

从软件研究者的视角认识 “软件定义”

关键词：软件定义 云计算 操作系统

梅宏 黄昱 曹东刚 等
北京大学

“软件定义”已成为当今计算机领域的热门术语，从“软件定义的网络”、“软件定义的存储”、“软件定义的数据中心”，到“软件定义的基础设施”、“软件定义的环境”、“软件定义的安全”，以至人们使用“软件定义的一切 (SDX)”来代指各种“软件定义”。另一方面，人们也开始对一些“软件定义”概念的炒作表示担忧。本文从软件系统的角度，通过分析几个有代表性的 SDX，探讨“软件定义”背后的技术本质、发展规律和未来趋势。

“软件定义的网络”(Software Defined Network, SDN)可以算是“软件定义”热潮的肇始。此概念的出现可追溯到2008年美国斯坦福大学承担的美国国家科学基金会重大项目“可编程开放移动互联网 (Programmable Open Mobile Internet)”^[1]，目的是为移动互联网提供一种新型的网络架构。该项目最重要的成果是设计出一套原型系统 OpenFlow，其基本思想是将网络设备（如路由器、交换机等）的管理控制功能从硬件中分离出来，形成一个单独的完全由软件构成的控制层。该控制层抽象了底层网络设备的具体细节，为上层应用提供了统一的管理视图和编程接口，从而支持用户通过软件来定义逻辑上的网络拓扑，满足上层应用对网络资源的不同需求，而无须关心底层网络的物理拓扑结构，实现灵活的网络控制。尽管软件定义的网络和 OpenFlow 在下一代互联网的研究中影响较大，但直到2012年前后，以“基础设施即服务”(Infrastructure as a

Service, IaaS)为代表的云计算，采用软件定义的网络实现数据中心网络的灵活管理后，软件定义的网络才开始大规模应用，成为学术界和产业界关注的热点。“软件定义”的概念也开始在云计算的大背景下迅速向存储、数据中心、运行环境等方向蔓延。

软件定义的存储 (Software Defined Storage, SDS) 是一种数据存储设备的管理方式，通过对存储设备虚拟化，实现高效管理的目标。目前各种各样的存储设备越来越多，如硬盘、U盘、磁盘阵列等，单独管理这些设备会带来存储空间不能充分利用、不方便做冗余备份等一系列问题。同时，高速网络的普及为统一管理存储设备创造了条件。软件定义的存储的主要目标是提供一个软件层，通过高速网络管理各种硬件存储设备，统一高效地实现负载均衡、备份恢复、配额管理等，并以虚拟存储设备的方式向用户提供服务。软件定义的存储已经出现了很多以分布式文件系统技术为核心的产品，如 IBM Storwize、HP StoreVirtual、EMC ViPR 等。

软件定义的数据中心 (Software Defined Data Center, SDDC) 认为，数据中心的计算、存储、网络等资源均应被软件定义，即所有资源均应虚拟化并通过可编程接口对资源进行个性化和自动化的配置和组合，以帮助应用系统快速而有效地响应环境变化。换言之，软件定义的数据中心是从整个数据中心的视角整合软件定义的网络、软件定义的存储和软件定义的计算等。在此基础上，软件定义的环境

(Software Defined Environment, SDE) 强调以虚拟化和可编程接口来实现基础设施资源对应用系统的灵活支撑, 如根据负载模式、业务规则、设施可用性等情况灵活组织和调度各种 IT 资源, 为客户定制和优化业务服务等, 并能够在服务投入运行后持续自动地进行动态优化和配置。

