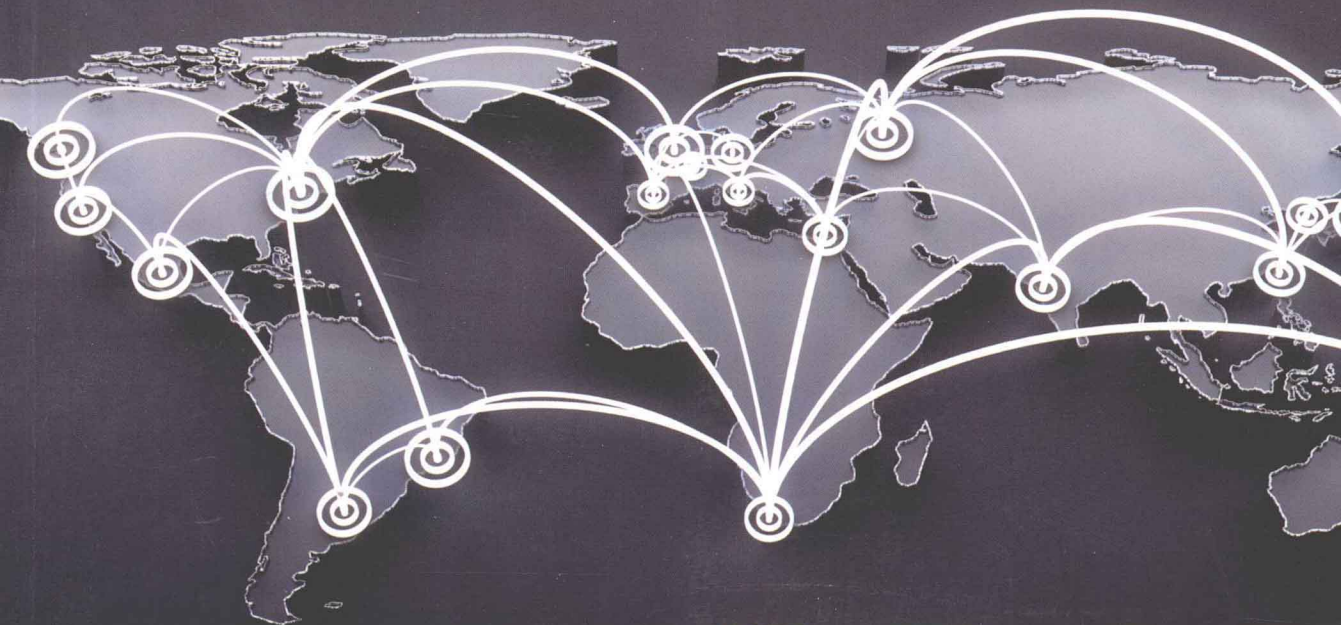


国内实践派专家精心撰写，全面深刻揭秘SDN

Broadview®
www.broadview.com.cn



深度解析SDN

利益、战略、技术、实践

张卫峰 著

SDN

Interest, Strategy, Technology and Practice

电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

本书仅提供部分阅读，如需完整版，请联系QQ: 461573687

提供各种书籍pdf下载，如有需要，请联系 QQ: 461573687

PDF制作说明：

本人可以提供各种PDF电子书资料，计算机类，文学，艺术，设计，医学，理学，经济，金融，等等。质量都很清晰，而且每本100%都带书签和目录，方便读者阅读观看，只要您提供给我书的相关信息，一般我都能找到，如果您有需求，请联系我 QQ: 461573687, 或者 QQ: 2404062482。

本人已经帮助了上万人找到了他们需要的PDF，其实网上有很多PDF,大家如果在网上不到的话，可以联系我QQ。因PDF电子书都有版权，请不要随意传播，最近pdf也越来越难做了，希望大家尊重下个人劳动，谢谢！

备用QQ:2404062482

●● 专家力荐 ●●

此书写得非常不错，深入浅出地介绍了SDN技术的发展历程、总体架构、工作原理，是从事网络演进研究和开发的很实用的好书。

中国移动通信研究院院长 黄晓庆 (@黄晓庆)

SDN是网络行业的一次“变革”，数据和控制的分离打破了传统网络厂商的封闭格局，使得更多的玩家可以按照自身的需求和理解来实现网络。本书作者作为中国SDN的领导厂商盛科的高级研究人员，根据自身丰富的从业经验和对行业的深入见解，编写了一本SDN专著，为我们解读了SDN的发展历程、技术细节、行业发展和未来趋势，必将为SDN在中国的发展起到很好的推动作用。软件定义网络，软件定义世界。

UCloud云计算 CEO 季昕华 (@季昕华)

这是中国市场上第一本真正站在产业高度全面解析SDN的书籍，唯有浸淫网络产业核心领域多年沉淀的资深功力和专业功底，才能融渊博翔实和缜密逻辑于轻松笔调中，全面又深入地解析SDN领域的标准利益纠葛、技术路线之争、厂商战略博弈、最佳实践总结，从市场到技术全面生动地呈现SDN的前生今世。好书一本难得，读来令人喜悦。

计算机世界副社长 张群英 (@葑菲女郎花)

卫峰是我们SDN QQ群的发起人之一，并且非常热心地为大家授道解惑。也正是因为卫峰在数据通信特别是SDN领域有深厚的技术和实践积累，才能通过朴实的语言和易懂的案例给大家娓娓道来SDN的故事并整理成书。看此书如同看袁腾飞老师讲历史课一般，让阅读者轻松理解并得到许多启发。

SDN联合播报(www.sdnapp.com) 创办人 吴应辉 (@SDNAP)

卫峰从多维度分解SDN的历史、标准来由、产业链，深刻分析软件定义网络不只是软件+网络，这是本不可多得的网络教科书。

阿里巴巴网络架构师 庞俊英 (@KPang)

上架建议：计算机网络



@博文视点Broadview



责任编辑：董英
封面设计：侯士卿

ISBN 978-7-121-21821-7



定价：59.00元

深度解析SDN

利益、战略、技术、实践

张卫峰 著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

SDN 是当前的热点技术,但是由于 SDN 技术相对还比较新,而且本身内涵并不清晰,所以极易产生误解,不仅初学者不太容易把握,即便是已经接触过一段时间的人,也仍然会造成混淆。

本书用通俗易懂的语言深入浅出地介绍 SDN 的概念本质,SDN 架构,产生原因,发展历史,各种对 SDN 的误解,SDN 对产业的影响和发展趋势预测,各种标准组织及企业的动机和利益诉求,各个公司的 SDN 战略、SDN 产品、在产业链中的位置,一些热门技术(如网络虚拟化、NFV、云计算等)跟 SDN 的关系,SDN 的热门技术 OpenFlow 的分析以及 OpenFlow 所面临的各种挑战和尝试,SDN Controller(控制器)的介绍,经过实践检验过的多个应用案例分析。由于作者具有很强的芯片公司的从业背景,对 SDN 转发面也有深入的分析。

本书内容涵盖范围较广,从战略、内幕、利益、技术到实践全部覆盖,对设备商、云服务提供商、数据中心、企业 IT 运维人员、科研工作者等多个领域的从业人员了解 SDN 都会大有裨益。希望看过本书之后,读者能够对 SDN 有一个全面而又深刻的理解。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

深度解析 SDN: 利益、战略、技术、实践 / 张卫峰著. —北京: 电子工业出版社, 2014.1
ISBN 978-7-121-21821-7

I. ①深… II. ①张… III. ①计算机网络—网络结构 IV. ①TP393.02

中国版本图书馆 CIP 数据核字(2013)第 264797 号

责任编辑: 董 英

印 刷: 北京中新伟业印刷有限公司

装 订: 北京中新伟业印刷有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×980 1/16 印张: 14.5 字数: 278 千字

印 次: 2014 年 1 月第 1 次印刷

印 数: 4000 册 定价: 59.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

推荐序 1

SDN（Software Defined Network）的浪潮在逐渐影响着数据中心、运营商，以及越来越多的企业、互联网公司。

SDN 的来龙去脉是什么？为什么工业界需要 SDN？其背后的技术背景、相关公司、非营利化标准组织、商业利益集团在 SDN 面前的策略是什么？另外，SDN 领域出现了许多分支，其各自布局和未来发展方向如何？

这些问题都是相关领域的国家技术发展政策研究人员、企业 CIO / CTO、高级研发人员、高校研究人员必须了解和能够回答的问题。

卫峰的这本《深度解析 SDN—利益、战略、技术、实践》非常好地深入浅出地解释了上述各种问题，是一本非常难得的好书。

SDN 相关的技术书籍已经存在一些，各有亮点。卫峰的这本 SDN 书能够从细节着手，也能站在宏观和产业界的角度来思考 SDN 的应用场景、技术和政策走向，是非常难能可贵的。

拿到该书的样章后，我一口气读完了所有的章节，深感学到了许多有价值的知识。特别是卫峰对网络虚拟化的各种演变做了许多独到的分析和阐述，体现了他在 SDN 领域非常扎实的技术功底，以及他对产业界风云变幻的精确把握，对此我深感赞许。

