银行业信息科技风险管理高层指导委员会
银行业信息化丛书

商业银行私有云设计与实现

金磐石　戴蕾　侯铮　等编著

The Design and Implementation of Private Cloud for Commercial Banks

机械工业出版社
CHINA MACHINE PRESS

本书内容简介

本书系统地介绍了云计算的概念、已有相关理论和产品，以及商业银行私有云的整体架构和设计，涵盖了中国建设银行在私有云的整体架构设计、云基础设施设计、云服务设计、云管理平台设计、云安全设计等方面的知识、技巧和技术。此外，本书还介绍了中国邮政储蓄银行开发测试云建设过程，以及浙江省农村信用社联合社在私有云方面的应用。

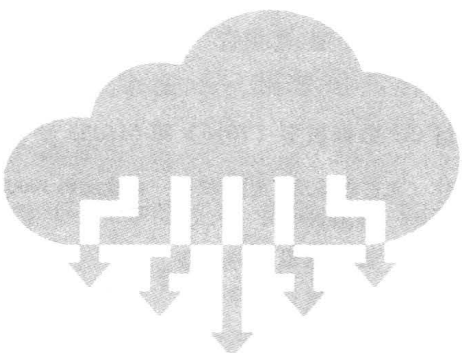
本书可供各银行私有云规划人员、设计人员以及建设人员等使用，也可供其他行业的相关人员参考。

978-7-111-54238-7
978-7-111-53087-5

银行业信息科技风险管理高层指导委员会
银行业信息化丛书

商业银行私有云设计与实现

金磐石　戴蕾　侯铮　等编著



The Design and Implementation of Private Cloud for Commercial Banks

机械工业出版社
CHINA MACHINE PRESS

本书系统地介绍了云计算的概念、已有相关理论和产品，以及商业银行私有云的整体架构和设计，涵盖了中国建设银行在私有云的整体架构设计、云基础设施设计、云服务设计、云管理平台设计、云安全设计等方面的知识、技巧和技术。此外，本书还介绍了中国邮政储蓄银行开发测试云建设过程，以及浙江省农村信用社联合社在私有云方面的应用。

本书可供各银行私有云规划人员、设计人员以及建设人员等使用，也可供其他行业的相关人员参考。

图书在版编目（CIP）数据

商业银行私有云设计与实现/金磐石等编著. —北京：机械工业出版社，2016.2

（银行业信息化丛书）

ISBN 978-7-111-52725-1

Ⅰ.①商… Ⅱ.①金… Ⅲ.①银行—管理信息系统—研究

Ⅳ.①F830.49

中国版本图书馆 CIP 数据核字（2016）第 016253 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

总 策 划：张敬柱 黄养成

策划编辑：林运鑫 责任编辑：林运鑫

责任校对：王建梅 封面设计：徐 超 责任印制：乔 宇

保定市中画美凯印刷有限公司印刷

2016 年 2 月第 1 版第 1 次印刷

184mm × 260mm · 19.25 印张 · 474 千字

0001 — 5500 册

标准书号：ISBN 978-7-111-52725-1

定价：79.80 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

电话服务

服务咨询热线：010-88361066

读者购书热线：010-68326294

010-88379203

网络服务

机 工 官 网：www.cmpbook.com

机 工 官 博：weibo.com/cmp1952

金 书 网：www.golden-book.com

教育服务网：www.cmpedu.com

“银行业信息化丛书”编委会

总　序

信息化是推动经济社会变革的重要力量。坚持走中国特色的新型工业化、信息化、城镇化、农业现代化道路，是党中央立足全局、放眼未来、与时俱进的战略决策。2014年2月27日，中央网络安全和信息化领导小组的成立，更加体现了中央保障网络安全、推动信息化发展、维护国家利益的决心。银行业作为国家经济体系的重要行业之一，是信息化的重要推动主体、参与主体和受益主体。银行业持之以恒地贯彻落实国家信息化战略，不仅是推动加快我国信息化进程的必然要求，也是银行业改革发展、转型升级和更好服务实体经济的内在需求。

近年来，我国银行业审时度势、积极作为，坚持基础建设与科技创新并重、提升服务与保障安全并举的科学发展导向，以推进信息化为契机，调整经营理念、优化经营机制、完善服务模式，在服务手段信息化、管理模式信息化、信息安全保障等方面取得积极进展，推动了银行业的核心竞争力、市场适应力和贴身服务能力的进一步提升。一是服务手段信息化发展迅速。电子银行、自助银行、智能支付终端等信息化服务渠道日渐普及，使得金融服务覆盖面更加广泛、服务方式更加便捷、服务产品更加丰富。二是管理模式信息化迈出实质性步伐。注重依托核心数据库、运用先进数据挖掘分析工具，推进银行经营决策逐步智能化，风险管理日趋精细化，产品创新逐渐体现个性化，银行业经营管理信息化水平不断提升。三是信息安全保障取得积极进展。银行业信息安全越来越受重视，相关科技基础设施建设步伐加快，多层次、立体化、全方位的信息安全保障体系正在逐步形成。

当然，我们也应该清醒地认识到，银行业信息化面临着复杂的内外部环境，核心技术受限、网络安全威胁、隐私保护和信息保密等挑战将长期存在，银行业自身认识不尽到位、技术储备不够充分、资源投入相对不足、过度依赖外包等问题仍较为突出，针对银行业特殊需求的信息化产品、工具和方法还比较单一，缺乏应对复杂需求的灵活创新能力。总的看来，银行业信息化还有很长的路要走，信息科技风险将成为当前和未来较长时期银行业的重要风险领域之一。

银行业信息化既不能因为成绩而骄傲自满，也不可因为差距而妄自菲薄，更不可因

为困难而畏首畏尾。各银行业金融机构要勇于直面困难、主动迎接挑战，坚决按照国家信息化总体战略部署，切实坚持“自主可控、持续发展、科技创新”的基本方向，紧紧抓住信息化发展机遇，推动信息服务和信息安全再上新台阶。一是借助信息化推动银行业金融机构治理能力现代化。积极引入先进的信息科技治理和管理理念，运用现代信息技术缓解治理中的信息不对称问题，推动流程银行建设，提高治理有效性。同时，理顺信息化建设的体制机制，加快信息化建设进程，为银行业转型发展提供有力保障。二是依托信息化推动金融服务智慧化。要充分利用互联网、移动计算蓬勃发展的大环境，积极应用大数据等新兴技术，创新思维模式，充分发挥金融数据和信息的价值，研发智能化、个性化、便捷化的产品和服务，灵活响应客户诉求，努力改善客户体验，尽力发掘潜在客户需求，增加产品和服务的吸引力，培育更为坚实的客户基础，形成新的业务和利润增长点。三是以自主创新增进安全可控能力。要坚持市场起决定作用的基本方针，探索形成以研发创新支持应用推广、以市场应用激发创新动力的良性正反馈机制。推动应用自主创新信息技术，建立自主创新信息技术落地银行业的配套机制，力争金融领域关键信息技术自主创新占比逐步提高，不断提升信息系统的开放性、灵活性和整体集约化水平。四是利用信息技术强化行业协作。要加强银行业信息化建设的统筹规划，促进信息化资源的集约共享，提升数据（灾备）中心布局的合理性，增强同业协同协作，共同应对外包集中度等风险。

为更好地推进落实银行业信息化战略，由银行业信息科技风险管理高层指导委员会指导推动，编著了“银行业信息化丛书”（简称“丛书”）。这套“丛书”致力于挖掘、研究、总结、提炼和传播国内外信息化最佳实践、宝贵经验和最新成果，内容涵盖银行业信息科技治理与管理、信息系统开发与应用创新、信息安全、基础设施与运行维护、信息科技监管等主要领域，可为银行业信息科技人才培养提供一些基础性、前瞻性、实用性的知识和信息。

展望未来，银行业信息化任务艰巨、时间紧迫。希望银行业在有关各方支持下，推动信息化工作更加积极主动、规范有效、科学前瞻，为我国银行业持续健康发展、提升服务水平提供坚实的支撑，为增强国家网络安全保障能力、提升信息化建设水平提供有力支持，为贯彻落实创新驱动发展战略、实现中华民族伟大复兴的中国梦做出积极贡献。

尚福林

前　言

2006 年 3 月，亚马逊推出弹性计算云（Elastic Compute Cloud；EC2）服务，2006 年 8 月，Google 首次提出“云计算”（Cloud Computing）的概念，云计算发展至今已近 10 年。据 Gartner 预测，到 2016 年全球云服务支出将达 6700 多亿美元，发展速度非常迅猛。

何为“云计算”？对于云计算的定义有多种说法，现阶段广为接受的是美国国家标准与技术研究院（NIST）给出的定义：云计算是一种按使用量付费的模式，这种模式提供可用的、便捷的、按需的网络访问，进入可配置的计算资源共享池（资源包括网络、服务器、存储、应用软件和服务），这些资源能够被快速提供，只需投入很少的管理工作，或与服务供应商进行很少的交互。

云计算拥有超大规模、虚拟化、高可靠性、通用性、高可扩展性、按需服务等特点。如亚马逊公有云计算平台已拥有上百万台服务器，而企业私有云通常也拥有几千台，甚至上万台服务器。“云”能赋予用户前所未有的计算能力。云服务是云计算的核心内容，具有三种服务模式，包括基础设施即服务（IaaS）、平台即服务（PaaS）和软件即服务（SaaS）。当前常见的云计算部署模式有公有云、私有云和混合云三种。

随着云计算理念的不断深入，数据中心和 IT 服务管理水平成为企业竞争力的一个重要组成部分，甚至能直接创造价值。应用云计算理念和云计算技术的数据中心成为新的发展趋势。

商业银行这样的企业同样关注成本，而随着业务迅猛发展，数据中心规模急剧增长，IT 系统数据量及复杂度呈几何级数上升，对系统的安全性、可用性要求越来越高，甚至“零容忍”，传统的基础设施面临着非常大的挑战，商业银行探索及采用新技术已经成为必然的选择。云计算、大数据等新技术的产生和逐步成熟为商业银行 IT 基础设施建设带来新的机遇，它们需要云服务。尽管公共云服务提供商遵守一定的行业法规，但是大型商业银行要兼顾行业、客户隐私，不可能将重要数据存放到公共网络上，因此更倾向于架设私有云计算平台。商业银行私有云也是现阶段各商业银行主推的云计算部署模式。

中国建设银行从2011年5月开始进行云计算技术的论证，2012年完成了基础设施规范设计、云服务设计、云管理平台设计，2013年正式启动了新一代云管理平台项目开发，在参考云计算最新技术趋势、国内外先进的云数据中心建设理念和实践经验的基础上，结合中国建设银行新一代的业务需求和研究成果，构建了以云管理平台为核心的一体化管理体系，完成了云计算数据中心一期的建设，到目前为止已顺利支持了新一代一期项目、新一代二期项目的投产，武汉数据中心建设及搬迁工作。据了解，中国工商银行、中国银行、中信银行、中国民生银行、中国邮政储蓄银行等商业银行也在私有云建设上进行了大量实践工作，并取得了良好的效果。

因此，中国建设银行受银行业信息科技风险管理高层指导委员会委托编著了《商业银行私有云设计与实现》一书。本书系统地介绍了云计算的背景，商业银行私有云设计与应用，阐述了大型商业银行数据中心面临的问题和挑战，介绍了云计算产业的各类服务、国内外云计算产品，深入探讨了商业银行私有云的技术实现、应用实践、实施收益、实施难点及建议，为商业银行的云计算从业人员、使用者全面了解私有云提供帮助和指导。

本书共分为三篇：即基础篇、设计篇、应用与探索篇，包括12章。

第1章：概要介绍云计算的国际背景、国内行业背景，分析商业银行数据中心面临的问题、挑战以及商业银行私有云应用研究背景，说明商业银行私有云建设的必要性。

第2章：介绍云计算相关理论和参考模型，包括ITIL最佳实践、IBM CCRA云计算参考框架、DevOps理论和云安全管理模型，说明探索和借鉴行业最佳理论、实践对云计算数据中心的意义重大。

第3章：介绍业内较为熟悉的主流厂商云计算解决方案、虚拟化技术和云管理方案，让读者能够更全面地了解各个产品的特性。

第4章：主要介绍业内主流的云计算领域的开源软件架构和相关技术。

第5章：介绍目前商业银行私有云平台使用的技术、私有云的特点，以及建设私有云平台应采用的技术要求、技术标准等。

第6章：全面介绍云基础设施的设计，涵盖了机房环境资源规划设计要点，计算资源规划设计，网络资源规划涉及的网络虚拟化关键技术、资源池网络设计，存储资源的存储服务级别设计、存储池规划设计等方面，让读者详细了解云计算中各类资源的设计要求。

第7章：介绍云服务的定义和设计原则，云服务与云管理平台的关系，以及云服务描述模型，并阐述在设计云服务过程中需要制定的服务规范、技术领域规范、云服务开发流程等，让读者了解云服务在云计算中的作用，以及云服务的设计、开发是云平台建设过程中非常重要的环节。

第8章：介绍云管理平台的设计思路及需要遵循的设计原则，并详细介绍了云管理平台总体功能、内部服务、服务质量等要求，以及云管理平台设计核心。

第9章：从IaaS、PaaS和SaaS三个层面进行安全分析，提出初步的保护策略，根据安全分析，构建私有云下的安全架构设计模型，并就各个防护模块进行简要阐述。

第 10 章：重点介绍中国建设银行私有云建设的几个成果，主要包括资源的全生命周期管理、弹性伸缩、发布和变更管理、运维大数据分析以及监控智能处置，总结中国建设银行私有云建设的实施收益和实施难点，为从业人员设计和实施私有云提供借鉴。

第 11 章：介绍中国邮政储蓄银行开发测试云建设情况，分析了开发测试环境现状，探索云技术在开发测试环境中的应用，全面阐述中国邮政储蓄银行开发测试云的建设过程、部署方案，以及给银行带来的价值。

第 12 章：介绍浙江省农村信用社联合社借助私有云实现在互联网金融上的创新，分析传统模式建立互联网金融所遇到的问题，建设以私有云为基础的全新互联网服务系统，详细阐述了私有云的建设方案及所取得的收益。

参加本书编写的有中国建设银行股份有限公司信息技术管理部的金磐石总经理、戴蕾、侯铮、张有才、高亚军、刘永福、刘涵、孟朝雄、侯岳、邱俊珺、赵子健、张明、宋育芳、李康、万威，中国邮政储蓄银行总行信息科技部的张振山、穆冬生、瞿红来、钟泽宇，浙江省农村信用社联合社科技信息处的顾亚明、傅祝庆、徐云飞、邓运高、许文胜。

本书大部分内容来自中国建设银行新一代云平台建设的实施经验和成果，在编写过程中得到了中国建设银行股份有限公司信息技术管理部朱玉红副总经理、刘延新副总经理、林磊明资深经理、刘爱辉处长，以及北京数据中心沈秋翔主任、吴险峰副主任、郭玉章资深经理、张翔处长等领导、同事的悉心指导与鼎力支持。

此外，感谢中国银监会银行业信息科技监管部梁峰、包倩、周航在本书编写过程中提出的宝贵意见和建议，并给予的指导和帮助。感谢 IBM、VMware、华为、阿里巴巴、青云公司为本书提供的宝贵技术资料。

由于本书知识面较广，虽经编者反复修改，但由于时间和水平有限，书中难免有疏漏和不当之处，敬请读者批评指正。

编　者

目　　录

设 计 篇

应用与探索篇

基础篇

第 1 章
云计算概述

1.1 云计算背景

1.1.1 云计算国际背景

2006 年美国的亚马逊推出了世界上第一个云计算系统 AWS（Amazon Web Services，亚马逊云服务），并获得了巨大成功。此后，相关技术得到了突飞猛进的发展。Gartner（高德纳咨询公司）在 2013 年的云计算报告中指出，2013 年全球公共云服务市场规模将从 2012 年的 1110 亿美元，增长 18.5% 至 1310 亿美元，IaaS（Infrastructure as a Service，基础设施即服务）依然是增长最快的云计算服务，2013 年 IaaS 的增长幅度将提高到 47.3%，市场规模将达到 90 亿美元。Gartner 预测，2013—2016 年，全球云服务支出总计将达 6770 亿美元。

随着云计算理念的不断深入，数据中心和 IT（Information Technology，信息技术）服务管理水平成为企业竞争力的一个重要组成部分，甚至能直接创造价值。应用云计算理念和云计算技术的数据中心成为新的发展趋势。越来越多的国外银行也开始考虑将传统 IT 基础设施迁移到云计算方案。

1.1.2 云计算国内行业背景

近些年，中国云计算从概念到实际应用有了实质性的发展，尤其是对于云计算基础架构的投资更是发展迅速。国内大中型企业对于建设云计算基础架构的兴趣越来越浓厚。各企业数据中心越来越多地采用虚拟化技术和自动化管理软件，并逐步向云计算基础架构过渡。云基础架构已经成为下一代数据中心（Next Generation Data Center，NG-

DC）建设的主要目标。IDC（International Date Corporation，国际数据公司）对中国云计算市场的研究数据显示，中国私有云计算基础架构市场将保持高速发展。中国云计算基础架构市场及其预测见表 1-1。

表 1-1　中国云计算基础架构市场及其预测　（单位：百万美元）

年　　度	2011	2012	2013	2014	2015	2016
公有云运营商基础架构支出	319.8	426.6	532.4	636.5	747.1	856.5
私有云用户基础架构支出	285.8	410.4	555.5	752.3	932.8	1157.1
总计	605.7	837.0	1087.9	1388.8	1679.9	2013.6
增长率(%)	—	38.2	30.0	27.7	21.0	19.9

腾讯、百度、阿里巴巴等国内领先的互联网公司，中国移动、中国电信等电信公司都开始相关的研究和实施，完成了自身的云计算数据中心和云管理平台的建设。经过近几年的不断发展，云计算数据中心已经从 IaaS 发展到 SaaS（Software as a Service，软件即服务），应用范围也从满足企业内部的需求扩展到满足外部客户的需求。

1.1.3　商业银行数据中心面临的问题和挑战

随着银行业务的迅速发展和规模的急剧扩大，银行 IT 系统的数量、规模及复杂度也呈几何级数上升，业务对 IT 系统的安全性、可用性与持续性的依赖程度也越来越高，但商业银行在完成数据大集中之后，由于基础设施仍主要采用传统技术方案，数据中心面临着如下困境与挑战：

1. 系统高可用性要求日益严格，运行风险日益突出

商业银行数据中心作为“金融业跳动的心脏”，稳定运行和控制风险是第一要务。一方面，基础设施故障、突发业务压力、频繁变更上线都可能影响系统的稳定和服务质量，随着业务部门和上级监管机构要求不断提高，银行对数据中心高可用性要求也日益严格；另一方面，数据中心对外部基础设施、外部技术和服务的依赖性不断增强，网络入侵、信息泄露等安全风险日益突出。