SDX本质分析

分析探究目前各种“软件定义”技术方案, 核心思想都是将传统的“一体式 (monolithic)”硬件设施分解为“基础硬件及其虚拟化+管控软件”两部分。基础硬件提供标准化的基本功能, 进而在其上新增一个软件层来替换“一体式”硬件中实现管控的“硬”逻辑, 为用户提供更开放、灵活、智能的系统管控服务。采用这种技术思路的直接动因是, 互联网环境下应用系统大量涌现, 迫切需要根据各自的需求灵活地共享硬件资源。以云计算为代表的新型互联网应用, 要求硬件基础设施能够以服务的方式提供灵活的计算资源, 而目前的主机管理、存储管理和网络管理在很大程度上是与应用业务脱离的, 几乎都是手工管理、静态配置、分割运行, 甚少发生变动, 难以满足上层应用对计算资源个性化定制、灵活调度、开放共享的需求。而要满足上述需求, 就必须改变目前应用软件开发和网络化资源管理各自分离的情况, 使得各种硬件资源能够根据应用需求自动管理、动态配置。因此, 软件定义就成为一种自然、必然的选择, 这也符合计算系统发展的历史规律。

计算系统由硬件和系统软件组成, 二者互相协作为用户提供计算服务, 支持应用程序的运行^[2]。硬件提供基本的计算资源, 传统上指计算机的主要部件, 包括中央处理器、存储器、外部设备等。随着网络的出现及其快速发展, 硬件资源开始不再局限于单台计算机, 可以是一台或多台通过网络连接的、规模更大的分布式计算系统。在硬件之上, 系统软件 (包括操作系统、编译器、中间件等) 负责管理协调硬件资源, 并抽象为一个软件实现的“虚

拟机”, 为用户提供编程接口和访问界面, 支持应用软件正确、高效的运行。回顾计算系统的发展历程可以发现, 从单机计算系统、网络计算系统到今天的互联网计算系统, 伴随着硬件能力的不断提升和应用需求的复杂多变, 硬件的管理配置功能越来越多地转由软件特别是系统软件来提供, 通过软件进行灵活地管理调度, 硬件的功能和性能才更能在具体应用中得以充分发挥。软件实现的“虚拟机”在功能上和硬件机器等价, 不仅能够为用户提供更为方便易用的界面和开发支撑, 还具有硬件机器所缺乏的灵活性, 可满足不同应用程序对资源的不同需求, 这在本质上正是“软件定义”的体现。

“软件定义”技术途径的核心是硬件资源虚拟化和管控功能可编程。硬件资源虚拟化是将硬件资源抽象为虚拟资源, 然后由系统软件对虚拟资源进行管理和调度, 如操作系统中虚拟内存对物理内存的虚拟、伪终端对终端的虚拟、套接字 (socket) 对网络接口的虚拟、逻辑卷对物理存储设备的虚拟等。硬件资源虚拟化支持物理资源的共享, 可提高资源利用率, 屏蔽不同硬件的复杂细节, 简化对资源的管理和调度, 通过系统调用接口可对上层应用提供统一的服务, 方便程序设计。应用软件和物理资源在逻辑上分离, 可分别进行独立的演化和扩展并保持整个系统的稳定。

管控功能可编程是对通用计算系统应用的核心需求, 主要表现在访问资源所提供的服务、改变资源的配置和行为两个方面。在硬件资源虚拟化的基础上, 用户可编写应用程序, 通过系统调用接口访问资源所提供的服务, 更重要的是能够灵活管理和调度资源, 改变资源的行为, 以满足应用对资源的多样需求。所有的硬件资源在功能上都应该是可以编程的, 这样, 软件系统才可以对其实施管控, 一方面发挥硬件资源的最佳性能, 另一方面满足不同应用程序对硬件的不同需求。从程序设计的角度, 管控功能可编程意味着计算系统的行为可以通过软件进行定义, 成为所谓的“软件定义的系统”。

操作系统作为计算系统中最为重要的系统软件, 一方面直接管理各种计算资源, 另一方面作为

“虚拟机”为应用程序提供运行环境。在此意义上，操作系统体现了“软件定义的系统”技术的集大成。当前出现的软件定义的网络、软件定义的存储等技术如同设备互联技术、磁盘存储技术之于单机操作系统一样，本质上反映了“网络化操作系统”对网络化、分布式设备的管理技术诉求，也将成为“网络化操作系统”核心的底层支撑技术，并在操作系统的整体协调下，发挥最高的功效。同样，未来互联网范围的操作系统也会通过各种“软件定义”的途径，有效管理互联网范围的计算资源。因此，回顾操作系统的发展，有助于从软件的视角更深入地理解“软件定义”的本质。