强烈推荐计算机网络、系统和相关专业的研发人员阅读此书，也强烈建议国家相关政策研究人员阅读此书。

陈怀临

《弯曲评论》创办人

www.valleytalk.org

2013 年 10 月 25 日于硅谷

推荐序 2

SDN 现在是一个大热的话题，目前看来它有引起一场网络革命的趋势，无论在企业网、互联网数据中心还是运营商网络。所有 IT 相关的研发人员、运维人员、市场销售人员以及公司决策人员都有必要深入了解 SDN 是什么、它能给企业带来什么、应该如何去面对 SDN 所带来的变化。

SDN 是把“双刃剑”，一方面它能够提升管道价值，降低流量成本，加速网络创新；另一方面，集中控制给整个网络的安全性、可靠性、扩展性也带来了巨大的挑战。如何用好这把“双刃剑”是我们应当认真思考的问题。

作为运营商，我们也在密切关注 SDN 的发展，并投入了很多人力资源去研究、实验 SDN 网络，同时积极推动相关标准的建立。目前看来，SDN 在运营商网络中大有用武之地。SuperPTN 就是这样的一个研究项目，并且有可能在不久的将来落地。

因为 SuperPTN 这个项目，我接触到了该书的作者张卫峰，并有机会阅读了卫峰写的《深度解析 SDN—利益、战略、技术、实践》一书。读下来第一个感觉是通俗易懂，相信稍微了解一点网络的人都可以读懂。第二个感觉就是作者对网络特别是 SDN 的认识非常深刻，非浸淫网络多年不足以写出这样的书。更难得的是，本书有很多作者自己独到的观点，这些观点完全体现了作者作为一个芯片

设计厂商、SDN 技术领先厂商的研发人员多年的技术功底。

本书还有一个很吸引人的地方，就是在第 7 章里面列出来的 SDN 应用案例，都是真实网络中部署的实际案例，很多都是作者亲身经历的，读来感觉真实可信，极有说服力。

希望有更多人能看到本书，及时拥抱变化，迎接 SDN 对网络的改变。

段晓东

中国移动研究院网络研究所所长

2013 年 11 月

前言

这本书的写作是因为一个偶然的的机会。2013 年 5 月，CSDN 的编辑在新浪微博上给我发私信，希望我为他们写一篇 SDN 科普型的文章。我觉得这是个很不错的的事情，就写了一篇 6 千多字的文章，发表在《程序员》6 月刊上。这篇文章系统地介绍了 SDN 的起源、发展、概念定义、跟一些技术之间的关系、它面临的挑战、对行业的冲击等。7 月初，又收到电子工业出版社编辑的私信，希望我在文章的基础上，进一步充实和扩展，写一本系统介绍 SDN 的书。

写书？说实话，在这之前我还真的从来没想到过，尽管我经常在微博上发 SDN 相关的技术文章，所以我的第一反应就是拒绝，没时间。后来出版社编辑跟我深入分析了现在市场 SDN 教育书籍的缺位和大家对这方面的强烈需求，我当天就改变了主意，Why not？SDN 的很多内容都装在我的脑子里面，与其我每天在微博上东讲一点，西扯一点，一个一个地去纠正一些对 SDN 的误解，不如系统地把它写出来，让更多的人看到。于是，这本书就诞生了。

在写作的过程中，我越来越觉得这个题材非常有意义，原因如下。

第一，SDN 是一种革命性的技术，它对整个网络甚至基于网络的很多应用都有深远的影响，而且它的影响是全方位的，它对设备商、运营商、互联网公司、软件提供商、企业用户甚至个人用户都有影响。而这样一种影响如此广泛的技术，

介绍它的书籍特别是中文书籍基本没有，网上虽然有很多文章，但是都不够系统。

第二，SDN 是一个框架，而不是一个具体的协议，而且跟其他技术相比，SDN 是一种由用户驱动的技术，这就决定了它的需求基础旺盛而技术基础薄弱，还远没达到成熟的地步。所以对它的误解很多，对它的质疑很多，对它的实现方式也很多，对于网络基础薄弱的大众来说，就会有些无所适从，不知道它为什么诞生，不知道它的架构是怎么样的，也不知道它背后的利益关系是怎样的，更不知道它会走向何方。甚至在那些对 SDN 已经有过较多研究的人群中，也仍然会有激烈的争论，更不用说普通大众了。所以，我觉得有必要有一本书系统地介绍一下，就算不能解决所有的疑问，至少让更多的人了解一些真相，为大家进一步深入研究和讨论奠定基础。

这本书的写作过程很快，三个月不到就写完了，之所以能这么快写完，是因为大多数内容都在我的脑子里，无须到处去查阅，无须费尽心思去编造。除了一些事实性介绍，基本上是一气呵成，而这一切都要归功于我所在的公司——盛科网络。盛科是国内 SDN 领域的领先厂商，有大量 SDN 客户，所以我有幸能够了解很多别人了解不到的各个领域的 SDN 实际部署案例。盛科很早就加入了 SDN 领域最有影响力的 ONF 标准组织，所以我有幸能够了解很多 ONF 的内部资料，还有机会跟 ONF 标准组织的大佬们通过 E-mail 或者面对面交流，了解到 ONF 的一些真实想法和未来动态。同时盛科又是世界上仅有的几家交换芯片厂商，还是 ONF 的芯片顾问委员会（CAB）的成员，并且还做了世界上第一个针对 SDN 进行优化的 ASIC 芯片，而我也参与了历代所有芯片的设计工作，所以我对整个 SDN/OpenFlow 的转发面的现状、存在的问题都了如指掌，也比较清楚芯片界对未来的转发面走向的一些看法。同时，十几年的数据通信生涯使我对整个数据通信行业的认识也比较深入。以上种种，都为我写这本书提供了极大的便利，这也

是我写这本书的最大优势吧！

本书最大的特点是什么？我的责任编辑看完本书之后，在 QQ 上发给我的第一句话就是“你写得太通俗了，连我都看懂了”，那我的目的就达到了。我看书的时候，一向都痛恨过于学术化的论述，己所不欲勿施于人，自然，我会刻意避免学术化表达。这在“SDN 真实应用案例分析”一章体现得特别淋漓尽致，里面的所有案例全部都是实践总结而非理论分析。

除此之外，考虑到 SDN 还远未到成熟阶段，当前对 SDN 的“**What**”、“**Why**”、“**Where**”、“**How**”等问题的研究比具体的技术分析重要得多，所以本书把重点放在下面几个方面，整体组织思路如下：

- 第 1 章澄清 SDN 的概念内涵，列举一些流传较广的误解并纠正，讲述 SDN 的发展历史。
- 第 2 章解释为什么我们需要 SDN，现在网络碰到了什么样的挑战，为什么现有架构解决不了，以及 SDN 能带来什么好处，并用两个案例来分别说明传统网络的问题和 SDN 的好处。
- 第 3 章详细介绍各个标准组织，他们想要达到的目的，分析隐藏在各个标准组织背后的利益纷争，看看屁股是如何决定脑袋的。
- 第 4 章详细分析 OpenFlow 标准，OpenFlow 在转发面和控制面所面临的挑战，芯片转发面的各种尝试和趋向，Controller（控制器）分析和介绍。
- 第 5 章介绍网络虚拟化的概念、历史和现状，分析 SDN 在网络虚拟化和云计算中的地位和作用。
- 第 6 章详细介绍 30 多个公司的 SDN 战略、产品和方案，包括传统厂商、

新型创业厂商和互联网公司，同时分析一些比较有影响力的 SDN 相关的收购，以进一步了解大公司的 SDN 战略和意图。

- 第 7 章分析了多个真实发生在不同领域的 SDN 应用案例，从中了解客户的真实需求及 SDN 在其中的作用。
- 第 8 章深入分析 SDN 对网络产业格局的影响，以及 SDN 未来发展方向的预测。

这本书能顺利写完，首先要感谢我的家人对我的支持，儿子三天两头地对我说的两句话就是：“爸爸你的书写完了吗？”“爸爸你赶紧写书吧！”其次要感谢我的老板@盛科孙剑勇对我的支持，是他鼓励我写这本书，而且在写完之后帮我评审了一遍并给出了很多很好的建议。还要感谢我的同事@盛科_朱坚和@弓长东亚，我从他们那里学到了很多。同时感谢我的编辑董英（@英子 DD），为我的写作提供了很多分析建议，表现出了优秀的专业精神，使这本书得以顺利出版。最后要感谢 SDNAP 的创办者吴总（@SDNAP），感谢他提供了一个很好的 SDN 网站和 SDN 交流平台（QQ 群：279796875），在跟大家的交流中了解到了很多信息。

由于作者水平有限并且 SDN 本身争议很大，本书难免会有错误和疏漏，肯定也会有读者对其中的观点不赞成，您可以通过新浪微博（@盛科张卫峰）或者 QQ（67278439）跟我进一步交流探讨。

