以在同一个网络的计算机之间传播并且进行复制，形成链条式的传播扩散。

但事实上，根据当前多数安全专家的观点是，被病毒加密的文档基本不可恢复，无论交不交赎金。除非你拿到病毒加密的密钥，但这基本是不可能的。

我们需要注意的是，这个病毒是蠕虫式的传播，利用的是 Windows 操作系统存在的安全漏洞，病毒攻击的那些电脑导致文件被锁定基本都是没有打过补丁的。用户想要开启文档时，勒索软件就会通过弹出窗口、对话框或生成文本文件等方式，向用户发出勒索通知，要求用户向指定帐户汇款来获得解密文件的密码。

我们从这次病毒攻击的目标对象（医院、学校、政府机构、企业）可以看出，病毒危害波及的受害者层面广，往往都涉及到国家企事业单位层面或者公立的医疗机构企事业单位，危及国家层面上的网络安全。

而我们也有必要了解下美国国家安全局（NSA），据说 NSA 旗下的“方程式黑客组织”使用的网络武器有十款工具最容易影响 Windows 个人用户，包括永恒之蓝、永恒王者、永恒浪漫、永恒协作、翡翠纤维、古怪地鼠、爱斯基摩卷、文雅学者、日食之翼和尊重审查。

这其中的网络武器，包括可以远程攻破全球约

70%Windows 机器的漏洞利用工具。这次事件是美国国家安全局 (NSA) 黑客武器库泄露出来的“永恒之蓝”实现的，实现方法是加密重要文件，这个加密算法以目前的科技水平基本上没法破解。无需任何操作，只要联网就可以入侵电脑，瞬间血洗互联网。

那么这次只是黑客无意间拿到了 NSA 其中的一款工具——“永恒之蓝”牛刀小试，就已经血洗全球互联网。这无疑在告诉大家，全世界的电脑这么多年其实一直都在裸奔，直到这次黑客终于把这些黑客工具从美国国家安全局 (NSA) 偷了出来加以利用，人们终于意识到了这种裸奔的风险。

那么如果没有这次黑客的行动，我们是否可以认为，人们的网络安全其实也是被捏在 NAS 手里？只不过，人们愿意相信，NAS 是不作恶的。

但这依然无法回避的一个事实是，许多大规模杀伤性网络武器事实上是掌握在 NAS 手里，因为全球 PC 互联网基本上都在 Windows 操作系统的覆盖之下，手握大量漏洞攻击工具的 NAS 它如果要作恶，全球的互联网安全会是多么脆弱？

尤其是当前国内，从 PC 互联网到移动互联网均无自主操作系统的前提下，国内的互联网安全脆弱程度可以想象。

网络安全无小事，全球互联网的操作系统基本上就掌握在微软、谷歌、苹果手里，人们的信息、账号、隐私、安全等各种隐私越来越多开始转交给科技巨头保管，安全不能只停留在各巨头的口头吹嘘，它们更应关注信息浪潮中革命进程的安全变革与组织结构是否能够支撑捍卫无数用户的安全隐私的权利。

这无疑也体现出，黑客对于傲慢的互联网巨头的施加颜色，对于他们而言某种程度是一次警钟的敲响。

在此次病毒事件之前，早前 XcodeGhost 入侵苹果 iOS 事件也是对傲慢的苹果在系统级安全层面

的一次嘲弄，让人们看到苹果公司在操作系统研发功底与沉淀上技术实力并非牢不可破，一次黑客的攻击可以打击他们的傲慢，挽救它们可能因后续更大的漏洞造成更大的伤害。

这次事件是一个警醒，也提醒国内的互联网巨头甚至国家安全机构自主操作系统的重要性。有网友提到，如类似漏洞在战时被利用，破坏力无疑相当于在敌国首都扔一颗核弹。

众所周知，当前从桌面到移动端，所有的操作系统巨头都掌握在硅谷巨头手里，在一个日渐信息化的世界里，未来各国之间爆发网络战的可能性也并非为 0。然而，在操作系统受制于人的情况下，拥有全球性自主操作系统权限的一方若发动信息网络安全战，其攻击力与杀伤力事实上是远大于传统战争。

国产操作系统起步比较晚，而且建立一个完整的软硬件融合的生态系统并不是一朝一夕之功，周期太长，设计、研发、资源、技术上要投入大量的人力，况且一个新的操作系统没有应用软件生态来支撑，就等同于一个空壳，是没有价值与意义的。

但有人也提到，国内做操作系统，人们更容易想到的是骗取补贴以及有更多的理由去开各种后门。但总的来说，这次病毒事件可能是一次警醒，对于用户来说，定时备份重要文件与资料或者使用云存储服务存储资料的重要性。

同时也提醒国内互联网操作系统方面的政策顶层设计层面要有长远战略规划，尤其是要关注国家重要机构、公共安全、医疗机构与机密单位的网络安全与自主操作系统的建设与推进，也需要在顶层政策方面整合各方资源，为国产操作系统做些准备与考虑，避免在特殊极端情况下受制于人。d

更多精彩内容，关注钛媒体微信号 (ID: taimeiti)，或者下载钛媒体 App



武汉光谷国产 Deepin 操作系统完美防御勒索病毒

◎ 湖北日报

一场史无前例的电脑病毒袭击正在席卷全球。5月12日起，勒索病毒 WannaCry（“想哭”）在 100 多个国家和地区爆发，国内高校校园网、政府部门、银行、中石油加油站等网络均遭受不同程度的感染。

据了解，本次大规模流行的“想哭”病毒，大多通过邮件传播，电脑用户往往会收到一封伪装成招聘信息或购物清单的电子邮件，一旦打开相关链接或附件就会中招。

被感染的电脑会在 10 秒内锁住，所有文件被全部加密，只有按照弹窗提示交纳一定数量的比特币作为赎金才能解锁。超过期限，黑客就会“撕票”销毁数据。

根据国家网信办数据，截至 5 月 14 日 10 时 30 分，国家互联网应急中心已监测到约 242.3 万个 IP 地址遭受该漏洞攻击，另外监测到 5471 个 IP 可能已感染该病毒，主要分布在广东、浙江、北京和上海等地。在武汉，武汉工程大学、中石油湖北销售分公司也受到波及。

5 月 15 日，中央网信办网络安全协调局负责人回应称，勒索病毒传播速度已明显放缓。

然而，病毒扩散虽已得到有效遏制，但不少电脑专家提醒仍不能掉以轻心。欧盟刑警组织认为，黑客组织已经开发出升级版病毒，或于今后数日发

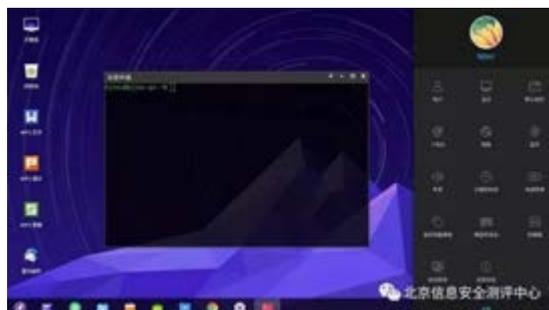
动新一轮攻击。

值得注意的是，此轮病毒肆虐最严重的，正是目前最为主流、装机量最大的 windows 系统。

网络安全专家指出，windows 系统由于自身存在很多未知安全漏洞，或已被“黑客”组织利用，用户只能通过被动安装安全防护软件，升级安全补丁保障安全。但是，由于病毒“变异”很快，面对勒索软件，侥幸躲过这一次，难免遇到下一次。

此外，由于业界暂未能有效破除该勒索软件的恶意加密行为，用户主机一旦被勒索软件渗透，只能通过重装操作系统或使用专杀工具的方式来清除勒索软件，但若用户重要数据文件没有备份，则很难完全直接恢复。

对于苹果用户而言，目前，Mac OS 系统能够躲避绝大多数针对 Windows 系统的病毒、恶意软件，不过，该系统对硬件成本要求很高，其封闭的生态保障了其安全系数，但开放性也成为其短板。



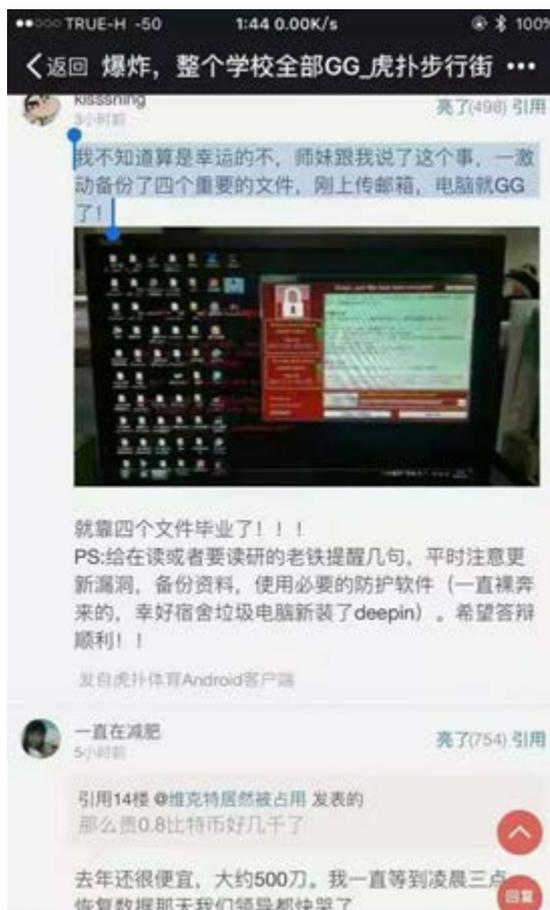
在本轮病毒袭击中，装机量并不占优的 Linux 系统 在安全、成本等方面表现出领先的综合优势。一组来自武汉深之度科技有限公司的数据显示，该公司基于 Linux 研发的 Deepin 操作系统在此次袭击中，零感染率。

深度科技网络安全专家张磊表示，对个人用户来说，使用 Deepin 操作系统，可以做到与 Mac OS 一样对 Windows 病毒、恶意软件免疫，不会被流行的各种病毒和木马感染。对于政府和企业用户，由于源代码开放可控，绝无后门隐患，安全性也更有保障，大大降低了维护代价和购买防护软件的费用。

数据显示，截止到 2016 年，深度操作系统下载超过 5000 万次，提供 30 种不同的语言版本，以及遍布 6 大洲的 70 多个镜像站点的升级服务。在全球开源操作系统排行榜上，深度操作系统已位居前 10 名，也是排名最高的中国操作系统产品。

深度科技作为国产操作系统生态的打造者，不但与各芯片、整机、中间件、数据库等厂商结成了紧密合作关系，还与 360、金山、网易、搜狗等企业联合开发了多款符合中国用户需求的应用软件。深度科技的操作系统产品，已通过了公安部安全操作系统认证、工信部国产操作系统适配认证、入围国管局中央集中采购名录，并在国内党政军、金融、运营商、教育等客户中得到了广泛应用。

深度操作系统让 linux 不再只是程序员的玩具，用户无需关心晦涩的命令行，通过全新的控制中心设计和交互，首页展示快速入口、常用快捷设置等，让操作更加方便和快捷。集成全新设计的安装器，让安装的过程也变成一种享受。自主开发的深度商



店持续集成丰富的应用软件，全面满足用户的工作和娱乐需求。

深度科技负责人表示：“只有技术上的自主可控，才能做到真正的安全风险可控。”面对勒索病毒 WannaCry，用户完全可以选择能够替代 Windows 的国产操作系统软件 Deepin OS。 [d](#)



深度原生 最美和声

两个或两个以上不同的音符按一定的法则同时发声而构成的音响组合称为和声，不同的和声经过感性的组合后形成音乐。

无论是气势恢宏的音乐，还是悠扬婉转的音乐，亦或是空灵震神的音乐，都是热情洋溢的自由艺术，像风、像四季、像海洋，像自然那样无边无际，又像自然生态那样生生不息。

一系列深度原生应用就像一段段和声，与深度操作系统一起，组成一首来自开源生态、来自极客文化，和源自用户心声的音乐。

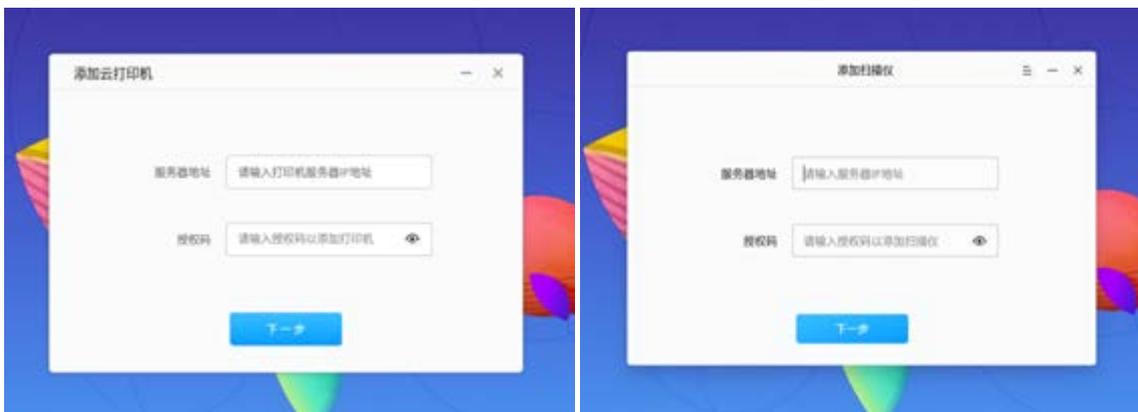


从深度截图、深度音乐、深度影音和深度商店开始，我们只是想为操作系统添加一抹弦乐和管乐的丝竹之音。伴随着用户的鼓励和鞭策，以及我们对真实需求的不断探索，丝竹之音已经无法满足用户对于我们的期许。所以，深度原生应用开始在不知不觉中深入到满足用户日常使用的方方面面，诞生了深度云扫描、深度云打印、深度远程协助等更多原生应用。毕竟匏土革木石金，搭配丝与竹，才是完整的八音。



深度云扫描 / 云打印解决方案以国产操作系统为基础，整合国内外打印和扫描设备资源，构建深度操作系统共享打印和扫描平台，向用户提供随时随地的打印和扫描服务。深度云扫描 / 云打印有效解决了打印机和扫描仪厂商对国产操作系统的驱动支持匮乏，并导致国产软硬件平台无法使用国内外打印机和扫描仪的局面。同时，深度云扫描 / 云打印适用于台式机、笔记本、部分平板电脑以及其它所有已授权的设备。

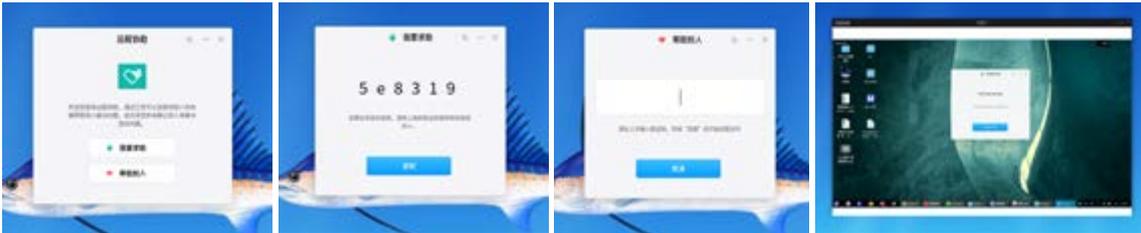
部署深度操作系统桌面版的客户端无需安装打印机和扫描仪驱动程序，只需要通过网络连接方式访问深度云扫描 / 云打印服务，即可使用远程打印机扫描仪。