2. 技术路线受制于人，数据中心建设成本不断攀升

大型商业银行一直以来采用最成熟、可靠的 IT 技术路线，通常使用国外主流厂商提供的信息技术和商业产品进行集中式部署，在信息技术实施、支持和保障上很大程度依赖信息技术供应商，存在技术标准不统一、新技术应用和技术创新缓慢、投入产出比低等问题，也使得银行自身缺乏核心技术积累，在技术路线选择上受制于国外厂商。随着基础设施规模的不断扩大，数据中心建设成本不断攀升，例如在淘宝“11·11”促销和电商秒杀等业务需求中，传统技术只能按照业务峰值配置基础设施资源，会造成巨大的资源浪费。因此，出于安全生产和企业降低 IT 成本的双重要求，银行亟需探寻自主可控的创新解决方案。

3. 传统 IT 基础设施无法高效支撑银行创新业务

在金融服务互联网化、移动化发展趋势下以及利率市场化挑战下，银行从战略和战术层面积极应对，提出了金融互联网、大数据、电子商务、客户体验等新的战略目标，

并以此快速推出创新业务。但这些目标所需要的海量信息技术处理能力往往无法通过传统 IT 基础设施解决方案有效满足，即使传统方案能够实现，企业也无法承受天文数字般建设成本以及漫长的建设周期。要兼顾新业务需求建设成本和响应速度，探索、采用新技术已经成为必然选择。

1.1.4 商业银行私有云应用研究背景

当前，银行业信息科技正面临新的机遇和挑战。一方面，互联网金融推动银行业务创新，迫使银行革新现有 IT 技术；另一方面，云计算等新技术的产生和逐步成熟为银行 IT 基础设施建设带来新的机遇。

国内的金融行业对于云计算的态度已经从研究、观望进入实质性的规划实施阶段，各大银行已开展内部私有云项目的尝试，引入云计算技术的数据中心已经成为大家的共识。但传统数据中心的云化将对数据中心运营服务的管理能力产生巨大影响，对传统 IT 运营团队提出了更高的要求，需要具备更先进、更全面、更高层次的技术、运营、管理和服务等综合能力。相对于互联网行业，金融行业的云计算数据中心建设还处于起步阶段，经验和方法还没有形成完整的最佳实践。

中国建设银行（以下简称建行）自 2006 年开始实施 ITIL（Information Technology Infrastructure Library，信息技术基础架构库）项目，按照规划逐步建设 ITIL 流程，成功实施了事件管理流程、问题管理流程、变更管理流程、发布管理流程、服务请求管理流程、配置管理模块及知识库管理模块。经过多年的持续改进，建立了完善的 ITIL 服务管理体系，积累了丰富的服务管理实践经验。2009 年建行开始全面推进数据中心自动化工具体系的建设，先后实现了服务器自动化、网络自动化、批处理调度自动化、存储自动化，在自动化方面积累了丰富的经验，培养了一大批专业人才。云计算的本质就是 ITIL 服务管理和自动化的融合。建行 ITIL 和自动化项目的实施，为向云数据中心的转型奠定了理论和技术基础。在此背景下，2012 年建行启动了新一代 IT 服务管理云平台项目，在参考云计算最新技术趋势、国内外先进的云数据中心建设理念和实践经验的基础上，结合建行新一代的业务需求和研究成果，构建了以云管理平台为核心的一体化管理体系，完成了云计算数据中心一期的建设。中国工商银行（以下简称工行）、中国邮政储蓄银行（以下简称邮储）、中信银行（以下简称中信）等商业银行也在私有云建设上进行了大量实践工作，取得了良好效果。

1.2 云计算的基础概念

1.2.1 云计算定义

美国国家标准与技术研究院（National Institute of Standards and Technology，NIST）

定义：云计算是一种按使用量付费的服务模式，这种模式提供可用的、便捷的、按需的网络访问，进入可配置的计算资源（包括网络、服务器、存储、应用软件、服务）共享池，这些资源能够被快速提供，只需投入很少的管理工作，或与服务供应商进行很少的交互。

云服务是由云计算平台提供者将 IT 能力以面向用户的服务形式来进行包装和集成，通过云管理平台和 Internet（互联网）或者 Intranet（内联网）渠道向云服务消费者（用户）来提供的一种服务。例如 IBM 计算云和亚马逊的 EC2（Elastic Compute Cloud，亚马逊弹性计算云）通过 Internet 实现计算资源的按需供给的 IaaS。新浪云通过 Internet 提供中间件平台的 PaaS（平台即服务）云服务。一个云服务可以集成和使用其他的云服务，往往一个高阶的云服务需要使用几个低阶的云服务进行底层的构建。

云服务是云计算的核心内容，同时是云计算技术实现和业务应用的结合点。开发好的云服务需要发布注册至云管理平台，云管理平台需要将所有的云服务面向云服务消费者（用户）实现交付和管理。当云服务消费者提交服务请求并被批准后，云服务提供者将创建一个云服务实例，向云服务消费者提供云服务。云服务实例是云服务在生产环境下存在的表现形式。

云计算基本原理是，通过计算分布在大量的分布式计算机上，而非本地计算机或远程服务器中的数据，企业数据中心的运行将与互联网更相似。这使得企业能够将资源切换到需要的应用上，根据需求访问计算机和存储系统。

云计算有以下特点：

（1）超大规模 “云计算管理系统”通常具有较大的规模，亚马逊公有云已经拥有上百万台服务器，微软、阿里云等的“云”均拥有几十万台服务器。企业私有云一般拥有数百上千台服务器。“云”能赋予用户前所未有的计算能力。

（2）虚拟化 云计算支持用户在任意位置、使用各种终端获取应用服务。所请求的资源来自“云”，而不是固定的有形的实体。应用在“云”中某处运行，但实际上用户无需了解，也不用担心应用运行的具体位置。只需要一台笔记本或者一个手机，就可以通过网络服务来实现我们需要的一切，甚至包括超级计算这样的任务。

（3）高可靠性 “云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性，使用云计算比使用本地计算机更可靠。

（4）通用性 云计算不针对特定的应用，在“云”的支撑下可以构造出千变万化的应用，同一个“云”可以同时支撑不同的应用运行。

（5）高可扩展性 “云”的规模可以动态伸缩，满足应用和用户规模增长的需要。

（6）按需服务 “云”是一个庞大的资源池，可按需购买；云可以像自来水、电、煤气那样计费。

1.2.2 虚拟化技术

虚拟化可以有很多种，如硬件虚拟化、桌面虚拟化、软件虚拟化、存储虚拟化、网

络虚拟化等，我们通常说的是服务器虚拟化，属于硬件虚拟化。将服务器物理资源抽象成逻辑资源，使一台服务器变成几台甚至上百台相互隔离的虚拟服务器，并且虚拟服务器可以在物理服务器之间动态迁移，不再受制于物理上的界限，使 CPU、内存、磁盘、I/O（Input/Output，输入/输出）等硬件变成可以动态管理的“资源池”，从而提高资源的利用率，简化系统管理，实现服务器整合，使 IT 对业务的变化更具适应力，这就是服务器的虚拟化。

服务器虚拟化的核心技术是 Hypervisor。Hypervisor 是运行在物理服务器和传统操作系统之间的中间软件层，可允许多个操作系统和应用共享硬件，也可叫作 VMM（Virtual Machine Monitor，虚拟机监控系统）。Hypervisor 是一种在虚拟环境中的“元”操作系统，通过它可以访问服务器上包括磁盘和内存在内的所有物理设备。Hypervisor 不但协调着这些硬件资源的访问，也同时在各个虚拟机之间施加防护，进行安全隔离，可以说 Hypervisor 是所有虚拟化技术的核心。

根据硬件的不同，虚拟化技术也分为小型机平台虚拟化和 X86 平台虚拟化，小型机平台虚拟化主要有 IBM PowerVM 和 HP vPAR 技术，在这里不做过多讨论，本文重点讲述 X86 平台虚拟化技术。

目前市场上各厂商、各产品的 Hypervisor 架构存在差异，X86 平台虚拟化技术常见的有两类：全虚拟化和半虚拟化。

1）全虚拟化也称为裸虚拟化或者原始虚拟化，即由 GuestOS（Guest Operation System，宿主级操作系统）和物理裸硬件之间的 Hypervisor 层完成 CPU、内存、I/O 之间的协调工作，一些受保护的指令必须由 Hypervisor 来捕获处理，完全虚拟化不需要修改 GuestOS 操作系统，GuestOS 无法感知它到底运行在硬件还是软件上。

2）半虚拟化使用 Hypervisor 分享存取底层的硬件，但是它的 GuestOS 操作系统集成了虚拟化方面的代码，因此半虚拟化必须对 GuestOS 做一些修改。

虚拟化技术可以看作是通过软件实现对操作系统的资源再分配；而半虚拟化技术则是通过代码修改已有系统，形成一种新的可虚拟化系统，调用硬件资源去安装多个系统。

目前主流的虚拟化平台有 VMware ESXi、Microsoft Hyper-V、Redhat KVM 和 XEN。在最初的 X86 虚拟化技术平台中，XEN 是典型的半虚拟化平台，ESXi 是典型的全虚拟化平台，但是全虚拟化是大势所趋，现在这几种主流虚拟化平台都支持全虚拟化。

1.2.3 资源池定义及特点

资源池（Resource Pool，RP），是指云计算数据中心中所涉及的各种硬件和软件的集合。资源池是数据中心重要的基础设施组成部分，在基础设施云架构下，计算资源、存储资源、网络资源被封装整合为资源池。资源池要符合集约化的特点，要能够进行各种资源的灵活调配，起到资源利用削峰填谷的作用。

资源池主要有以下特点：

（1）统一管理　资源池中的各类资源要进行统一管理、统一调配，不能在单一资源上出现瓶颈。

（2）负载均衡　资源利用率要能够自动达到平衡，不能出现无法提供服务的局部热点。

（3）资源可控　资源池可设定资源高低水位线，初始化时，含有资源对象的最小数目称为低水位线，资源对象所能达到的最大数目称为高水位线。资源请求达到资源池供给能力一定比例时，启动扩容，超过供给能力时，要能够动态迁移至安全线以下。

（4）自我修复　资源池要有完备的高可用措施，一般故障能够无需人为介入即可自动故障恢复，重大故障能够保证在 RPO（Recovery Point Objective，恢复点目标）、RTO（Recovery Time Objective，恢复时间目标）。

（5）成本风险兼顾　考虑成本与风险的平衡。

1.2.4　云计算服务模式

当前，几乎所有的知名 IT 提供商、互联网提供商，甚至电信运营商都在向云计算进军，都在提供相关的云计算服务。但归纳起来，从用户体验的角度出发，当前云计算主要分为三种服务模式。云计算服务模式如图 1-1 所示。

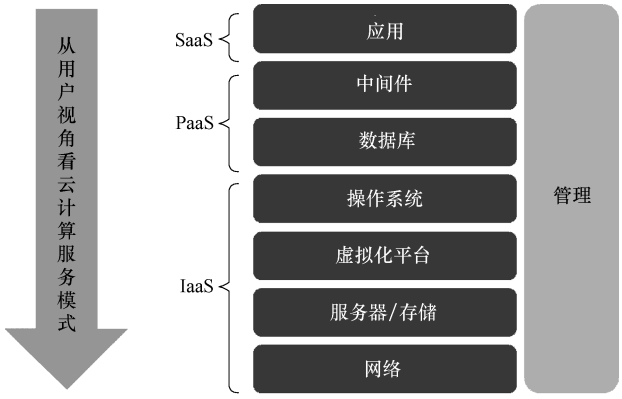


图 1-1　云计算服务模式

1. IaaS（基础设施即服务）

IaaS 是将对所有计算基础设施的使用提供给消费者的服务。它包括处理 CPU、内存、存储、网络和其他基本的计算资源，用户能够按需任意部署和运行各类软件，包括操作系统、数据库、中间件以及各类应用程序。消费者不控制也不管理任何云计算基础设施，但可以控制操作系统的选择、计算能力、存储空间以及部署的应用，也可能获得有限的网络组件（例如路由器、防火墙、负载均衡）的控制。

业界比较典型的 IaaS（基础设施即服务）如 Amazon 为个人和企业客户提供虚拟服务器和虚拟存储的服务。值得一提的是，IaaS 很好地实现了云计算按需付费的理念，通过“弹性云”用户可只在需要时才接入这些基础设施资源，并只为自己使用的部分。

2. PaaS（平台即服务）

PaaS 实际上是指将软件研发的平台作为一种服务，提交给用户。PaaS 可以将现有各种业务能力进行整合，具体可以归类为应用服务器、业务能力接入、业务引擎、业务开放平台，向下根据业务能力需要测算基础服务能力，通过 IaaS 提供的 API 调用硬件资源，向上提供业务调度中心服务，实时监控平台的各种资源，并将这些资源通过 API

（Application Programming Interface，应用程序编程接口）开放给 SaaS 用户。用户或者厂商基于 PaaS 平台可以快速开发自己所需要的应用和产品。同时，通过 PaaS 平台开发的应用能更好地搭建基于 SOA（Service-Oriented Architecture，面向服务的体系结构）架构的企业应用。

PaaS 所提供的服务与其他的服务最根本的区别是 PaaS 提供的是一个基础平台，而不是某种应用。在传统的观念中，平台是向外提供服务的基础。一般来说，平台作为应用系统部署的基础，是由应用服务提供商搭建和维护的，而 PaaS 颠覆了这种概念，由专门的平台服务提供商搭建和运营该基础平台，并将该平台以服务的方式提供给应用系统运营商。

对于小型企业和初创型企业来说，PaaS 也是比较有用的，因为这些企业并没有广泛的、具有较高依赖性的旧应用程序需要迁移。PaaS 的多租户特性可实现应用程序和数据资源的最大数量共享，同时让开发资源继续专注于应用程序的交付和连接，而不是开发和支持数据库资源。PaaS 的未来发展空间似乎在小型企业和初创企业，这类公司由于不依赖于与旧应用程序的集成而更适于在云计算中进行应用程序开发。

3. SaaS（软件即服务）

SaaS 是一种通过 Internet 提供软件的模式，厂商将应用软件统一部署在自己的服务器上，客户可以根据自己实际需求，通过互联网向厂商定购所需的应用软件服务，按定购的服务多少和时间长短向厂商支付费用，并通过互联网获得厂商提供的服务。用户不用再购买软件，而改用向提供商租用基于 Web 的软件，来管理企业经营活动，且无需对软件进行维护，服务提供商会全权管理和维护软件，软件厂商在向客户提供互联网应用的同时，也提供软件的离线操作和本地数据存储，让用户随时随地都可以使用其定购的软件和服务。对于许多小型企业来说，SaaS 是采用先进技术的最好途径，它消除了企业购买、构建和维护基础设施和应用程序的需要。

SaaS 软件应用服务经过多年的发展，已经开始从 SaaS1.0 的阶段慢慢进化到 SaaS2.0 的阶段。类似于 Web1.0 与 Web2.0 的概念，SaaS1.0 更多地强调由服务提供商本身提供全部应用内容与功能，应用内容与功能的来源是单一的；而 SaaS2.0 阶段，服务运营商在提供自身核心 SaaS 应用的同时，还向各类开发伙伴、行业合作伙伴开放一套具备强大定制能力的快速应用定制平台，使这些合作伙伴能够利用平台迅速配置出特定领域、特定行业的 SaaS 应用，与服务运营商本身的 SaaS 应用无缝集成，并通过服务运营商的门户平台、销售渠道提供给最终企业用户使用，共同分享收益。

SaaS 通过租赁的方式提供软件服务，省略了软件安装实施过程中一系列专业并复杂的环节，使软件的实施使用变得简单易掌握。SaaS 模式软件的开发基于“能完全替代传统管理软件功能”这样的要求，并提供在线服务和先进的管理思想，实现销售、生产、采购、财务等多部门多角色在同一个平台上开展工作，实现信息可管控的高度共享和协同。正是由于这些优势，SaaS 发展非常迅速。

1.2.5 云计算部署模式

对于提供者而言，云计算包括以下三种部署模式：公有云、私有云和混合云模式。

1. 公有云

公有云通常指第三方提供商为用户提供的能够使用的云，公有云一般可通过 Internet 使用，可能是免费或成本低廉的。公有云的核心属性是共享资源服务，它所有的服务是供别人使用，而不是自己用。目前，典型的公有云有微软的 Windows Azure Platform、亚马逊的 AWS，以及国内的阿里云等。

对于使用者而言，公有云的最大优点是，其所应用的程序、服务及相关数据都存放在公有云的提供者处，自己无需做相应的投资和建设。目前最大的问题是，由于数据不存储在自己的数据中心，其安全性存在一定风险。同时，公有云的可用性不受使用者控制，这方面也存在一定的不确定性。

2. 私有云

企业自己使用的云，它所有的服务不是供别人使用的，而是供自己内部人员或分支机构使用的。私有云的部署比较适合于有众多分支机构的大型企业或政府部门。随着这些大型企业数据中心的集中化，私有云将会成为他们部署 IT 系统的主流模式。

相对于公有云，私有云部署在企业内部。因此其数据安全性、系统可用性都可由自己控制。

3. 混合云

它是指公有云和私有云的混合。混合云提供既在公共空间又在私有空间中的服务。当公司需要使用既是公有云又是私有云的服务时，选择混合云比较合适。混合云把公有云模式与私有云模式结合在一起，混合云有助于提供所需的、外部供应的扩展。用公有云的资源扩充私有云的能力，可用来在发生工作负荷快速波动时维持服务水平。混合云也可用来处理预期的工作负荷高峰。混合云需要考虑的问题是数据和处理资源之间的关系。

1.3 云计算产业应用

自 SaaS 在 20 世纪 90 年代末出现以来，云计算服务已经经历了 10 多年的发展历程。云计算服务真正受到整个 IT 产业的重视是始于 2005 年亚马逊推出的 AWS 服务，产业界认识到亚马逊建立了一种新的 IT 服务模式。在此之后，谷歌、IBM、微软等互联网和 IT 企业分别从不同的角度开始提供不同层面的云计算服务，云服务进入了快速发展的阶段。云服务正在逐步突破互联网市场的范畴，政府、公共管理部门、各行业企业也开始接受云服务的理念，并开始将传统的自建 IT 方式转为使用云服务方式，云服务将真正进入其产业的成熟期。

最近几年，中国云计算从概念到实际应用有了实质性的发展，尤其是对于云计算基础架构的投资更是发展迅速。国内大中型企业对于建设云计算基础架构的兴趣越来越浓厚。各企业数据中心越来越多地采用虚拟化技术和自动化管理软件，并逐步向云计算基础架构过渡。云基础架构已经成为下一代数据中心（NGDC）建设的主要目标。阿里巴巴、腾讯等国内领先的互联网公司已经开始加紧云计算布局。