希望这本书能对读者有所帮助，这是我的最大心愿。

张卫峰

2013 年 12 月

九载耕耘奠定专业地位

博文视点诚邀精锐作者加盟

以书为证彰显卓越品质

《代码大全》、《Windows内核情景分析》、《加密与解密》、《编程之美》、《VC++深入详解》、《SEO实战密码》、《PPT演义》……

“圣经”级图书光耀夺目,被无数读者朋友奉为案头手册传世经典。

潘爱民、毛德操、张亚勤、张宏江、咎辉Zac、李刚、曹江华……

“明星”级作者济济一堂,他们的名字熠熠生辉,与IT业的蓬勃发展紧密相连。

九年的开拓、探索和励精图治,成就博古通今、文圆质方、视角独特、点石成金之计算机图书的风向标杆:博文视点。

“凤翱翔于千仞兮,非梧不栖”,博文视点欢迎更多才华横溢、锐意创新的作者朋友加盟,与大师并列于IT专业出版之巅峰。

英雄帖

江湖风云起,代有才人出。

IT界群雄并起,逐鹿中原。

博文视点诚邀天下技术英豪加入,

指点江山,激扬文字

传播信息技术,分享IT心得

专业的作者服务

博文视点自成立以来一直专注于IT专业技术图书的出版,拥有丰富的与技术图书作者合作的经验,并参照IT技术图书的特点,打造了一支高效运转、富有服务意识的编辑出版团队。我们始终坚持:

善待作者——我们会把出版流程整理得清晰简明,为作者提供优厚的稿酬服务,解除作者的顾虑,安心写作,展现出最好的作品。

尊重作者——我们尊重每一位作者的技术实力和生活习惯,并会参照作者实际的工作、生活节奏,量身制定写作计划,确保合作顺利进行。

提升作者——我们打造精品图书,更要打造知名作者。博文视点致力于通过图书提升作者的个人品牌和技术影响力,为作者的事业开拓带来更多的机会。



联系我们

博文视点官网: <http://www.broadview.com.cn>

新浪官方微博: <http://weibo.com/broadviewbj>

投稿电话: 010-51260888 88254368

CSDN官方博客: <http://blog.csdn.net/broadview2006/>

腾讯官方微博: <http://t.qq.com/bowenshidian>

投稿邮箱: jsj@phei.com.cn

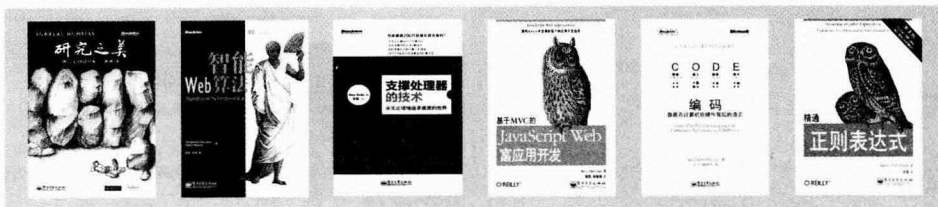
关于本书用纸的温馨提示

亲爱的读者朋友: 您所拿到的这本书使用的是 **环保轻型纸**!

环保轻型纸在制造过程中添加化学漂白剂较少,颜色更接近于自然状态,具有纸质轻柔、光反射率低、保护读者视力等优点,其成本略高于胶版纸。为给您带来更好的阅读体验并与读者共同支持环保,我们在没有提高图书定价的前提下,使用这种纸张。愿我们共同分享纸质图书的阅读乐趣!

博文视点精品图书展台

专业典藏



移动开发



物联网



云计算



数据库



Web 开发



程序设计



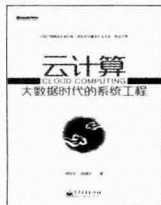
办公精品



网络营销



博文视点本季最新最热图书



《云计算：大数据时代的系统工程》

姚宏宇 田潮宁 著

定价：32.00元

工程院院士倪光南，用友软件董事长王文京
联合力荐！



《超越想象—— Windows 8应用设计与开发》

朱宏，元光宇，李孟晓编著。

定价：59.00

第一本 Windows 8 设计开发的原创著作用代码
创造神奇，用应用创造财富微软官方写作，内容详
尽清晰



《高质量程序设计指南：C++/C语言》

林锐，韩永泉编著

定价：65.00元

《大学十年》感动一代 IT 人，林锐亲述多年一线经验。
本书提供内建高质量代码必须熟练掌握的编程技术与
规范。
“高质量”试图挽回的是——投入大量人、材、物力的
事后检测和补救。

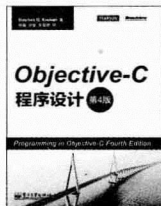


《支撑处理器的技术 ——永无止境地追求速度的世界》

[美]海撒安藤著；李剑译

定价：69.00元

为程序员打开处理器黑盒，深入理解构造与原理。



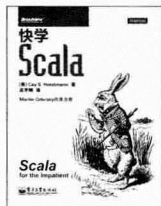
《Objective-C 程序设计(第4版)》 (全球最畅销Objective-C编程书籍)

[美]Stephen G. Kochan (斯蒂芬·G·科昌) 著

林冀 范俊 朱奕欣 译

定价：89.00元

这是一本 Objective-C 编程领域最畅销的书籍，内
容涵盖 Xcode 4.2 和自动引用计数 (ARC)。



《快学Scala》 (Core Java作者Horstmann最新力作)

[美]霍斯曼 (Horstmann, C.S.) 著

高宇翔 译

定价：79.00元

Scala 是一门以 Java 虚拟机 (JVM) 为目标运行环
境并将面向对象和函数式编程语言的最佳特性结
合在一起的编程语言。你可以使用 Scala 编写出更
加精简的程序，同时充分利用并发的威力。



《Android系统源代码情景分析》

罗升阳 著

定价：109.00元 (含光盘1张)

全面、深入、细致地掌握 Android，引领移动互联网
新时代！

在内容上，本书结合使用情景，全面、深入、细致地
分析了 Android 系统的源代码，涉及到 Linux 内核
层、硬件抽象层 (HAL)、运行时库层 (Runtime)、应
用程序框架层 (Application Framework) 以及应用
程序层 (Application)。



《x86/x64体系探索及编程》

邓志著

定价：119.00元

对 x86 处理器介绍得最详尽又最具实践指导意义
的一本书。

本书是对 Intel 手册所述处理器架构的探索和论证。
全书共五大部分，从多个方面对处理器架构相关的
知识进行了梳理介绍。书中每个章节都有相应的测
试实验，所运行的实验例子都可以在真实的机器上
执行。



《大道至简： 软件实践者的思想(典藏版)》

周爱民 著

定价：49.00元

本书提出了审视软件工程的全新视角和软件工程的
体系模型 (EHM，软件工程层次模型)。本书用非
工程的方式重新解析软件工程现象，全面、细致而
深刻地分析了工程中各个环节的由来、价值及其内
在关系。



《项目中的.NET》

李天平 编著

定价：108.00元

本书别无它用，只能用来求取、做项目！

直指 .NET 核心优势，以商业实战指导新手入行！

目 录

第 1 章 认识 SDN	1
1.1 什么是 SDN	1
1.2 SDN 不是什么	6
1.3 SDN 架构	7
1.4 SDN 发展历史	10
1.5 对 SDN 的误解	16
1.6 不该被遗忘的 SDN 先烈	19
第 2 章 我们为什么需要 SDN	21
2.1 网络业务发展趋势	21
2.2 传统网络碰到了瓶颈	24
2.3 SDN 如何来解决这些问题	25
2.4 SDN 适用的网络	27
2.5 让我们来看两个案例	28
2.5.1 案例一：使用传统交换机的案例	29
2.5.2 案例二：使用 OpenFlow 交换机的案例	31
第 3 章 SDN 相关组织及利益纷争	35
3.1 ONF（Open Networking Foundation）	35
3.1.1 ONF 性质和职责	35
3.1.2 ONF 组织结构	37

3.1.3	ONF Plugfest	38
3.1.4	ONS (Open Networking Summit)	41
3.2	ODL (OpenDayLight)	42
3.2.1	ODL 诞生的原因分析	42
3.2.2	ODL 主要目标介绍和分析	45
3.2.3	ONF 和 ODL 的比较	46
3.2.4	ODL 和 ONF 的利益冲突	47
3.2.5	ODL 内部利益之争及 Big Switch 的退出事件	48
3.2.6	OpenDayLight 系统架构	51
3.3	OCP (Open Computer Project)	53
3.4	NFV (Network Function Virtualization)	55
3.4.1	NFV 性质	55
3.4.2	NFV 主要的工作目标	56
3.4.3	NFV 所带来的好处	58
3.4.4	NFV 跟 SDN 的关系	58
3.4.5	NFV 目前进展	60
3.5	ONRC	61
3.6	IETF	62
3.6.1	Software Driven Network	62
3.6.2	I2RS	63
第 4 章	详解 OpenFlow	65
4.1	OpenFlow 标准分析	67
4.1.1	交换机转发面详解	67
4.1.2	Controller 和交换机之间消息	75
4.1.3	OpenFlow Channel	78
4.1.4	Controller 角色和选举	78
4.1.5	OpenFlow 系统性能指标	79
4.1.6	OpenFlow Controller 和交换机工作流程	80