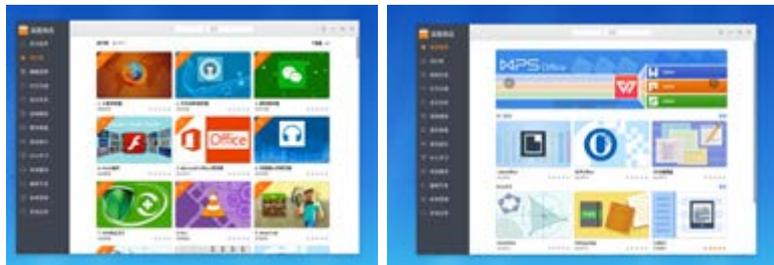
深度远程协助

远程协助是深度科技及时响应用户的求助，帮助用户和技术人员解决各种问题，同时也能够对他人发起求助的一款应用。深度团队已经将远程协助整合在操作系统中，不需要您关注服务端和客户端，无论您身在何处，只要有互联网就能够获得或提供远程帮助。远程协助操作简洁之余，通过六位数字与字母随即组合而生成的授权码保障用户安全，防止出现未经授权的远程连接。并且能够根据不同的网络质量选择画质，从而在绝大多数网络环境中得到良好的表现。



深度商店

深度商店是将软件安装和卸载等单一功能与社区相结合，集应用展示、下载、安装、评论、评分于一体的平台软件。为用户精心筛选和收录了不同类别的应用，每款应用都经过人工安装验证，保障用户可以通过商店搜索热门应用，并一键下载自动安装。深度商店以极简、扁平化的设计风格，获得了更为简约、精致的外观。其中应用专栏与应用专题每月定期更新，为用户呈现最新、最热的精品应用。目前，深度商店已与社区全面打通，用户可在深度商店中使用社区账号进行登录。用户评分与应用评论功能进行合并，确保应用评论与评分信息的真实、可靠，在不同用户之间提供更多的应用软件参考信息。

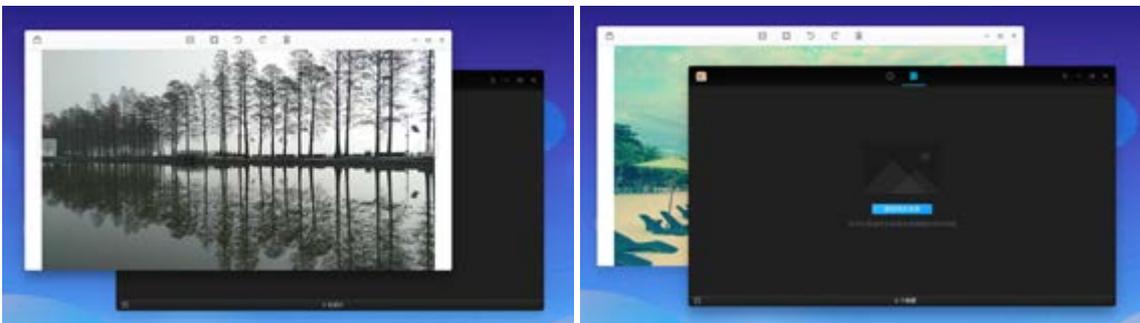




为解决 Linux 操作系统中截图软件匮乏，且使用复杂、功能单一的问题，深度团队发布了深度截图软件。深度截图通过选取方式的多种多样、工具栏的精心设计、编辑功能的全面优化和截图状态的智能记忆等特性，将截图这个说起来简单，用起来复杂的应用程序真正做到了简单易用，且功能全面。



深度看图是深度科技精心打造的一款图片查看和管理应用，外观时尚、性能流畅，支持多种图片格式。深度看图的图片管理功能，让您可以将图片按日期排列在时间线上，还可以将图片添加到收藏夹或归类到不同的相册中，将图片管理得井井有条。深度看图采用黑色系主题，不受边框色系的视觉影响，让用户更多专注其图片本身；多种图片浏览方式，全屏、旋转、幻灯片等不同的视角感受图片的美；一键收藏心爱的图片，快速定位图片位置，图片秒变桌面壁纸，导出成需要的图片格式；每一本相册记录成长和生活中的精彩瞬间，当你翻开相册时，总能勾起记忆中美好的时刻。





深度影院

深度影院提供了直观易用的操作界面和丰富齐全的快捷键，依靠键盘您就能完成所有的播放操作，让您彻底摆脱鼠标点击的束缚。您可以使用深度影院播放多种格式的视频文件，利用流媒体功能轻松欣赏到网络视频资源。得益于最新的 QML 技术，深度影院与前身深度影音相比，拥有了全新的外观和更多动态特效，搭配各种贴心功能（如进度条智能加粗、视频窗口自适应等），使其完美的接替了深度影音的使命——让用户更为专注视频所带来的快乐！



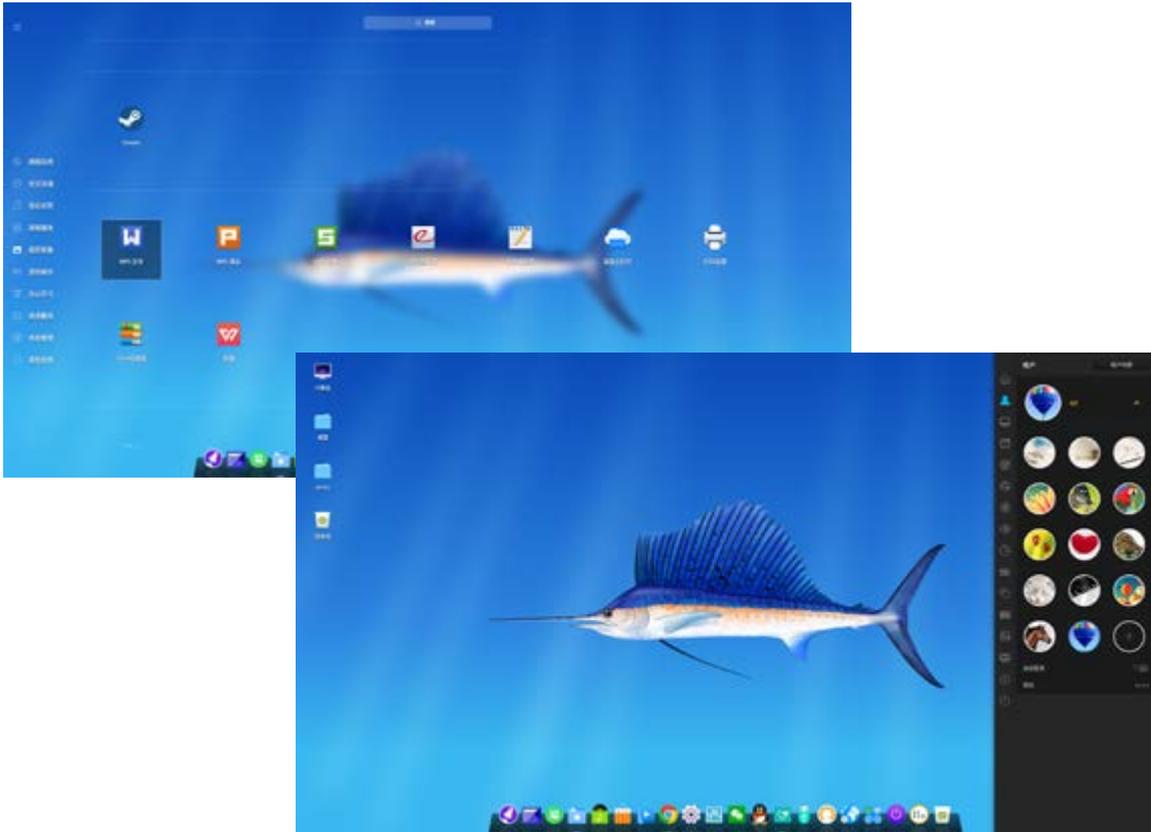
深度终端

深度终端是深度科技团队精心打造的一款终端模拟器。它拥有简单的界面，丰富而强大的功能。您可以瞬间启动和关闭，使用起来像普通文件窗口一样流畅，让您在输入命令时心无旁骛，得心应手。试试雷神模式，一边看页面一边输入命令；当有多任务进行时，您可以像在浏览器中增加页签一样建立多个工作区或者将工作区分割成多个窗口；远程管理，上传下载文件再也不是问题。更多强大功能，静待您的发掘！



深度桌面环境

作为中国唯一一个独立构建操作系统桌面环境的团队，深度团队开发的桌面环境已经与 Redhat 领导的 Gnome、Suse 领导的 KDE 和 Ubuntu 领导的 Unity 一起，成为全球范围内主流的操作系统桌面环境，并被开发者移植到其他更多的操作系统中。深度桌面环境追求的美观易用、极简操作的实用体验，主要由桌面、启动器、任务栏、控制中心、窗口管理器等组成。它既能让您体验到丰富多彩的娱乐生活，也可以满足您的日常工作需要。



深度团队的产品设计和规划已经从软件安装、听音乐、看视频等单个需求，向满足用户办公、娱乐、学习、开发等的场景需求转变，最终将会形成一套完整的、闭环的应用生态。从音乐单曲的从无到有，到向用户上演一场场史诗级的音乐剧。 **d**

这系统，吓到我了



guodapeng / 2017-4-3 13:24

浏览: 2143 / 回复: 44

我是在头条上看见一篇文章后发现这个系统的。

安装了虚拟机，打开的时候感觉一种安卓手机的视感。

当我打开 shell 的时候。我 cao，这么漂亮，比 MacOS 的漂亮多了。我一下字喜欢上了这系统。

看了看应用，居然有 QQ 和 Steam，这.....有追求的系统。

请允许我灌个水，这系统，我用定了。

关于志愿者的事情，我没太了解只前，还是先围观。

PS：第一次发关于系统架构介绍的中文文档，如此细节，有情怀。

PS：这个终端有源码没？我想安装到 MacOS 上，时在是太漂亮了。



kingskill 发表于 2017-4-3 13:34:40 | 只看该作者

欢迎入坑。12.12 版开始入坑的老白菜向 lz 致敬~~

15.4 马上就要出了，不过建议还是虚拟机立先试一下。等过段时间稳定之后再考虑实盘安装。俺 15.3 的，目前系统已经被调教得很好了。



jany_wu 发表于 2017-4-3 15:27:53 | 只看该作者

没有最完美，只有更完美，坚持做下去，坚定支持您将 DEEPI 做得更好，更强大！



xy_god 发表于 2017-4-4 08:58:27 | 只看该作者

deepin 终端的确是极品来的



hiboxdy 发表于 2017-4-5 14:43:54 | 只看该作者

一直跟踪着深度，期待能替代我的 Mac 和 Windows，我是 3 个系统都在用，看着深度在不断进步



lrbrfx 发表于 2017-4-7 09:43:11 | 只看该作者

欢迎入坑，虽然 deepin 还有很多让人吐槽的地方，谁用谁知道吖，不过，deepin 确实挺漂亮。



linuxbaby 发表于 2017-4-21 18:52:41 | 只看该作者

我用了四年了，还是一如既往地，从未都是支持，加油

加油！！deepin 进排行榜前3位不是问题（谦虚）



luoyeqingzhou / 2017-1-2 23:01

浏览: 649 / 回复: 13

截止 2017 年 03 月 31 日 21:54:25 <http://distrowatch.com> 的排名

名次	发行	HPD*
1	Mint	2682▼
2	Manjaro	2181▲
3	Debian	1792▼
4	Ubuntu	1598▼
5	deepin	1553▲
6	openSUSE	1075▼
7	Antergos	981▼
8	Fedora	960▼
9	Zorin	858▼
10	Plamo	793▲
11	elementary	712▼

我看好你哟



mjchow 发表于 2017-3-31 22:11:04 | 只看该作者

这就厉害了



zwq133 发表于 2017-3-31 22:46:17 | 只看该作者

哟！一个不小心上到第五位了



wtz 发表于 2017-3-31 23:14:33 | 只看该作者

有希望。

想当年，Manjaro 还在 10 名开外。



yiisaa 发表于 2017-3-31 23:32:26 | 只看该作者

厉害了我的深度



lupin 发表于 2017-4-2 02:29:36 来自移动端 | 只看该作者

感谢 deepin 团队

热烈祝贺 15.4 系统国人的世界骄傲



qwert123yuiop / 2017-4-19 18:47
浏览: 402 / 回复: 10

我从 14 版 - 15.4 版一直单系统，几年没有用 win 系统了，我每天早中晚看 deepin 论坛，我自认为现在
是世界 3 大系统之一了。苹果、微软、深度，加油巅峰是中国的。



Hiseico 发表于 2017-4-19 19:31:38 | 只看该作者

祝贺 deepin 在系统 UI 上面又迈进了一大步



ychost 发表于 2017-4-19 19:55:19 | 只看该作者

不得不说，深度做的真的很好



rekols 发表于 2017-4-19 19:56:57 来自移动端 | 只看该作者

jingle 在被酱油哥的威胁下终于发布了 15.4



llyy007 发表于 2017-4-19 21:16:41 | 只看该作者

普天同庆，国人的骄傲！深度，不忘初心！

对于 Deepin 15.4, 我想说



lingw168 / 2017-4-12 17:12

浏览: 1222 / 回复: 16

之前也有体验过 linux 的其他发行版本, 基本用 1 天就删除掉了, 从 IT 新闻上看到 Deepin 15.4 的介绍, 然后到官网下载安装, 体验和使用一天后感觉完全出乎我的预期。5.4RC2 的界面非常漂亮, 在普通办公的实用性上要比其他 linux 发行版本好得多, 基本能完全应付我日常的工作 (我是搞业务的), 我自己办公用的两台电脑和家里面的电脑都安装了 Deepin, 感觉完全可以长期使用 (QQ、google Chrome、Deepin 软件等太棒了), 今天特地注册了论坛账号, 想发声感谢 Deepin 开发人员们的努力, 让我们体验到了这么好的一款产品! 接下来我会给其它同事推广, 让他们也使用! 也希望应用商城有更多优秀的软件加入!



doudou586 发表于 2017-4-12 17:19:02 | 只看该作者



看来会是一个长期用户



winter110 发表于 2017-4-12 17:22:14 | 只看该作者

欢迎



jingle 官方管理员 发表于 2017-4-12 17:28:35 | 只看该作者

谢谢对我们的支持, 有好的应用欢迎推荐



versee 发表于 2017-4-13 09:39:33 | 只看该作者

单奔学生党路过

大爱 Deepin! 我的 chromebook C720 终于能体验 15.4



howgor / 2017-5-16 20:56
浏览: 529 / 回复: 15

可能我是菜鸟的原因，我的 chromebook acer c720 2G 内存版本，一直无法安装 64 位的 Deepin 15.4。只能给它装 15.3 用。

然而 15.3 在这台机器上体验不太好，经常卡（当然，这不能怪 Deepin，毕竟配置在那儿）。

没想到，今天给它升级了一下，竟然升级到了 15.4 桌面。感动！！！！

到了 15.4 之后，之前蛋疼的触控板流畅了，和 Chrome OS 下体验无差异。系统也更流畅了。

Deepin, I love you! 祝 Deepin 越来越好，一个用心做人性化操作系统的团队，一定成事儿！就像老罗的 Smartisan OS，你们都能成事儿。



yimshingchan 发表于 2017-5-16 21:58:27 | 只看该作者

同 chromebook



luoyeqingzhou 发表于 2017-5-17 00:02:01 | 只看该作者

打消了我装黑 苹果的念头，真的挺漂亮的

每天来坛子里看看，已成了习惯。。。。



superbingcn / 2017-5-27 10:46
浏览: 321 / 回复: 26

看看大家交流啥问题，互相学习一下，或者有更好的发现。。。



zfskun 发表于 4 天前 | 只看该作者

每天必来！



tsugumi 发表于 4 天前 | 只看该作者

一天不逛论坛就不舒服



myfsbch 发表于 4 天前 | 只看该作者

习惯就好



moonlimb 发表于 4 天前 | 只看该作者

我也是，习惯了，每天打开电脑，就必来。



huaifeng 发表于 4 天前 | 只看该作者

看看有没有新发现吗，话说搜狗输入法越来越好用的感觉，还有 qq 也不卡了也不崩溃了



Guumi 发表于 4 天前 | 只看该作者

我也是滴，只要有条件有时间开机第一件事毕竟是先上 deepin.org，那怕登录后挂着也要上



fengying6_6 发表于 4 天前 | 只看该作者

以前习惯性打开百度检查网络是否通畅，现在习惯性打开深度论坛



junzipeng 发表于 4 天前 | 只看该作者

每天都来看看，虽然有时候不说话。



重庆市渝中区文化委 桌面操作系统国产化试点

● 深度科技 重庆办事处/文

国产化，是一个沉重的话题，操作系统国产化，特别是桌面操作系统国产化，那无疑是一个“深痛”的话题。有人说过：如果 Linux 桌面与 Windows 桌面按使用量进行柱状图统计对比，那么 Linux 桌面会处在几乎看不见的状态。