云计算已被确定成为国家重点支持项目，这将加速云计算在国内的落地，互联网、运营商以及手机厂商等都将自发性地进入云计算产业，后续扶持政策的陆续出台与产业资金的进入，也将加速整个行业的快速发展。本章节将介绍几个国内外比较典型的云计算平台。

1.3.1 国外云计算应用

1.3.1.1 亚马逊云计算服务

亚马逊是互联网上最大的在线零售商，但是同时也为独立开发人员以及开发商提供云计算服务平台。亚马逊将其云计算平台称为弹性计算云，它是最早提供远程云计算平台服务的公司。

1. 开放的服务

亚马逊将自己的弹性计算云建立在公司内部的大规模集群计算的平台之上，而用户可以通过弹性计算云的网络界面去操作在云计算平台上运行的各个实例（Instance），而付费方式则由用户的使用状况决定，即用户仅需要为自己所使用的计算平台实例付费，运行结束后计费也随之结束。

弹性计算云从严格意义上来看，并不是亚马逊公司推出的第一项云服务，它由亚马逊网络服务的现有平台发展而来。早在 2006 年 3 月，亚马逊就发布了简单存储服务（Simple Storage Service，S3），这种存储服务按照每个月类似租金的形式进行服务付费，同时用户还需要为相应的网络流量进行付费。亚马逊网络服务平台使用 REST（Representational State Transfer，表述性状态传递）和简单对象访问协议（Simple Object Access Protocol，SOAP）等标准接口，用户可以通过这些接口访问到相应的存储服务。

2007 年 7 月，亚马逊公司推出了简单队列服务（Simple Queue Service，SQS），这项服务使托管主机可以存储计算机之间发送的信息。通过这一项服务，应用程序编写人员可以在分布式程序之间进行数据传递，而无需考虑信息丢失的问题。通过这种服务方式，即使信息的接收方还没有模块启动也没有关系。服务内部会缓存相应的信息，而一旦有信息接收组件被启动运行，则队列服务将信息提交给相应的运行模块进行处理。同样的，用户必须为这种信息传递服务进行付费使用，计费的规则与存储计费规则类似，依据信息的个数以及信息传递的大小进行收费。

在亚马逊提供上述服务的时候，并没有从头开始开发相应的网络服务组件，而是对公司已有的平台进行优化和改造，一方面满足了本身网络零售购物应用程序的需求，另一方面也供外部开发人员使用。

在开放了上述的服务接口之后，亚马逊公司进一步在此基础上开发了 EC2 系统，并且开放给外部开发人员使用。

2. 灵活的工作模式

亚马逊的云计算模式沿袭了简单易用的传统，并且建立在亚马逊公司现有的云计算基础平台之上。弹性计算云用户使用客户端通过 SOAP over HTTPS 协议来实现与亚马逊弹性计算云内部的实例进行交互。使用 HTTPS（Hyper Text Transfer Protocol over Secure Socket Layer，网络协议）协议的原因是为了保证远端连接的安全性，避免用户数据在传输的过程中造成泄露。因此，从使用模式上来说，弹性计算云平台为用户或者开发人员提供了一个虚拟的集群环境，使得用户的应用具有充分的灵活性，同时也减轻了云计算平台拥有者（亚马逊公司）的管理负担。

而弹性计算云中的实例是一些真正在运行中的虚拟机服务器，每一个实例代表一个运行中的虚拟机。对于提供给某一个用户的虚拟机，该用户具有完整的访问权限，包括针对此虚拟机的管理员用户权限。虚拟服务器的收费也是根据虚拟机的能力进行计算的，因此，实际上用户租用的是虚拟的计算能力，简化了计费方式。在弹性计算云中，提供了三种不同能力的虚拟机实例，具有不同的收费价格。例如，其中默认的也是最小的运行实例是 1.7GB 的内存，1 个 EC2 的计算单元，160GB 的虚拟机内部存储容量，是一个 32 位的计算平台，收费标准为每小时 10 美分。在当前的蓝计算平台中，还有两种性能更加强劲的虚拟机实例可供使用，当然价格也更加昂贵一点。

由于用户在部署网络程序的时候，一般会使用超过一个运行实例，需要很多个实例共同工作。弹性计算云的内部也架设了实例之间的内部网络，使得用户的应用程序在不同的实例之间可以通信。在弹性计算云中的每一个计算实例都具有一个内部的 IP（Internet Protocol，网络互联协议）地址，用户程序可以使用内部 IP 地址进行数据通信，以获得数据通信的最好性能。每一个实例都具有外部的地址，用户可以将分配给自己的弹性 IP 地址分配给自己的运行实例，使得建立在弹性计算云上的服务系统能够为外部提供服务。当然，亚马逊公司也对网络上的服务流量计费，计费规则也按照内部传输以及外部传输进行分开。

总而言之，亚马逊通过提供弹性计算云，减少了小规模软件开发人员对于集群系统的维护，并且收费方式相对简单明了，用户使用多少资源，只需要为这一部分资源付费即可。这种付费方式与传统的主机托管模式不同。传统的主机托管模式让用户将主机放入到托管公司，用户一般需要根据最大或者计划的容量进行付费，而不是根据使用情况进行付费，而且可能还需要保证服务的可靠性、可用性等，付出的费用更多，而很多时候，服务并没有进行满额资源使用。而根据亚马逊的模式，用户只需要为实际使用情况付费即可。

在用户使用模式上，亚马逊的弹性计算云要求用户要创建基于亚马逊规格的服务器映像（名为亚马逊机器映像即 Amazon Machine Image，AMI）。弹性计算云的目标是服务器映像能够拥有用户想要的任何一种操作系统、应用程序、配置、登录和安全机制，但是当前情况下，它只支持 Linux 内核。通过创建自己的 AMI，或者使用亚马逊预先为

用户提供的 AMI，用户在完成这一步骤后将 AMI 上传到弹性计算云平台，然后调用亚马逊的应用编程接口（API），对 AMI 进行使用与管理。AMI 实际上就是虚拟机的映像，用户可以使用它们来完成任何工作，例如运行数据库服务器，构建快速网络下载平台，提供外部搜索服务甚至可以出租自己具有特色的 AMI 而获得收益。用户所拥有的多个 AMI 可以通过通信而彼此合作，就像当前的集群计算服务平台一样。

在弹性计算云的将来发展过程中，亚马逊也规划了如何在云计算平台之上帮助用户开发 Web2.0 应用程序。亚马逊认为除了它所依赖的网络零售业务之外，云计算也是亚马逊公司的核心价值所在。可以预见，在将来的发展过程中，亚马逊必然会在弹性计算云平台上添加更多的网络服务组件模块，为用户构建云计算应用提供方便。

1.3.1.2 微软云计算服务

在互联网时代，微软的愿景是希望借助互联网和软件的力量，为用户创造跨越不同设备的无缝体验。云计算时代的开启加速了这个新愿景的实现。

微软认为，未来的互联网世界将会是“云 + 端”的组合，在这个以“云”为中心的世界里，用户可以便捷地使用各种终端设备访问云中的数据和应用，这些设备可以是计算机和手机，甚至是电视等大家熟悉的各种电子产品，同时用户在使用各种设备访问云中的服务时，得到的是完全相同的无缝体验。云计算平台是现有 IT 和互联网技术以及业务模型逐渐演变的结果，而一个成功的云计算平台可以最大限度地发挥现有软件开发经验、能力和各种资源。长期以来，微软致力于云计算技术和服务的不断创新，在动态数据中心、私有云以及公有云等方面开展了卓有成效的探索和实践。

2008 年 10 月，微软发布了自己的公有云计算平台——Windows Azure Platform，由此拉开了微软的云计算大幕。

1. 微软的云计算战略及特点

微软的云计算战略包括三大部分，目的是为自己的客户和合作伙伴提供三种不同的云计算运营模式：

① 微软运营：微软自己构建及运营公有云的应用和服务，同时向个人消费者和企业客户提供云服务。例如，微软向最终使用者提供的 Online Services（在线服务）和 Windows Live（Web 服务平台）等服务。

② 伙伴运营：ISV/SI（Independent Software Vendor/System Integration，独立软件供应商/系统集成）等各种合作伙伴可基于 Windows Azure Platform 开发 ERP（Enterprise Resource Planning，企业资源计划）、CRM（Customer Relationship Management，客户关系管理）等各种云计算应用，并在 Windows Azure Platform 上为最终使用者提供服务。另外一个选择是，微软运营在自己的云计算平台中的商务办公在线套装软件（Business Productivity Online Suite，BPOS）产品也可交由合作伙伴进行托管运营。BPOS 主要包括 Exchange Online、SharePoint Online、Office Communications Online 和 LiveMeeting Online 等服务。

③ 客户自建：客户可以选择微软的云计算解决方案构建自己的云计算平台。微软可以为用户提供包括产品、技术、平台和运维管理在内的全面支持。

与其他公司的云计算战略不同，微软的云计算战略有三个典型特点：即软件 + 服务、平台战略和自由选择。

（1）软件 + 服务　在云计算时代，一个企业是否就不需要自己部署任何的 IT 系统，一切都从云中计算平台获取呢？或者反过来，企业还是像以前一样，全部的 IT 系统都自己部署，不从云中获取任何的服务呢？

很多企业认为有些 IT 服务适合从云中获取，如 CRM、网络会议、电子邮件等；但有些系统不适合部署在云中，如自己的核心业务系统、财务系统等。因此，微软认为理想的模式将是“软件 + 服务”，即企业既会从云中获取必需的服务，也会自己部署相关的 IT 系统。

“软件 + 服务”可以简单描述为两种模式：

① 软件本身架构模式是软件加服务。例如，杀毒软件本身部署在企业内部，但是杀毒软件的病毒库更新服务是通过互联网进行的，即从云中获取。

② 企业的一些 IT 系统由自己构建，另一部分向第三方租赁、从云中获取服务。例如，企业可以直接购买软硬件产品，在企业内部自己部署 ERP 系统，而同时通过第三方云计算平台获取 CRM、电子邮件等服务，而不是自己建设相应的 CRM 和电子邮件系统。

“软件 + 服务”的好处在于，既充分继承了传统软件部署方式的优越性，又大量利用了云计算的新特性。

（2）平台战略　为客户提供优秀的平台一直是微软的目标。在云计算时代，平台战略也是微软的重点。

在云计算时代，有三个平台非常重要，即开发平台、部署平台和运营平台。Windows Azure Platform 是微软的云计算平台，其在微软的整体云计算解决方案中发挥关键作用。它既是运营平台，又是开发、部署平台；既可运行微软的自有应用程序，也可以开发部署用户或 ISV 的个性化服务；既可以作为 SaaS 等的应用模式的基础，又可以与微软线下的系列软件产品相互整合和支撑。事实上，微软基于 Windows Azure Platform，在云计算服务和线下客户自有软件应用方面都拥有了更多样化的应用交付模式、更丰富的应用解决方案、更灵活的产品服务部署方式和商业运营模式。

（3）自由选择　为用户提供自由选择的机会是微软云计算战略的第三大典型特点。这种自由选择表现在以下三个方面：

1）用户可以自由选择传统软件或云服务两种方式。自己部署 IT 软件、采用云服务或者两者都用，无论是用户选择哪种方式，微软的云计算都能支持。

2）用户可以选择微软不同的云服务。无论用户需要的是 SaaS、PaaS 还是 IaaS，微软都有丰富的服务供其选择。微软拥有全面的 SaaS，包括针对消费者的 Live 服务和针对企业的 Online 服务；也提供基于 Windows Azure Platform 的 PaaS；还提供数据存储、计算等 IaaS 和数据中心优化服务。用户可以基于任何一种服务模型选择使用云计算的相关技术、产品和服务。

3）用户和合作伙伴可以选择不同的云计算运营模式。微软提供多种云计算运营模

式。用户和合作伙伴可直接应用微软运营的云计算服务；用户也可以采用微软的云计算解决方案和技术工具自建云计算应用；合作伙伴还可以选择运营微软的云计算服务或自己在微软云平台上开发云计算应用。

2. 微软云计算解决方案

微软主要有三类云计算解决方案，即 Live 和 Online 解决方案、Windows Azure Platform 解决方案，以及动态云解决方案。

（1）Live 和 Online 解决方案　微软的云计算应用既有针对消费者的服务，也有针对企业的服务。对于用户而言，这些云计算解决方案对应的客户自有软件（即客户自己购买或构建的软件并安装运行在自己的环境中）都是需求最广、用户最熟悉的应用软件，微软提供相应的云计算应用模式，为用户提供更多的应用模式选择，让应用这些软件服务的用户可以缩减系统建设投资、降低软件升级运维成本、随需随用，而这恰恰是云计算模式的应用优势。微软当前提供的云计算解决方案已包括操作系统、办公软件、即时通信、邮件、中间件、应用管理软件等系列产品，为消费者和企业用户提供了全面的云计算应用选择。

（2）Windows Azure Platform 解决方案　Windows Azure Platform 是一个运行在微软数据中心的云计算平台，它包括一个云计算操作系统和一个为开发者提供的服务集合。开发人员创建的应用既可以直接在该平台中运行，也可以使用该云计算平台提供的服务。相比较而言，Windows Azure Platform 延续了微软传统软件平台的特点，能够为客户提供熟悉的开发体验。

用户已有的许多应用程序都可以相对平滑地迁移到该平台上运行。另外，Windows Azure Platform 还可以按照云计算的方式按需扩展，在商业开发时可以节省开发部署的时间和费用。

Windows Azure Platform 包括 Windows Azure、SQL Azure（数据库服务）和 Windows Azure Platform AppFabric（开发服务）。Windows Azure 可看成一个云计算服务的操作系统；SQL Azure 是云中的数据库；AppFabric 是一个基于 Web 的开发服务，它可以把现有应用和服务与云平台的连接和互操作变得更为简单。AppFabric 让开发人员可以把精力放在应用逻辑上而不是在部署和管理云服务的基础架构上。

1）Windows Azure。Windows Azure 是一个云服务的操作系统，它提供了一个可扩展的开发环境、托管服务环境和服务管理环境，这其中包括提供基于虚拟机的计算服务和基于 Blobs、Tables（表存储）、Queues（队列存储）、Drives 等的存储服务。Windows Azure 为开发者提供了托管的、可扩展的、按需应用的计算和存储资源，还为开发者提供了云平台管理和动态分配资源的控制手段。Windows Azure 是一个开放的平台，支持微软和非微软的语言和环境。开发人员在构建 Windows Azure 应用程序和服务时，不仅可以使用熟悉的 Microsoft Visual Studio、Eclipse 等开发工具，同时 Windows Azure 还支持各种流行的标准与协议，包括 SOAP、REST、XML（Extensible Markup Language，可扩展标记语言）和 HTTPS 等。

Windows Azure 主要包括三个部分，一是运营应用的计算服务；二是数据存储服务；

三是基于云平台进行管理和动态分配资源的控制器（Fabric Controller）。

① 计算服务。计算服务能够运行多种不同的应用，并支持大量并发用户的应用。Windows Azure 提供计算服务的方式是根据需要把计算任务同时分配到多台虚拟服务器上。Windows Azure 虚拟机运行 64 位的 Windows Server 2008，由 Hyper-V 产品进行云中改造而来。开发者只要通过浏览器接入 Windows Azure 门户，用 Windows Live ID 进行注册登录，就可以开始使用平台提供的服务。

② 存储服务。Windows Azure 存储不是一个关系型数据系统，并且它的查询语言也不是 SQL（Structured Query Language，结构化查询语言），主要被设计用来支持建于 Windows Azure 上的应用，它提供了更简单容易扩展的存储。

存储服务应用可以通过很多不同方式来运用数据，Windows Azure Storage（微软云存储）服务提供了多种选择，包括 Blobs、Tables、Queues 和 Drives。Blobs 非常便于存储二进制数据，比如 JPEG 图片或 MP3 文档等多媒体数据；Tables 是可扩展存储，通过多个虚拟机对分布式数据进行扩展和收缩，这比使用一个标准的关系型数据库更为有效；Queues 的主要功能是提供一种 Web Role Instance 和 Worker Role Instance 沟通的方式；Drives 的主要作用是为 Windows Azure 应用程序提供一个 NTFS 文件卷，这样应用程序可以通过 NTFS（New Technology File System，Windows NT 环境的文件系统）API 来访问存储的数据。无论数据以 Blobs、Tables、Queues 或 Drives 任何方式存储，Windows Azure Storage 都会将所有数据复制三次，任何一个拷贝的丢失都不是致命的，任何一个应用都能够保证立即准确读取原始数据信息。

2）Windows Azure Platform AppFabric。AppFabric 为本地应用和云中应用提供了分布式的基础架构服务。在云计算中存储数据与运行应用都很重要，但是我们还需要一个基于云的基础架构服务。这个基础架构服务应该既可以被客户自有软件应用，又可以被云服务应用。AppFabric 能够使客户自有应用与云应用之间进行安全连接和信息传递。AppFabric 目前主要提供互联网服务总线（Service Bus）和访问控制（Access Control）服务。

3）SQL Azure。SQL Azure 是一个云的关系型数据库，可以在任何时间提供客户数据应用。SQL Azure 基于 SQL Server 技术构建，由微软基于云进行托管，提供的是可扩展、多租户、高可用性的数据库服务。SQL Azure 为用户提供了内置的高可用性和容错能力，且无需客户进行实际管理。SQL Azure Database 支持 TDS（Tag Data Standard，标签数据标准）和 Transact-SQL（SQL 在 Microsoft SQL Server 上的增强版，T-SQL），客户可以使用现有技术在 T-SQL 上进行开发，还可以使用与现有的客户自有数据库软件相对应的关系型数据模型。SQL Azure Database 提供的是一个基于云的数据库管理系统，它能够整合现有工具集，并提供与客户自有软件的对应性。

（3）动态云解决方案　动态云解决方案是微软提供的基于动态数据中心技术的云计算优化和管理方案。企业可以基于该方案快速构建面向内部使用的私有云平台，服务提供商也可以基于该方案在短时间内搭建云计算服务平台对外提供服务。微软动态云能够让用户自己动态管理数据中心的基础设施（包括服务器、网络和存储等），包括开

通、配置和安装等。其核心价值在于，它可以帮助用户提高 IT 基础设施资源的利用效率，提升基础设施的应用和管理水平，实现计算资源的动态优化。

微软动态云解决方案能够帮助企业创建虚拟环境来运行应用，用户可以按照需要弹性分配适当的应用配置，并且支持动态扩展。具体功能特点包括部署、24h ×7 监控、优化、保护和灵活适配五个方面。其中，部署功能包括部署服务器、网络和存储服务等资源；灵活的自我管理。24h ×7 监控功能包括收集运行情况数据来更好地满足 SLA（Service Level Agreement，服务等级协议）需要，监控资源利用情况；客户自我监控。优化功能包括持续监控和在不影响或少影响应用运行的情况下主动根据运行需要来调整和迁移服务器；根据需要分配“合适”的资源，不超配和低配。保护功能包括防病毒、垃圾访问过滤和防火墙等；应用和数据备份；保证 99.9% 正常运行时间和基础设施的物理安全。灵活适配功能包括容易调整环境、部署新资源；存储、带宽等根据需要可以动态调整；支持不同虚拟技术，并可以管理不同类型的虚拟机。