4.1.7 对 OpenFlow 标准的总结	81
4.2 OF-Config 介绍	81
4.3 Controller (控制器)	84
4.3.1 Controller 架构介绍	84
4.3.2 Onix 分布式 Controller 模型	86
4.3.3 FloodLight	87
4.3.4 Ryu	88
4.3.5 NOX/POX	88
4.3.6 Trema	89
4.4 OpenFlow 所面临的挑战	89
4.4.1 OpenFlow 控制面的挑战	89
4.4.2 OpenFlow 转发面的挑战	91
4.4.3 芯片厂商的犹豫	94
4.5 OpenFlow 转发面的各种尝试和创新	99
4.5.1 NPU 和 FPGA 方案	99
4.5.2 TTP/NDM 方案	101
4.5.3 ONF 心目中的理想方案	104
4.5.4 芯片厂商的折中方案	109
4.6 OpenFlow 非技术面的阻力	111
4.7 Hybrid 交换机	113
第 5 章 网络虚拟化和 SDN	115
5.1 什么是虚拟化	115
5.2 服务器虚拟化	116
5.3 网络虚拟化	118
5.3.1 什么是网络虚拟化	118
5.3.2 网络虚拟化的价值	121
5.3.3 网络虚拟化的战争	122
5.3.4 网络虚拟化中的三种 Tunnel 技术比较	123

5.3.5 网络虚拟化的三种组网方案	126
5.4 云计算跟网络虚拟化的结合	128
5.5 SDN 在网络虚拟化和云计算中的作用	129
第 6 章 各公司的 SDN 战略及解决方案	130
6.1 元老派厂商	130
6.1.1 Arista	130
6.1.2 Brocade	131
6.1.3 Citrix	132
6.1.4 Cisco	133
6.1.5 Dell	134
6.1.6 Ericsson	136
6.1.7 Extreme	136
6.1.8 HP (H3C)	137
6.1.9 HuaWei	138
6.1.10 IBM	140
6.1.11 Juniper	141
6.1.12 NEC	143
6.1.13 Radware	144
6.1.14 RedHat	145
6.1.15 VMware	145
6.2 新生代厂商	146
6.2.1 Big Switch	146
6.2.2 Centec	148
6.2.3 ConteXtream	150
6.2.4 Cumulus	151
6.2.5 Embrane	153
6.2.6 Midokura	154
6.2.7 Nicira	155

6.2.8	NoviFlow	156
6.2.9	Nuage	157
6.2.10	Pica8	158
6.2.11	Plexxi	160
6.2.12	PlumGrid	161
6.2.13	Vello Systems	162
6.3	互联网/云服务公司	164
6.3.1	Google	164
6.3.2	Facebook	164
6.3.3	Amazon	165
6.3.4	MicroSoft	165
6.3.5	百度/阿里/腾讯	166
6.4	跟 SDN 相关的收购	167
6.4.1	Intel 收购 Fulcrum Microsystems	167
6.4.2	VMware 收购 Nicira	168
6.4.3	Oracle 收购 Xsigo System	168
6.4.4	Cisco 收购 vCider	169
6.4.5	Brocade 收购 Vyatta	170
6.4.6	Juniper 收购 Contrail	170
6.4.7	F5 收购 LineRate	171
6.4.8	Cisco 收购 Insieme	172
第 7 章	SDN 真实应用案例分析	173
7.1	用 SDN 改造 Google WAN 网络	173
7.1.1	背景介绍	174
7.1.2	具体实现	176
7.1.3	B4 网络改造项目总结	179
7.2	ADVA 基于 SDN 的虚拟光传输网络	181
7.3	SDN 跟网络虚拟化的完美结合	184



7.4	用 SDN 设备做安全分析和负载均衡	187
7.5	运营商基于 SDN 的 SuperPTN	189
7.6	Hybrid 交换机促进传统网络跟 SDN 的融合	194
7.7	SDN 在无线接入领域的应用	195
第 8 章	SDN 对行业的影响和发展趋势	198
8.1	SDN 对网络产业的影响	198
8.2	产业链利益分析	201
8.2.1	大型设备商	202
8.2.2	中小设备商和创业型设备商	202
8.2.3	软件提供商（包括个人）	203
8.2.4	大中型芯片厂商	203
8.2.5	小芯片商和新兴芯片厂商	203
8.2.6	设备用户	204
8.3	SDN 未来发展方向分析	205
8.3.1	SDN 必将兴起，但不会一统江湖	205
8.3.2	南向接口会开放化，但不会标准化	206
8.3.3	北向接口将百花齐放，特定场景可能标准化	206
8.3.4	OpenFlow 不会消亡，但仅是 SDN 的一部分	207
8.3.5	大型设备商仍会有优势，但是影响力弱化	207
8.3.6	没有一个 Controller 可以一统天下	208
8.3.7	转发表会有 OpenFlow 优化，但是不会有纯 OpenFlow ASIC 芯片	209
8.3.8	白牌设备将大发展	209
8.3.9	厂商锁定问题只会缓解，无法根除	210
8.3.10	开源趋势势不可挡	210
8.4	总结	211
8.5	SDN is dead, Long live SDN!	212
附录 A	跟 SDN 相关的网站	213

第 1 章

认识 SDN

1.1 什么是 SDN

SDN 是 Software Defined Network 的缩写，译成中文就是软件定义网络。当然，笔者如果就只是这样简单来解释一下 SDN，估计会被骂得狗血淋头，Google/百度谁不会用啊？讲英文翻译的话这几个单词初中生都能翻译出来。

在 2013 年的 ONS 大会上，有人开玩笑说 SDN = Stanford Defined Network，或 Sexy Defined Network，或 Still Don't kNow。玩笑的背后隐藏的含义就是，大家对 SDN 的理解不太一致，并且对 SDN 的前景看法也不一致。

笔者在十几年的通信技术职业生涯中，总结出了一条快速学习新技术的经验，在 IT 技术领域，特别是通信领域，对一项新技术的命名是非常严谨的，很多技术的名字都起得非常精准到位，如果能把该技术的名字领悟透彻，对理解该技术帮助会很大，比如 VLAN、STP、OSPF、MPLS 等都是其中的典型代表，能够绝佳地体现这些技术的内涵精髓。笔者非常喜欢在研究一项新技术的时候，先

把该技术的名字研究清楚，SDN 也不例外。

网上关于 SDN 的争论铺天盖地，褒贬不一。在质疑 SDN 的声音中：

- 最常听到的就是——“我为什么需要 SDN？有什么是 SDN 能做到而传统网络做不到的？”
- 具体到转发面，通常是——“SDN 交换机本质上也不过是把传统交换芯片包装一下而已，有什么转发行为是传统交换机支持不了的？”
- 具体到控制面，通常是——“传统交换机是用网管软件或者命令行来进行管理的，SDN 交换机无非是换成 Controller（控制器）来管理而已，又有什么本质区别？”

这里我们暂且不讨论为什么需要 SDN，这是后面章节的话题。就这些质疑，笔者想说的是，提这些问题的人并没有真正理解 **Software Defined** 的意思。没错，就转发面而言，SDN 交换机跟传统交换机并无本质区别，也许以后慢慢会进行芯片技术创新来更好地支持 SDN，但是本质并无不同。而控制面，也没错，无论是 SDN 交换机，还是传统交换机，都有一大堆软件在支撑、在管理。但是 **Software Defined Network** 跟传统网络是有本质区别的。

传统网络的转发行为，是受各种网络协议控制的，尽管它们也间接地反映了管理员的意志，但是：

第一，它们是逐设备单独控制的，是纯分布式控制。

第二，控制面跟转发面在同一个设备中，紧密耦合。

第三，管理员无法直接操控转发行为（管理员配置网络协议，网络协议通过自身的运行再去影响转发行为，管理员无法改变协议本身的行为）。

第四，网络协议对转发行为的影响是有固定模式的，比如路由协议只能靠目的 IP 地址来进行转发，MPLS 协议只能靠 MPLS Label 来进行转发，且不同情况下的转发只能对报文进行固定模式的修改。比如路由转发的时候只能修改报文的 MacDa、MacSa、VlanTag、TTL、DSCP，改不了其他东西，且修改的方式也是固定的，比如 TTL 通常只能减一。

而 SDN 呢？它要求集中式控制（也有分布式，但是那是先集中再分布，后面会详细讲到），要求转发跟控制分离，要求管理员可以直接操控设备的转发行为，可以不用通过各种网络协议（但是 SDN 并没有拒绝使用动态协议，只不过这些协议是要运行在 Controller 上的，而不是运行在设备上的），而是直接通过应用程序（也就是说 Software）来控制转发行为，而且这种控制是直截了当的，不受任何协议影响的，比如管理员不希望仅通过“目的 IP”来转发而是希望看“目的 IP+源 IP”，转发的时候直接修改报文的“目的 IP”等。

总结一下就是，传统网络虽然也有大量软件参与，但是这些软件不是网络的使用者设计的（是设备提供商写死在设备中的），无法完全让管理员随心所欲地体现自己的意志，管理员不能清楚地知道发生了什么事情，自然我们就不能说这些软件（协议）定义了这个网络。而 SDN 中的软件可以完全是管理员所在的组织设计的，可以让管理员随心所欲地利用这些软件来规划自己网络中的任意转发行为，管理员完全通过这些软件来定义了整个网络，也就是所谓的“Software Defined Network”。