市场开拓时，客户听见国产操作系统就纷纷摇头，说之前响应国产化号召有买过，但是完全用不了，并且有问题也没有人解决。所以在与渝中区文化委的初期沟通中，客户的首要期望是能真正的使用起来，而且能提供必要的技术支持服务。而这亦正是我们的期望。值得欣慰的是，客户的相关负责人明白国产化不可能一蹴而就，所以愿意给我们时间进行测试和适配。于是，就开始了我的跌跌撞撞的项目推进之路。

第一次的测试在周末，刚巧遇上另外一个项目的首次实施，所以公司总部安排了乔老师到重庆进行技术支持。这次测试的电脑只有一台，就是客户相关负责人的电脑，测试的状态还算顺利，尽管中间出现了很多操作系统以外的问题，比如不支持硬盘隔离卡等等，但我们都逐一的进行了解决或采用替代方式进行了解决。相对于另外一家国产操作系统兄弟厂商之前连安装都没有安装上，客户对我们的测试结果还算满意。

乔老师回去了，我的测试之路也开始了，区域仅我一人（其实重庆还有一个大神，但是是负责内核维护的，属于核武器般的存在，打蚊子的小事情总不能放个原子弹吧？），作为一个技术小白，我开始了跌跌撞撞的挣扎。

一个多月后，客户打电话说前段时间太忙了，你马上再过来给我们委办分管领导装个系统吧，让他体验一下，如果他认可，基本上就可以走采购流程了，同时我再把其它需要正常使用的外设再给你说一下。当天晚上，我把 U 盘安装工具又检查了一下，确认工作正常无误，第二天一早就在客户的带领下到了委领导的办公室。递了名片，简单的汇报





了一下，就开始干活。委领导还顺便问了一句，你在公司是从事销售工作的吧，为什么还来装电脑？咳，领导，这不是咱们刚起步，人手不足嘛，不过这绝对不会影响服务质量，同时，装系统很简单，我自己用的电脑都是自己装的，很快就好了。

海口是夸下了，没想到刚开始就遇到了麻烦，插上了装机 U 盘，选择 U 盘启动却不能进入安装界面，试了几次都不行，仔细检查了一下，发现了端倪；这台 HP Pro Desk 还划了 UEFI 安全分区并做了相应的设置，赶紧进入 bios 禁止 UEFI 启动，再用分区工具把 UEFI 分区删除并将硬盘重新划分了分区以绝后患。重新启动后正常，顺利的装上了操作系统。但新的问题出现了，以前 Windows 的时候，这台电脑单网卡通过一个网线转接盒实现内外网的切换上网，也就是说 1 张网卡上得配上两套不同网段的 IP、网关和 DNS；修改网络配置文件后，内网的 DNS 根本不生效，其区政府 OA 完全无法访问。这可是会影响办公的，怎么办？还有这个打印机驱动装好之后怎么无法打印呢，怎么办？好吧，继续远程求助乔老师，大神出马，一切顺利解决，欣欣然收工回家。

第二天一早，领导又打电话来了，昨天的有些图标不见，桌面的下菜单栏不见了，还有昨天你给我装的那个网易云音乐我觉得不好，装的那个酷狗安卓版没法用啊，另外我们办公的有个网站窗口弹不出来了……好吧，2 小时内现场响应。去了之后，正好客户负责人也在，说顺便再把后面要用的外设给我讲讲，先做一下准备。于是好几台打印机，扫描仪进入了我的手机相册，老款的打印机好几台，其中一台后面的标签上妥妥的标着“september 2006”，另外还有一台打票据用的老款的针式打印机……工程部的同事有得忙了。

就这样不断的折腾和挣扎，采购后，到下面科室实施时还遇到了各种各样的问题，总之在公司同事的大力支持下都一一得到了解决或采取了替代方案进行了处理，客户很满意，特别是服务态度，磨难亦是成长！

经常有熟人和客户问我我觉得现在的工作怎么样？我常答：做咱们中国人自己的操作系统，你不觉得很有意义吗？ d



Linux 全局事件监听技术

● 王勇/文

应用场景

开发应用程序的过程本质就是通过图形库获得用户的输入事件（鼠标、键盘或者触摸屏等）和数据以后，对这些用户的事件和数据进行处理后，通过界面或其他交互形式展现给用户结果。

应用程序完成后，拥有美观的界面和简洁易用的使用逻辑，让用户在使用过程中感到舒服和爽快，这样的应用程序我们就可以称为交互体验优秀的产品。

一般来说，应用程序窗口的所有事件都可以通过图形库 (Gtk+、Qt 等) 自己来获取的，但是有时候我们需要一种技术来获取整个操作系统的事件，来满足以下场景：

- 监听用户输入的鼠标事件，比如屏幕取词
- 监听用户输入的键盘事件，比如全局快捷键

这时候 Gtk + 和 Qt 就无法做到，需要 X11 相关的技术才能做到对系统的事件进行监听。

X11 相关的技术有两种方案：

- 通过 XGrabPointer 和 XGrabKeyboard 抓取系统的焦点后监听全局事件
- 通过 XRecord Extension 非侵入式的监听全局事件

XGrabPointer 和 XGrabKeyboard 一般主要用于菜单的实现，而且这种方法必须要抢占用户的鼠标或键盘焦点，导致一旦焦点被抢占时，别的程序就无法正常使用（比如菜单弹出时，其他程序就无法输入字符或响应鼠标事件了）。

大部分应用程序监听事件时往往并不需要抢占系统的事件焦点，希望在监听事件的时候用户

可以正常操作系统。所以，今天讲解一下怎么用 XRecord 这个 X11 的扩展库来进行鼠标事件以及键盘事件的监听。

技术原理

X11 是 Linux 下最古老和通用的技术，不论用户的输入事件还是最后画到屏幕的绘制动作其实都是 XServer 来实现的。

Linux 下所有图形应用的底层消息都是按照下面的顺序来执行的：

硬件产生事件→XServer 发送输入事件给图形库→图形库 (X Client) 包装输入事件传递给应用程序→应用根据输入事件产生绘制命令→图形库 (X Client) 根据应用绘制命令产生绘制消息→XServer 接受绘制消息→绘制图形到屏幕上。

上面顺序中的 X Client 就是我们通常说的 Gtk+、Qt 这些图形库，通过 xcb/xlib 和 XServer 进行输入输出通讯，保证输入事件和输出绘制都可以及时响应，同时图形开发库提供高级的 API 封装，让开发的同学不用直接编写复杂的 Xcb/Xlib 通讯代码和参数细节。

而 XRecord 就是一个 XServer 端的扩展，你可以想象 XRecord 就像一条寄生虫寄生到 XServer 里面，只要 XServer 从硬件那里接收到所有输入事件都会告诉一下 XRecord，我们只需把对应的代码挂到 XRecord 循环中，只系统一有输入事件产生，XServer 就会告诉 XRecord，XRecord 接着就通过事件循环告诉我们写的应用程序，应用程序再利用实时截获到的输入事件进行处理。

这一切都发生的悄无声息，既监听了系统上所有的输入事件又不会影响系统中的任何应用，是不是听着很邪恶？（刀能切菜也能伤害别人，千万不要做坏事哟）

代码讲解

输入事件监听的核心代码都在 `event_monitor.cpp` 中，下面我一个一个函数的讲解：

```
// 因为 XRecord 的事件循环会阻塞当前线程，避免监听事件的时候应用程序卡位
// 我们建立一个继承于 QThread 的 EventMonitor 类，通过线程进行事件监听操作
EventMonitor::EventMonitor(QObject *parent) :
    QThread(parent)
{
    // 鼠标按下标志位，用于识别鼠标的拖拽操作
    isPress = false;
}
```

```
void EventMonitor::run()
{
    // 创建记录 XRecord 协议的 X 专用连接
    Display* display = XOpenDisplay(0);

    // 连接打开检查
    if (display == 0) {
        fprintf(stderr, "unable to open display\n");
        return;
    }

    // 初始化 XRecordCreateContext 所需的 XRecordClientSpec 参数
    // XRecordAllClients 的意思是 "记录所有 X Client" 的事件
    XRecordClientSpec clients = XRecordAllClients;

    // 创建 XRecordRange 变量，XRecordRange 用于控制记录事件的范围
    XRecordRange* range = XRecordAllocRange();

    // 记录事件范围检查
    if (range == 0) {
        fprintf(stderr, "unable to allocate XRecordRange\n");
        return;
    }

    // 初始化记录事件范围，范围开头设置成 KeyPress，范围结尾设置成 MotionNotify 后
    // 事件的类型就包括 KeyPress、KeyRelease、ButtonPress、ButtonRelease、MotionNotify 五种事件
    memset(range, 0, sizeof(XRecordRange));
    range->device_events.first = KeyPress;
    range->device_events.last = MotionNotify;
}
```

```
// 根据上面记录的客户端类型和记录事件范围来创建 "记录上下文"
// 然后把 XRecordContext 传递给 XRecordEnableContext 函数来开启事件记录循环
XRecordContext context = XRecordCreateContext (display, 0, &clients, 1, &range, 1);
if (context == 0) {
    fprintf(stderr, "XRecordCreateContext failed\n");
    return;
}

// 释放 range 指针
XFree(range);

// XSync 的作用就是把上面的 X 代码立即发给 X Server
// 这样 X Server 接受到事件以后会立即发送给 XRecord 的 Client 连接
XSync(display, True);

// 建立一个专门读取 XRecord 协议数据的 X 链接
Display* display_dataLink = XOpenDisplay(0);

// 连接打开检查
if (display_dataLink == 0) {
    fprintf(stderr, "unable to open second display\n");
    return;
}

// 调用 XRecordEnableContext 函数建立 XRecord 上下文
// XRecordEnableContext 函数一旦调用就开始进入堵塞时的事件循环，直到线程或所属进程结束
// X Server 事件一旦发生就传递给事件处理回调函数
if (!XRecordEnableContext(display_dataLink, context, callback, (XPointer) this)) {
    fprintf(stderr, "XRecordEnableContext() failed\n");
    return;
}
}
```

```
// handleRecordEvent 函数的 wrapper，避免 XRecord 代码编译不过的问题
void EventMonitor::callback(XPointer ptr, XRecordInterceptData* data)
{
    ((EventMonitor *) ptr)->handleRecordEvent(data);
}

// 真实处理 X 事件监听的回调函数
void EventMonitor::handleRecordEvent(XRecordInterceptData* data)
{
    if (data->category == XRecordFromServer) {
        // 得到 xEvent 对象
        xEvent * event = (xEvent *)data->data;
        switch (event->u.u.type) {
            case ButtonPress:
                // 过滤掉滚轮事件后，发送 buttonPress 信号
                if (filterWheelEvent(event->u.u.detail)) {
                    isPress = true;
                    emit buttonPress(

```



```
        event->u.keyButtonPointer.rootX,
        event->u.keyButtonPointer.rootY);
    }

    break;
case MotionNotify:
    // 只有在按下鼠标的时候移动, 才发送 buttonDrag 信号
    if (isPress) {
        emit buttonDrag(
            event->u.keyButtonPointer.rootX,
            event->u.keyButtonPointer.rootY);
    }

    break;
case ButtonRelease:
    // 过滤掉滚轮事件后, 发送 buttonRelease 信号
    if (filterWheelEvent(event->u.u.detail)) {
        isPress = false;
        emit buttonRelease(
            event->u.keyButtonPointer.rootX,
            event->u.keyButtonPointer.rootY);
    }

    break;
case KeyPress:
    // 发送 keyPress 信号, 附带按键的 code
    emit keyPress(((unsigned char*) data->data)[1]);

    break;
case KeyRelease:
    // 发送 keyRelease 信号, 附带按键的 code
    emit keyRelease(((unsigned char*) data->data)[1]);

    break;
default:
    break;
}

// 资源释放
fflush(stdout);
XRecordFreeData(data);
}

// 过滤滚轮事件
bool EventMonitor::filterWheelEvent(int detail)
{
    return detail != WheelUp && detail != WheelDown && detail
    != WheelLeft && detail != WheelRight;
}
```

```
mkdir build
cd build
qmake ..
make
./xrecord-example
```

编译完成以后, 会弹出一个 Qt 窗口, 可以实时查看鼠标和键盘的事件信息, 大家可以基于上面的代码进行改造, 以融合到自己的项目中。



我对开发者学习一项新技术的建议是:

先拷贝现有代码→精简提炼出核心代码→融合到自己的项目中, 先会用→用的熟练以后再研究 API 和每一个参数细节→最后查看底层库源代码

只有先实践才能真正理解开源项目的原作者为什么这么写, 最后才能真正吸收这些技术, 做好开源贡献。d

代码下载

可编译的代码请在 <https://github.com/WHLUG/xrecord-example> 下载后, 执行下面的命令来测试:

deepin-wm 的动态背景模糊实现

● 北京 研发部/文

这篇文档的目的是对深度窗口管理器动态背景模糊实现技术做一个大致的描述。

简介

kwin 从很早开始就支持类似 windows 窗口的菜单栏毛玻璃的效果。其实现机制在 kwin 良好的插件架构下相对容易，然而要在基于 mutter 的窗口管理器上实现同样的效果却不容易。这主要在于 mutter 依赖的技术之一 clutter 的设计思路跟 kwin 完全不同。两者的共同点都是基于 OpenGL 实现（当然，kwin 有多个后端，可以不依赖 OpenGL，但是其他后端不能开启很多依赖硬件支持的特效）。kwin 对 OpenGL 没有做过度封装，因此可以直接使用 OpenGL 的特性。而 clutter 依赖 cogl 这个 OpenGL 封装库，为了提供最大的兼容性，cogl 暴露的功能相对有限，尤其是对于需要 OpenGL 扩展的情况。

kwin 的特效是全局的，在每一帧绘制中，所有激活的特效按顺序对当前的 draw framebuffer 内容进行处理，因此 kwin 里大量的上下文信息对于所有特效来说是共享的，并且容易访问。而 clutter 的设计是以 Actor 为单元来进行处理的。每个窗口，每个可视元素对应一个 Actor，因而有针对 Actor 的 Effects（等价于窗口特效），而且 Effect 只能操作 Actor 的内容，并没有一个机制来对全局场景进行处理。

为了实现标题栏或窗口背景等的动态毛玻璃效果，必须对 framebuffer 进行两次渲染。首先是渲染

所有不带背景模糊的对象（Actor），然后是渲染有背景模糊的对象的模糊部分，从 framebuffer 中截取背景部分的内容作为材质（texture）进行模糊运算，然后再把结果贴回 framebuffer 被截取的区域作为对象的背景，再与对象进行 alpha 混合（没有透明度的窗口自然不能做背景模糊）。