1.3.2 国内云计算应用

1.3.2.1 腾讯云计算服务

腾讯做云计算不是人云亦云，也不是孤立地看待云计算本身的商业价值，而是腾讯开放共享战略的一个重要组成部分。

早年的 QQ 空间用户可能经历过这样的体验：进入 QQ 空间后，界面上有一个小地球在不停地跳动，有时候在这个界面上需要等几个小时才能进入个人空间。这是早期 QQ 空间应对海量访问考验时为了避免雪崩而采取的策略，将用户引导到一个等待界面。

今天互联网用户对产品的要求比那时高得多，很难想象他们再像当年那样对一个优秀的应用如此包容和谅解。而应对海量访问的能力，绝不是一朝一夕能够形成的，QQ 空间今天可以轻松地支撑起千万级的同时在线用户，是长时间技术积累的结果。

腾讯打造云计算平台，就是要把腾讯多年积累下来的海量访问技术和运营能力，和所有互联网行业的创业者分享，让他们少走弯路，更容易创业成功。这种能力包括了：海量运维、海量计算、海量存储、海量数据分析、云安全、支付营销以及客服等各种能力。在分享的同时，腾讯不希望把开发者捆绑在自己的云平台上，而是希望能够用一种非常快捷、简单的方式，把这些能力分享给各种各样架构、开发语言和应用场景的互联网应用，能够去适应业界主流的开发模式，让大多数开发者可以在已经开发好的应用基础之上，不经过大的改动，就可以接入到腾讯的云计算平台。

从产品的角度出发，腾讯云计算平台针对广大开发者推出了以下产品：

1. 云存储

每款应用都有其生命周期，而社交游戏的生命周期相对较短，往往数周之内迅速达到用户峰值，接着步入稳定期和衰减期。在此期间，开发者往往疲于奔命，需要在访问量不断增加时进行持续的数据层重构，需要频繁地进行机器扩容和数据迁移，在后期又

要进行数据合并和资源退出。

腾讯根据数据层研发过程中的实际经验，以及应用从小到大的发展过程中遇到的各类技术问题，推出了自助化和运维透明化的海量存储服务。腾讯相继推出了基于 Key/Value（分布式存储）的全内存持久化存储方案（Cloud Memcache，CMEM）和兼容 MySQL（关系型数据库管理系统）的存储方案 CDB（Cloud Database），以帮助接入腾讯社区开放平台的第三方应用应对同时在线用户数在短时间内迅猛增加时面临的访问压力。

CMEM 和 CDB 针对不同的服务场景，开发者可以根据成本和性能来选择不同的实例。使用 CMEM 和 CDB，开发者无需关注数据层具体实现，通过云存储解决数据层性能、容量、安全及可用性问题，而将更多的精力聚焦于应用本身的逻辑开发和产品运营。

2. 虚拟设备

腾讯推出了基于虚拟机的设备管理方案，除了适应开发者的各种架构需求外，也提供了针对应用或流量的监控、扩容、缩容功能，并提供多种方式保护用户计算资源的安全，包括防 DDOS（Distributed Denial of Service，分布式拒绝服务）攻击、访问控制、业务隔离等。

目前的虚拟机方案是基于 Xen（开放源代码虚拟机监视器）技术，其管理简单、接入便捷。该方案以多种不同配置类型的虚拟机为服务单元，用户在 Web 页面上即可进行计算资源的申请、使用和退还。

即将推出的基于 Linux Container（内核虚拟化技术，LXC）的虚拟机方案，将提供性能更高、安全性更好的服务，并使资源管理更加精细化。

3. 应用安全

一个成功的互联网应用在安全方面会面临很多挑战。如果不加以保护，外挂、恶意信息等很快会纷至沓来。

腾讯基于自身多年运营社交网站（Social network site，SNS）应用的经验，为开发者提供的数据安全和实时数据分析服务，可解决大部分开发者在安全方面意识淡薄或经验不足的短板，对保护用户安全，提升用户体验非常重要。

腾讯云计算平台目前为开发者提供的安全服务包括：

1）运维安全。腾讯建立了完备的应用安全评估体系，对第三方应用进行安全评级，并提供网络攻击防护、防止入侵、挂马检测、漏洞扫描等服务。

2）业务安全。腾讯提供 OpenAPI（开放接口）给开发者调用，根据应用上报的数据进行实时分析，开发者可获取外挂用户信息并针对用户账号以及 IP 实施精准的限制策略。

3）信息安全。腾讯提供 OpenAPI 给开发者调用，以实现垃圾消息过滤，评论频率控制，以及验证码下发等功能。

4. 运营数据分析

腾讯产品拥有 6 亿注册用户和强大的用户黏性，这些都依托于公司多年积累下来的

海量数据收集和处理分析能力，以及海量运营方法。这些能力和方法，将会逐渐开放给第三方开发者。

目前开发者已经可以看到应用的一些简单的运营分析数据，开发者可以进行初步的产品挖掘，为产品优化提供决策和支持。

未来，腾讯将提供一些成熟的运营分析模式，例如传播模型、用户画像、漏斗模型、营销模型等。这些工具可以让开发者可以深度挖掘产品，持续优化产品特性，提升用户体验，并通过快速、高效的分析研究如何推广应用、存留用户和提升利润，使应用更好更快地发展。

5. 自动化运维

腾讯为开发者提供的自助化运维平台，可以让开发者自主维护自己的应用，实施各种资源申请，应用上线、发布、版本升级、故障处理、回滚、扩容、缩容等。这些流程的自动化可以极大地减轻开发者的运维负担。

6. 支付营销体系

腾讯成熟的用户账户体系，以及依据这个体系建立的支付营销体系，为产品的在线营销提供强有力的支持。

已经开放的腾讯支付体系让开发者可以用 Q 币兑换应用中的虚拟货币。同时腾讯也在推动一种让开发者可以在应用中直接使用 Q 币的方式。

后续，活动营销将被打包成服务，让开发者可以快捷地进行各种活动推广。

7. 客服系统

腾讯向第三方应用开发者传递“一切以用户为依归”的理念。让这个理念得以实施的，以及让几亿注册用户获取优质服务的，是腾讯完备的客服系统。开发者接入这个客服系统后，不仅可以使用完备的系统工具，也可以复用腾讯积累的优秀实践、方法论、标准和规范，从而在提高服务质量、吸取用户反馈、优化产品、加强用户黏性上做出进一步的努力。

1.3.2.2 阿里云计算服务

阿里云，阿里巴巴集团旗下云计算品牌，全球领先的云计算技术和服务提供商，创立于 2009 年，致力于成为“DT 时代中国商业发展的基础设施”，成为阿里巴巴集团未来 10 年“云 + 端”的重要战略。

如图 1-2 所示，在最近 10 年，中美两国的三家超级云服务商（亚马逊、谷歌、阿里云）持续推动全球云计算市场和技术的不断发展。2006 年，谷歌、亚马逊推出的云服务标志着“美国云计算元年”。2009 年阿里云成立，并面向商用市场改造淘宝分布式集群技术，则标志着“中国云计算元年”的到来。阿里云持续多年投入大量研发资源，是世界上第一个对外提供 5K 云计算服务能力的公司，由 5000 台服务器组成的飞天 5K 服务器集群，拥有超过 10 万核计算能力、100PB 存储空间，可处理 15 万并发任务数，承载亿级别文件数目。100TB 排序能在 30min 完成，超过了当时 Yahoo 在 Sort Benchmark（排序基准评估机构）排序测试 Daytona Gray Sort 所创造的世界纪录——71min。

阿里云全球数据中心遍及北京、杭州、青岛、深圳、香港、美国硅谷、内蒙（筹

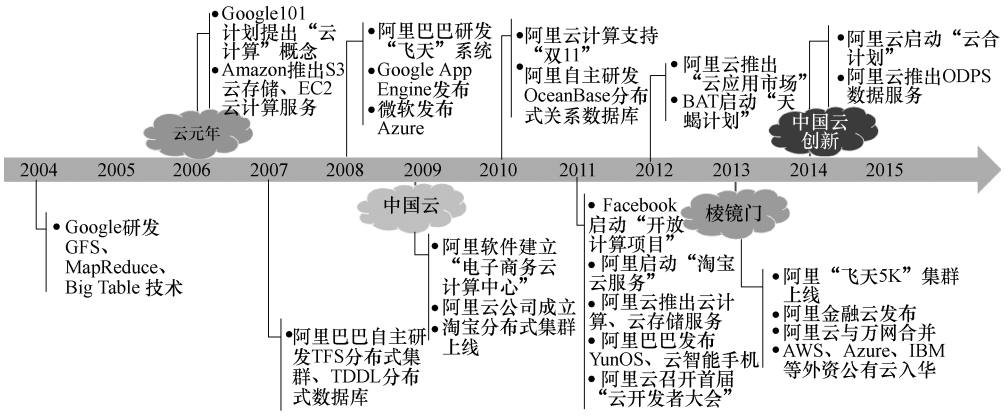


图 1-2　中美三大云服务商发展历程

注：Google：谷歌公司；GFS：可扩展的分布式文件系统，用于大型的、分布式的、对大量数据进行访问的应用；MapReduce：一种编程模型，用于大规模数据集（大于 1TB）的并行运算；BigTable：Google 设计的分布式数据存储系统，用来处理海量的数据的一种非关系型的数据库；Amazon：亚马逊公司；S3：一个公开的服务，Web 应用程序开发人员可以使用它存储数字资产，包括图片、视频、音乐和文档；EC2：亚马逊弹性计算云（EC2，Elastic Compute Cloud）是一个让使用者可以租用云端电脑运行所需应用的系统；TFS：Taobao File-Syste，是一个高可扩展、高可用、高性能、面向互联网服务的分布式文件系统；TDDL：Taobao Distributed Data Layer，淘宝分布式数据库层，主要解决了分库分表对应用的透明化以及异构数据库之间的数据复制；Google App Engine：是一种让您可以在 Google 的基础架构上运行您的网络应用程序；Azure：微软的专业的国际化公有云平台；OceanBase：一个支持海量数据的高性能分布式数据库系统；FaceBook：脸书公司；YunOS：阿里巴巴集团旗下的一款智能设备操作系统产品；BAT：百度、阿里巴巴和腾讯三家公司简称；ODPS：Open Data Processing Service，由阿里云自主研发，提供针对 TB/PB 级数据、实时性要求不高的分布式处理能力，应用于数据分析、挖掘、商业智能等领域；AWS：Amazon Web Services，亚马逊公司提供的云服务；IBM：国际商业机器公司。

建中）、欧洲（筹建中）、西亚（筹建中）、新加坡（筹建中）等地，针对不同行业的特点，阿里云提供了政务云、游戏云、金融云、电商云、移动云、医疗云、交通云等行业解决方案。根据 IDC 调研报告，阿里云是国内最大的公有云计算服务提供商，2014 年服务客户数超过 140 万，目前每天注册成为阿里云会员企业达到 4000 多家，高速增长的客户群体遍布互联网、移动 App、音视频、游戏、电商等各个领域。

其中，阿里金融云是为金融行业量身定制的云计算服务，具备低成本、高弹性、高可用、安全合规的特性，帮助金融客户实现传统 IT 向云计算的转型，并为客户实现与支付宝、淘宝、天猫的直接对接，助力金融客户业务创新，提升竞争力。阿里金融云数据中心位于杭州、青岛、深圳（筹建中），属于完全独立的合规集群（经过监管部门的合规检查），也是目前业界独有的为金融行业创建的专属云。天然的两地三中心架构，具备 IOS（International Organization for Standardization，国际标准化组织）27001、CSA Star、（云安全国际认证金牌）等保三级、可信云等安全认证。阿里云全部都是自主开发，拥有云操作系统、弹性计算服务（Elastic Compute Service，ECS）、关系型数据库服务（Relation Database Service，RDS）、开放存储服务（Open Source，Software，OSS）、负载均衡（Server Load Balancing，SLB）、开放数据处理服务（Open Data Processing

Service，ODPS）等数十项云技术专利。目前，阿里云拥有数十万台服务器，阿里金融云拥有数万台服务器，还在按需逐渐扩展。

图 1-3 展示了阿里金融云总体架构，通过飞天操作系统，阿里金融云能把多达数十万的大规模计算机集群资源有机连接起来协同工作，对外形成一台通用功能的巨型计算机，在此之上，对外提供了包括弹性计算服务（ECS）、负载均衡服务（SLB）、关系数据库服务（RDS）、开放存储服务（OSS）、内容分发网络（Content Delivery Network，CDN）和具有 6h 处理 100PB 级别数据能力的开放数据处理服务（Open Date Processing Service，ODPS）。

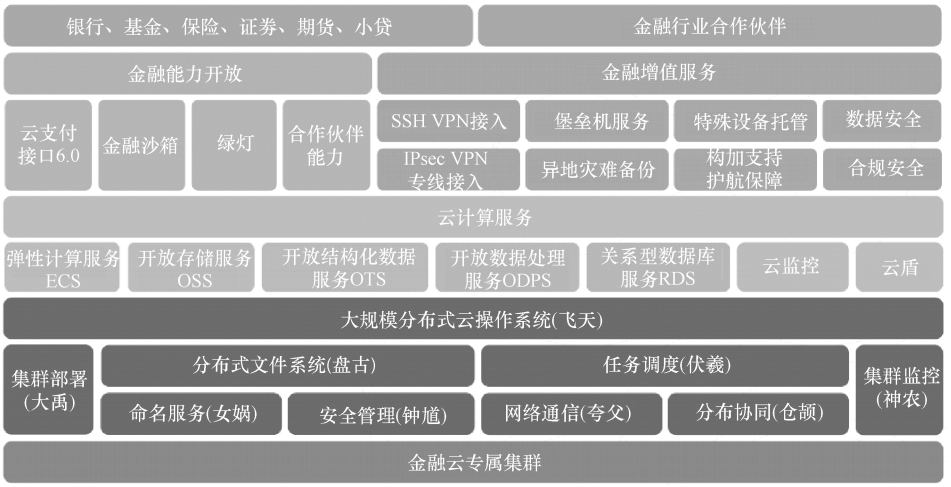


图 1-3　阿里金融云总体架构

注：SSH VPN：采用 SSH（Secure Shell，安全层）协议来实现远程接入的一种 VPN（虚拟专用网络）技术；IPsec VPN：采用 IPSec（因特网协议安全性）协议来实现远程接入的一种 VPN（虚拟专用网络）技术；ECS：弹性计算服务；OSS：开放存储服务；OTS：开放结构化数据服务；ODPS：开放数据处理服务；HDS：关系型数据库服务。

阿里金融云为广大金融企业设计了“系统上云”、“数据融通”、“金融生态”三步走策略，先“金融上云”，再“云上金融”的云计算实施路线（见图 1-4）。当金融企业将自己的业务应用部署在阿里金融云之上时，该企业作为阿里巴巴生态圈的重要客户，即获得在业务层面与金融信息服务平台对接，并利用大数据提升金融业务水平的领先能力，标志着与阿里巴巴全面合作共建金融生态圈的产业序幕已经拉开。

目前金融企业客户通过阿里金融云能够快速对接如下业务资源：

1）芝麻信用。

2）大数据实验室。

3）支付宝服务窗（覆盖 8 亿实名客户、4 亿活跃用户）。

4）招财宝。

5）微贷平台。

全球 IT 市场中，政企市场客户约占 70%，与中小企业大量使用公有云、行业云不

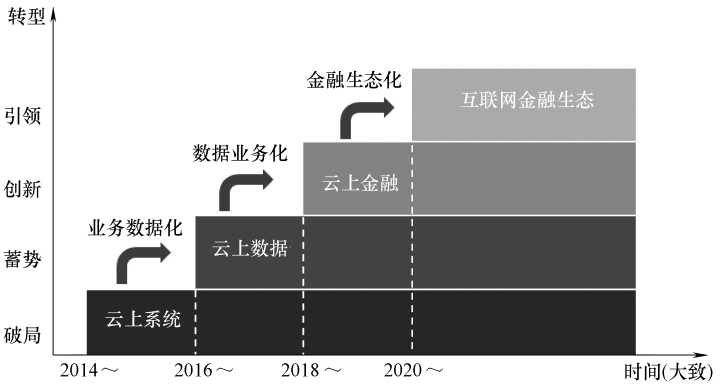


图 1-4　上云路线

同，政企客户希望采用互联网行业“云计算 + 大数据”的 Scale-Out（模向扩展）架构重构基于传统 IOE（IBM、Oracle，EMC）的 Scale-Up 架构，实现 IT 架构水平扩展、按需获得 IT 资源，同时满足行业安全合规和自主可控技术要求，这需要在客户机房独立建设一套“云计算 + 大数据”的基础平台，满足客户“云计算平台 + 专有使用”的独占要求。阿里专有云利用客户自建的机房，将阿里云专有云产品通过软件方式部署在客户机房内的通用 X86 服务器上，为客户应用系统赋予领先成熟的云计算、大数据服务能力。阿里云服务分类如图 1-5 所示。

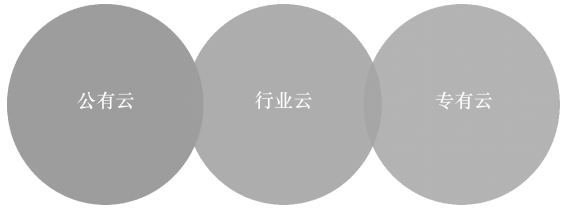


图 1-5　阿里云服务分类

如图 1-6 所示，阿里云专有云位于客户 IT 基础设施硬件层之上，将物理服务器的计算和存储能力、网络设备虚拟化成虚拟计算、分布式存储、软件定义网络，在此基础上提供云数据库、大数据处理、分布式中间件服务，并和客户现有账号体系、运维系统对接融合。

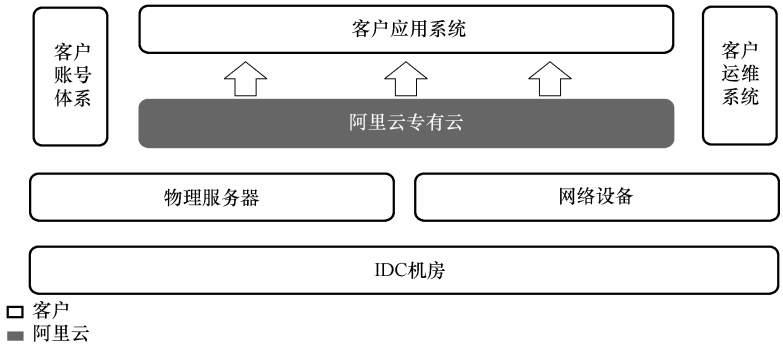


图 1-6　阿里云专有云部署

在表 1-2 中，“公有云”的特点是通过共享资源来实现最优性价比，而“专有云”是通过承担更多基础设施投资、运营工作来换取独立性与控制力，责任、控制权与收益是相匹配的，企业客户针对自身行业业务具体情况选择对自己最有利的云采购/运营模