说到这里，有人会问：我明白传统网络中的软件，是指各种协议和网管，那么 SDN 中的软件到底是什么？其实就是各种应用程序（Application），比如一套视频监控的管理软件，这个软件通过图形界面操作来控制交换机，规定哪个摄像头的数据要发送到哪个服务器去，走哪条路径。又比如一个防火墙控制软件，这个软件通过图形化操作界面来控制交换机，规定符合什么条件的数据报文要被送到监控服务器去进行深度分析。这些软件不一定都是要管理员去操作了才起作用，也可以是根据管理员预先配置好的策略动态去配置防火墙，比如某个应用程序设了个定时器，定时去让防火墙启动一个过滤器。还有更复杂的应用，比如网络虚拟化的自动化部署软件、流量工程中的自动化流量测试、路径规划软件等。这些软件有一个共同特点，它们可以不依赖于任何路由交换协议，而且是网络的用户可以定制设计的，可以用来实现自动化控制的，完全体现用户意志。从这个意义上说，SDN 可以有一个等价的名字，也许这个名字更贴切，叫作 ADN（Application Defined Network）。事实上，已经有人在提这个名字了。如果你看到有人在跟你吹嘘 ADN 的概念，不要被忽悠了，本质上其实就是 SDN，谁能说 Application 不是 Software 呢？

SDN 并不是一个具体的技术（相比较而言 OSPF、BGP、MPLS、Trill 等都是一个具体的技术），它其实是一个框架，一种网络设计的理念。SDN 框架中的网络，控制面和转发面必须是分离的，在转发面这一角度，它希望是协议行为无关的（尽管并不是所有人都这么认为），管理员的意志最重要，软件可以完全地体现管理员的意志，管理员通过它来控制转发行为，驱动整个网络流量。

除了上述的直接含义之外，SDN 还有一些其他的潜台词。最广为人知的就是对硬件转发面配置接口的标准化（网管术语中的南向接口），因为如果软件想随

心所欲地控制转发行为，就应该尽量不依赖于特定的硬件，否则软件就无法通用。还有一个潜台词则是集中控制，因为是应用程序定义的网络，该网络中的设备都需要受相关应用程序的统一控制。

另外还有一些跟 SDN 关系很紧密、但是并不必需的属性，比如硬件转发面的标准化（不是硬件配置接口，而是内部逻辑实现），比如开源等。对这些属性认知的不同，造成了不同标准组织和公司在行为上的差异，后面我们会详细介绍。

总结一下，目前一般比较认可的 SDN 的特征属性包括以下几点：

- 控制面与转发面分离。
- 开放的可编程接口。
- 集中化的网络控制。
- 网络业务的自动化应用程序控制。

其中前两点是 SDN 的核心属性，只要符合这两点，无论它具体用了什么样的实现技术，都可以宽泛地认为是 SDN 架构。理解到这一点，就可以抓住 SDN 的本质，可以在各种纷纭复杂的产品和方案宣传中透过现象看到本质。

注意，这里面并不包括硬件编程接口的标准化（但是必须要开放化），而硬件编程接口的标准化恰好是 OpenFlow 所刻意追求的方向（OpenFlow 不仅要求编程接口要标准化，它更进一步，要求硬件内部转发行为也要标准化）。这一点也是很多公司，特别是传统设备商跟 OpenFlow 的标准组织 ONF 之间严重的分歧之一。笔者认为，硬件编程接口甚至转发行为若能标准化，若能灵活可编程，肯定是好的，对网络的发展绝对是个助力。但是这不应该是现在就要追求的目标，现

在为了稳步推进 SDN 的发展,应该先把目光聚焦到上述 SDN 的几个本质属性上,那才是 SDN 的重点所在。至于最终能否做到硬件编程接口标准化,笔者对此不报什么期望,尽管笔者也认为它挺重要。至于硬件转发行为的标准化,笔者对它更是不报什么期望,而且也认为其实不是那么重要。

1.2 SDN 不是什么

跟 SDN 相关的技术很多,后面我们会一一介绍,在这里只是先来做一些它们关系上的澄清。SDN 就是 SDN,它不是其他任何技术。

SDN 不等于 OpenFlow, SDN 不等于网络虚拟化 (Network Virtualization), SDN 不等于网络功能虚拟化 (NFV, Network Function Virtualization), SDN 更不等于云计算或者大数据或者数据中心网络 (这一点相信很多人都能理解)。

但是更进一步,SDN 不意味着一定要用 OpenFlow,网络虚拟化/云计算/大数据/数据中心网络也不一定必须要用 SDN。

一些初窥当前热点技术门径的人,还会听说过其他一些词汇,什么 OpenStack, CloudStack, OVS, HyperVisor 等,SDN 跟它们也都没有任何直接关系。

SDN 也不是一个网络协议,不是一个网管工具,如前所述,它仅仅是一种网络架构的理念,它规划了网络的各个组成部分 (软件和硬件、转发面和控制面) 及相互之间的互动关系。

本书的重点是 SDN，虽然也涉及一些其他的技术，但是本书将不会花很多篇幅去介绍这些技术，要了解其他跟云计算、数据中心相关的技术，请参考别的资料，推荐徐立冰同学写的那本《腾云：云计算和大数据时代网络技术揭秘》。

1.3 SDN 架构

说了这么多什么是 SDN，不如用一张图来得直观。前面讲过了，SDN 不是一项具体的技术，而是一种网络设计理念，一种网络架构，那这里我们就图解一下 SDN 架构，请看图 1-1。

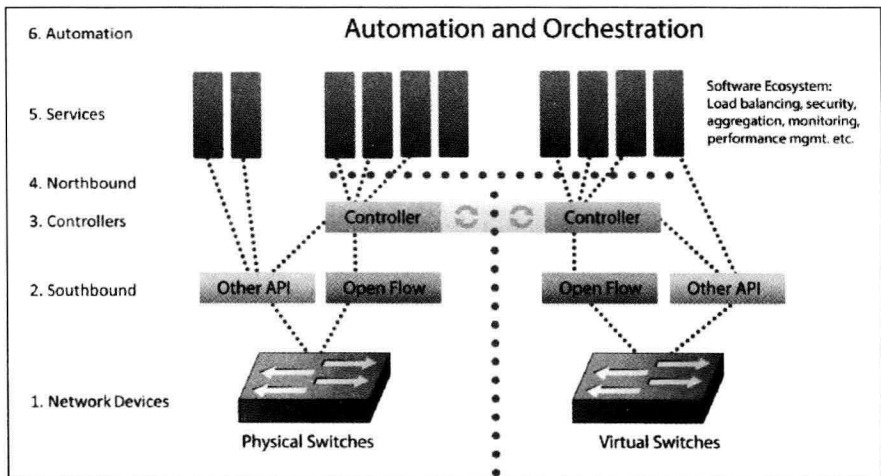


图 1-1

让我们具体来解释一下图 1-1。

1. 网络设备（Network Devices）

这里的网络设备其实可以抽象为转发面（Forwarding Plane 或者另外一个名

字 Data Plane), 它不一定是硬件交换机, 也可以是虚拟交换机, 比如 OVS, 当然也可以是别的物理设备, 比如路由器。所有的转发表项, 都存储在网络设备里面, 用户数据报文在这里面被处理、转发。

网络设备通过南向接口 **Southbound Interface** 接收 Controller 发过来的指令, 配置位于交换机内的转发表项, 并可以通过南向接口主动上报一些事件给 Controller。

2. 南向接口 (Southbound Interface)

南向和北向是传统网络中的术语, 这里被借用过来。南向接口是指控制面跟数据转发面之间的接口, 传统网络的南向接口并没有什么标准化, 而且都存在于各个设备商的私有代码中, 对外也不可见, 也就是说既不标准也不开放。而在 SDN 架构中, 希望南向接口是标准化的 (当然这是个理想, 未必能变成现实), 只有这样, 才能让软件摆脱硬件的约束, 尽可能地做到随心所欲, 做到应用为王, 否则 SDN 到最后还是特定软件只能在特定硬件上运行。

从第 2 层开始向上, 已经看不到硬件交换机和虚拟交换机的区别, 看到的只是抽象的转发面。

我们注意到图 1-1 中在这一层有 OpenFlow 和 Other API 两种接口。这是因为目前 OpenFlow 是最有影响力的南向接口标准, 但是并不是唯一的, 有些公司和组织并不买账, 准备或者已经另起炉灶。这对 SDN 的发展未必是坏事, 尽管我们都希望最终看到有一个唯一的, 大家都认可的标准出现, 一统江湖, 但是同样, 这是一种理想, 未必能实现, 不要对此期望太大, 后面我们会详细分析这个问题。

3. 控制器 (Controllers)

Controller 也就是中文里面说的控制器，一个 SDN 网络里面的 Controller 可以有多个。Controller 之间可以是主从关系（只能有一个主，可以有多个从），也可以是对等关系。一个 Controller 可以控制多台设备，一台设备也可以被多个 Controller 控制。通常 Controller 都是运行在一台独立的服务器上，比如一台 x86 的 Linux 服务器或者 Windows 服务器。