分层实现

因此为了能支持动态背景模糊，具体下来要做几件事情：

- 首先修改 cogl，让其支持 glCopyTexSubImage 2D 这是最主要的基础功能。可以从当前的绘图 buffer 中取出已经绘制的内容来作为材质，以便进行模糊处理，再进行后续的混成。
- 其次，修改 clutter。让其保证那些特殊的带有模糊背景材质的 Actor 在每帧都能被绘制。
- 最后就可以在 mutter 里根据上面的基础设施实现 MetaBlurActor。
- 定义 `_NET_WM_DEEPIN_BLUR_REGION_MASK` 属性，所有有此属性的透明窗口都能自动实现动态背景模糊。

下面具体描述一下。修改 cogl 很直观，没什么好说的。至于为什么要修改 clutter 要做一个说明。clutter 的绘制模型大致是这样的：首先，clutter 里所有的 actor 跟 X11 一样是一个单根的树形结构。顶层的 actor 叫 stage，每个 actor 都可以有任意数



量的子 actor。在每一帧绘制的开始，stage 搜集所有需要重绘的 actors（一般通过 `queue_redraw(_with_clip)` 来通知）。stage 根据这些 actor 查找所有路径上的父节点的 actor（即子 actor 重绘会引起父 actor 的重绘），加入待重绘列表。然后根据 clip 等信息确定列表里真实需要绘制的 actors，并调用 `clutter_actor_paint` 自顶向下绘制这些 actor。这里面有很多优化细节，但是对于这里的讨论没有太大影响。总的来说，在每一帧，只有自己请求了重绘的那些 actor 才会发生重绘。

我设计并实现了 `MetaBlurActor`，当我们需要对一个代表窗口的 actor 做动态背景模糊时，将此 `MetaBlurActor` 作为需要动态模糊背景的 actor 的子 actor，这样保证每次 actor 需要 redraw 时，clutter 也会同时重绘 `MetaBlurActor`。在 actor 发生移动和 resize 时也能按需更新。`MetaBlurActor` 本身实现了使用 `glCopyTexSubImage2D` 取得这个 actor 背后区域的内容，生成材质，进行模糊处理的功能。

现在的问题是作为动态背景，只要 actor 后面的任何内容有变化 `MetaBlurActor` 都应该发生重绘，而不仅仅是 actor 本身请求重绘时才重绘。但是 `MetaBlurActor` 本身并没有明显的方式确定是否需要重绘，这是由 clutter 的绘制机制决定的。一种可能的实现方式是 `MetaBlurActor` 自己监视每个在其后面的 actor 的 `damage`（X11 有 window 的 `damage` 机制，但是 actor 并没有），但是当有大量 `MetaBlurActor` 时，这种方式估计开销很大，而且 clutter 也没有机制方便跟踪 actor 之间的层次变化（类似 X11 的 `window stacking change` 事件）。因此目前修改了 clutter 加入了一种能快速实现的方式，保证所有的 `MetaBlurActor` 在每帧都能得到绘制。当然目前这种方式显然在性能方面还不完善，后面还会进行优化。

比如通过全局追踪每帧 `damage` 的 region，来判断和裁剪 `MetaBlurActor` 需要更新的材质区域，减少不必要的模糊运算。目前使用的模糊算法近似高斯模糊，对模糊的速度和效果有一个比较好的折中。

MetaBlurActor 的实现

上面已经简单说明了 `MetaBlurActor` 的基本功能：通过 `glCopyTexSubImage2D` 取得这个 actor 背后区域的内容，生成材质，对材质进行离线模糊处理，然后重新绘制出来。因此当 `MetaBlurActor` 发生移动或改变大小时，都要实时的追踪背后的变化。模糊算法是近似高斯模糊，但是由于高斯模糊的速度问题，要快速的进行大半径的高斯模糊，需要使用一些技巧。目前通过 `downscale` 材质来实现（这也造成了一些问题）。`MetaBlurActor` 不仅要实现动态模糊，而且还需要支持异形窗口的背景模糊。窗口根据需求定义了 `_NET_WM_DEEPIN_BLUR_REGION_MASK`，这是一个 alpha 位图，可以定义任何形状，`deepin-wm` 根据这个属性生成一个 `shape mask` 来处理模糊的区域，以实现异形模糊窗口。比如：深度录屏的模糊效果，OSD 提示，深度通知，dock 等等。d

好奇的同学可以参考这里的实现：

<https://github.com/linuxdeepin/deepin-mutter/blob/release/3.20/src/compositor/meta-blur-actor.c>

RPM 包制作与 YUM 仓库基础

● 北京 技术部/文

制作 RPM 包

什么是 RPM 包

RPM 包源自于 Red Hat Linux 分发版，是 Linux 下常见的软件包格式之一，RPM 包有两种包格式：

- 扩展名为 .rpm 封装完成的 RPM 二进制安装包
- 扩展名为 .src.rpm 包含编译控制文件的 SRPM 源代码包

准备工作环境

构建 RPM 软件包需要做如下准备工作：

- 一是系统中安装好 rpmbuild 打包工具，执行命令：`yum install rpm-build -y`
- 二是编写一个扩展名为 .spec 文件，该文件指导 rpmbuild 命令如何构建和打包软件。这个文件可以任意地给它命名并把它放到任何地方，RPM 对此没有限制。

修改上游源代码包

本文作为基础篇，不深入讲解如何编写 spec 文件，我们可以通过获取上游 SRPM 包重新编译构建来体会 rpm 的过程。执行命令：`rpm -ivh http://mirrors.ustc.edu.cn/fedora/releases/22/Server/source/SRPMS/b/bc-1.06.95-13.fc22.src.rpm`

rpm 会把 srpm 包解压到 `~/rpmbuild/` 目录，其中：

- spec 文件 解压到 `~/rpmbuild/SPECS/` 目录中
- 补丁和源码解压到 `~/rpmbuild/SOURCES/` 目录中

重新编译源代码包

```
rpmbuild -ba ~/rpmbuild/SPECS/bc.spec
```

编译完成后，结果会存放在 `~/rpmbuild/SRPM/~/rpmbuild/RPM/` 目录中，在这里需要了解一下 rpm 的环境变量，查看 rpm 的环境变量 `rpm--showrc`，其中 `_topdir` 定义了工作目录位置，默认是 `$HOME/rpmbuild/`，该目录下有五个目录：

- SPECS 放置 .spec 文件
- SOURCES 放置套件的源码及补丁等
- BUILD 用于存放解后压合并布补丁的源码目录
- BUILDROOT 用于存放封装生成的 RPM 安装包的文件
- RPMS 放置二进制 RPM 安装包 (.rpm)
- SRPMS 放置源码格式的 RPM 包 (.src.rpm)

rpmbuild 工作流程

下面总结了在您运行 `rpm -ba filename.spec` 时，RPM 都做些什么：

- 读取并解析 filename.spec 文件
- 运行 %prep 部分来将源代码解包一个临时目录 (`~/rpmbuild/BUILD/XXXX`)，并应用所有的补丁程序
- 运行 %build 部分来编译代码
- 运行 %install 部分将代码安装到一个临时目录 (`~/rpmbuild/BUILDROOT/XXXX`)
- 读取 %files 部分的文件列表，收集文件并创建二进制和源 RPM 文件。
- 运行 %clean 部分清楚临时构建目录



创建 YUM 仓库

yum 主要用于自动安装、升级 rpm 软件包，它能自动查找并解决 rpm 包之间的依赖关系。使用 yum 就需要有添加一个包含各种 rpm 软件包的 repository (软件仓库)，这个软件仓库我们习惯称为 yum 源，下面我们就讲述如何创建自定义的软件仓库。

创建仓库

执行命令 `yum install createrepo -y`，安装一个名为 `createrepo` 的软件包，然后使用 `createrepo` 就可以完成 yum 仓库的创建，示例如下：`createrepo -g dvd-comps.xml -u Packages/ /repo`

- `-g dvd-comps.xml` 指定分组配置文件
- `-u Packages/` 使用 `Packages/` 这个 rpm 包存放目录
- `/repo` 仓库根目录，默认生成的索引文件存放在这个目录下

各个版本软件分组参考可以从这里获取：`git clone git://git.fedorahosted.org/git/comps.git`

更新仓库

在一个已创建好的 yum 仓库目录下，添加或删除 rpm 包后，使用 `--update` 参数就可以完成仓库的更新：`createrepo -g dvd-comps.xml -u Packages/ --update /repo`

YUM 仓库索引

在创建好的 yum 仓库目录下会创建 `repodata` 目录，里面存放 XML 格式或 sqlite 数据库的仓库索引文件，执行命令 `yum update`，就是在同步 yum 源的索引，下面是 `repodata` 索引部分的概述：

```

repomd.xml           描述的其他元数据文件的文件
primary.[xml/sqlite].gz 主要元数据信息文件，记录软件包报名，版本，预配置文件，依赖关系等
filelists.[xml/sqlite].gz 软件包文件，目录列表描述信息
other.[xml/sqlite].gz 目前只记录存储数据的变更记录
comps.xml.[gz]       用于记录软件包组分类等信息
(需要创建仓库的时候指定分组文件)

```

更多细节可参考文档 <http://createrepo.baseurl.org/>

添加软件源

以上文提到的创建好的 yum 的软件仓库为例，添加一个软件源。创建配置文件：`/etc/yum.repos.d/local.repo`

```

[local]
name=local
baseurl=file:///repo/
gpgcheck=0

```

重新执行命令 `yum update` 之后就可以使用这个软件仓库了。

软件源格式

yum 仓库配置文件扩展名是 `.repo`，配置文件存放目录：`/etc/yum.repos.d/`。一个 `repo` 文件可以添加多个 repository 配置，`repo` 文件中的 repository 配置遵循如下格式：

```

[serverid]
name=Some name for this server
baseurl=url://path/to/repository/
其他可选配置

```

- serverid 是用于区分不同的 repository，必须有一个独一无二的名称；
- name 定义 repository 的描述部分
- baseurl 定义 repository 的访问方式

baseurl 的格式

baseurl 指向是 repository 服务器设置中最重要的部分，一个 repository 配置中只能有一个 baseurl，只有设置正确，yum 才能从上面获取软件。它的格式是：

```
baseurl=url://server1/path/to/repository/
url://server2/path/to/repository/
url://server3/path/to/repository/
```

其中 url 支持的协议有 http:// ftp:// file:// 三种。baseurl 后可以跟多个 url。

打包辅助工具

实际开发过程中必须模拟用户的环境或是构建一个“干净”的环境（一个仅仅满足编译构建的最小系统环境）使用 mock 命令就达到在一个“干净”的环境重新编译构建。

- 安装软件包：yum install mock -y

YUM 仓库的配置

使用 mock 工具编译软件包，需要在 yum 仓库中建立了一个 sysbuild 分组，该分组包含了一个最小化系统所需的基础软件包。以下是 comps.xml 参考：

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE comps PUBLIC "-//CentOS//DTD Comps info//EN" "comps.dtd">
<comps>
<group>
<id>sysbuild</id>
<name>mock build base group</name>
<name xml:lang="zh_CN"> 自动编译基础系统 </name>
<name xml:lang="zh_TW"> 自动编译基础系统 </name>
<description>mock build base group</description>
```

```
<description xml:lang="zh_CN"> 自动编译基础系统 </description>
<description xml:lang="zh_TW"> 自动编译基础系统 </description>
<default>false</default>
<uservisible>>true</uservisible>
<packagelist>
<packagereq type="default">bash</packagereq>
<packagereq type="default">gawk</packagereq>
<packagereq type="default">rpm</packagereq>
<packagereq type="default">rpm-build</packagereq>
<packagereq type="default">bzip2</packagereq>
<packagereq type="default">gcc</packagereq>
<packagereq type="default">sed</packagereq>
<packagereq type="default">coreutils</packagereq>
<packagereq type="default">git</packagereq>
<packagereq type="default">deepin-release</packagereq>
<packagereq type="default">tar</packagereq>
<packagereq type="default">cpio</packagereq>
<packagereq type="default">gnupg2</packagereq>
<packagereq type="default">texinfo</packagereq>
<packagereq type="default">curl</packagereq>
<packagereq type="default">grep</packagereq>
<packagereq type="default">unzip</packagereq>
<packagereq type="default">diffutils</packagereq>
<packagereq type="default">gzip</packagereq>
<packagereq type="default">redhat-rpm-config</packagereq>
<packagereq type="default">util-linux-ng</packagereq>
<packagereq type="default">findutils</packagereq>
<packagereq type="default">make</packagereq>
<packagereq type="default">patch</packagereq>
<packagereq type="default">which</packagereq>
</packagelist>
</group>
<category>
<id>sysbuild</id>
<name>sysbuild</name>
<description>mock mini require</description>
<display_order>60</display_order>
<grouplist>
<groupid>sysbuild</groupid>
</grouplist>
</category>
</comps>
```

mock 工具的配置

根据需要修改默认配置 /etc/mock/default.cfg，



配置文件中 `config_opts['chroot_setup_cmd'] = 'install @sysbuild'` 分组名称 `sysbuild` 需要和仓库配置保持一致，参考配置如下：

```
config_opts['root'] = 'deepin-15-x86_64'
config_opts['target_arch'] = 'x86_64'
config_opts['legal_host_arches'] = ('x86_64',)
config_opts['chroot_setup_cmd'] = 'install @sysbuild'
config_opts['dist'] = 'deepin15' # only useful for
--resultdir variable subst
config_opts['yum.conf'] = """
[main]
keepcache=1
debuglevel=2
reposdir=/dev/null
logfile=/var/log/yum.log
retries=20
obsoletes=1
gpgcheck=0
assumeyes=1
syslog_ident=mock
syslog_device=
mdpolicy=group:primary
best=1

# repos
[deepin]
name=deepin server enterprise linux repo - amd64
baseurl=http://10.1.10.21/server-dev/dsee-15-amd64/
main/
enabled=1
priority=1
gpgcheck=0
"""
```

mock 的基本使用

一切准备就绪，开始使用 `mock` 重新编译一个软件包。

```
mock --init
mock --rebuild pkg.src.rpm
```

小技巧：使用 `mock` 编译辅助脚本，并行编译软件包

```
#! /bin/bash
pkg=$1

cat > /etc/mock/$1.cfg <<EOF
config_opts['dist'] = 'deepin15'
config_opts['root'] = '$pkg'
config_opts['target_arch'] = 'x86_64'
config_opts['legal_host_arches'] = ('x86_64',)
```

```
config_opts['chroot_setup_cmd'] = 'install @buildsys-
build'
config_opts['yum.conf'] = """
[main]
keepcache=1
debuglevel=2
reposdir=/dev/null
logfile=/var/log/yum.log
retries=20
obsoletes=1
gpgcheck=0
assumeyes=1
syslog_ident=mock
syslog_device=
mdpolicy=group:primary
best=1

[deepin]
name=deepin server enterprise linux repo - amd64
baseurl=http://10.1.10.21/server-dev/dsee-15-amd64/
main/
enabled=1
priority=1
gpgcheck=0
#gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-
release