式，是一种预判未来的战略决策。

表 1-2　阿里云的“专有云”和“公有云”对比

对　　比	专有云	公有云
购买方式	软件 License 或者服务	服务
硬件投资	客户投资	阿里云投资
运维模式	客户或者 ISV 运维	阿里云运维
安全隔离方式	物理隔离	逻辑隔离或物理隔离

如图 1-7 所示，阿里云专有云系统架构充分考虑了分布式系统的高可用性、开放性、健壮性和可扩展性。主备架构的“云中控系统”由阿里云账号、权限、云基础服务（全局位置信息等）、控制台入口、计费计量等关键调度模块组成；企业客户根据业务分布的不同地域建设不同的 Region 区域，每个 Region 至少部署两个可用区（Available Zone，AZ），实现同城双活或者灾备；“飞天”是阿里云的分布式操作系统，负责将标注的 X86 物理服务器连接成一个超级计算机，为上层所有云服务提供基础支撑，具有大规模平滑扩展、高可用性的设计特点。

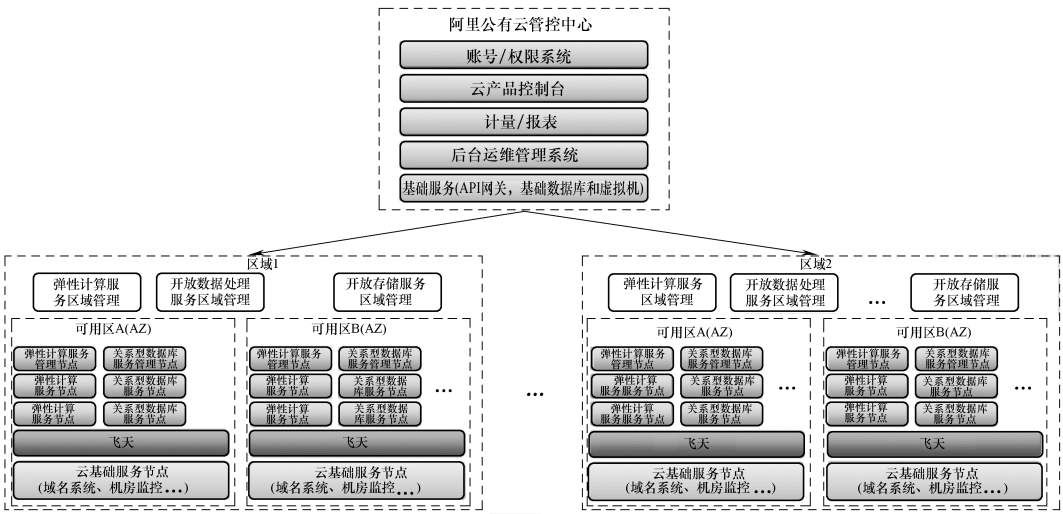


图 1-7　阿里云专有云系统架构

阿里云拥有百万以上客户，金融云拥有 2000 多家客户，其中银行、保险、证券客户 200 多家；P2P（Peer To Peer，点对点网络借款）、小贷、众筹、资管公司等微金融客户 1000 多家。在上云客户中，日交易量超过 5000 万的包括天弘基金（余额宝，全系统上云）、众安保险（全系统上云）；银行包括阿里网商银行；保险包括阳光保险、永安保险、华夏人寿等；证券包括海通证券、银河证券、宏源证券等。专有云包括云上贵州、海关、海淀政务云、证通等。

1.3.2.3　青云云计算服务

青云（QingCloud）成立于 2012 年 4 月，是全球首家实现资源秒级响应并按秒计费

的基础云服务商，致力于为企业用户提供安全可靠、性能卓越、按需、实时的 IT 资源交付平台。其服务涵盖了计算、存储、网络、安全等 IT 基础设施层所需的各个组件以及数据库、缓存等 Half PaaS（通用型 Paas）层服务。QingCloud 采用分布式块存储系统确保高性能 I/O，部署实时异地副本保障数据安全，通过云端 SDN（Software Defined Network，软件定义网络）实现灵活易用的专享私有云服务（Virtual Private Cloud，VPC），提供块设备级的备份与恢复，设计实时 P2P 机器人社区协作确保故障无害，基于自主研发的 IaaS 平台提供原生通用 PaaS，并无限制开放全部功能 API。

自 2013 年 7 月正式开通以来，QingCloud 公有云平台性能多次全面升级，真正实现 IaaS 层的稳定、可靠、安全、高性能、实时、弹性和开放，并在此基础上快速开发和推出基于 QingCloudIaaS 层的 Technical PaaS。除公有云（Public Cloud）业务外，QingCloud 还提供受管云（Managed Cloud）服务，帮助大型企业用户构建和管理私有云平台，并支持数据中心运营商向云计算转型。目前已经部署和运营了 4 个受管云的区域。未来 QingCloud 还将提供包括数据中心基础设施、传输网络、超融合设备和云固件在内的全套云计算解决方案以满足不同类型企业用户的需求。秉持开放的心态，QingCloud 联合产业链上众多优秀的 PaaS、SaaS 与 DevOps（开发运维一体化）服务，整合与构建完整的云计算生态，为未来的科技创新提供稳定、可靠、安全、高性能、弹性、丰富的 IT 资源与开发工具与服务。

1. 公有云服务

自 2013 年 7 月正式开通以来，QingCloud 已经为逾万家企业用户提供服务，帮助企业用户有效提升系统效能，大幅降低 IT 成本与运维投入；帮助初创团队轻资产创业，真正做到 IT 资源平民化和像用水用电一样方便。

QingCloud 始终以满足苛刻的企业级 IT 需求为目标，其客户群既包括来自于视频、游戏、电商、社交、大数据、物联网、移动 App（Application，应用程序）等互联网和移动互联网领域的企业，也包括政府、金融、教育、制造业、建筑业等传统行业的客户。

2. 私有云服务

QingCloud 私有云系统是一套纯自主研发，面向新一代云数据中心架构，软件定义的 IT 资源交付与管理平台。它面向严肃企业，提供稳定可靠、集成统一、弹性伸缩、智能高效并自主可控的云基础架构及服务。QingCloud 私有云系统构建及管理充分体现下述原则：

1）搭建统一的基础架构云计算平台，将这项业务系统融入统一的动态基础架构中，提升服务器资源的利用率。

2）遵循兼容原则，支持计算、存储、网络和安全四个方面的标准接口、规范与协议，同现有传统系统无缝兼容，充分保护企业现有 IT 资产。

3）提供 IaaS 层和 Half PaaS 层的标准化组件，避免厂商锁定。

4）无限接近甚至超越物理设备的稳定性、性能及安全，满足金融行业对 IT 系统的可靠性需求。

5）资源调用实现充分弹性，资源调用极致敏捷。

6）Design for failure（设计时为故障做好准备）通过 P2P 架构摆脱系统对单台物理设备可靠性的依赖。

7）通过云计算平台实现统一管理，改善 IT 资源的运营，提升管理和维护效率，充分借助机器智能（Machine Intelligence）实现系统自运维，最大限度降低运维投入的同时保障故障无害。

8）为今后其他更多系统进行统一化管理基础设施提供良好的基础。

3. 混合云服务

商业银行不同类型的业务对 IT 系统有着不同的需求，通常情况下，私有云解决方案适用于存储核心数据和部署关键业务，而公有云则可以在投资最少、效率最高的前提下为非关键业务提供开发测试环境。QingCloud 支持多种 VPN 及隧道协议，例如：OpenVPN（开源虚拟专用网络）、PPTP（Point to Point Tunneling Protocol，点对点隧道协议）、GRE（Generic Routing Encapsulation，通用路由封装）IPsec（Internet Protocol Security，Internet 协议安全性）和 IPsec 等，帮助企业设计及实现混合云（Hybrid Cloud）部署，便捷地将多地、多形式的 IT 环境协调成一个整体的 IT 环境进行统一管理与调度。除 Internet 外，QingCloud 还将建设与运营骨干网络，支持企业用户通过专线链接位于不同数据中心的 IT 环境。

青云 QingCloud 的核心技术优势体现在以下几个方面：

（1）完整的 IaaS 平台和构建在自有 IaaS 平台上的标准 Half PaaS 服务

1）自主产权的 IaaS 平台：覆盖 IaaS 层计算、存储、网络、安全的全部组件。功能与服务包括主机、高性型硬盘与容量型硬盘、基础网络与私有网络、弹性公网 IP、支持透明代理的负载均衡器、防火墙、映像、备份与恢复、SSH（Secure Shell，建立在应用层和传输层基础上的安全协议）密钥、关系型数据库服务、缓存服务等。

2）构建在自有 IaaS 平台上的标准 Half PaaS 服务：有了完整、稳定、高性能的 IaaS 平台保障，QingCloud 自 2014 年 3 季度起陆续推出了关系型数据库服务、缓存服务等标准 Half PaaS 层服务，未来还会快速推出 AutoScaling（Application Engine）、对象存储（Object Storage）、共享存储（Shared Storage）、消息队列（Messaging）、大数据处理和分析平台等更加丰富、完善的 Half PaaS。

（2）稳定可靠

1）通过实时多重副本保障数据安全。

2）确保为每个用户提供的资源满足 QoS（Quality of Service，服务质量）保障。

3）设计实时 P2P 机器人社区协作确保故障无害，并确保 QingCloud 的系统可以无限水平扩展。

（3）安全　QingCloud 帮助用户阻隔来自云内部和公网上的攻击和非授权访问，并避免因误操作或应用逻辑 bug（系统缺陷）导致的数据丢失。

1）提供防火墙以阻隔来自公网的攻击；默认防火墙端口关闭，以保障用户开发环境不暴露在公网之上。

2）提供块设备级的备份与恢复功能，用户可对主机和多张硬盘（包括系统盘和数据盘）进行在线或离线备份，而不会影响或中断业务；可选增量或全量备份；可从任意一个备份点恢复数据。

（4）软件定义网络

1）通过云端 SDN 实现了私有网络（VPC），可以用虚拟路由器以及交换机对主机进行组网，支持复杂的端口转发功能，并与其他用户实现 100% 的二层隔离。

2）支持多种 VPN（Virtual Private Network，虚拟专用网络）及隧道协议，例如：OpenVPN、PPTP、GRE 和 IPsec 等，帮助企业设计及实现混合云（Hybrid Cloud），便捷地将现有 IT 环境同云端资源协调成一个整体的 IT 环境。

3）基于 QingCloud 的 SDN 网络，实现支持全透明代理等高级功能的负载均衡器。

（5）软件定义存储

1）虚拟资源的性能可以达到或接近物理设备的水平，通过分布式块存储系统实现高性能 I/O（性能型硬盘 I/O 吞吐 128MB/s）。

2）通过实时异地副本保障数据安全。

（6）卓越的性能　QingCloud 倡导云计算资源性能要媲美硬件设备，现已具备支持 90% 以上企业级 IT 应用的能力。

1）虚拟资源性能达到或接近物理设备水平，虚拟化效率 99.98%。

2）存储高性能 I/O（性能型硬盘 I/O 吞吐 128MB/s），未来还将基于 SSD（Solid State Drives，固态硬盘）硬盘推出超高性能存储服务。

3）全部 Half PaaS 层的数据库和缓存等服务全部基于 QingCloud 自有 IaaS 平台部署。

（7）实时与按需　QingCloud 倡导“自由计算”，以极致敏捷的资源交付和完全自由的服务加载为用户带来充分的弹性。

1）所有资源秒级响应并按秒计费，最大程度发挥云资源的弹性优势。

2）资源间采用松耦合关系，用户可完全按需要加载各种类型和数量服务，轻松实现“水平”与“垂直”的扩展与收缩（scale in/out，scale down/up）。

（8）开放　QingCloud 无限开放全部功能 API，从而支持更高效、更有创造力的资源使用和云端生态链的构建。

（9）易用

1）QingCloud 支持菜单界面和图形界面操作，通过控制台快速创建与管理资源，并随时查看资源使用情况、操作日志及消费记录。

2）通过工单（Tickets）系统，用户可以直接同 QingCloud 开发人员进行沟通，咨询和解决使用中的各种问题。

第 2 章
云计算相关理论和参考模型

探索和借鉴行业最佳理论和实践，对云计算数据中心的意义重大。本章将归纳和总结业界对云计算和云管理最新的理论成果。

2.1 ITIL 最佳实践

2.1.1 ITIL 理论发展历程

ITIL 是 Information Technology Infrastructure Library 的缩写，即信息技术基础架构库，是一套基于 IT 业界最佳实践的 IT 服务管理框架。自 20 世纪 80 年代中期由英国政府部门 CCTA（Central Computing and Telecommunications Agency）提出以来，其内容得到多次扩展并优化，是当前全球 IT 服务领域最受认可的系统且实用的管理方法，成为 IT 服务管理事实上的国际标准，得到了全球 IT 企业及组织的大力支持。ITIL 目前由英国商务部 OGC（Office of Government Commerce）负责管理，它由一套客观、严谨、可量化的标准和规范组成。ITIL 最初是作为英国政府 IT 部门的最佳实践指南，问世后不久便被推广到英国的私营企业，逐步传遍欧洲，随后在美国兴起。

到目前为止，ITIL 已经发布了三个版本，分别是 ITIL V1、ITIL V2 和 ITIL V3 版本，ITIL 从低版本到高版本所涵盖的管理范围和内容逐步扩充，管理体系逐步完善。ITILV1 版本于 1986 年发布，主要用于政府信息化项目；2000 年升级到 ITIL V2，迅速在全世界得到广泛响应和推广实践；2007 年引入 IT 服务管理生命周期概念后升级为 ITIL V3；随后根据业界的最新进展和意见反馈，于 2011 年推出了最新的 ITIL 2011 修订版，包含服务战略、设计、转换、运营及持续改进等 5 个生命周期阶段，26 个流程，确保 IT 服务管理整个生命周期能够有机地融入业务中，把 IT 上升到企业战略资产高度，展示 IT 服务的价值。以下对这三个版本分别加以介绍。

1. ITIL V1

20世纪80年代中期，人们开始总结在IT服务方面产生的经验和教训，不断研究如何提供可靠优质的IT服务，寻找质量可测量、成本可计量的管理手段，逐步摸索出一套IT服务的规范化方法。在政府、IT组织及专家的共同努力下，80年代后期至90年代初期，CCTA（英国中央计算机与电信局）陆续发布了通过流程组织IT服务管理的最佳实践ITIL V1.04，共包含40个流程，主要是基于职能型的实践。

2. ITIL V2

ITIL树立的以流程为中心的IT服务管理方法，使IT服务管理不再是一个“虚无缥缈”的概念，人们逐步认识到IT服务管理的重要性，ITIL已经成为一个独立的有着巨大发展潜力的行业。从20世纪90年代初开始，越来越多的公司进入ITIL领域并加大对ITIL的投入，试图占据一个有利的位置，开发基于ITIL的管理模型和系统软件，例如BMC公司通过收购Peregrine公司的Remedy解决方案切入这个市场；荷兰Vrije SERC（大学软件工程研究中心）组织开发IT服务能力成熟度模型（IT Service CMM）；微软公司为所有微软产品开发了MOF（管理运营框架）；HP公司开发了该公司实施IT服务管理的方法论HP ITSM Reference Model（惠普IT服务管理参考模型）；CA公司的Unicenter服务管理解决方案已经可以实现ITIL各核心流程的集成管理。世界上越来越多的企业认识到了IT服务管理的重要性，并已经开始或正准备开始实践IT服务管理。这些企业实施IT服务管理的经验和教训有力地促进了IT服务管理方法的改进、提高和发展。基于此种情况，英国政府商务部（OGC，CCTA于2001年并入了该机构）在2000~2003年期间发布了ITIL的全新版本ITIL V2。ITIL V2包含7个体系：服务支持、服务提供、实施服务管理规划、应用管理、安全管理、基础架构管理及ITIL的业务前景。它已经成为IT服务管理领域全球广泛认可的最佳实践框架。

3. ITIL V3

2007年5月30日，ITIL V3的核心读物由OGC正式出版发布。ITIL V3引入了服务生命周期的概念，整合了ITIL V1和ITIL V2的精华，并与时俱进地融入了IT服务管理领域当前的最佳实践。ITIL V3拥有三个组件：核心组件、补充组件和网络组件。核心组件由5个领域组成，替代了ITIL V2中的服务支持和服务提供，涵盖了IT服务从设计到退役的整个生命周期，包括关键概念和相对稳定、通用化的最佳实践。补充组件包括不同情况、行业和环境的详细内容和目标。补充组件是ITIL V3的新特色，该部分为在不同市场、不同规模企业或行业、不同技术或规范环境中的应用提供了有针对性的指导，例如COBIT（信息系统审计和控制联合会，Control Objectives for Information and related Technology）、六西格玛、萨班斯法案等。补充组件将每年或每季度不定期地根据需求进行变更。补充组件中的指导可以帮助用户进行客户化定制ITIL，使其满足用户的特定需求，并且也为如何将ITIL与其他最佳实践和标准相结合提供指导。如果ITIL V3能同COBIT及其他的标准和最佳实践结合得更加紧密，这将更易于ITIL的实施且可以带来更加成功的结果。网络组件提供共同所需的动态资源和典型资料，例如流程图、定义、模板、业务案例和实例学习。

ITIL 的发展印证着 IT 服务管理方面的巨大进步，从早期 ITIL V1 职能化、组织化管理到 ITIL V2 流程化管理，再到 ITIL V3 的全生命周期管理，IT 服务管理逐渐成为一个系统，更具有系统性。展现了 IT 服务管理整个发展过程和对应管理模式的变化过程。

随着云计算技术的不断发展，给 IT 服务管理也带来了巨大的变革。企业部署新业务的时间缩短，业务敏捷性提升，满足市场需求的速度更快，数据规模达到了新的层次，系统管理也遭遇了前所未有的复杂性。ITIL 在云计算管理领域面临新挑战，传统的基于 ITIL 的 IT 服务管理模式，要适应云管理“更快”、“更大”的特点，提高流程效率，加强流程的自动化水平，逐步降低 IT 服务的风险。采用云计算的企业急需形成一套能够支撑云管理体系的 IT 服务管理模型，来有效管理本企业或组织的 IT 服务。在云计算时代，如何建立基于 ITIL 理论的，能够充分利用云计算提高流程管理效率，同时能够有效管理云计算提供的 IT 服务，充分降低风险，成为企业和组织面临的新挑战。