操作系统发展回顾

操作系统是计算系统中最为关键的一层系统软件。按照《中国计算机科学技术百科全书(第二版)》的定义，“操作系统是管理硬件资源、控制程序运行、改善人机界面和为应用软件提供支持的一种系统软件”。它的主要功能是：向下管理资源（包括存储、外设和计算等资源），向上为用户和应用程序提供公共服务。

最早的计算机没有操作系统，只是一台硬件裸机，用户可以在某个指定的时间段单独占用计算机上的所有资源，通过打孔带（卡）或者磁带来手动输入程序和数据。随着处理器的速度越来越快，手动的任务切换方式会浪费大量时间，从而出现了批处理功能来对任务进行自动切换，以提高处理器的利用率。计算机的操作也逐步从最早的由程序员动手，变成了专业的操作人员负责，进而出现了自动监控程序。这些包含了基本硬件管理、简单任务调度和资源监控功能的管理程序形成了操作系统的雏形。随着新的应用需求的不断出现，越来越多的管理功能逐渐添加到操作系统中，进而积累出操作系统的标准功能。

从操作系统的出现、发展和功能基本定型的过程可以看出，操作系统实际上就是对计算系统进行“软件定义”的产物。相对于最早的硬件计算机，操作系

统可视为一种“软件定义”的“虚拟计算机”，屏蔽了底层硬件细节，由软件对硬件资源进行管理，用户不再直接对硬件进行编程，而是通过应用编程接口改变硬件行为，实现更优的灵活性、通用性和高效性。

为了更好地支持网络应用的开发和运行，作为对单机操作系统的补充，还出现了“中间件”(middleware)的概念。中间件是位于应用程序和操作系统之间的一层系统软件，凝聚了与网络资源管理紧密相关的一组服务，如远程调用、负载均衡、容错管理、访问控制等，进而为应用程序提供相应的编程接口，以辅助其以更加透明的方式访问网络资源。

近年来随着网络的迅速发展和不断延伸，以互联网为主干，正在形成融合其他多种异构网络（包括电信网、广电网、传感网）的复杂网络计算环境。与运行于传统单机计算环境下的操作系统相比，运行于复杂网络计算环境下的互联网操作系统面临若干挑战^[3]，需要在现有单机系统和中间件的基础上，实现若干新的功能^[4]，包括结构化（不再一次加载软件的全部功能，而是按需加载部分功能）、服务化（用户在不拥有软件代码的前提下，随时随地访问和使用软件所提供的功能）、可伸缩（统一管理虚拟化的计算和存储资源，根据用户需求弹性、透明地调度和分配）和可定制（支持对共性资源的个性化定制，以更加灵活地构建面向公众、部门、行业、家庭或个人的网络操作系统）。为了支持上述功能，需要通过软件手段对互联网操作系统的能力进行扩展、提升甚至重新定义。

目前已有很多针对互联网操作系统的研发工作，如学术界从软件的基本规律和特性角度开展的网构操作系统和虚拟计算环境的研究，产业界以云计算为背景开展的云计算管理系统和数据中心操作系统的研发等。其中，云计算管理系统是一种互联网环境下的新型“网络化操作系统”，通过“软件定义”技术对网络化、规模化的各种计算资源进行高效灵活的调度管理，是各种“软件定义”技术的综合展示平台。了解云计算管理系统的发展，有助于从软件的视角更准确地把握“软件定义”的发展趋势。

云计算管理系统

云计算是互联网时代的一种计算模式，统一管理服务器、存储、网络甚至软件等各种资源，并以服务的形式供用户和应用使用。这些资源聚集为共享资源池，用则进行分配，而且种类和数量均可按需配置并按实际用量计费，不用则回收资源池中等待下次分配^[5]。