"""
EOF

mock -r $pkg --nocheck --cleanup-after --rebuild
$pkg --resultdir=/data &>/tmp/$1.log &
```

GRUB 与系统引导

● 武汉 技术部/文

计算机引导过程

固件

确切地说，固件（Firmware）也是一种软件，但是它比一般的软件（例如操作系统或者运行在操作系统上的应用软件）更接近硬件，一般是由硬件厂商在硬件出厂前直接固化到硬件内部的芯片上。有些固件是可以升级的，例如大部分计算机主板的固件，这里我们讨论的也主要是计算机主板的固件。

现在我们能看到的计算机主板固件主要上分为两种：BIOS（Legacy BIOS）和 UEFI，BIOS 历史悠久，数十年没有太大的变化，已经不太适应计算机的发展，因而催生了更加先进和适应时代需求的 UEFI。

BIOS 是 Basic Input Output System（基本输入输出系统）的缩写，如前文所说，它也是可执行的程序代码，计算机启动时会首先将 BIOS 载入到内存并执行，并由 BIOS 来完成硬件检测和初始化，然后启动磁盘上的操作系统。

UEFI 是 Unified Extensible Firmware Interface（统一可扩展固件接口）的缩写，它是 BIOS 的替代者，并且本着向下兼容的原则，大部分 UEFI 都包含 BIOS 的兼容模块（Compatibility Support Module/CSM），在其设置中也能找到相关的选项。

BIOS 与 MBR

一般来说，BIOS 需要配合 MBR 分区表来引导系统（GPT 分区表也可以通过 PMBR 兼容 BIOS 引导），对于使用 BIOS 的计算机，其系统引导过程如下：

- BIOS 被加载执行，完成加电自检（Power On Self Test, POST），并检查和初始化需要的硬件设备

- BIOS 根据预设的顺序查找引导设备（能从中读取数据，且首扇区以 0x55AA 结束）

- BIOS 执行第一个找到的引导设备首扇区前 440 字节代码，即主引导记录（MBR）

- MBR 通常是启动管理器（Boot Loader）的第一阶段代码，由它引导启动管理器的第二阶段

- 启动管理器完成操作系统的引导启动

MBR 是 Master Boot Record 的缩写，又叫做主引导扇区，是计算机开机后访问硬盘时所必须要读取的首个扇区。在深入讨论主引导扇区内部结构的时候，有时也将其开头的 446 字节内容特指为“主引导记录”（MBR），其后是 4 个 16 字节的“磁盘分区表”（DPT），以及 2 字节的结束标志（0x55AA）。因此，在使用“主引导记录”（MBR）这个术语的时候，需要根据具体情况判断其到底是指整个主引导扇区，还是主引导扇区的前 446 字节。

UEFI 与 GPT

UEFI 的出现就是为了代替 BIOS，GPT 分区表（GUID Partition Table）是同 UEFI 一起诞生的，它突破了 MBR 分区表 4 个主分区及最大 2.2T 容量的限制。

UEFI 与 BIOS 相比还有很多特性，例如：

- 支持文件系统，可以直接读取分区上的文件
- 可以存储启动管理器的路径并直接引导之



- 可以加载并运行 EFI 应用
- 提供 EFI Shell

主流的 UEFI 都支持 MBR 和 GPT 分区表，并且支持 FAT12/FAT16/FAT32、ISO9660、UDF 等文件系统。

一般来说，UEFI 下系统的引导的过程如下：

- UEFI 被加载执行，完成加电自检（Power-On Self Test, POST），并检查和初始化需要的硬件设备
- UEFI 根据预设直接从引导设备上加载 UEFI 应用（有可能是启动管理器，如 GRUB）
- 启动管理器完成操作系统的引导启动

UEFI 的引导依赖于分区表上的一个特殊分区，叫 EFI 系统分区（EFI System Partition），UEFI 标准中规定 EFI 系统分区为 FAT32，同时支持 FAT12/FAT16 作为移动介质的文件系统。

UEFI 的多重引导

每个操作系统或厂商都可以维护自己的 EFI 系统分区中的文件，同时不影响其他系统，一般来说各个操作系统或厂商都会将文件置于 EFI 系统分区中的 /EFI/<VENDOR NAME>/ 目录，例如 /EFI/deepin 或 /EFI/Microsoft，这样各家的文件互不影响。UEFI 的多重启动只是简单的运行不同路径下的 UEFI 程序，避免了依赖链式启动机制（Chain Loading，通过一个启动引导程序加载另一个引导程序），这些程序的路径要添加到 UEFI 设置中才能在计算机开机时自动运行或进行选择。

安全启动

UEFI 提供了安全启动（Secure Boot）的功能，安全启动功能启用时，UEFI 会验证启动管理器的合法性，一般来说验证是通过数字签名来实现的，

但很多计算机出厂时只预装了微软的证书，有的 UEFI 固件也支持基于路径的认证，还有将启动管理器路径写死的。

大多数 UEFI 固件提供了关闭安全启动的选项，还有部分可以在 UEFI 设置界面手动添加受信任的路径。

常见启动引导器

BIOS 或 UEFI 加载并初始化硬件完成后，一般会启动某个启动管理器来接管硬件设备，引导操作系统启动的工作将由启动引导器来完成。

引导程序引导方式及程序视应用机型种类而不同。例如在普通的个人电脑上，引导程序通常分为两部分：第一阶段引导程序位于主引导记录（MBR），用以引导位于某个分区上的第二阶段引导程序，如 NTLDR、BOOTMGR 和 GNU GRUB 等。

NTLDR/BOOTMGR

NTLDR（NT Loader 的缩写）是微软的 Windows NT 系列操作系统（包括 Windows XP 和 Windows Server 2003）的引导程序。

NTLDR 可以从硬盘以及 CD-ROM、U 盘等移动存储器运行并引导 Windows NT 系统的启动。如果要用 NTLDR 启动其他操作系统，则需要将该操作系统所使用的启动扇区代码保存为一个文件，NTLDR 可以从这个文件加载其它引导程序。

NTLDR 主要由两个文件组成，这两个文件必须放在系统分区（大多数情况下都是 C 盘）：

- NTLDR，这是引导程序本身
- boot.ini，这是引导程序的配置文件当 boot.ini

丢失时，NTLDR 会启动第一块硬盘第一个分区上的 Windows 目录中的系统。

bootmgr（Windows Boot Manager）是从 Windows Vista 开始引进的新一代开机管理程序，用

以替换 NTLDR。

当电脑运行完 POST (Power On Self Test) 后, 传统型 BIOS 会根据引导扇区查找开机硬盘中标记 "引导" 分区下的 bootmgr 文件; 若是 UEFI 则是 bootmgr.efi 文件, 接着管理程序会读取开机配置数据库 BCD (Boot Configuration Database) 内的引导数据, 接着根据其中的数据加载与默认或用户所选择的操作系统。

GNU GRUB 及其使用

GNU GRUB (简称 GRUB) 是一个来自 GNU 项目的启动引导程序。GRUB 是多启动规范的实现, 它允许用户可以在计算机内同时拥有多个操作系统, 并在计算机启动时选择希望运行的操作系统。GRUB 可用于选择操作系统分区上的不同内核, 也可用于向这些内核传递启动参数。

GRUB Legacy/GRUB2

新的 GRUB2 (GRUB 第二版) 为 GRUB 的重写版本, 它是 GRUB 的大革新。GRUB2 对 Linux

系统做了更多的优化, 支持更多的功能, 如动态的载入模块 (而在之前的 GRUB 中, 新增或删除模块要重新编译 GRUB) 等。GRUB2 的版本号为 0.98 或更高; 旧的 GRUB 的版本号则为 0.97 或更低, 也被称为 "GRUB Legacy" 或 "GRUB1" 等。GRUB2 的配置、命令等较 GRUB Legacy 有一定的不同, 如未指明, 本文所说 GRUB 均指 GRUB2。

GRUB 的安装

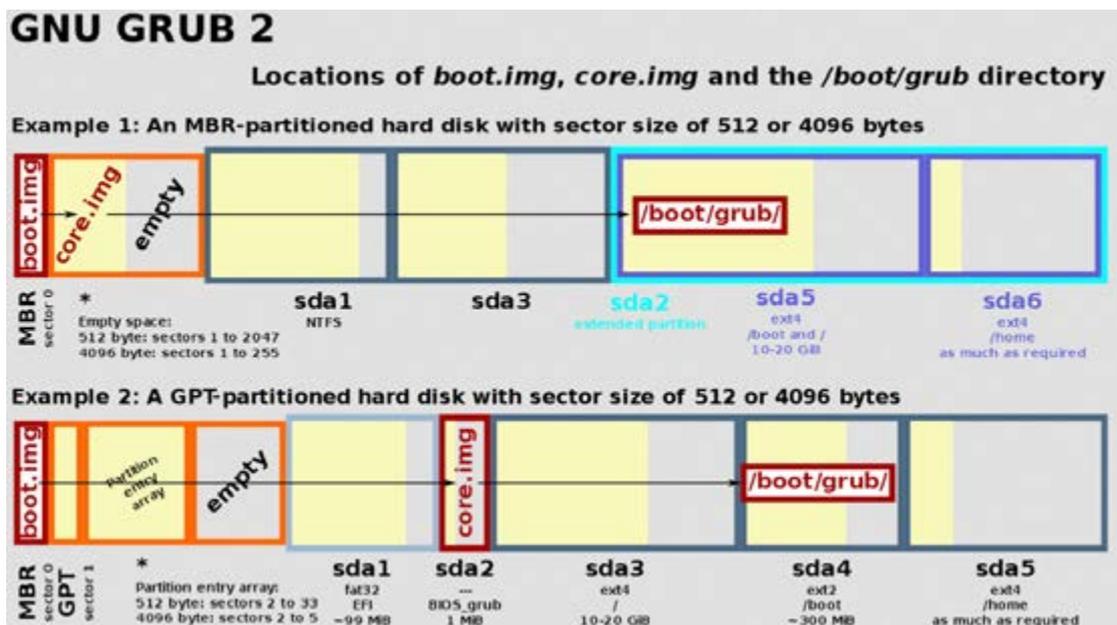
GRUB 提供一个安装程序 grub-install 用以安装 GRUB 到磁盘, 下面是 grub-install 的一些常用参数及选项:

- --boot-directory=DIR

指定 boot 目录, 用于存放 GRUB 的相关文件, 包括其镜像、模块及字体等, 同时 GRUB 也会从该目录读取配置文件, 此参数缺省为 /boot, 则 GRUB 的相关文件都在 /boot/grub 下, 配置文件路径为 /boot/grub/grub.cfg。

- --target=TARGET

指定要安装的 GRUB 类型 (对应不同的 BIOS





启动模式)，常用的可选参数有 i386-pc、x86_64-efi、i386-efi 等，对应的文件会被安装到 <boot directory>/grub/<target>/（<boot directory> 为 --boot-directory 参数指定的 boot 目录，默认为 /boot）下

- o --force

即使遇到报错仍然强制安装

- o UEFI 特有参数及选项：

- --efi-directory=DIR

指定 EFI 系统分区的根目录，即 EFI 系统分区的挂载点

- --removable

为移动设备安装，将 GRUB 安装到 U 盘时必须使用此选项

- --bootloader-id=ID

指定启动项 ID

GRUB 安装实例

1. 将 BIOS 模式的 GRUB 安装到 /dev/sda

```
sudo grub-install --target=i386-pc --boot-directory=/boot /dev/sda
```

上面的 --boot-directory=/boot 一般可以省略。

注意：要安装 i386-pc 模式的 GRUB 需要安装软件包 grub-pc-bin

```
sudo apt-get install grub-pc-bin
```

2. 将 BIOS 模式的 GRUB 安装到 U 盘 (/dev/sdb)

假设 /dev/sdb 只有一个分区，文件系统为 FAT32，如果 U 盘已经挂载，先卸载之：

```
sudo umount /dev/sdb1
```

安装 GRUB：

```
mkdir /tmp/sdb1
sudo mount /dev/sdb1 /tmp/sdb1
sudo grub-install --target=i386-pc --boot-directory=/tmp/sdb1/boot /dev/sdb
sudo umount /dev/sdb1
```

上面的 --boot-directory=/tmp/sdb1/boot 不可省略。

3. 将 x64 EFI 模式的 GRUB 安装到计算机 UEFI（EFI 系统分区为 /dev/sda1）

假设 /dev/sda1 已经被挂载到 /boot/efi，如果没有先手动挂载之：

```
sudo mount /dev/sda1 /boot/efi
```

安装 GRUB：

```
sudo grub-install --target=x86_64-efi --boot-directory=/boot --efi-directory=/boot/efi
```

上面的 --boot-directory=/boot--efi-directory=/boot/efi 一般可以省略。

4. 将 x64 EFI 模式的 GRUB 安装到 U 盘 (/dev/sdb)

同样假设 /dev/sdb 只有一个分区，文件系统为 FAT32，如果 U 盘已经挂载，先卸载之：

```
sudo umount /dev/sdb1
```

安装 GRUB：

```
mkdir /tmp/sdb1
sudo mount /dev/sdb1 /tmp/sdb1
sudo grub-install --target=x86_64-efi --boot-directory=/tmp/sdb1/boot --efi-directory=/tmp/sdb1 --removable
sudo umount /dev/sdb1
```

上面的 --boot-directory=/tmp/sdb1/boot --efi-directory=/tmp/sdb1 --removable 不可省略。

GRUB 的配置文件

GRUB 配置文件的文件名和位置随系统的不同而不同；常见为 /boot/grub/grub.conf。

修改 GRUB 的配置文件后，不必把 GRUB 重新安装到 MBR 或者某个分区中。在 Linux 中，grub-install 命令是用来把 GRUB 的步骤 1 安装到 MBR 或者分区中的。GRUB 的配置文件、步骤 2 以及其它文件必须安装到某个可用的分区中。如果这些文

件或者分区不可用,步骤 1 将把用户留在命令行界面。

GRUB Shell 常用命令

在 GRUB 界面下按 C 可进入 GRUB Shell 界面,它提供了一个简单的类 Bash 的命令行环境,下面介绍一些常用的命令:

- ls: 不加参数使用时,列出当前的存储设备及识别出来的分区 可以接目录路径为参数,列出目录内的内容
- cat: 查看文件内容
- echo: 显示字符串或环境变量
- set: 设置环境变量,例如 set root=hd0,gpt1
- insmod: 载入模块
- loopback: 映射回环设备
- search: 搜索分区
- linux: 载入 Linux 内核
- initrd: 载入初始化内存盘
- boot: 启动系统
- chainloader: 链式启动其他文件
- source: 载入 GRUB 配置文件

GRUB Shell 操作实例

手动启动硬盘上的 deepin 系统

假设硬盘分区表为 GPT,第一个分区为 FAT32 格式的 EFI 系统分区,第二个分区为 deepin 的根分区,使用安装了 GRUB 的 U 盘启动电脑,进入 GRUB Shell (在 GRUB 菜单界面按 C),可按如下操作:

1. 载入 GPT 分区表支持模块: insmod part_gpt
2. 查看存储设备及分区: ls
3. 假设计算机硬盘被识别成 hd1,设置 hd1,gpt2 为根分区: set root=hd1,gpt2
4. 查看根分区内容: ls /
查看 /boot 目录内容: ls /boot
查看 /boot/grub 目录内容 ls /boot/grub
5. 载入 Linux 内核: linux /boot/vmlinuz-4.9.0-

deepin2-amd64 root=/dev/sda2 (示例,也可以用 linux /vmlinuz root=/dev/sda2)

6. 载入初始化内存盘: initrd /boot/initrd.img-4.9.0-deepin2-amd64 (或 initrd /initrd.img)

7. 启动系统: boot

如果一切正常,系统会正常启动。此外,在上面第 4 步的时候,也可以用 source /boot/grub/grub.cfg 命令载入硬盘上的 GRUB 配置文件,然后按 Esc 返回 GRUB 菜单界面,会看到原本电脑上 GRUB 菜单,直接选择 deepin 系统启动即可。

在 LIVE 环境修复 GRUB 引导

进入 LIVE 环境

使用系统修复光盘或 U 盘启动计算机即可进入 Live 桌面环境。

进入 CHROOT 环境

在 Live 桌面环境中打开终端

确定硬盘上系统的各个挂载点的分区,以下表所示情况举例:

挂载点	分区	备注
/boot/efi	/dev/sda1	UEFI 下特有
/	/dev/sda2	
/home	/dev/sda3	如未单独分区则忽略
/var	/dev/sda4	如未单独分区则忽略

挂载文件系统

```
mkdir /tmp/mnt # 建立临时挂载目录
cd /tmp/mnt # 进入目录
sudo mount /dev/sda2 ./ # 挂载根分区
sudo mount /dev/sda1 boot/efi # 仅在 UEFI 模式下挂载 EFI 分区
sudo mount /dev/sda3 home # 如果仅修复引导此分区可不挂载
sudo mount /dev/sda4 var # 如果 /var 单独分区,必须挂载
sudo mount --bind /sys sys
sudo mount --bind /proc proc
sudo mount --bind /dev dev
sudo mount --bind /dev/pts dev/pts
```



进入 chroot 环境