Controller 是 SDN 网络中的核心元素，是各个大公司都想抢占的制高点。因为它向上提供应用程序的编程接口，向下控制硬件设备，处于战略位置。从 iOS、Android 的火热程度大家就能知道为什么这些公司都要争夺 Controller 的市场。

4. 北向接口 (Northbound Interface)

传统网络里面，北向接口是指交换机控制面跟网管软件之间的接口，比如电信网络里面耳熟能详的 SNMP、TL1 等标准协议。在 SDN 架构中，它是指 Controller 跟应用程序之间的接口，目前该接口尚无标准化，这也是一些标准组织所想推进的事情。但是这个事情绝对要比南向接口复杂得多，因为转发面毕竟是万变不离其宗，更容易抽象出通用接口，而应用层面则变数太多。

5. 应用服务 (Services)

Services 也就是我们说的应用层面，之所以用 Service 而不是 Application 这个词，是因为 Service 比 Application 更能表达出网络的本质，是要为用户提供服务的，这里的 Service 就有很多了，包括 load balancing（负载均衡）、security（安全）、monitoring（网络运行情况监测）、performance management（包括拥塞、延时等网络性能的管理和检测）、LLDP（拓扑发现）等很多服务。这些服务最终都以软件应用程序的方式表现出来，代替传统的网管软件来对网络进行控制和管理。它们可以跟 Controller 位于同一台服务器上，也可以运行在别的服务器上通

过通信协议来跟 Controller 通信。

6. 自动化 (Automation)

自动化算不上一个层次，其实是对应用程序的封装和整合。它通常是跟 Orchestration 这个词一起出现的，甚至 Orchestration 比它出现的频率更高。本质上，两个词说的是同一回事，只是 Automation 是目的，Orchestration 是手段。Orchestration 是管弦乐、编排、和谐的结合的意思，实质上就是说把各种各样的技术糅合在一起，组成一个和谐一体的系统，共同来达到一个目的，就像管弦乐是把各种乐器组合在一起演奏出优美的曲子一样。这个 Orchestration 要达到的目的就是业务的自动化部署。

提到 SDN，很多人都以为是用管理员手工配置代替了动态网络协议。代替动态网络协议没错，但是并不意味着一定要手工配置，可以通过强大的软件应用，让软件来自动帮管理员做事情。举个最简单的例子，让软件定期读取设备链路负载情况，自动生成链路负载曲线图，这中间会涉及多个应用和服务，被整合在一个系统管理框架里面。更复杂的软件系统则是云计算平台，通过将各种应用融合在一起，通过 Controller 来对资源进行控制，最终达到自动化业务部署的目的，这个云计算平台就可以认为是一个 Orchestration 系统。

1.4 SDN 发展历史

在介绍 SDN 发展历史之前，先介绍一下 Clean Slate 项目。

从 1969 年 DARPA 试验网开始，Internet 已经走过了 40 多年。基于 Ethernet

和 TCP/IP 的 Internet, 由于它设计的松散性、简单性, 获得了巨大成功, 积累起了数量庞大的技术, 用于解决在不同时期出现的不同问题。

与此同时, Internet 也有很多与生俱来的缺陷, 主要集中在可扩展性、安全性、移动性和 QoS 上。这些问题早已经被发现, 技术专家们也发明了不少技术来尝试完全解决或者部分缓解这些问题, 如无类域间路由 CIDR、网络地址翻译 NAT、MPLS、Traffic Engineering 等, 这些技术在不同程度上帮助整个 Internet 及各种私有网络一路发展到今天。但是中国有一句俗语叫“按下葫芦起来瓢”, 不管这些添砖加瓦的机制发展得多么完美, 总是会有新问题出现, 而且越来越难以解决, 究其原因, 这些技术都是在现有网络框架内进行修补的。现在有越来越多的研究者相信, 重新定义网络架构也许是根本的解决方案, 尽管也有很多人认为新的架构也许会有新的问题——至少笔者认为, 肯定会有新的问题。

这种根本的方案, 学术界形象地称之为 Clean Slate 方案, 有点推倒重来的意思。这一次, 他们希望尽可能地考虑当前的各种需求及未来可能的需求。

Clean Slate 在互联网研究上有广义和狭义之分。广义上泛指各种各样的下一代网络(NGN)项目, 如美国 NSF(National Science Foundation)通过 FIND(Future Internet Network Design)计划在学术界和工业界推动的 GENI(Global Environment for Network Innovations), 欧盟在 FP7(Seventh Framework Programme)的 ICT 方向下资助的 FIRE(Future Internet Research and Experimentation), 以及日本国家信息通信技术所(NICT)资助的 AKARI 项目和对应的下一代试验床 JGN2+。狭义的 Clean Slate 则是由斯坦福大学 Nick McKeown 教授牵头的实验室研究计划。SDN 就诞生于这个狭义的 Clean Slate 计划中。

SDN 公认的发源地是美国斯坦福大学，它起源于校园网，发扬光大于数据中心。它的首创者是一个叫 Martin Casado 斯坦福研究生（确切地说是 OpenFlow 的首创者而不是 SDN 概念的首创者）。Casado 之前曾在一家不具名的美国情报机构工作过，该机构的网络被他认为是有史以来最安全的计算机网络。但该网络的问题在于这个网络过于复杂，无论是建设还是维护都异常困难，一旦想要改动网络，就会带来一连串的问题。这个事情给 Casado 的触动很大，他觉得现在的网络架构到了需要变革的地步。他举了个例子，改动一台计算机就得进行 8 项不同的配置变更，一不小心就会出错，所以基本上这样的网络搭建好之后什么都不能动。

网络设备制造商不允许对硬件进行重新编程，代码都是直接写进交换机或者路由器的。因为大型设备商不可能允许代码开源，况且就算开源，那些网络设备的代码都复杂无比，一般人根本就不敢改动。这就给用户改造和控制网络带来了很大麻烦，因为设备商提供的控制接口再灵活，也总有不能满足的需求。

所以 Martin Casado 到了斯坦福大学研究生院后就开始着手建设一种灵活的、能够像计算机一样可编程的网络。Casado 领导了一个关于网络安全与管理的科研项目 Ethane，并据此写了一片博士论文《一种名为 Ethane 的网络架构》。这个 Ethane 项目属于 Clean Slate 计划的一部分。

了解 Ethane 项目有助于了解后面的 OpenFlow 以及 SDN。这个项目是安全相关的，涉及一些安全策略。该项目试图通过一个集中式的控制器，让网络管理员可以方便地定义基于网络流的安全控制策略，并将这些安全策略应用到各种网络设备中，从而实现对整个网络通信的安全控制。其实 SDN 后续的发展也证明，安全领域是非常适合 SDN 部署的，因为安全领域本身强调配置管理而不是动态网络协议，也强调集中控制，这跟 SDN 的特点非常吻合。Casado 同学凭借这个

项目起步，慢慢地创下来一番事业，成了名副其实的高富帅（此是后话，暂且略过不表）。但是作为 SDN 的鼻祖级人物，我们有必要贴一下他的帅照（图 1-2），让大家瞻仰一下。



图 1-2

Casaso 的导师就是大名鼎鼎的 Nick McKeown。我们
有必要在这里先介绍一下 McKeown，介绍他不是为了猎
奇，而是埋下一个伏笔，让大家知道现在 McKeown 花了
很多时间在 OpenFlow 通用芯片模型设计上是有原因的，
并非不自量力。McKeown 在利兹大学拿到了本科学位，
然后在著名的 Hewlett-Packard 公司工作了三年多。之后又跑去读书，在加州大学
伯克利分校拿到了他的硕士和博士学位。他的博士论文题目是“Scheduling Cells
in an Input-Queued Cell Switch”，是关于 Fabric（交换网）芯片的。毕业后短暂地
在 Cisco 工作过一段时间，帮助 GSR 12000 路由器的系统架构设计（偏芯片）。
在 1997 年，他跟别人一起成立了 Abrizio 公司，他担任 CTO。该公司在 1999 年
以 4 亿美元的价格被 PMC-Sierra 公司收购。2003 年他又跟别人一起成立了 Nemo
Systems 公司，他担任 CEO，到了 2005 年该公司被 Cisco 以 1250 万美元收购。
McKeown 最广为人知的杰作是 2007 年跟 Martin Casado、Scott Shenker 一起成立
的公司 Nicira，该公司在 2012 年被 Vmware 公司以 12.6 亿
美元的天价收购，被收购的时候公司甚至都没有赢利。鉴于
McKeown 是个帅哥以及他在 SDN 领域无与伦比的影响力，
我们也给他上一张养眼帅照（图 1-3）。