云计算管理系统在早期发展阶段主要关注于虚拟机的使用，通过对虚拟机管理器 (hypervisor) 的协同控制，支持用户按需使用具有不同的 CPU、内存和磁盘配置的虚拟机。随着虚拟机使用场景的不断普及和深化，网络的个性化配置、存储的分布式备份等管理需求会接踵而至。

网络作为一种重要的资源也需要实现个性化定制，例如，有的用户要求虚拟机的网络能够保证百兆网速，而有的用户则指定 IPv6 路由协议栈。由于虚拟机网络最终由物理网络承担，这些需求自然地转变为对物理网络进行个性化配置的需求。面对庞大的租户群及其多样的个性化配置需求，网络硬件有限的配置方式让人捉襟见肘。因此，软件定义网络成为云计算系统管理网络资源的自然选择，如 VMware 最新的 5.1 版本中通过 Distributed vSwitch 来管理网络，OpenStack 也已经考虑融合 OpenFlow 的网络功能。

与网络类似，存储作为虚拟机中数据的存取介质也需要灵活、可靠、高效的配置和管理。从业务角度，云用户既不关心存储介质的接入方式是 NAS 还是 FC SAN，也不关心存储格式是 ext3、FAT3T 还是 NTFS，他们关心的是数据能否按设想的文件系统格式持久保存、快速访问、定期快照和备份等。然而，存储设备一般由专业厂商提供，其私有的访问协议和独特的硬件架构使得数据的业务操作功能与底层持久化机制互相绑定，难以支撑云计算系统中大规模租户数据的按需存取。因此，迫切需要通过软件来实现数据的业务操作任务，打破业务操作与持久化机制的绑定，屏蔽硬件差异，提高数据存取的灵活性，即“软件定义的存储”。如 VMware

给出了 Virtual SAN 的解决方案，OpenStack 给出了 Swift 的分布式存储解决方案。

云计算管理系统通过“软件定义”的途径，一方面实现资源虚拟化，达到物理资源的共享和虚拟资源的隔离；另一方面实现管理功能的可编程，打破传统硬件配置能力有限的桎梏，为用户的业务需求提供高效灵活、随需而变的支撑。因此，云计算管理系统作为一种新兴的网络化操作系统，是一个典型的软件定义的系统。

结语

“软件定义”是近期出现的热点概念，但其蕴含的硬件资源虚拟化和功能可编程两个核心思想一直都是操作系统设计与实现的目标与准则。“按需使用、随需随用、无需停用”将成为常态。可以预料，“软件定义”将扮演日益重要的角色，这给软件技术特别是系统软件技术的发展带来了一系列新挑战和新机遇。■



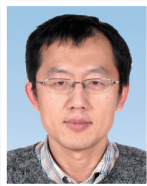
梅 宏

CCF理事、会士、系统软件专业委员会主任。中国科学院院士，上海交通大学、北京大学教授。主要研究方向为软件工程和系统软件等。meih@pku.edu.cn



黄 晔

CCF会员、CCF青年科学家奖获得者。北京大学教授。主要研究方向为系统软件和互联网计算等。hg@pku.edu.cn



曹东刚

CCF会员。北京大学副教授。主要研究方向为系统软件与分布式计算等。caodg@pku.edu.cn

其他作者：郭 耀 张 颖 刘譞哲 熊英飞

参考文献

[1]POMI 2020 project website. <http://pomi.stanford.edu/>.

[2]Brandal E. Bryant, David R. O' Hallaron, Computer Systems: A Programmer's Perspective (2nd Edition), 2010, Addison-Wesley.

[3]梅宏,刘譞哲. 互联网时代的软件技术: 现状与趋势. 2010; 55(31): 1214~1220.

[4]梅宏,郭耀. 面向网络的操作系统—现状与挑战. 中国科学信息科学. 2013(3).

[5]Michael Armbrust, Armando Fox, Rean Griffith, and et al.. A view of cloud computing. Communications of the ACM, April 2010, vol. 53, no. 4.