```
sudo chroot /tmp/mnt
```

修复 GRUB 引导

切记，修复本机 GRUB 引导需要在 chroot 环境下操作！

BIOS 下：

```
grub-install --boot-directory=/boot /dev/sda  
update-grub
```

UEFI 模式下：

```
grub-install --boot-directory=/boot --efi-  
directory=/boot/efi  
update-grub
```

参考资料

1. ArchWiki: Arch boot process (简体中文)
2. 维基百科: 主引导记录
3. 维基百科: 统一可扩展固件接口
4. 维基百科: GNU GRUB
5. ArchWiki: GRUB (简体中文)
6. GNU GRUB Manual 2.02
7. 维基百科: Windows Boot Manager
8. 维基百科: NTLDR

深度·讲坛 征稿啦

是什么让你豁然开朗？是什么让你灵感迸发？又是什么让你百思不解？

这里汇聚了深度大神，专门凑在这里搞技术，从今往后，这里专治各种疑难杂症，你可以来提问，可以来传授技术，总之，这里都是爱凑热闹的技术控。

分享专业知识，方便学习交流，内刊“深度·讲坛”栏目长期向各位技术控们征集技术稿件啦。

字 数：1000+，图文并茂

内 容：必须为原创

投稿邮箱：deepin-magazine@deepin.com

注明“部门 + 姓名”





志愿者装机活动

深度操作系统的发展离不开社区的支持，同时我们也会及时响应社区用户的需求。为了更好的服务、推广、宣传深度操作系统，让更多的社区用户参与线下互助和传播，通过社区用户反馈的建议，我们推出了免费装机地图活动。

本着奉献、友爱、互助、进步的志愿者精神，志愿者们利用业余时间，帮助希望了解 deepin 操作系统的爱好者们，免费上门安装系统，过程中不仅丰富了自己的专业知识，也结识了更多 Linux 爱好者。

活动开始至今，已覆盖全国 31 个省市，参与志愿者近 300 余人。

捐助渠道

深度操作系统的发展离不开社区的支持，为了进一步完善深度操作系统社区生态环境的建设，深度科技对外正式开通捐助渠道。

捐助秉承着完全自愿的原则，不管你是个人还是组织、来自何处、捐助金额多少，都是对社区的发展贡献出自己的力量，我们对你的捐助深表感谢。

捐助定位：

- 深度操作系统是一个致力于为全球用户提供美观易用、安全可靠的 Linux 发行版。
- 深度操作系统的发展离不开社区的支持，为了进一步完善深度操作系统社区生态环境的建设，深度科技内部通过决策，对外正式开通捐助渠道。
- 捐助秉承着完全自愿的原则，不管你是个人还是组织、来自何处、捐助金额多少，都是对社区的发展贡献出自己的力量，我们对你的捐助深表感谢。

捐助目的：

- 调查各个项目参与捐助的金额和人数，决定项目开发的方向和优先级
- 持续和深入的开发
- 社区活动和建设（例如：内测活动奖品、志愿者奖励以及社区礼品）





使用深度 deepin 国产操作系统的一点心得与体会！

● 用户 杨兴山 / 文

近期，勒索病毒肆无忌惮的传播着。同时根据电力公司文件精神，关于检查电站二次安防及继电保护专项工作，使我更进一步地认识到，我们的电力系统对因特网攻击，对网络安全、信息安全的高度重视与重要性。

我相信在我们生活、工作越来越依赖网络的今天，我们的国家、我们的公司和我们自己也会越来越注重我们自身数字数据信息的安全。

当然，站在国家的层面上，国家已经采取行动，2014年中央国家机关明令禁止所有计算机类产品不允许安装 Windows8、windows10 操作系统。出现这样的局面，当然是因为微软为了自身知识产权利益着想，不允许把操作系统源代码给政府部门审核的原因了。

因为谁都不知道他们不可控的操作系统里面有没有留后门，会不会后台悄悄上传我们相关敏感的信息文件，这是很难说的，更何况美国还出现过棱镜门事件（电子监听计划）。

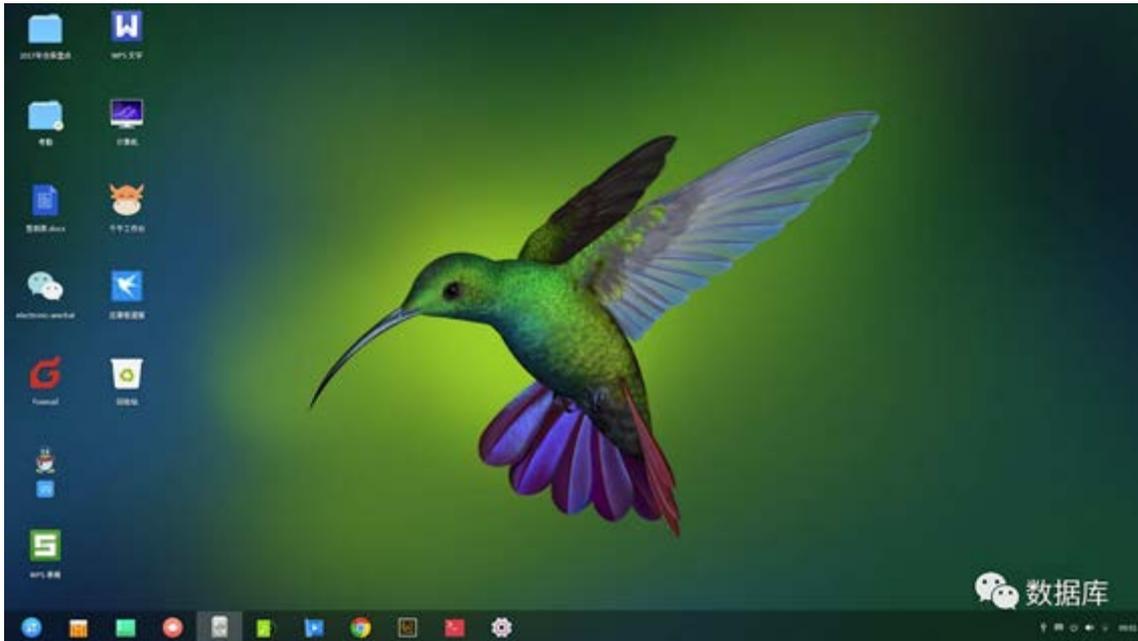
未来是大数据时代，谁掌握人民的衣食住行，谁就能及时地把控经济利益的方向，甚至一个国家命脉！虽然，国家政府并没有明令禁止或要求每个公民都将 Windows 操作系统更换掉，当然，这样不符合公民自由权利的法律法规，也不符合 WTO 相关立法。但是，国家发自内心的还是需要我们每个人都配合和行动起来的。

出于这样的思想和目的，配合电力公司的这次二次安防的检查工作，加上近期出现的勒索病毒的出现。故此，我把自己的电脑更换安装上了深度 deepin15.4 国产操作系统。

虽然深度 deepin 国产操作系统是基于国外开源软件 Linux 系统内核开发而成。但相对其他各个版本的 Linux 国产操作系统发行版来说，已经前进了一大步。

尽管 linux 操作系统在桌面的占有率只有 2% 左右，大部分市场被微软 Windows 所占据。但 Linux 系统在服务器领域，嵌入式系统以及手机操作系统的内核中占达 90% 以上市场占有率。这一方面，是它免费。另一方面是它安全可靠，运行稳定。有资料显示，一台几年前的硬件配置的电脑安装 Linux 系统，连续不停机运行几年，每次操作还像刚安装一样的流畅。这一点，Windows 就做不到，它只要运行一段时间就会被安装的应用程序垃圾文件，注册表给拖累，变得越来越慢。同时会被各种安全卫士，杀毒软件占用较多资源。

目前，使用 Linux 操作系统的用户很少，其中最重要的一个原因是，能够供他可运行的应用程序很少。用的人少，应用程序开发厂商就更不会投入资金，人力物力，去开发 Linux 系统版本应用程序的产品，因为他们赚不到钱。所以在 Linux 系统中运行的大多数应用程序都是开源免费的软件。



深度 deepin 国产 Linux 操作系统系统的出现，对这一局面有所改变。深度团队也致力于改变国产 Linux 操作系统生态建设，不紧实现在 Linux 系统中可直接运行 Windows 应用程序，同时还可以实现安卓系统的直接运行的原生态般体验。也就是说，Windows 操作系统能做的应用功能，我们国产操作系统同样能实现！

深度 deepin 国产操作系统通过其深度应用商店，不紧使安装 Linux 系统应用程序变得无比简单，同时使我们不必去各个应用软件官方网站或软件下载网站到处去查找下载，也就给我们节省了时间，避免下载中病毒风险与机率。深度应用商店其中每款软件都是深度操作系统专业人员收集测试可供完美下载运行的。

经过多月以来的使用与测试，深度 Linux 国产操作系统完全可以代替微软 Windows10 操作系统在日

常工作中的使用。甚至他的桌面环境，主题界面设计都丝毫不输给微软 Windows10，甚至有些地方远超过了 Windows 符合国人的操作的习惯。它完全能给我们耳目一新，操作流畅，运行稳定的，化繁为简，节省操作时间，提升工作效率的期望。同时，我们公司大部分中控室后台监控系统也是使用 linux 操作系统系统，我们还可以通过深度国产操作系统不断提升维护我们监控操作系统业务能力！

在工作中，我们完全可以通过深度国产操作系统使用 WPS、QQ、微信、谷歌浏览器、迅雷下载、winrar、搜狗输入法、邮箱客户端、打印机等等 Windows 操作系统中有的它基本也可以代替实现。

当然，如果你想利用它打传奇，英雄联盟等网络游戏，那么，可以明确告诉你，它做不到！服务商也不支持！**d**



要想完全放弃 Windows 使用 Linux 需要多少勇气?

● 用户 朱广彬 / 文

作 为一名资深码农，大部分时间都在用 Linux 做开发，但不得不因为 IE、Outlook、网银、QQ 等选择 Windows+Linux 双系统，直到尝试了 Deepin OS，一切似乎完美的解决了，从此放弃 Windows，再也不受一堆病毒和流氓软件的困扰，安心的做该做的事情。

我为什么只推荐 Deepin OS，而不是其他比如 Ubuntu(事实上我之前一直用 Ubuntu)、Linux Mint、Arch Linux 等，因为 Deepin OS 简单方便。

Deepin 默认即将常用的软件装好了，基本上满足了大部分人的日常办公需求。试想，当我终于鼓足了勇气备份好资料重装系统，并花费了半个小时安装完毕后，还要再一个一个安装搜狗输入法、Chrome、Office、QQ、邮件等等，是不是心好累……Deepin 则默认装好了这一切，真正做到开箱即用。以我的经验，重装系统需要花费 20 分钟，剩下的就是装一些自己特殊需求的工具软件了，而且就在 Deepin 提供的深度商店里面，这种类似于 App Store 的软件安装方式是我喜欢 Deepin 的又一个理由。

下面来介绍下我常用的软件吧：

浏览器：Chrome(装机自带的，这个算必备了吧)

输入法：搜狗 (自带，我只用搜狗)

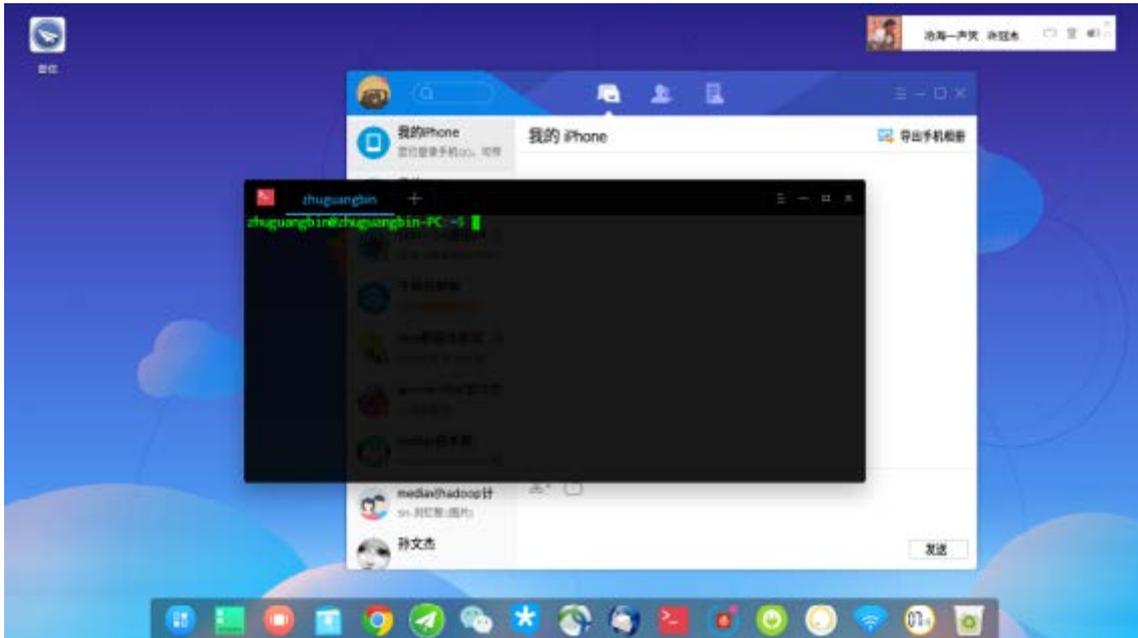
文档办公：WPS、福昕 PDF 阅读器、Thunder Bird (自带，Thunder Bird 可以装一个 Exchange 的插件来收发 MS Exchange 邮件)

社交：QQ、TIM、Electronic WeChat (QQ 是自带的，其他可以去深度商店里下载。还是忍不住在这里着重说明下，Deepin 自带的 QQ 稳定性很赞，之前用 Ubuntu 折腾各种 Wine 的版本，折腾鸡肋的 WebQQ，浪费了多少人生大好光阴……)

娱乐：自带网易云音乐 (大爱)、深度音乐、深度影院等，对我来说够了

网盘：OverDrive (自己去官网下 deb 包安装，同步 Google Drive 的 Linux 下最可靠方案，值得花





4.99 刀)

开发类：Intellij IDEA、Eclipse、Sublime Text、Atom (太多了，自己去深度商店安装，哦，最重要的也用的最多的 Terminal，深度终端足够了，支持多 tab、支持远程会话管理，还有强大的雷神模式，让我放弃了 putty 和 SecureCRT)

其他：不得不说深度商店真的很强大，里面有各种你想到的和想不到的软件，如果找不到，请看下一行 CrossOver，功能很强大还比较稳定，如果在商店里找不到想要的 Windows 软件，用这个来安装创建容器，我装的蓝信使用下来还是比较稳定的。

Deepin 的桌面也美得让我敲代码都舒服，谁说

写代码就要干巴巴的 IDE 和远程终端，别搞那些花里胡哨？我就习惯听着音乐敲代码，程序员的枯燥生活需要美好的桌面来增加乐趣。

码了这么多字，最后我只想说，我选择 Deepin 是因为我的工作生活大部分是在 Linux 下，因为我是个码农，Windows 已经几乎没有我依赖的了。如果你严重依赖 Windows，就不要为了 Linux 而 Linux，工具是为了方便工作生活的，切勿为了折腾而浪费青春。我也曾经为了折腾 Ubuntu 下 QQ，Ubuntu 下主题美化方案浪费了大把青春，现在，有了 Deepin 我可以安心一边听网易云音乐、一边敲代码，有问题通过 QQ 来沟通，开发生活两不误，放弃了 Windows 的垃圾软件和弹窗、病毒，一切变得和谐美好多了…… [d](#)



和深度一起度过的时间

● 用户 码农 BTS/ 文

写在前面

有时候使用 Linux 是一种信仰，是一种发自内心的喜爱，是一种许多人无法理解的事。Linux 在很多人的生命中仿佛都是过客，他们活在虚拟机里、活在服务器里、活在你常年不会碰的第二个分区里。Linux 的命运仿佛就是这样，默默地在一个角落，想起来就去碰碰，然后就让它被灰尘掩埋。

Linux 不该是这样的，一直不应该是这样的，Linux 应该被大众所接受，应该像 Windows、像 Mac OS 一样走进所有家庭。Linux 的光芒不应该局限于服务器、虚拟机，它应该成为一个出色的家用操作系统的。

但是谁都知道，这一切并非如此简单，毕竟 Windows 已经在国人心中扎下了根，是一种不可改变的认知，甚至连 Mac Book 的用户都希望把自己的笔记本刷上 Windows 系统。几乎大多数人连 Windows 的操作都还不熟悉，也不愿意去熟悉，也没有必要都熟悉。乍一看，国内对系统的需求似乎到达了一个“饱和”的状态，连新版本的 Windows 都没有人愿意去接受，Linux 将如何在国内立足？只有让一个全新的、更易用的、更人性的、更自由的系统出现，才可能改变这一切的现状。