2.1.2 ITIL V2 理论

2.1.2.1 IT 服务管理的基本理念

（1）IT　包括技术基础设施（硬件、系统软件和通信设施）、应用基础设施（应用软件和数据库）和设施以及文档等。

（2）服务（Service）　由 IT 服务提供商支持的、以让客户感觉协调一致的方式满足客户的一种或多种需求的可用系统或功能。

（3）IT 服务（IT Service）　综合利用人、资源和程序以满足客户的信息需求。

（4）管理（Management）　在提供和交付服务中使用的战略级、战术级和运营级的概念和实践，它涉及使用各种资源，包括设备、人力、流程和理念等来实现某个目标，在这里是指交付恰当的服务。

（5）IT 服务管理（IT Service Management，ITSM）　国际 IT 领域的权威研究机构 Gartner 认为，ITSM 是一套通过服务级别协议（SLA）来保证 IT 服务质量的协同流程，融合了系统管理、网络管理、系统开发管理等管理活动和变更管理、资产管理、问题管理等许多流程的理论和实践。而 ITSM 领域的国际权威组织 ITSMF（国际 IT 服务管理论坛）则认为 ITSM 是一种以流程为导向、以客户为中心的方法，它通过整合 IT 服务与组织业务，提高组织 IT 服务提供和服务支持的能力及其水平。

ITSM 的核心思想是，对于 IT 组织，不管它是企业内部的还是外部的，都是 IT 服务提供者，其主要工作就是提供低成本、高质量的 IT 服务。ITSM 也是一种 IT 管理。不过与传统的 IT 管理不同，它是一种以服务为中心的 IT 管理。

2.1.2.2 ITIL V2 体系介绍

ITIL V2 主要包含 7 个模块，这 7 个模块是业务和技术之间进行交流的桥梁，7 个模块中，服务支持和服务交付模块中的 11 个过程，是 IT 服务管理的核心过程。图 2-1 为 ITIL V2.07 个模块的体系架构，能够展示各模块之间的关系。各模块的具体介绍

如下：

1. 业务视野

ITIL 所强调的核心思想是从客户视野理解 IT 服务需求。在提供 IT 服务时首先应该根据业务需求来确定 IT 需求。本模块帮助服务提供方深入了解 IT 基础架构支持业务流程的能力及 IT 服务管理在提供端到端 IT 服务过程中的作用，以协助他们更好地处理与业务部门之间的关系，实现商业利益的最大化。

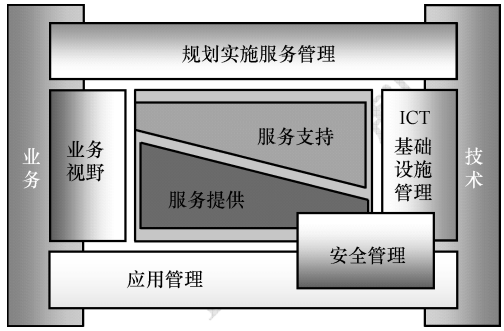


图 2-1　ITIL 体系流程

2. 服务管理

服务管理是 ITIL 的核心模块，包含服务提供和服务支持两部分。它把 IT 管理活动归纳为 10 个核心流程和一些辅助流程。服务管理的 10 个核心流程分为服务提供和服务支持两组。其中，服务提供由服务级别管理、IT 服务财务管理、IT 服务持续性管理、可用性管理和能力管理 5 个服务管理流程组成；服务支持由事件管理、问题管理、配置管理、变更管理和发布管理 5 个流程及服务台职能组成。

3. 基础设施管理

IT 基础设施管理覆盖 IT 基础设施的各个方面，包括识别业务需求、实施和部署、对基础设施进行支持和维护等活动。IT 基础设施管理的目标是确保 IT 基础架构是稳定可靠的，能够满足业务需求和支撑业务运作。

4. 应用管理

应用管理包括对应用系统从最初的业务需求到应用系统的交付、支持、维护、运作及下线。为了确保应用系统满足客户需求并方便对其进行支持和维护，IT 服务管理的职能应介入应用系统的开发、测试和部署。应用管理模块指导 IT 服务提供方有效协调应用系统的开发和维护，以使它们做为一个整体的为客户的业务运作提供支持和服务。

5. 安全管理

安全管理的目标是保护 IT 基础架构，使其避免未经授权的使用。安全管理模块为如何确定安全需求、制定安全政策和策略及处理安全事件提供了全面指导。ITIL 的安全管理模块侧重的是从政策、策略和方法的角度指导如何进行安全管理，更加侧重于安全管理原则的指导。

6. 规划实施服务管理

规划实施服务管理的作用是指导服务提供者如何整合和实施上述模块中的各个流程。它指导客户确立远景目标，分析和评价现状，确定合理的目标并进行差距分析，确定任务的优先级，同时对流程的实施情况进行测量和评审。

2.1.2.3　服务提供

ITIL V2 的服务提供模块更多地覆盖了 IT 服务提供的前期需要进行管理的内容，它包括服务级别管理、IT 服务财务管理、IT 服务持续性管理、可用性管理和能力管理。

这些过程主要关注于为改进所提供的 IT 服务的质量开发计划。

1. 服务级别管理（Service Level Management）

服务级别管理是定义、协商、订约、检测和评审提供给客户的服务的质量水准的流程。服务级别协议规定了服务双方各自的责任、权利和义务，是 IT 服务成功运营的重要保障。本流程的目标是确保服务级别协议是根据客户需求而不是服务提供者的技术能力确定的，保证服务级别协议得到有效执行，并在服务双方出现争议时提供有效的证据和解决争议的指导规则。

2. IT 服务财务管理（Financial Management of IT Services）

IT 服务财务管理是指负责预算和核算 IT 服务提供方提供 IT 服务所需的成本，并向客户收取相应服务费用的管理流程。其目标是通过量化服务成本减少成本超支的风险、减少浪费，确保所提供的 IT 服务符合成本效益的原则。本流程包括 IT 投资预算、IT 服务成本核算和服务计费三个子流程。IT 服务财务管理流程产生的信息可为服务级别管理、能力管理、IT 服务持续性管理和变更管理等管理流程提供决策依据。

3. IT 服务持续性管理（IT Service Continuity Management）

IT 服务持续性管理是指发生灾难后有足够的技术、财务和管理资源来确保 IT 服务持续性的管理流程。本流程关注的焦点是在发生服务故障后仍然能够提供预定级别的 IT 服务从而支持组织的业务持续运营的能力。

4. 可用性管理（Availability Management）

可用性管理是根据用户和业务方的可用性需求优化和设计 IT 基础架构的可用性，从而确保以合理的成本满足不断增长的可用性需求的管理流程。可用性管理是一个前瞻性的管理流程，它通过对业务和用户可用性需求的定位，使 IT 服务的可用性设计建立在真实需求的基础上，从而避免过度的可用性级别，节约了 IT 服务的运营成本。

5. 能力管理（Capacity Management）

能力管理是指在成本和业务需求的双重约束下，通过配置合理的服务能力使组织的 IT 资源发挥最大效能的服务管理流程。能力管理包括业务、服务和资源能力管理三个子流程。其中业务能力管理主要关注当前及未来的业务需求，服务能力管理主要关注当前 IT 服务的绩效支持正常业务运营的情况，资源能力管理主要确保 IT 基础设施中所有组件能发挥最大的效能。

2.1.2.4 服务交付

服务支持由事故管理、问题管理、配置管理、变更管理和发布管理 5 个流程及服务台职能组成。对这些服务流程和服务职能的含义分别说明如下：

1. 服务台（Service Desk）

服务台作为 IT 服务提供方与 IT 服务客户和用户之间的统一联系点。当客户或用户提出服务请求或报告事故时负责记录并尽量解决，在不能解决时可以转交给相应的支持小组并负责协调各小组和用户的交互。同时，把支持小组的处理进展及时通报给用户。此外，服务台还为其他管理流程如变更管理、配置管理、发布管理、服务级别管理及 IT 服务持续性管理提供了接口。

2. 事故管理（Incident Management）

事故是任何不符合标准操作且已经引起或可能引起服务中断和服务质量下降的事件。本流程的目的是在出现事故时尽可能快地恢复服务的正常运营，避免造成业务中断，以确保最佳的服务可用性级别。事故管理流程必须最佳地利用资源支持业务、开发和维护有效的事故记录以及设计和应用统一的事故报告。

3. 问题管理（Problem Management）

问题是发现事故的潜在原因，并防止它们重复发生的过程。问题管理与事故管理有明显的不同，事故管理的目标是尽可能快地恢复服务，而问题管理的主要目的是找出事故产生的根本原因，为此，它甚至可能要求中断服务。问题管理如果发现事故产生的原因并找到解决方案，需提交变更请求（RFC）以消除事故或问题产生的根本原因。

4. 配置管理（Configuration Management）

配置管理是识别和确认系统的配置项，记录和报告配置项的状态和变更请求，检验配置项的正确性和完整性等活动构成的过程。其目的是提供 IT 基础架构的逻辑模型，支持其他服务管理流程特别是变更管理和发布管理的运营，为其他流程提供准确的信息，验证基础架构记录并在必要时纠正有关记录。

5. 变更管理（Change Management）

变更是对已批准构建或实施的、已在维护的或作为基准的硬件、网络、软件、应用、环境、系统及相关文档所做的增加、修改或移除。本流程的目标是在最短的中断时间内完成基础架构或服务变更而对其进行控制的过程。变更管理的目的是使用标准方法和规程来快速有效地处理所有变更，以减少任何有关事故对服务的影响。

6. 发布管理（Release Management）

发布管理是指一组经过测试后导入实际运营环境的新增的或经过改动的配置项。发布管理的目的是为了保证发布的成功，主要应用于大型或关键的硬件、主要软件及打包或批处理一组变更。

2.1.3 ITIL V3 理论

从 ITIL V3 理论可以看出，随着经济发展以及 IT 服务的不断扩展，人们对 IT 的看法已经发生了巨大转变。人们逐步意识到管理 IT 所需的不仅仅是一系列流程，而是需要通过管理服务生命周期来创造业务价值。因此，如何充分利用技术来提高业务价值，成为人们关注的焦点。借助 ITIL V3，服务管理可以与不断变化的业务需求、优先级不断变化的业务类别、日新月异的技术以及新的治理模式保持协调一致。更新版的 ITIL V3 旨在加速并简化服务管理流程的应用与实施，从而优化业务成效。

2.1.3.1 ITIL V3 特性

加强对服务生命周期的管理是 ITIL V3 与前两版 ITIL 的重大区别。在过去 5 年甚至更长的时间中，服务生命周期管理的理念发生了转变并逐渐成熟。IT 成为企业越来越重要的战略组成部分，为满足业务需求，IT 管理包含的内容从流程和功能的实施。逐

步扩展到 IT 服务周期的各个阶段提供的各项 IT 服务。ITIL V3 基于这一新理念，根据服务生命周期阶段及业务成效目标构建，保留了 ITIL V2 的流程，并明确了利用这些流程来提供业务支持服务，以及实施和管理这些流程的方法。

（1）ITIL V3 服务生命周期的主要阶段划分在 ITIL V3 中，服务生命周期的 5 个主要阶段划分如下：

1）面向实际决策制定的服务战略阶段：该阶段定义了 IT 的角色和需求，以确保整体业务的成功。

2）规划实际服务蓝图的服务设计阶段：IT 部门将对服务进行设计，使之能够通过功能和性能满足企业的需求，同时具备可管理性和经济高效性。

3）旨在改善管理变更、降低风险并保证质量的服务转换阶段：对服务进行测试，并以可控的方式在基础设施中实施。

4）旨在提高服务稳定性和响应能力的服务运营阶段：本阶段真正提供服务并为服务提供支持。服务运营将提供所定制的服务，同时积极应对业务和 IT 方面的各种变化，提高服务的稳定性和灵活性。

5）具备可行的衡量标准的持续性服务改进阶段：该阶段将对服务的质量和成本进行持续监控和衡量，从而改进服务的质量并降低其成本，同时使其与不断变化的业务需求保持一致。

（2）ITIL V3 新增特性

1）让用户看到 ITIL 实践带来的业务优势。

2）在原有 ITIL 版本上改进了 ITIL 的有效性和适用性。

3）使 ITIL 更加易于实施。

4）引入了原有 ITIL 版本的实际改进，其中包括工具、技术和关系类型。

2.1.3.2 ITIL V3 框架

ITIL V3 的核心架构是基于服务生命周期的。如图 2-2 所示，生命周期模型的引入改变了 ITIL V2 模块之间相互割裂、独立实施的局面，从战略、战术和运作三个层面提出服务管理实践方法。ITIL V3 以服务战略作为总纲，通过服务设计、服务转换和服务运作加以实施，并借助持续服务改进不断完善和优化整个过程，使 IT 服务管理的实施过程被有机整合为一个良性循环的整体。其中，服务战略是服务生命周期的轴心；服务设计、服务转换和服务运营是服务实施和运营阶段；服务改进则是基于服务战略对服务的定位、进程和项目进行优化和改进。

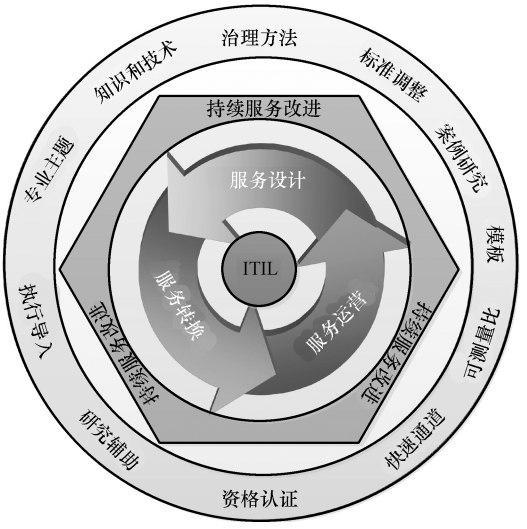


图 2-2　服务生命周期框架

接下来，我们将简单介绍一下各模块所包括的内容：

1. 服务战略

该模块提供了关于如何设计、开发和实施服务管理的指导。服务战略是服务设计、服务转换、服务运营和服务改进的基础和核心，它的主题包括了市场开发、内部和外部的服务提供、服务资产、服务目录以及整个服务生命周期过程中战略的实施；同时，还包括了财务管理、服务投资组合管理、组织的制定和战略风险等一些重要的主题。服务战略确保组织能处理与服务投资组合相关的成本和风险，建立运营的有效性和实现出色的绩效。

2. 服务设计

该模块为服务和服务管理流程的设计和开发提供指导。服务设计的范围不仅限于新的服务，还包括为了保持和增加客户价值，而实行服务生命周期过程中必要的变更和改进、服务的连续性、服务水平的满足，以及对标准、规则的遵从性。它指导了组织如何开发设计服务管理的能力。

3. 服务转换

服务转换是指导如何将新的或变更的服务转换到运营体系中的能力。服务战略需要通过服务设计进行编码，而服务转换则是探讨如何将这种编码有效地导入到服务运营的体系中，并在此革新的过程中避免出现不良的结果。此外，它还提供了客户与服务提供商之间转换过程中对服务控制的指导。

4. 服务运营

对如何达到服务支持和交付的效果和效率以确保客户与服务供应商的价值提供了指导。战略目标最终需要通过服务运营来实现，因此，它是一种非常重要的能力。它提供了在设计、规模和服务水平变化的情况下保持服务运营稳定性的方法。同时为经理和实践者如何利用知识管理对服务可用性、控制需求、优化使用能力、操作安排和问题修复等方面做出更优的决策提供了指导。

5. 服务改进

它结合了质量管理、变更管理和能力改进方面的原则、实践和方法。为创造和保持客户价值，而用更优化的服务设计、导入和运营提供了指导。组织在服务质量、运营效率和业务连续性方面要有不断提高和改进意识。此外，服务改进还为改进所取得的成就与服务战略、服务设计和服务转换之间如何建立关联提供了方法。对建立基于 PDCA 模型（Plan、Do、Check 和 Act）闭环反馈系统提供了指导。

2.1.4 ITIL V3 与 ITIL V2 的特征比较

ITIL V3 是对 ITIL 最佳实践的巩固和提高，也是“当前最佳实践”的精髓。OGC 对 ITIL V2 中的重要内容加以精简，然后将其收录到 ITIL V3 中。ITIL V3 的结构框架和内容来源于大量的公众评议及行业管理者的意见。同时，它也保留了 ITIL V2 中仍被 ITSM 团体广泛实践和运用的内容。ITIL V3 增加了部分新概念，尤其引入了“生命周期”这个概念。借助于“生命周期”的贯穿，ITIL V3 将 ITIL V2 中的各个流程有机地整合在

一起，同时增加了一些营销方法与流程，讲解了 ITIL 在不同的行业该如何切入，使得 ITIL 跟企业的关系更加紧密。ITIL V2 与 ITIL V3 的比较主要体现在五个方面，见表 2-1

表 2-1　ITIL V2 与 ITIL V3 的特征比较

ITIL V2 的特征	ITIL V3 的特征
关注诸如服务台、事件、问题、变更、配置和风险管理的流程	关注服务，因为流程只是服务的附属物
关注业务与 IT 的结合	强调业务和 IT 的整合
关注价值链管理	强调价值网络的集成
关注线性的服务目录	强调动态的服务投资组合
关注流程一体化的集成	强调全面服务管理的生命周期

2.1.5　云计算对 ITIL 产生的影响

在云计算高速发展的时代，企业的 IT 服务管理面临着巨大的挑战：业务要快速上线运营，IT 基础架构日益复杂，IT 部门的管理负担加重，对业务的连续性提出了更高的要求，运营成本不断增加。以 ITIL 理论为基础的 IT 服务管理最佳实践为云计算优质服务的提供发挥了更重要的作用；同时，云计算对流程的自动化、高效化，降低流程风险也起到了巨大的推动作用。

1. 提供标准化的 IT 服务

在云计算环境下，IT 部门为业务部门或 IT 部门内部提供的 IT 服务呈现出更加标准化、规范化、可量化的特点。服务的申请与提供流程更加简洁。例如：原有基础设施及平台资源的安装与分配因基础设施软硬件环境的非标准化导致安装过程极为繁琐。云计算环境下通常为用户提供标准的资源服务，使流程环节得到简化，流程成本逐步降低，流程处理效率大幅提升。

2. 自动化提升流程效率，降低流程风险

基于云计算环境的自动化管理水平大幅提升，已知事故的处理、标准化变更的实施、版本分发都可通过自动化方式进行，标准化服务通过自动化方式提供，在减少人力成本的同时，减低人为操作风险，提高流程执行效率，减少流程风险。