图 1-3

McKeown 对 Casado 的这个项目很重视，给与了很多指
导。受此项目（及 Ethane 的前面的一个项目 Sane）启发，

Casado 和 McKeown(时任 Clean Slate 项目的 Faculty Director)发现,如果将 Ethane 的设计更一般化,将传统网络设备的数据转发 (data plane) 和路由控制 (control plane) 两个功能模块相分离,通过集中式的控制器 (Controller) 以标准化的接口对各种网络设备进行管理和配置,那么将为网络资源的设计、管理和使用提供更多的可能性,从而更容易推动网络的革新与发展。但是他们并没有一开始就提出 OpenFlow 或者 SDN 的概念,而是着手开发一个名叫 NOX 的 Controller,用来对网络中的交换机进行集中控制,OpenFlow 其实是 NOX 的一个副产品,因为他们在使用 NOX 对交换机进行集中控制的时候发现,如果每台交换机能够对 Controller 提供一个标准的统一接口,那么控制起来会很容易,于是 OpenFlow 就诞生了。后来 McKeown 等人于 2008 年在 ACM SIGCOMM 发表了题为 OpenFlow: Enabling Innovation in Campus Networks (该论文下载地址为 <http://ccr.sigcomm.org/online/files/p69-v38n2n-mckeown.pdf>) 的论文,首次详细地介绍了 OpenFlow 的概念。该论文除了阐述 OpenFlow 的工作原理外,还列举了 OpenFlow 几大应用场景,包括:

- 校园网络中对实验性通信协议的支持。
- 网络管理和访问控制。
- 网络隔离和 VLAN。
- 基于 WiFi 的移动网络。
- 非 IP 网络。
- 基于网络包的处理。

当然，目前关于 OpenFlow 的研究已经远远超出了这些领域。

基于 OpenFlow 为网络带来的可编程的特性，McKeown 和他的团队（包括加州大学伯克利分校的 Scott Shenker 教授）进一步提出了 SDN 的概念。不过也有人说 SDN 的概念最早是由 KateGreene 于 2009 年在 TechnologyReview 网站上评选年度十大前沿技术时提出的。SDN 入选 2009 年 TechnologyReview 十大前沿技术。图 1-4 是当年的 TechnologyReview 网站的截图，SDN 技术位列第 10 名。



图 1-4

从 SDN 诞生的历史我们就可以看出，SDN 最本质的特点就是控制跟转发的分离。本书后面章节会不断强调这一点，避免读者对 SDN 的理解进入误区（比如笔者就经常听到有人在讲没有标准的 OpenFlow 芯片，就没办法推行 SDN，这是不理解 SDN 的表现）。

斯坦福大学成立的一个 OpenFlow 特别工作小组负责撰写 OpenFlow 1.0 的标准，Casado 同学是其中的主力。值得一提的是，这里面还有另外一个叫 Guido Appenzeller 的助教（Consulting Assistant Professor），他继 Casado 成立 Nicira 公司之后，也成立了一个公司，就是现在也很有名气的 BigSwitch。按照现在

BigSwitch 官方网站的介绍，说他是 Clean Slate 实验室的负责人，带领一个团队开发了 OpenFlow 1.0 并开发了 OpenFlow 的参考交换机和 NOX Controller(head of the Clean Slate Lab where he led the research team that developed the OpenFlow v1.0 standard and the reference switch and Controller implementations)，真实情况如何，我们也不得而知。有意思的是，被 Juniper 以 1.76 亿美元收购的 Contrail 公司的 CEO Ankur Singla 也是斯坦福大学的学生，跟 Martin Casado 和 Guido Appenzeller 一起参与了 OpenFlow 项目（好像是负责 NetFPGA 的设计），关于 Contrail 公司以及本次收购，本书后面有详细介绍。该收购发生的时候，Contrail 甚至一毛钱都没赚，SDN 的火爆由此可见一斑。

很快 OpenFlow 技术引起了工业界的关注，很多公司相继参与进来，2011 年 3 月 21 日，德国电信、Facebook、Google、微软、Verizon、Yahoo!发起成立了 ONF (Open Networking Foundation) 组织，旨在推广 SDN，同时开始了 OpenFlow 的标准化工作。该组织陆续制定了 OpenFlow Specification 1.1、1.2、1.3、1.4 的标准（注意：OpenFlow 1.0 在 ONF 成立之前就已经完成了），目前仍在继续发展完善中。随之更多的公司开始加入这个组织，ONF 以及 SDN 的影响力迅速扩大。

很多公司都看到了其中的机会，也有很多公司看到了这场技术变革给自己带来的负面影响，伴随着利益之争，ONF 之外的一些组织成立了，战争才刚刚开始。这场战争对有些公司来说是进攻战，有些是积极防御战，还有一些则是被动防御战。

1.5 对 SDN 的误解

由于 SDN 目前还没有得到普及，仍处于生命前期，所以网上有很多对 SDN

的误解，这里我们做一下澄清。

误解一：SDN 一定要使用 OpenFlow 协议来配置转发面

这可以说是最经典的误解了，以至于虽然前面我们已经做过了澄清，这里仍然要把它拿出来，并且列为首位。OpenFlow 只是发展最早、目前影响力最大的南向接口，但是并不是唯一的，实际上，在写作本书的过程中，OpenDayLight 已经提出了另外一些南向接口，包括现存的以及新定义的，如 PCEP、NETCONF、SNMP 等，后面我们会详细讲到。

误解二：SDN 要求硬件转发面的标准化

这只是 OpenFlow 的要求，并不是 SDN 的要求。实际上，目前很多厂商，包括某些标准组织都没有刻意去追求硬件转发面的标准化，尽管有些人确实在做这个工作，但至少这不是所有人的共识。哪怕是 OpenFlow 的标准组织 ONF，目前也在考虑一种折中方案。

误解三：SDN 设备可以代替所有设备

至少目前看来，这是极端不靠谱的说法，就算最激进的 SDN 鼓吹者也不会说这样的话。在笔者看来，第一，SDN 并不是适合所有网络；第二，就算在适合的网络中，SDN 也不能替换所有层次的设备。

误解四：SDN 得到了所有厂商的支持

至少目前看来没有，有很多公司表面上看来挺积极，但是实际上他们是在被动前行而不是主动前进。举个不太恰当的例子，过年的时候很多人都去给领导送礼，其实这其中有相当一部分人认为送了也没什么用，但是如果不送，那就成异类了，落后了。很多厂商，特别是一些市场份额很大的设备商/芯片商就有这种心态。这很容易理解，因为他们的市场份额已经很大了，再折腾一次，也不会再大。但是他们不得不跟着折腾，因为万一别人做起来自己没跟上，市场份额就缩水了。

看清楚这一点，可以更好地去解读一些公司的市场行为和宣传。

误解五：SDN 是设备商发起的

这个问题不能说有误解，而是说很多人可能没去想过这个问题，但是这个问题很重要。Internet 发展的这 40 多年来，基本上绝大多数的高新技术都是设备制造商提出和引导的，包括 IETF、IEEE、ITU 的各种标准，但是 SDN 不同，SDN 的提出者和最初的推动者都是网络设备的用户而不是网络设备制造商（当然现在在利益驱动下，厂商都参与进来并开始引领潮流）。这一点非常重要，我们后面会多次提到这个问题，这决定了利益导向。

误解六：SDN 主要用在数据中心网络

应该说，SDN 之所以能迅速崛起，主要的驱动力来自于数据中心，更准确地说，是来自于数据中心中的网络虚拟化。如果没有网络虚拟化，数据中心跟其他网络对 SDN 而言没啥区别。而随着 SDN 的发展，人们发现很多别的网络，包括运营商、企业局域网、无线网络、安全领域等都适用。

误解七：SDN Controller 都是集中式控制

集中式控制会有可扩展性问题，所以集中式控制的 Controller 只适合中小型网络。对于大型网络，需要分布式控制，即多个 Controller 协同工作，每个 Controller 负责网络的一部分，彼此之间又有协调。还可能有混合式控制（局部集中、全局分布）。

误解八：SDN 设备需要特殊的 SDN 芯片支持

经常听到有人在说什么 OpenFlow 芯片、SDN 芯片，作为一个芯片设备商里面的资深技术人员，笔者可以明确地告诉大家，至少目前市场上还没有专门用于 OpenFlow/SDN 的 ASIC 商用芯片。SDN 强调转发面和控制面的分离，接口的标准化，但是并不意味着必须用特殊芯片。只不过，如果芯片方面能够针对 SDN

的需求进行创新定制，会更好地支持 SDN，目前我们有充分的理由相信各个芯片厂商都已经有所动作了。但是在没有专门的芯片之前（也许永远都不会有所谓的“专门”芯片，只会有针对 SDN 优化过的芯片），设备商仍然可以对传统芯片进行包装配置，向上提供出部分符合 SDN 需求的南向接口。

误解九：SDN 设备都是靠静态配置的

前面讲过了，SDN 设备是应用程序通过 Controller 来配置的，这些应用程序完全体现了管理员的意志，但是可以是自动运行的，并不一定都要静态配置。比如仍然允许路由协议在 Controller 之上运行，计算的结果通过 Controller 下发送到交换机。

误解十：SDN 只适用于交换机

应该说目前大多数人谈论 SDN 的时候，主要谈论交换机，毕竟与 SDN 密切相关的 OpenFlow 标准是专门针对交换机的，而且 SDN 也起源于 OpenFlow 交换机。但是实际上，SDN 的核心理念是转发面和控制面分离，软件定义网络，它并不局限于交换机，也就是说 SDN 的理念完全可以用于路由器、防火墙、无线产品等一切可以用来组建网络的设备。