在我认识 Deepin 之后，仿佛对 Linux 的立足有了信心。

我与深度 Deepin 的相识，像是一种必然。

从小学开始，我对脚本编程开始有了兴趣，从小学三年级在记事本上的复制粘贴，到后来的自己写愚人脚本，慢慢接触高级语言，对编程的一切开

始感兴趣，精益求精地寻找更优算法，一直到现在，初三了，确定了短期 C 嵌入式开发的研究方向，每天自信地打开笔记本，拥抱友好的 Linux 系统。

虽然我现在已经不只是一个深度用户了，但这只会让我更加珍惜使用深度 Deepin 的那段时间，只会让我更加地去爱这个系统。如果没有上一次的崩溃意外，我肯定不会选择临时放弃 Deepin，如果还有下一次，我肯定会重新选择 Deepin。

为什么？

我初识 Linux 的时候是初一，当时使用的发行版，也是现在所使用的发行版是 Ubuntu，当时开始使用 Linux 只有一个原因，一个程序猿亲戚告诉我，我还没有进入真正的编程世界。于是接下来，我唯一要做的一件事，就是去进入“真正的编程世界”，就这样，我认识了 Linux。

与很多人一样，我选择了安装双系统，Windows 10 + Ubuntu 15.04。这是很愉快的一段时间，开始慢慢接触不一样的目录风格，接触全新、振奋的终端操作。是的，是很愉快的时间——但我的愉快并不全在 Linux 上，我还是需要 Windows，我还是需要 Windows 上的一切，去填补一个不完美的 Linux 的其他功能。而且很明显地感到不舒服的一件事就是：你需要在两个系统之间切换。

日子平淡如水，我继续我的学习生活，只不过身边多了 Linux 的陪伴。

一年多之后，我的笔记本走向了老年，在双系统的折磨下也已经大不如当时了。一个热爱计算机、

热爱代码的人，很不能接受的一件事情，就是缓慢的运行速度。所以我换了笔记本——这次更换笔记本之后，我陷入了很纠结的事情：还要装双系统吗？

我已经离不开 Linux 了，体验过 Linux 的便利的人都能够理解，那 Linux 是否能够胜任我的日常需求就成为了很重要的决定因素。终于，我遇到了 Deepin。

我了解 Deepin，是很久以前听一个同样喜欢折腾 Linux 的同学说的，只不过看起来他对 Deepin 的印象并不怎么好，他的原话是：“千万不要去用 Deepin！它会颠覆你对 Linux 的认知！”但是这次，我选择了 Deepin，因为这一次我就是需要一款颠覆性的 Linux 发行版，我就是需要一个与众不同的、能够胜任日常工作的 Linux。

初识

第一次亲眼看到 Deepin，我的第一感受、第一印象是美到极致。当时用的可能是 15.1 的版本，喜欢它那种极简的风格设计，欣赏其扁平的设计风格，欣赏那种友好的感觉。程序猿好像都是只看中性能，对桌面环境并没有很高的要求，但是看到 Deepin 的那一刻，我真的是有点被震撼到了。这可能不是国内最优秀的操作系统，但绝对是国内最好的桌面环境。

我当时推荐很多人入了 Deepin 的坑，都是我身边最好的朋友。为什么？WPS、QQ、Chrome 什么都有，如此友好的操作界面，谁会不动心？谁会认为这不是一个近乎完美的操作系统？反正不管是什么样的印象，至此我永远告别了 Windows，开始接触 Deepin。

其实在安装 Deepin 的时候我也有遇到过一些小麻烦，但是这个小麻烦反而给我带来了深度更大的好感。当时的 Deepin 15.1 并不支持我新笔记本的触控板，我凭着对焦点的观察用键盘完成了安装

过程，但是我不可能用键盘过一辈子吧？虽说用命令行居多，但绝不全是，于是我开始在深度社区上发表求助帖，让我怎么都没想到的是，王勇（深度科技副总经理）居然会亲自回复我的帖子，然后就给了我他的联系方式。我未曾想到过一个问题能引起一个团队这么大的关注，曾经在国内外的帮助社区从未出现过这样的现象。更让我惊讶的是，居然会有一个工程师团队为了解决这个驱动问题，与我反复确认数据、抓取日志、远程操作，定制一枚驱动给我，这是一个多么有热情的团队才能够做到的！

Deepin 是很酷的系统，我从一开始就认定了。我是一个忠诚的 Vim 用户，自然不会少用终端。在使用终端的过程中，我才真正明白了“高效”的定义，并且我不停的感叹，这是怎么样的一个团队，能够想到这么多？能够为用户的使用想到这么多细节——透明度的调整、完美的横竖分屏、完美的补全和快捷操作，搭配 Vim 使用是一种什么样的感受？我曾经一度想着要双显，要分屏，要 21:9 屏幕，直到遇到深度终端，这一切的需求显得有点多余。真正震撼到我的，是当我知道这个所谓的深度终端居然是王勇在“空余时间”写出来的作品。

这样一个震撼的初识，让我心中已经有了决定——我必将和 Deepin 走过一段很长的路。

生态

我的师傅，红盟的一位技术顾问，曾跟我们聊起过一个系统最重要的是什么。有人说是安全，有人说是文件系统，有人说是 UI，而他给出的见解是——生态。一个系统没有生态就没有未来，我们国内完全可以做出自己的内核，自己的系统，可以做得更好，但为什么没有人去踏出这一步？

原因很简单，因为没有生态。你如果不是主流，就不会有程序猿来为你编写基于你的系统的应用，



没有应用就不会有人买你的账，没有人买你的账你就无法成为主流，这是一个死循环。Windows 也好，Linux 也好，都已经将自身的生态发展得很完善，但一个新的系统并不会。

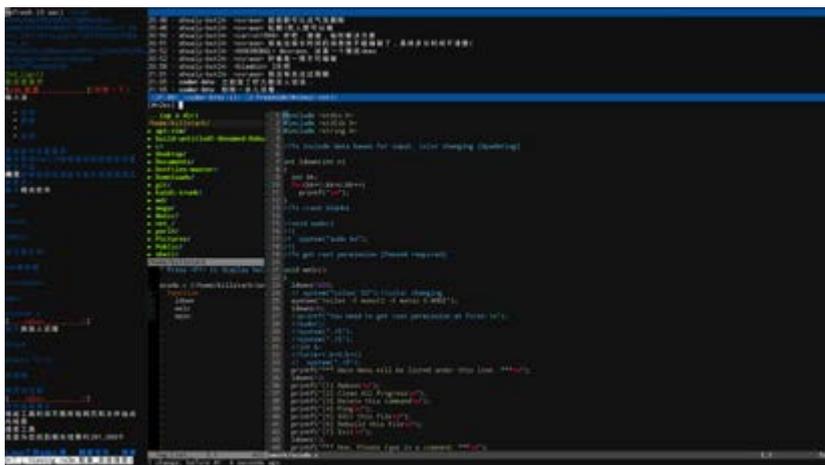
深度的生态打造让我感到非常友好——这可能是最合适的形容词了。如果没错的话，Deepin 应该是全国唯一一个为用户付

费购买 Crossover 的系统，为用户着想，尝试着去发现用户的需求。就凭我当时的使用感受来说，至少 Deepin 是第一个让我能够在 Linux 上完美地使用 QQ、WPS 的系统。说 Deepin 是最适合家用的 Linux 系统，可能未免太过了，但是它绝对是最适合中国人家用的 Linux 系统！这是一个为国人量身定制的系统，好像是 Linux 界的 MIUI。

这一切都来自于一群充满热情的工程师，精细的软件移植，深度的产品合作，大量的原生软件——这是最重要的一点，用过 Linux 的朋友很可能都有一个自己惯用的软件清单，因为毕竟在 Linux 上找到一个自己很喜欢的软件并不容易，当然也有像我一样不挑剔的码农，看每一个软件，只要能运行，但是遇到深度，你会遇到全新的元素：自己打造的播放器、自己打造的图片浏览器、自己打造的音乐软件……每一项都是为了最适合国人的生态。

桌面环境

很多人跟我说：深度太花哨了，它的桌面环境占用的资源实在太多了，会严重拖慢效率……其实并不是每一个人都是技术宅，不是每一个人都喜欢 KDE 早期那种一片灰蒙蒙的“极简”界面，深度的



目标是让 Linux 走进更多人的生活和工作，去普及 Linux，当然如果你真的那么嫌弃深度的桌面环境，完全可以关闭动画渲染，或者干脆用控制台。

普及 Linux 是一个大工程，尽管深度有适合国人的最完美的生态环境，但是要让国人从根本没听说过 Linux 到普及使用，还需要很多其它的付出。于是，深度的 DDE 横空出世，比 Elementary OS 还要美。是的，我承认很像 Mac OS，但是如果一项设计真的非常好，真的很完美，为什么不采用这项设计？Mac 证明了这种设计的优点，为什么不把他延续到 Linux 上？为什么不去给 Linux 用户带来一个更友好的体验呢？

深度的桌面环境很诱人，特别是 15.2 的时候，任务栏的位置实现了可以自定义，这让 Deepin 变得更加自由、更加人性化，和深度一起的时光中，每天看到一种清新的桌面风格真的是一件很愉快的事情，每一次轻击触控板带来完全不一样的体验。说真的，在这种桌面环境下，学习 Linux 比学习 Windows 要简单的多了，更加人性、更加优雅。

DDUC

当然，使用的过程中，深度也存在一些不足，



作为一个年轻的系统，它的稳定性还没有那么好，控制台可能会假死，可能会有各种各样出人意料的bug。但是这都挡不住开发团队的热情，因为他们会定期停下脚步，仔细反思、仔细聆听、收集各种各样的建议，为自己找到全新的灵感——就这样，便有了DDUC——深度开发者与用户交流大会(Deepin Developers & Users Conference)。

最能展现一个团队最真实的一面的，就是在与用户间心与心、脸对脸零距离沟通的时候。去年的10月我参加了第六届深度开发者与用户交流大会，看到王勇自信的笑容，看到整个团队自信的笑容，看到所有开发者们充满了动力的讲述，我更加确定这就是我所崇敬的那个团队，更加确定这就是那个最棒的国产操作系统团队。

让开发者与用户有一次最近距离的交流，让用户了解开发背后的故事，让开发者聆听用户的感受，这是其它很多系统都做不到的。这种线下技术交流是其他团队所缺乏的。会议中也有邀请Intel的工程师来讲述最新的研究成果、也有邀请嘉宾来分享自己学习的历程，每个人都是那么认真的听着，语调虽然不会多样，但是内容绝对不会枯燥。开源，让一切变得美好，是开源才让Deepin进步如此之快。

结语

使用深度Deepin的过程中，与其说是享受，更应该是惊喜。我从未想象过国内会有如此优秀的操作系统，未曾想过会有如此自信的开发团队。跟这样一群人在一起的感受不像是用户与开发者，更像是朋友之间的感情，是一种心与心之间的交流。

深度还很年轻，他还需要很多改进，需要很多的汗水，很多的时间。我很自豪能够作为一个中国人看着这样一个国产品牌成长起来，很感动看到它日益强大，更期待它让Linux变得普及的那一天。我也还很年轻，身边的朋友都还很年轻，新一代的精英都很年轻，我很自豪能够为深度、为开源、为国家做出力所能及的贡献，也希望自己像深度那样保持自信和活力！

他不是浮在表面上的UI定制，而是从最底层的优化，从用户出发的改进，为国人打造的系统环境，这不是翻版，这不是照葫芦画瓢，也不是抄袭，这是一次真正从心出发的旅程。用户不是仅仅在使用，他们是在用心与系统交流，体验它的魅力。开发者不仅仅是在完成任务，Deepin就像是他们的孩子一样，受他们的爱护，如此优秀的系统，应该在国内流行起来！看好深度！[d](#)



与众不同的 deepin

● 用户 didar/ 文

要 做国产操作系统的口号已经喊了有十几年的时间了，至今没有任何一个国产操作系统能真正在生产与生活中完全替代 windows。媒体上时不时的会冒出几个关于国产操作系统的新闻，而这些新闻大多无法引起公众太多的关注，原因大概缘于民众无法感受到新闻所述的众多国产操作系统取得的诸多进步。

记得前几年参加某国产操作系统的新品发布会时，该厂商代表发言时提到要在 2020 年完成在国内市场占比超过 windows 操作系统。每每想起这段发言，我总不知不觉地记起在读初中的时候看到的电视上当红组合 F4 领取某媒体奖项时宣称要做亚洲的 U2 乐队的豪言壮语。不管怎样，在 2017 年的今天来看，留给那个厂商的时间不多了。

从 2014 年的时候开始我便在办公电脑上使用 deepin 系统，对这家厂商的看法也从‘不太一样’逐渐变成了‘我看能行’。

Deepin 操作系统是由武汉深之度科技有限公司开发的，目前在国际上排名前十的 linux 发行版。深度操作系统是中国唯一一家被国际发行版排行榜收录的厂商，从产品角度来看不一样的地方可以分为：桌面环境与应用生态。

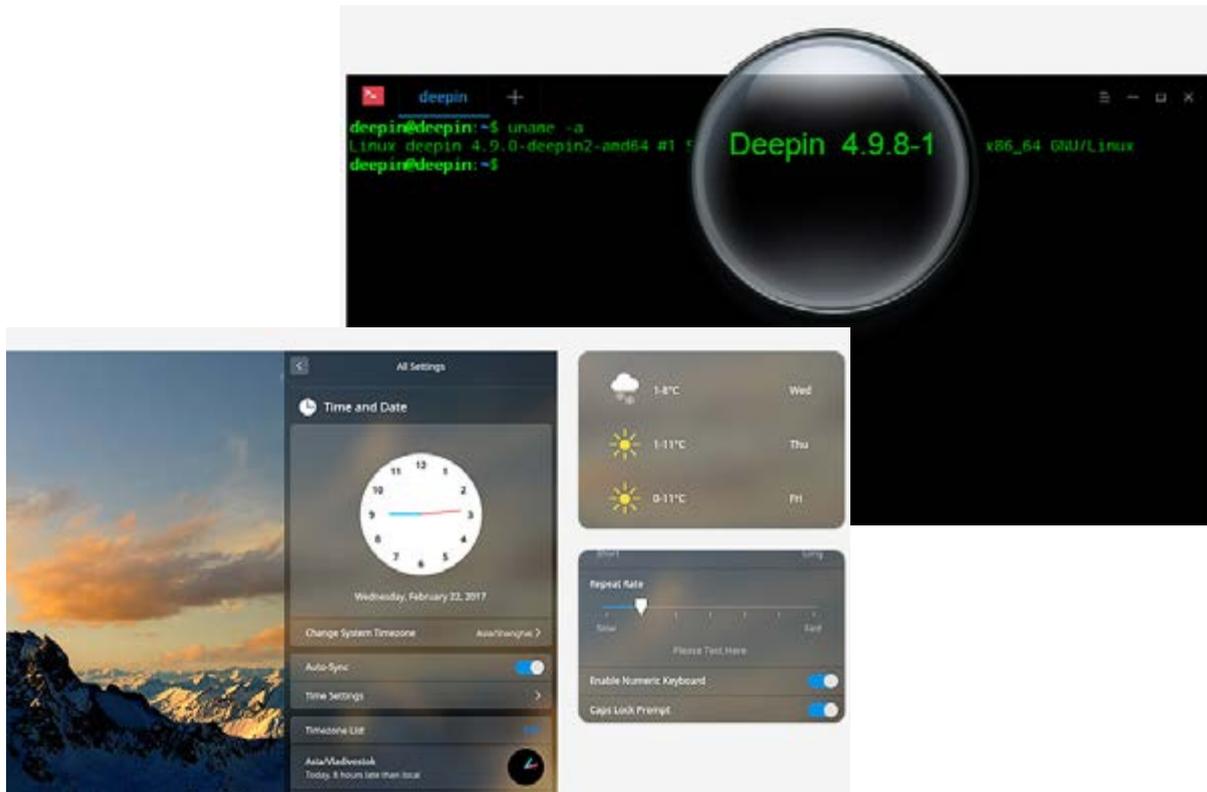
不同于从开源社区拿来的 kde 或 gnome，deepin 系统的桌面环境为自行研发的 dde。从使

用中我越来越相信：要摆脱 linux 是技术人员的专属的尴尬现状，必须使用由一家商业公司驱动开发的一套桌面环境。

在这个动不动谈体验的年代，对普通用户而言，几乎所有 linux 发行版都是半成品。技术人员硬要给普通用户解释 Ubuntu 的优越性的结果往往是彼此互相怀疑智商。好比一个程序员满心欢喜地拿着新款 nexus (pixel) 送给女朋友的结局可能会是分手，普通用户和女朋友没有义务和理由去理解这些半成品里透出来的情怀。由开源社区驱动生成的桌面环境因其社区的自身特性，是无法做到对普通用户而言的简单易用。从日常使用而言，dde 桌面环境是我使用过的最贴近普通用户的桌面环境。