3. 量化服务管理达到业务目标

云计算提供的标准化服务，为服务量化提供了便利。结合这一优势，量化交付的 IT 服务，做到成本可追踪，性能与可用性可被监控管理，进行量化管理能够达到降低成本、提升服务质量、提高生产力、掌控风险与持续改进的目标。

2.2　IBM CCRA

2.2.1　IBM CCRA 概述

目前很多公司都在制定自己的云计算参考框架，根据各自的理解定义了云计算的主

要构成组件，为构建企业级私有云提供了有益的参考。

CCRA 是 IBM 根据多年为客户咨询、设计与实施云计算解决方案的经验定义的一个云计算参考框架，是一个云计算的架构蓝图和实现指南。IBM 云计算参考架构为云计算平台的实现提供指导性原则和技术工作产品，如服务和部署模型，并定义了具体实现的采用模式（Adoption Pattern）。一个采用模式具体表达了体系结构模式，具体说明了企业单位可以用它来实施云计算解决方案的方法，可以指导企业单位定义与设计满足自己需求的云计算解决方案。

IBM 云计算参考架构定义了构成云计算环境的基本架构元素，下面进行简要说明。

1. 角色

该架构定义了三个主要角色，即云服务消费者（Cloud Service Consumer）、云服务提供者（Cloud Service Provider）和云服务创建者（Cloud Service Creator）。每一个角色可以由单人执行，也可以由一组人或一个组织团体执行。一个云服务消费者可以是一个组织、一个人或一个 IT 系统，他们消耗着特定的云服务实例。云服务提供者有能力为云服务消费者提供云服务。云服务创建者的目的是创建一个能够被云服务提供商运行并提供给云服务消费者的云服务。通常情况下，云服务创建者利用云服务提供商提供的服务功能来创建自己的云服务，就如同云服务提供商和云服务消费者一样，云服务创建者可以是一个组织或一个人。

2. 服务

在 IBM 云计算参考架构中设计了四种云服务模式：基础设施即服务（IaaS）、平台即服务（PaaS）、软件即服务（SaaS）和业务过程即服务（BPaaS）（业务过程即服务是 IBM 自己定义的，在美国国家标准中只定义了 IaaS、PaaS 和 SaaS）。

3. 基础设施

在 IBM 云计算参考架构中，基础设施所有的元素都在云服务提供商里面，包括服务器、存储、网络资源和机房设施。基础设施元素仅限于硬件基础设施，它不包括系统管理程序，也不包括任何虚拟化管理软件。

4. 公共云管理平台（Common Cloud Management Platform，CCMP）

在 IBM 云计算参考架构中，公共云管理平台（CCMP）的功能是通过由 CCMP 内部组件公开的 AP 来访问的。CCMP 被定义为通用云管理平台，以支持跨越 IaaS、PaaS、SaaS、BPaaS 任何类别的云服务管理。CCMP 分为两个主要元素：运营支持服务（Operational Support Services，OSS）和业务支持服务（Business Support Services，BSS）。

（1）运营支持服务（OSS） 代表了一系列运营管理和相关技术服务，这些服务将由 CCMP 提供给服务消费者，这些服务需要通过云服务的创造者来实现。OSS 主要包括平台与虚拟化管理（Platform and Virtualization Management）、监控与事件管理（Monitoring and Event Management）、IT 资产与授权管理（IT Asset and License Management）、容量与性能管理（Capacity and Performance Management）、自动化部署（Provisioning）、配置与变更管理（Configuration and Change Management）、服务自动化管理（Service Auto-

mation Management）、事件与问题管理（Incident and Problem Management）、IT 服务水平管理（IT Service Level Management）、服务交付目录管理（Service Delivery Catalog Management）、服务请求管理（Service Request Management）、镜像生命周期管理（Image Lifecycle management）、备份与恢复管理（Backup and Restore Management）、安全合规性管理（Security Compliance Management）和补丁管理（Patches Management）。

（2）业务支持服务（BSS） 代表了一系列与业务相关服务，这些服务将由 CCMP 提供给服务消费者，这些服务需要通过云服务的创造者来实现。BSS 主要包括客户账户管理（Customer Account Management）、合同与合约管理（Contracts and Agreement Management）、计量（Metering）、定价管理（Pricing）、评级管理（Rating）、账单管理（Billing）、订单管理（Order Management）、服务目录（Service Offering Catalog）、服务管理（Service Offering Management）、授权管理（Entitlement Management）、服务请求管理（Service Request Management）、预定管理（Subscription Management）、结算管理（Clearing Settlement）、应付账款（Account Payable）和应收账款（Account Receivable）。

5. 安全性、弹性、性能和易用性

安全性、弹性、性能和易用性横跨了基础设施、公共云管理平台（CCMP）和三个云服务角色等方面。这些非功能性要素必须从端到端的角度来看，包括 CCMP 本身的组成结构、硬件基础设施的设置（例如从隔离、网络分区方面划分，为数据中心配置灾难恢复等）以及如何实现云服务。

6. 服务创建工具

云服务创建者使用服务开发工具开发新的云服务，包括开发运行工件（Runtime Artifacts）和有关管理组件（如监测、计量、配置等）。

2.2.2 资源池的定义和目标

资源池是由云计算引申出来的基础设施资源管理的逻辑概念，通常来讲是将计算资源、存储资源和网络资源等传统的基础设施资源的运算能力通过云管理平台统一整合在一个池内，再进行统一分配及管理。

资源池打破了单一设备的限制，将所有的 CPU、内存、存储和网络等资源解放出来，并汇集在一起形成一个可分配管理的池，当用户提出需求时，便从这个池中配置能够满足需求的组合。

资源的池化使得用户不再关心计算、存储和网络等资源的物理位置和存在形式，只需要按需获取，其可共享的特征也使 IT 部门能够灵活地对资源进行配置，进而实现资源的弹性供给和有效利用。

2.2.3 云服务的定义和目标

云服务是指通过网络以按需、易扩展的方式获得所需服务。这种服务可以是 IT 服

务，也可以是其他服务，这里主要是指 IT 服务。目前业界流行的云服务模型主要包括 IaaS（基础设施即服务）、PaaS（平台即服务）和 SaaS（软件即服务）。

对于以上云服务模式可根据企业特点进行重新定义，示例如下：

1. IaaS

IaaS 提供给消费者的服务是对基础设施的利用，包括处理、存储、网络和其他基本的计算资源。用户能够部署和运行任意软件，包括操作系统和应用程序。消费者不管理或不控制任何云计算基础设施，但能控制操作系统的选择、存储空间、部署的应用，也有可能获得有限制的网络组件（防火墙、负载均衡器等）的控制。

此外，将 IaaS 界定为预装操作系统和必要的管理软件的标准部署单元，此部署单元以虚拟机或实体机形式供给，包含已按标准规范配置的操作系统及相关管理软件，用户在其上部署应用。除了计算资源，IaaS 还提供对应的存储服务和虚拟网络服务，即以虚拟机或实体机形式提供的计算资源服务，预装标准操作系统以及必要的管理软件，同时提供磁盘存储服务和网络连通服务。

2. PaaS

业界流行的公有云 PaaS 通常只支持某种特定的、由提供商规定的编程语言与工具创建的应用，提供商往往在提供应用运行环境和部署服务的同时为开发人员提供完整的开发框架，这使得开发人员可以快速地通过统一的接口获取云平台的各种服务而无需了解底层实现的相关细节，然而这种模式很难直接应用到企业私有云环境中。

因此，在企业私有云构建中，首先应用系统开发需要使用多种编程语言，运行在多种中间件之上，基础设施云必须支持这些异构的技术平台，且支持不同技术架构的部署单元之间的集成。

其次，在企业私有云环境下，应用的部署只是个开始，更重要的是对整个应用生命周期的管理。因此，基础设施云除了支持应用的自动化部署之外，还应该支持应用与其他通用技术服务的集成，以及通过应用系统与 IT 服务管理系统的集成实现对应用运行环境生命周期管理，即基于标准部署模式的应用部署与管理服务，应用运行环境中将预装操作系统、中间件、数据库和应用框架，且预集成通用技术服务、IT 管理服务等。应用运行环境可由单个部署单元组成，或由多个不同部署单元相互集成所形成。

云服务是基础设施云的资源供给形式，其主要目的是实现资源供给的标准化，其核心目标是将最佳实践规范化并以标准服务的形式共享，进而提高服务的可靠性和资源的可管理性。

2.2.4 云管理的定义和目标

云管理是以实现 IT 能力服务化为目的，是系列软硬件应用方案的总和。其核心理念是 SOA（面向服务架构），其核心技术是自动化流程管理。

云管理的目标是实现 IT 能力的服务化供应，并实现云计算的各种特性，例如：资源共享、自动化、按使用付费、自服务、可扩展等。

2.2.5 云管理的主要功能组件

云管理平台通常以服务目录为导向梳理各类 IT 资源服务模板，并将其以部署模式的形式与后台供给类资源进行映射，将服务项目与后台实体资源池进行关联，用户通过服务目录选择不同的资源请求项目，云平台接收服务请求后通过服务策略管理模块依照部署模式的描述协调调度各类资源池的分配以及环境的搭建部署，并最终交付用户。资源池的整体使用情况由服务策略管理进行判断及分配，由资源管理模块进行实体操作。云管理平台主要功能组件包括服务目录、部署模式、服务策略管理和资源管理四大模块。

1. 服务目录

通过标准化的服务目录对基础设施服务进行管理，建立 IT 服务的使用者与 IT 资源之间的标准接口，管理员在服务目录中可以定义、发布、更新和终止 IT 服务，对 IT 服务的名称、描述、资源类别、资源规模、费用等做出规定，同时可以设定不同用户访问服务目录的权限。

2. 部署模式

通过定义部署模式，管理员可构成特定服务目录的 IT 组件（虚拟机、软件、应用和存储），运行环境（如网络要求、巡检要求和监控要求）及其相互关系。

3. 服务策略管理

服务策略管理是池化资源管理的关键环节，该部分功能主要负责用户所请求的服务资源如何动态在池化资源池中进行资源的分配，具体将用户的配置要求落实在与其 SLA 要求、地域要求、容量要求、性能要求、用户类型等相匹配的物理资源上，指导下一步资源管理的具体部署。

4. 资源管理

自下向上实现计算资源、存储资源、网络资源的池化和纳管，自上而下实现最终的资源部署及配置操作。

2.3 DevOps 理论

2.3.1 DevOps 理论概述

2.3.1.1 DevOps 含义

DevOps 是开发（Development）和运维（Operations）的组合，是一组过程、方法与

系统的统称，用于促进开发（应用程序/软件工程）、技术运营和质量保障（QA）部门之间的沟通、协作与整合，如图 2-3 所示。它的出现是由于软件行业日益清晰地认识到，为了按时交付软件产品和服务，开发和运营工作必须紧密合作。

随着信息化深入到企业生产经营的各个领域，IT 技术已经成为企业降低成本、提升效率和提高竞争力的重要手段，业务部门对产品交付的频率要求越来越高，同时伴随着虚拟化、云计算和敏捷开发方法的日益普及，传统的软件交付方式如何适应新的需要成为一个重大的课题。在传统工作方式下，软件开发、IT 运营和质量保障都属于各自分离的部门，开发与运营部门的诉求点即工作目标和工作方式也是存在差异的。

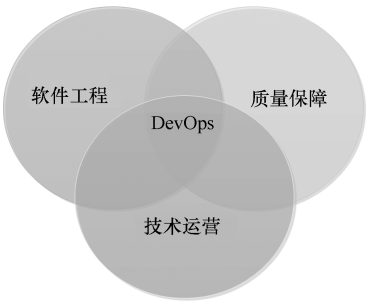


图 2-3　DevOps

1）开发部门关注的是快速实现业务部门的需求，以及交付新的产品。

2）开发人员很少考虑新的代码是否会对运维造成影响，同时运维人员基本上也不参与架构决策或代码评审。

3）开发人员在开发环境或者测试环境中手工修改配置，没有使用相应的自动化工具支持，在生产环境部署上又需要重新手工配置一遍，很容易出现纰漏；而且一旦需要回退时，往往也很难恢复到上一个稳定的状态。

4）在开发过程中，系统在开发者的本地机器上运行。在运营过程中，系统经常分布在多台服务器上，例如 Web 服务器、应用服务器、数据库服务器等。

5）运营部门关注的是如何保证系统能够持续提供服务，确保系统的稳定性和可靠性。

6）运营人员希望尽量避免系统修改或者减小变更频率，从而保证系统稳定运行。

7）如果减小了变更频率，但在给定时间内业务需要的总量不变，那么就意味着每次变更的规模变大。变更规模越大，理论上变更的风险也越大。

显然在这种传统的企业组织或者开发方式下，开发部门与运营部门之间存在着信息“鸿沟”，而这种信息鸿沟就是最容易出问题的地方。一方面可能延缓业务需求实现的交付速度，降低业务部门的满意度；另一方面由于运营人员对应用程序的架构缺乏了解，从而难以正确地选择运行时的环境和发布流程，而开发人员可能对运行环境缺乏了解，也难以正确地对代码进行调整和优化。DevOps 就是在这种背景下发展起来的，它提倡开发团队与运维团队之间的高度协同，在完成高频率部署发布的同时，提高生产环境的可靠性、稳定性、弹性和安全性。

2.3.1.2　DevOps 应用模式

1）将开发延伸至生产中，包括拓展持续集成和持续交付能力，集成 QA 和信息安全至整个工作流，确保代码和环境可在生产中直接部署。

2）向开发中加入生产反馈，持续的反馈促进了持续的改进。DevOps 工作流中，反馈时间很短，所以用户可以尽早地、更经济地进行调整。在整个生命周期中，开发和交

付团队持续监控运行质量来验证软件。软件一旦投入生产，这些度量标准就会捕获客户体验。

3）开发融入 IT 运维中，包括开发投入到整个生产问题处理链，分配开发资源用于生产问题管理，而且开发为 IT 运维提供交叉培训，增加 IT 运维处理问题的能力，从而降低升级问题的数量。

4）将 IT 运维融入开发中，包括分配 IT 运维资源至开发，帮助开发创建为 IT 运维，确保在产品的开发阶段就考虑到产品的非功能性需求。

2.3.1.3 DevOps 价值

DevOps 的引入能对企业带来深远的影响，对产品交付、测试、软件开发和运维产生积极的改变和改进。具体体现在以下几个方面：持续集成与交付，缩短开发周期和更高的部署频率，产品能够快速推向市场；持续质量保证，提高产品的质量，提高产品的可用性，变更成功率，减少故障等；提高团队工作效率，减少团队间的沟通成本，能够将时间用在更有价值的活动中。

1. 持续集成与交付

DevOps 是对软件交付模式的一种根本性变化，DevOps 的目标是通过良好定义的部署流水线将代码完全自动化地交付到一个经常使用的基础设施上。其好处是可以更加快速频繁地交付质量更高的软件。用一种结构性的方法将应用系统视为一个统一的整体、能帮助开发团队进行协调合作。

持续集成与持续交付贯穿了整个应用生命周期。持续集成与持续交付提供了一组务实的原则和实践，可用于实现产品的频繁变更诉求，并使它们变得更加可靠。频繁但是小的变更往往比集成一组变更要容易，而且如果一个小变更无法工作，从中回退也要比一组变更失败并从它们中回退要容易得多。持续集成有效地降低了复杂性并使得识别集成事故的根本原因变得更加简单。持续交付也以同样的方式运作。为了帮助一个团队从发布和部署中解放出来，应当让他们更加经常地交付小的代码包，而不是过一段时间就交付一个大的代码包。在部署任务方面，每周发布两次版本的团队要比每个月只发布一次版本的团队要轻松得多。如果您每次都发布整个系统，一个小集合的变更风险也明显要低，即使问题发生也更容易定位。

从整个应用生命周期的角度看，持续集成和持续部署最大的回报，就是提供了最新的代码可以被所有利益相关者进行评审的能力。里程碑发布版本或甚至测试发布版本都可以从敏捷开发（Scrum）迭代中获益，并使得验证需求、建立持续测试和更新改善系统设计成为可能，并让每一个利益相关者得以积极地参与到应用生命周期管理当中。同时，这些变更也能够帮助持续改善过程。

维持高部署率的能力转化为企业业务价值就是向市场快速推出满足客户需求的产品，会首先获得最大的市场份额。这是快速软件生产的一个重要驱动因素。如果率先向市场推出产品，企业会获得很大的收入。要满足市场需求，DevOps 和快速软件生产都必不可少。企业不仅获得了重大的竞争优势，还获得了增强的操作可预测性、改善的操作一致性和更低的成本，从而可以实现更高的利润。

2. 持续质量保证

通过 DevOps 关于自动化配置和持续部署方面的最佳实践，开发人员、QA 专业人员和其他参与者可以在一个以前并未曾有的低成本，而且与生产环境相似的测试环境中进行测试。虽然云计算大大减小了基础设施供给的难度，但即使是私有云，配给虚拟服务器也经常会显现一些特别的挑战。系统专业人员了解一个虚拟机并不能完全准确具备一台物理服务器的行为，这些风险需要被理解和避免。DevOps 关于配给测试环境的能力和使用云资源来交付新的能力，如果已经具备使用自动化规程配给和维护这些环境的 DevOps 最佳实践，可使得生产力和质量两个方面都得到加强。

软件开发团队通常使用最新的技术来不断开发产品，确保实现无错误的配置和交付结果。为了能够将用户和系统需求合并到一致的软件交付结果中，软件开发流程必须始终如一地保证质量，永久地支持产品迭代和修改，并不断采用有效的软件开发方法。DevOps 通过涵盖整个开发、测试、部署和运维生命周期端到端的流程，实现交付可靠、安全和没有缺陷的软件和系统。

3. 提高工作效率

DevOps 旨在更好地集成 IT 操作和软件开发，从而提高对业务更改的联合响应能力。通过建设 DevOps 能力，企业能够明显感受到软件产品发布和运营过程中的质量与效率。

更快的服务交付，与传统的瀑布式开发模型相比，采用迭代的工作方式意味着更频繁的发布、每次发布包含的变化更少。由于部署经常进行，每次部署不会对生产系统造成巨大影响，应用程序会以平滑的速率逐渐生长。与传统开发方法中大规模的、不频繁的发布相比，具备 DevOps 能力的企业大大提升了服务交付能力。

在开发与运营团队之间建立更加高效的协作关系，减少发布和运营中时间的浪费，提高了运营团队的工作效率。DevOps 实践强调改善开发和运营之间的沟通与协作这一目标。DevOps 实践增强开发团队、运维团队和其他关键利益相关者之间的沟通。通过强有力的发布协调机制来弥合开发与运营之间的技能鸿沟和沟通鸿沟；采用电话会议、即时消息等协作工具来确保所有相关人员理解变更的内容，使用统一的流程和工具，例如在线项目管理工具、配置管理工具可有效地减少或降低沟通成本，提高工作效率。