1.6 不该被遗忘的 SDN 先烈

Casado、Mckeown 等人其实并不是最早提出控制面跟转发面分离的人，早在 2004 年，Intel、北得克萨斯大学、Nokia 公司的科研和技术人员就一起提出了一个 RFC 3746 Forwarding and Control Element Separation (ForCES) Framework，只是这个 RFC 一直处于 Informational 状态（表示公而告知，并不算是正式被 Internet

社区广泛认可的标准)。在这个 RFC 里面，他们首次公开提出了控制面和转发面分离的概念，以及一个网络框架，在这个框架里面已经有现在 SDN 的雏形了，但是这个 RFC 并没有定义具体的实现标准。

到了 2010 年，浙江工商大学、吕勒奥理工大学、IBM、Intel、Nokia、Znyx 的科研和技术人员又提出了一个新的 RFC 5810 Forwarding and Control Element Separation (ForCES) Protocol Specification，在这个新的 RFC 里面，他们实现了一个类似于 OpenFlow 的协议标准，目前这个 RFC 已经变成了 Standard Track 状态（表示已经得到认可，成为正式标准）。据说浙江工商大学的相关科研人员早在 2003 年前后就已经开始这方面的研究了。有中国的机构/公司参与制定的 RFC 标准真的是屈指可数。

可惜的是 ForCES 无论是之前还是现在，都没有什么影响力，笔者也只是看到在 Nick Mckeown 的一个论文里面提了一下，这是名副其实的先烈。之所以成为先烈，主要原因是当时的网络环境还不成熟，网络用户没有足够的动力去做网络创新。另外一个原因估计是这个 RFC 的制定者没有斯坦福大学的人这么能折腾。此外，大家难道不觉得 OpenFlow/SDN 这两个名字比 ForCES 要响亮得多，且表意性更强吗？所以给一个技术起一个好的名字是多么重要啊！

第 2 章

我们为什么需要 SDN

2.1 网络业务发展趋势

每一种新技术的出现都是因为网络业务需求发生了变化，所以要理解一种新技术，一定要首先去理解业务需求发生了怎样的变化。SDN 的出现，也不外如此。所以首先就让我们来看看网络业务需求在近几年发生了什么样的变化。

全球化竞争的压力迫使各个企业和组织要不断利用技术创新来提高自己的竞争力，这些技术包括但不限于服务器虚拟化、存储虚拟化、云计算以及相配套的一些自动化工具和业务流程化工具，通过对这些工具的使用来加速产品推出时间和提高服务质量，从而扩大自己的竞争优势。这个过程中，IT 技术不断演进、转变，来适应这种需求。

IT 技术的这种演进和转变无论从云服务提供商、电信运营商还是企业网来看都在显著地发生着。比如数据中心里面的多租户环境的创建，从应用的角度来看很好、很强大，带来了很多好处，但是不可否认也很复杂，给早已不堪重负的传

统网络架构带来了沉重的压力，传统的网络架构其实已经很难适应这种新的需求，但是今天的很多数据中心仍然在传统架构的基础上步履维艰地勉强支撑着，落后的网络架构已经拖了业务发展的后腿。为了进一步看清楚这些问题，我们需要来分析一下影响数据中心/企业网的一些显著的变化趋势：

- 数据中心的合并。现在越来越多的企业减少自己内部网络的投入，将部分网络或者全部网络都移到了公有云提供商处，可以认为越来越多的企业自己的中小数据中心被合并到了一个很大的数据中心，对这些大的数据中心来说，意味着有更多的设备、更复杂的布线和更多的网络流量。
- 服务器虚拟化。为了充分利用资源，降低成本，并且减少宕机事件，越来越多的数据中心里面部署了服务器虚拟化，大量的虚拟服务器和用于访问它们的虚拟网络被广泛地集成到了物理网络基础架构中。研究数据表明，服务器虚拟化是未来一两年来 IT 部署和管理领域的 12 大高优先级事件（图 2-1）之一，排名第三。如何来有效地管理这些数量巨大的虚拟机和虚拟网络是一个很大的问题。
- 新的应用架构。数据表明，现在有超过 60%的组织正在大量部署基于服务和基于 Web 的应用，这些应用促使数据中心要创建大量服务器到服务器之间的通信连接，而且也要求不同应用之间要相互隔离。数据中心从传统的基于数据转发的模式转换到了基于服务的递交模式，这使得数据中心变得更趋向于动态、复杂，传统的网络架构不再适合。
- 云计算。事实上，云计算是近几年 IT 领域发展最迅速的技术。它要求企业能够用更加敏捷有效的技术来快速响应云计算业务需求，也对网络架构

提出了新的需求。

- BYOD。BYOD 是 Bring Your Own Device 的意思，就是说现在有越来越多的员工携带自己的 IT 设备到公司上班，包括笔记本、PAD、智能机等，这给现有企业无线网络的流量、安全、管理带来了更大的压力。

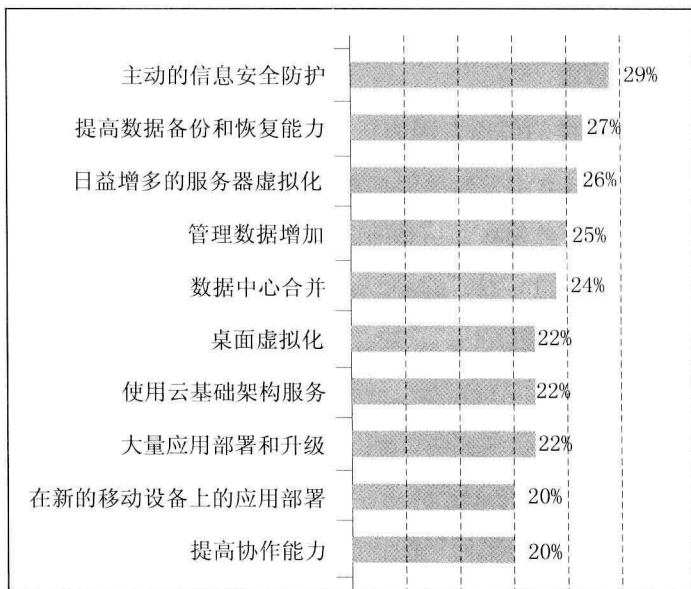


图 2-1

总结一下就是，网络业务的发展要求管理员能够管理越来越复杂的网络和设备，部署各种各样复杂的应用，以及应对越来越大的数据流量，这样，如何能够方便快捷地部署和管理这些应用、设备、网络，减少操作失误和操作时间，减少网络故障概率和恢复时间，就变得尤为重要。以上变化趋势主要发生在数据中心、大型企业网内部、运营商，特别是数据中心。所以说是数据中心激发了 SDN 的兴起（只不过后来延伸到了更广阔的领域）。

2.2 传统网络碰到了瓶颈

当更多小的企业网数据中心合并到更少的但是更大更复杂的网络中去，更多的服务器和应用模型开始部署，更多的员工带各种各样移动设备到企业网中去，传统的网络越来越不堪重负，管理员不得不疲于奔命来应付这些问题。

为什么会存在这样的问题？因为传统网络中都是一个设备一个设备地去管理的，而且管理员对网络中发生的很多事情不可见，网络里面有很多各种各样不同厂家的设备，很难有一个统一的管理平台，有时候出现了新的业务需求，要求对网络中的部分设备进行改动，但是很多时候哪怕是改动一个命令都不可能，因为设备对管理员来说是个黑盒子。不仅设备本身是黑盒子，就网络而言，转发路径都是通过动态协议计算的，管理员很难去知道某个业务的报文走了哪条路径，也比较难去搞清楚哪里发生了拥塞，是不是有更优的路径存在。而且如果有，也很难快速把这个业务切换到更优的路径上去，因为路径不是管理员指定的，而是协议计算出来的。网络里面涉及很多各种各样的协议，还有一些是厂家私有的，管理员很难去把这些知识都学会，这也导致了网络出问题的时候管理员难以定位。基于网络拓扑的自动化业务部署就更加无从谈起。

管理员每天把大量时间放在维护网络、升级设备、管理新的入网设备等网络运营维护的工作上，根本没时间来进行网络创新，这样很容易形成恶性循环。对大型网络来说，最需要关注的并不是一次性设备成本，而是运营成本。图 2-2 是研究机构的一份调查报告，里面列举了网络管理员最常碰到的一些问题。

本书仅提供部分阅读，如需完整版，请联系QQ: 461573687

提供各种书籍pdf下载，如有需要，请联系 QQ: 461573687

PDF制作说明：

本人可以提供各种PDF电子书资料，计算机类，文学，艺术，设计，医学，理学，经济，金融，等等。质量都很清晰，而且每本100%都带书签和目录，方便读者阅读观看，只要您提供给我书的相关信息，一般我都能找到，如果您有需求，请联系我 QQ: 461573687, 或者 QQ: 2404062482。

本人已经帮助了上万人找到了他们需要的PDF，其实网上有很多PDF,大家如果在网上不到的话，可以联系我QQ。因PDF电子书都有版权，请不要随意传播，最近pdf也越来越难做了，希望大家尊重下个人劳动，谢谢！

备用QQ:2404062482