Linux 应用生态极其孱弱，甚至要差过移动端的 windows phone。Deepin 系统为解决日常使用的不便而开发了十几个如深度商店、深度影音、深度录屏等嫡系应用。这些应用对应在开源社区都有不下十套的现有方案，但其设计体验毫无现代软件产品的概念。过去用户为看一个 avi 格式的视频可能就需要配置半天，而这种类似的“配置”在安装完成到正常使用过程中不计其数。如果你经历过把一款世界知名的 archlinux 打造成自己的办公终端，你一定会怀疑自己是否身处在一个软件行业高度发达的时代。

日常应用之外 deepin 还有一些联合应用厂商



开发的应用如网易云音乐、有道词典等。这些应用都能提供给用户等同于 windows 系统下的使用体验，而不是阉割版或乞丐版。除了原生 linux 应用，deepin 系统还提供了目前最成熟稳定的 wine 方案——crossover。通过 crossover 可以使用 QQ、微信、迅雷等常用应用，甚至用户可以用 Photoshop 修下图片和 Foxmail 收发邮件，相比去年某 Linux 生产厂商的发布会上把打包的网页版微信讲了三页 PPT 高出很多段位。

这几年移动端的如火如荼让人们们对桌面端的关注少了很多，其实就操作系统而言，linux 和 Android 都是舶来品。不同的是 Android 从 2008 年发布第一个版本到现在，九年不到的时间坐稳了移动端的第一把交椅。美国人不会想到他们发明

Android 的同时，全世界范围内进行定制开发玩的最好最溜的会是中国人。目前中国众多的安卓深度定制版在一定程度上引领着 Android 的发展方向。这一过程中，中国的众多手机厂商并不是在技术上强过了谷歌公司，而是通过不断的面向用户把在中国国情下的半成品 Android 加工打磨定制成为了如 MIUI、EMUI、flyme 的优秀的安卓定制版。同样，我相信未来引领 Linux 在更广大的普通用户中应用的会是那些面向用户的商业公司。而开源社区以其离散性决定了自身在做大众产品方面的先天不足，更不要说单纯依靠开源社区获取代码进行打包来发布一个发行版的厂商了。

正是基于这些，我才敢相信 deepin 系统会成为世界范围内的成功的 linux 发行版。d



深度户外拓展

● 深度科技 市场部 / 文

今年的春天来的特别早，外面热闹的小草和争艳的花朵，还有温暖的太阳都在等着深度的小伙伴们热情的拥抱，这些平日里噼里啪啦敲键盘的同学们也是时候该好好活动下筋骨了。

5月6日，深度科技武汉公司的同学们在武汉市江夏区东篱老屋生态度假区，开展了主题为“踏实·拼搏·责任，诚信·共赢·开创”的户外拓展活动，并特别设计了一些团队合作的小游戏，让大家的汗水与欢笑交织在一起，放肆笑，放肆玩。

早上9点深度科技的小伙伴们便在公司集合，

9:15 准时出发去东篱老屋。一路上大家望着窗外的美景，有说有笑。目的地有一个宽阔的大草坪，到处都是新鲜的青草香，大家可能还不知道，在这里，即将开始一系列紧张刺激的户外拓展活动。

开场热身活动——教练说

拓展教练将兴奋地小伙伴们面对面分两排站立，并根据教练的指令来做动作，这个游戏的难点就在于，教练快速发出指令，大家要在最短的时间做出反应。指令前教练会说“教练说……”视为指令有效，大家则按指令做出相应动作；若发出指令时，教练只是说了指令，没有说“教练说……”，则指令无效，



那么遵照无效指令做动作的同学就要遭殃了……

这个游戏考验了小伙伴们的应变能力，还能让“遭殃的”同学施展才华，表演节目，是不是一箭双雕啊。

第二个游戏——一字成语

小伙伴们每人先说一个带有“一”字的成语，等到大家都说完之后。教练将话筒交给第一个说成语的队员，让他把刚刚自己说的成语放在一句话的后面，如：洞房花烛夜，我 + 成语（所有成员都照此说一遍）。然后一些奇葩的句型就纷纷出炉了：洞房花烛夜我一心一意；洞房花烛夜我一鸣惊人；洞房花烛夜我一箭双雕……

第三个游戏——不倒长城

“万里长城永不倒”是一项很有趣的拓展训练项目，也称为木桶原理。为什么又叫木桶原理呢？因为大家都知道，一个木桶的盛水量取决于木桶上最短的那块木板的长度，以及每块木板之间的紧密度。万里长城永不倒其实就是这个原理。

小伙伴们围成一个圈圈，双手搭在前面一个队员的肩膀上面，培训师说开始，团队所有学员围着圈走动起来，听到培训师“停”的口令时，前面的学员立刻坐在后面学员的双腿上，大家都在想如何完成不倒长城的造型，经过多次尝试和调整，最终成功的，如果没有坐稳就算犯规，并且要接受惩罚。不过还好 deepin 的小伙伴们没有集体倒在地上的。



通过这个活动团队沟通更加默契、团队凝聚力更强、提高团队间的信任度。

万里长城永不倒项目总结：没有完美的个人，只有完美的团队。

第四个活动——团队文化建设

Deepin 全体小伙伴打散分成 6 个组，每个组给自己起一个队名、个性的口号以及团队的造型。下面大家来欣赏下我们的 6 组战斗小分队吧！

- 黑色衣服： 大白队 口号：我不是胖嘟嘟，只是气很足
- 天蓝色衣服：王者荣耀队 口号：不吃饭不睡觉，大气精神玩荣耀
- 深蓝色衣服：一马当先队 口号：狭路相逢勇者胜，一马当先我第一
- 绿色衣服： 伐木累队 口号：天霸动霸 tua
- 红色衣服： 小红队 口号：小红，小红，我最红
- 黄色衣服： 正黄旗队 口号：我们黄的一尘不染

每个队展示自己的造型，并大声喊出队名和口号，造型保持 5 秒钟。





后面还有你比我猜、看图猜小学课文等有趣的小游戏，通过这些需要团队合作才能完成的小游戏，大家部门与部门之间加深了了解，也让一些整日闷头敲代码的程序员得到了放松，deepin 作为一支年轻充满活力的国产操作系统团队，每一位同学都付出了自己的努力，希望大家在这次拓展训练中，能够释放压力，让deepin 这个大家庭充满朝气，为国产操作系统的发展，创造辉煌。

最后，deepin 的小伙伴们，we are 伐木累！ 



深度科技 · 制度规范小贴士

1. 财务管理制度变更：

1) 在差旅费报销环节，为鼓励员工在出差前提早填写出差申请，凡出差申请提交日期早于出差出发日期3天（含）以上的，差旅报销单审批通过后即可给予报销，不必等到每月报销日。

2) 针对先提交付款申请，后报销的情况，由于有的同事总会忘记索要发票进行报销，现对相应流程新增了补充规定。在付款申请逾期（以申请提交之日起2个月）未拿到发票的，不超过5000元的，直接作为申请人借款处理。超过5000元的，财务经理上报总经办暂停办理其借款、报销和相关付款，直至完成相关全部手续。

2. 活动、福利与假期规定变更：

1) 今后公司年会举行时间会安排在每年春节前，无特殊情况，全员参加。年会形式会在每年举行之前确认。

2) 每年的公司旅游，时间安排在每年7、8月前后，活动时长3天左右，无公司假。地点和项目结合实际选定，每人预算1500元，可以携带家属参加，家属费用自理。公司旅游以实际参加活动的人活动当天是否在职确认是否有该人的预算。旅游费用超公司预算的部分由员工本人自理。

3) 假期、倒休及请假相关：在此制度中所提的“年”即“公司财年”。

a) 员工年假由法定年假和公司年假组成，

公司年假为转正满一年后第二年的4月1日开始获得1天，之后每年增加1天，累计不超过10天。法定年假和公司年假综合不超过15天。

b) 员工入职后，综合部会按照国家规定核算员工当年的可休年假基数。10月1日前入职的，从入职当月至年底最多可休上述年假基数的1/2，10月1日至12月31日入职的，可休1/5，12月31日后入职的，当年无年假。

c) 员工如愿意放弃年休假，未休法定年假按日工资的2倍计算年假补贴，未休公司年假按日工资的1倍计算年假补贴。

d) 员工请事假时，如有未使用的年假，应当优先使用年假。

e) 从2017年财年开始，财年结束时的剩余可倒休天数折算为加班工资，随年末最后一个月工资一次性发放。员工辞职或被辞退的，从员工书面提出辞职通知或开始办理辞职手续之日起，剩余可倒休天数清零。所有有效加班，均可在填写加班登记时选择换倒休还是换加班费。要换加班费的，在加班记录里@直属领导后，再写上“换加班费”四个字即可，没写的或写成其他文字的，作为换倒休处理。

3. 考勤管理制度变更：

1) 北京员工采用知人打卡的考勤方式，员工登录知人HR系统，可在公司内通过蓝牙方式打卡。

2) 新增弹性工作考勤制度，分出勤率和在岗率两个考勤指标。考勤记录方式通过知人打





卡和外勤确认单共同完成。目前只有北京、武汉、上海技术部门中的部分员工适用弹性工作制。相关考勤规则详情，在 Seafile 系统公共制度及内部公告目录下的《考勤管理制度》文档中可以查阅。

4. 面试管理制度变更

1) 修改武汉地区面试流程可酌情由之前的两次面试合并为一次，在首次面试时 HR 就可同面试官共同讨论对面试者意见，并确定是否录用。

2) 调整全国销售部门总监岗位的面试官为史锦江副总。

5. 入职管理制度变更

1) 新员工的办公电脑，除设计岗位之外，默认只安装深度操作系统。

2) 新增新员工导师制度。旨在帮助新员工尽快适应新的工作和生活环境，消除陌生，熟悉公司管理氛围和常见工作习惯。制度详情，在 Seafile 系统公共制度及内部公告 / 人事规定目录下的《入职管理制度》文档中可以查阅。

6. 新增《绩效考核及优秀员工评定办法》

文档对于员工绩效考核及绩效奖金的评定标准，评分原则和对员工的影响做了详细阐

述，对优秀员工的评定办法和奖励方法做了详尽解释。

7. 新增《内部办公系统培训及考试办法》

公司对部分岗位员工要求必须使用深度操作系统作为日常办公系统，该文档对内部办公系统的培训和考试等相关信息，做了详细解释说明。

8. 新增《员工个人技能培训奖励办法》

为提升员工的专业能力和提高公司竞争力，特制定本办法对员工参与外部培训给予补贴，以鼓励各员工积极的学习和提升自己。当前针对红帽认证工程师 (RHCE) 和项目管理师 (PMP) 予以支持。详情在 Seafile 系统公共制度及内部公告目录下的《员工个人技能培训奖励办法》文档中可以查阅。

9. 新增《政府补贴申请工作奖励办法》

为了公司整体利益，鼓励市场部及其他部门相关人员申报适合企业经营发展需要的各类项目并同时争取到政府补贴资金，针对成功拿到项目补贴资金的给予现金奖励。详情在 Seafile 系统公共制度及内部公告目录下的《政府补贴申请工作奖励办法》文档中可以查阅。

I Want You

深度书籍社区团队 期待您激情加入

经过一段时间的考虑和规划，为了更好的宣传和了解深度操作系统，同时更好的分享深度操作系统方方面面的知识，我们成立了深度书籍社区团队，该团队主要是为了参与编辑深度书籍、内容大纲（参与定制大纲和讨论）没有太多的限制，希望您的参与!!!

PS: 当深度书籍（书名暂定：《给所有人的深度》）编写的程度达到印刷出书要求时，如果发售，我们会将发售收入的一部分用于社区活动奖励，所有参与撰写的用户根据贡献多少给予 1~10 套书籍的赠送。



扫二维码
填写深度书籍团队申请



deepin 集结 | 征稿启事

《deepin 集结》是深度科技内部刊物，它不仅是记录公司发展历程的一本“画册”，也是公司对外形象和企业建设的窗口。

自 2016 年伊始至今，《deepin 集结》已有定期发放给热爱深度的用户及合作伙伴，并得到了大家的认可。在收集各位读者的反馈意见后，我们开设了新栏目——深度伙伴，主要针对深度员工以外的人员投稿，使《deepin 集结》成为一个企业与企业沟通的刊物，彼此增加交流，分享开源技术。

内部征稿

投稿内容及要求：

形式：摄影作品

Ps. 不要叫你的单反在家睡大觉了，赶紧出来发挥作用吧！

绘画书法

Ps. 你不发出来怎么能知道原来你还这么有才！

诗歌散文、游记、人生感悟

Ps. 知道你原来话不多，但是肚子里面还是有墨水的！

经验技术分享

Ps. 专业达人们，把你们的专业知识拿来 show 一下吧，科普一下啦！

以及能落实到纸上的任何才艺

Ps. 还有多少是我意想不到的，快点告诉我！

外部征稿

投稿内容及要求：

1. 内容：a 技术分享

b 用户体验

c 项目评价

2. 要求：a 与行业相关

b 字数 1000-4000

c 文中图片需清晰

深度的同学可以告知身边爱好开源的发烧友积极投稿哦！

投稿邮箱主题需标明：外部投稿（字样）+ 姓名 + 手机号码

要求：我们很看重有图有真相哦，图片和文字说明一样重要。

稿酬：现金（微信红包）或精美礼品

投稿方式：deepin-magazine@deepin.com 邮件名称一定要注明“所在城市 - 部门 - 姓名”

诸位同学有任何问题，都可以立即马上咨询我们：

qindi@deepin.com

deepin

we do we change...