2.3.2 DevOps 各类管理工具

DevOps 是软件开发生命周期（SDLC）从瀑布式到敏捷再到精益的发展。DevOps 超越了敏捷发展方式，实现了 DevOps 涉及软件开发生命周期的各个方面。

1. 构建、打包和部署依赖管理

实现产品的自动化构建、打包和部署是 DevOps 工作的核心关注点。许多软件开发人员完全埋头在集成开发环境（Integrated Development Environments，IDE）中进行

工作，例如 Eclipse 及 Visual Studio。问题是他们并不能真正的知道和理解他们所有的构建依赖，对于开发人员来说，仅仅特定了解他们自己的构建和运行时依赖是十分常见的。当这些开发人员转移到下一个项目时，或者出现由于笔记本电脑崩溃而导致的意外时，组织也许会发现由于软件开发人员缺失理解构建、打包和部署代码所必需的知识而导致工程无法被构建。这就是构建工程师经常被行业法规规定为一个独立职责的现实原因，因为他们可以通过捕获所需的知识来构建自动化部署流水线，以增强产品可靠性。

脚本和自动化构建确保了编译和运行时依赖的核心知识可以被发现和被文档化。开发人员可能由于时间太久遗忘了被配置在集成开发环境里的所有环境设定，但是编写在编译和运行环境中的构建脚本提供了关于构建、打包及部署代码核心配置的一个清晰和准确的视图。

可以通过一个连贯的方式进行可靠的构建、打包和部署，这对于确保系统可以在不被意外和严重后果影响的前提下被支持和修改。除了可以可靠地构建代码之外，我们还需要确保已经验证正确的代码被部署了，以及更加重要的、任何来自恶意目的、非授权变更或人为错误都能立即被识别出来。

2. 持续集成

DevOps 的指导思想是“精益生产”。随着软件开发复杂程度的不断提高，团队开发成员彼此之间如何更好地协同工作以确保软件开发的质量已经慢慢成为开发过程中不可回避的问题。持续集成正是针对这一类问题的一种软件开发实践。它倡导团队开发成员必须经常集成他们的工作，甚至每天都可能发生多次集成。而且每次的集成都是通过自动化的构建来验证，包括自动编译、发布和测试，从而尽快发现集成错误，使团队能够更快地开发内聚软件。

持续集成的核心价值在于：

1）持续集成中的任何一个环节都是自动完成的，无需太多的人工干预，有利于减少重复过程以节省时间、费用和工作量。

2）持续集成保障了每个时间点上团队成员提交的代码是能成功集成的。换句话说，任何时间点都能第一时间发现软件的集成问题，使任意时间发布可部署的软件成为可能。

3）持续集成还能利于软件本身的发展趋势，这一点在需求不明确或是频繁性变更的情景中尤为重要，持续集成的质量能帮助团队进行有效决策，同时建立团队对开发产品的信心。

业界普遍认同的持续集成的原则包括：

1）需要版本控制软件保障团队成员提交的代码不会导致集成失败。常用的版本控制软件有 IBM Rational ClearCase、CVS、Subversion 等。

2）开发人员必须及时向版本控制库中提交代码，也必须经常性地从版本控制库中更新代码到本地。

3）需要有专门的集成服务器来执行集成构建。根据项目的具体实际，集成构建可

以被手工直接触发，也可以定时启动，如每半个小时构建一次。

4）必须保证构建的成功。如果构建失败，修复构建过程中的错误是优先级最高的工作。一旦修复，需要手动启动一次构建。

3. 应用发布自动化

DevOps 提供了对于创建自动化应用程序部署的能力来说最核心的原则和规程。使用 DevOps 关于持续集成和持续部署的实践来创建完全自动化部署流水线，对于贯穿整个应用程序部署生命周期来说非常重要。随着敏捷开发模式的推出，应用发布规模逐渐小型化，发布周期大幅缩短，导致发布任务密度大幅增加，以往缺乏工具支撑的管控模型不能有效满足当前的工作任务要求，而且往往由于发布问题导致业务故障，为解决此类问题，需要一个应用发布平台以更好地支持应用部署任务，降低部署应用程序的复杂性。

1）消除手动的基于脚本的部署方式，采用基于工作流的自动化部署方式。

2）可以批量在多台服务器上并行部署，大大提高部署的效率。

3）提高发布团队与开发、质量和运维团队之间的沟通协调能力，实现部署流程与 ITTL 流程无缝衔接。

4）与持续集成工具进行整合，实现一键式应用发布和服务启停。

4. 自动化测试

敏捷软件开发人员被称为质量感染者，这是因为他们关注于编写高质量的代码，渴望测试越早开始越好。自动化的回归测试是敏捷团队普遍采用的实践。该实践有时又被扩展为测试先行的方式，比如测试驱动开发（TDD），以及行为驱动开发（BDD）。由于敏捷团队经常一天多次运行他们的自动化测试集，并且能够马上修复发现的问题，所以他们比普通团队能达到更高的质量。对于运营人员而言，在同意一个业务变更发布到生产环境前，坚持足够的质量审查，这是一个很好的习惯。

开源领域存在很多优秀的工具，合理地使用或者借鉴这些工具的经验能够帮助我们更好地实践 DevOps 理论。

2.3.2.1 Jenkins

Jenkins 是一个开源项目，提供了一种易于使用的持续集成系统，使开发者从繁杂的集成中解脱出来，专注于更为重要的业务逻辑实现上。同时 Jenkins 能够实施监控集成中存在的错误，提供详细的日志文件和提醒功能，还能用图表的形式形象地展示项目构建的趋势和稳定性。

Jenkins 提供了丰富的管理和配置的功能，包括系统配置、管理插件、查看系统信息、系统日志、节点管理、Jenkins 命令行窗口、信息统计等功能。

此外，Jenkins 还可以安装大量功能强大的插件。如 Subversion 插件可以从代码版本控制库（SVN）库中获取代码，Maven 插件可以读取 pom. xml 分析依赖包、编译、测试，Deploy Weblogic 插件可以部署编译测试包。同时，Jenkins 也可以安装 ClearCase 插

件，从 IBM CC 中获取代码。

2.3.2.2 Puppet 与 MCollective

Puppet 是一种 Linux、Unix、Windows 平台的集中配置管理系统，使用自有的 Puppet 描述语言，可管理配置文件、用户、调度任务、软件包、系统服务等。Puppet 把这些系统实体称之为资源，Puppet 的设计目标是简化对这些资源的管理以及妥善处理资源间的依赖关系。

Puppet 采用客户端/服务端（C/S）星状结构，所有的客户端和一个或几个服务器进行交互。每个客户端周期性地（默认为半个小时）向服务器发送请求，并获得其最新的配置信息，以保证与该配置信息同步。每个 Puppet 客户端每半小时（可以设置）连接一次服务器端，下载最新的配置文件，并且严格按照配置文件来配置服务器。配置完成以后，Puppet 客户端可以反馈给服务器端一个消息，如果出错，也会给服务器端反馈一个消息。

MCollective 是一个构建服务器编排（Server Orchestration）和并行工作执行系统的框架。

首先，MCollective 是一种针对服务器集群进行可编程控制的系统管理解决方案。在这一点上，它的功能类似于 Func、Fabric 和 Capistrano。

其次，MCollective 的设计打破了基于中心存储式系统和像安全 Shell SSH 这样的工具，不再仅仅使用 SSH 的 For 循环。它使用发布订阅中间件（Publish Subscribe Middleware）这样的现代化工具和通过目标数据而不是主机名（Hostnames）来实时发现网络资源的现代化理念，提供了一个可扩展的、迅速的并行执行环境。

在实际应用中，Puppet 客户端的默认定时同步功能往往不能适应企业的实际情况，而通过 MCollective 能够更加安全地实现 Puppet 的推送更新功能。

2.3.2.3 Selenium

Selenium 也是一个用于 Web 应用程序测试的工具。Selenium 测试直接运行在浏览器中，就像真正的用户在操作一样。支持的浏览器包括 IE7、IE8、IE9，Mozilla Firefox，Mozilla Suite 等。这个工具的主要功能包括：测试与浏览器的兼容性——测试应用程序是否能够很好地工作在不同浏览器和操作系统之上；测试系统功能——创建衰退测试检验软件功能和用户需求；支持自动录制动作和自动生成 Net、Java、Perl 等不同语言的测试脚本。Selenium 是 ThoughtWorks 专门为 Web 应用程序编写的一个验收测试工具。

1）框架底层使用 JavaScript 模拟真实用户对浏览器进行操作。测试脚本执行时，浏览器自动按照脚本代码做出点击、输入、打开、验证等操作，就像真实用户所做的一样，从终端用户的角度测试应用程序。

2）使浏览器兼容性测试自动化成为可能，尽管在不同的浏览器上依然有细微的差别。

3）使用简单，可使用 Java、Python 等多种语言编写用例脚本。

2.3.2.4 Docker

Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。容器是完全使用沙箱机制，相互之间不会有任何接口（类似 iPhone 的 App）。几乎没有性能开销，可以很容易地在机器和数据中心中运行。最重要的是，它们不依赖于任何语言、框架或系统。

Docker 以其集装化技术为应用程序带来便携性，在 Docker 中，应用程序可以跨平台运行自给系统。Docker 是由 Docker 引擎和 Docker 集线器组成的，前者是一个轻量级的运行时间和包装工具，后者则是应用程序共享和工作流程自动化的云服务。

2.3.3 DevOps 在企业私有云中的应用前景

基础设施部署与应用程序部署一直是软件开发生命周期中两个重要的环节，或者说是软件快速交付的制约。云计算的快速发展使得基础设施的供给周期从过去以月、周为单位下降到目前的以小时、分钟为单位，大大提高了基础设施部署的效率。云计算也给应用程序带来更多的可能性或者选择性，但是只有当应用程序部署效率同样提升后才能实现软件开发生命周期的缩短，而 DevOps 的目标就是提供持续部署的能力，强大的部署自动化手段确保部署任务的可重复性、减少部署出错的可能性。

企业私有云是一个企业专用的云基础设施，DevOps 对私有云而言其核心价值就是实现企业的“业务敏捷性”和“IT 融合”。提高业务敏捷性使企业获得快速适应市场和环境变化的能力。IT 融合使得企业能够借助高效的 IT 技术来提高公司业绩或市场竞争力。为了实现 DevOps 战略，以下三方面需要关注：

1. 统一团队认识

首先要消除开发团队与运维团队之间的“鸿沟”，运维团队必须明确地认识到，只有将产品快速部署到生产环境中才能实现给企业带来最大的价值，提升企业的竞争力。开发团队也需要认识到开发代码或者更改配置时，可能会对整个系统稳定性和性能带来影响。其次开发、运维团队所有当事方需要明白，在更大的企业流程中自己只是其中一部分。个体和团队的成功都要放在整个开发运维生命周期内来进行评价。团队应该围绕业务系统而不是职责来组织工作，这就是 DevOps 打破 IT 分组壁垒的寓意。对于许多机构来说，这是一个转变，每一个团队不再是基于自己的团队来评价和判断业绩好坏。

2. 端到端的标准化流程

DevOps 涵盖整个开发、测试、部署和运维整个生命周期，必须被看作是一个端对端的流程。在流程的不同阶段可以采取不同的方法，只要这些流程可以被组合到一起创建一个统一的流程。不同的企业实现这个流程可能采取不同的方法和手段，但是必须要保证流程的完整性。

3. 自动化的工具支持

DevOps 的思想是提升 IT 服务的敏捷性，选择最佳技术方案，并同时利用自动化工具尽量减少业务生命周期内的人工干预程度是企业实现 DevOps 必然之路。除非整个流

程都拥有理想工具作为底层支持，否则自动化工作流程根本就是纸上谈兵。从本质上讲，这些工具在某一阶段中的输出内容都要能够为下一阶段的工具所接纳，也就是说各类工具之间要能够顺畅衔接。对于企业而言，关键是要选择一套适合自己的工具，确保自足可控应该放在首要的考虑因素。

2.4 安全管理模型

目前业界尚无一个被广泛认可和普遍遵从的国际性云安全标准，主要有 CSA（Cloud Security Association，云安全联盟）和 ENISA（European Network and Information Security Association，欧洲网络和信息安全研究所）两大组织发布的云安全管理框架可供参考。

2.4.1 CSA 云安全管理模型

CSA 发布的“云计算安全指南”，从云服务模型角度提出了一个云计算安全参考模型，该参考模型描述了三种基本云服务的层次性及其依赖关系，并实现了从云服务模型到安全控制模型的映射，如图 2-4 所示。

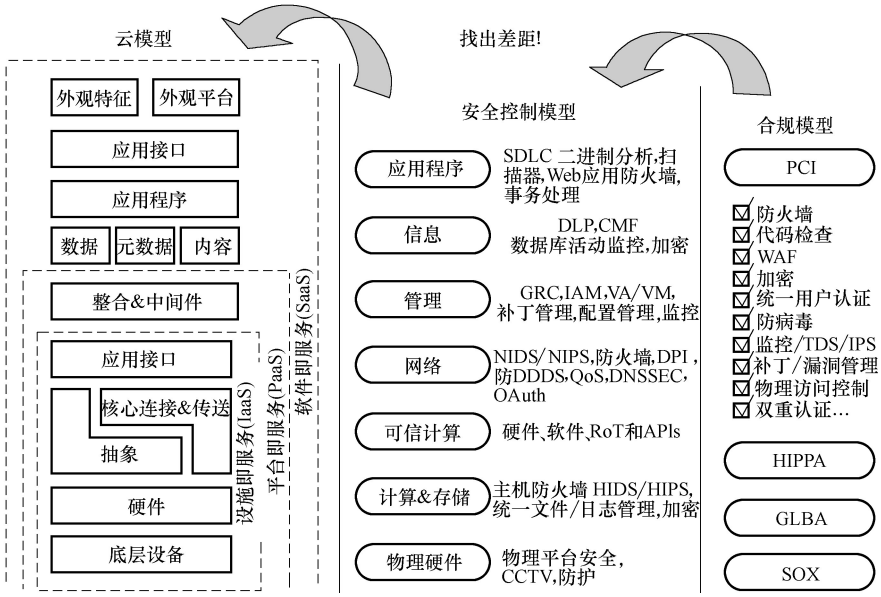


图 2-4 云计算安全参考模型

注：SDLC：软件生命周期；DLP：数据防泄漏；CMF：内容监控和过滤；GRC：企业治理、风险管理及合规审查；IAM：身份识别及访问管理；VA/VM：虚拟机；NIDS/NIPS：网络入侵检测系统/网络入侵防护系统；DPI：深入报文分析 DDOS：分布式拒绝服务；QoS：服务质量；DNSSEC：域名服务器安全扩展；OAuth：安全认证；RoT：信任根；APIs：应用接口；HIDS：主机入侵检测系统；HIPS：主机入侵防护系统；CCTV：闭路电视监控系统；PCI：支付卡产业；WAF：网站应用级入侵防御系统；IDS：入侵检测系统；IPS：入侵防护系统；HIPPA：健康保险可能性与定责法案；GLBA：金融现代化法案；SOX：萨班斯法案。

该安全参考模型的重要特点是云服务提供商所在的等级越低，云服务用户所要承担的安全能力和管理职责就越多。

在 SaaS 情况下，对服务本身和提供商的服务水平、安全、管控、合规性以及责任期望等有明确要求。在 PaaS 或 IaaS 情况下，这些内容的管理责任是用户自己的系统管理员，提供商对于安全保护底层平台和基础设施组件以确保基本服务的可用性和安全性，其具体要求可能会有一些相关的出入。

云计算中的安全控制机制与传统 IT 环境中的安全控制机制没有本质的不同。不过，云计算环境下存在其特有的安全风险，因此具有特别的安全关注领域。CSA 对云服务的主要安全关注点分为治理和运行两个领域，共涉及 12 个具体的关键域，见表 2-2。

表 2-2　云服务安全关注领域

治理域	运行域
治理和企业风险管理	传统安全、业务连续性和灾难恢复
法律和电子证据发现	数据中心运行
合规与设计	应急相应通告和补救
信息生命周期管理	应用安全
可移植性和互操作性	加密和密钥管理
	身份和访问管理
	虚拟比

2.4.2　ENISA 云安全管理模型

ENISA 也发布了云安全风险评估与云安全框架等相关的研究成果，在其发布的“云计算信息安全保障框架”中，建立了图 2-5 所示的 ENISA 云安全管理模型，重点关注 10 个领域，为云发展和云安全建设提供了一定的依据。

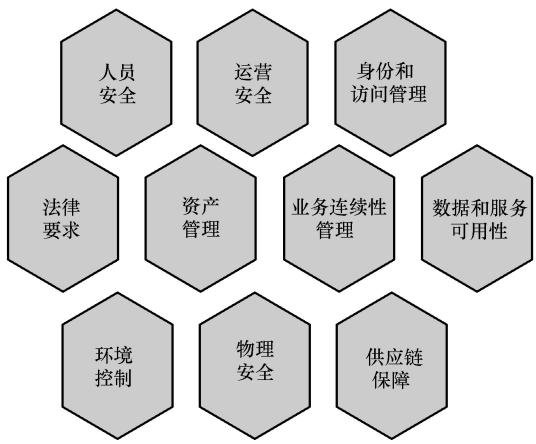


图 2-5　云计算信息安全保障框架

第 3 章
商业云计算产品

3.1 IBM 私有云解决方案

3.1.1 IBM 虚拟化技术

在当今的云计算时代，“虚拟化”这个词大家是耳熟能详了。但是，要问谁是虚拟化的鼻祖，可能有人会回答错误。其实，IBM 才是服务器虚拟化技术的先驱。因为早在 20 世纪 60 年代，IBM 就首次在其大型机上提出并实现了服务器虚拟化，利用该技术实现对属于稀有而昂贵资源的大型机硬件的分区。迄今为止，虚拟化和系统分区已在 IBM 大型机系统上存在了几十年，从运行 OS/390、z/VM、z/OS 的 IBM 大型机，到近些年的 IBM iSeries 和 pSeries 平台。最早使用虚拟化技术的是 IBM 7044 计算机，之后 20 世纪 60 年代还开发了型号为 Model 67 的 System/360 主机。Model 67 主机通过虚拟机监视器（Virtual Machine Monitor，VMM）虚拟所有的硬件接口。在早期的计算中，操作系统被称作 Supervisor（监管者），能够运行在其他操作系统之上的操作系统被称作 Hypervisor（超级监管者）。其中，VMM 直接运行在底层硬件上，允许执行多个虚拟机（VM），每一个 VM 运行自己的操作系统实例（Conversational Monitor System，CMS）。

1999 年 IBM 公司又在 AS/400 基础上提出了“逻辑分区（LPAR）”技术和新的高可用性集群解决方案。在 POWER 管理程序上运行的 AS/400 LPAR 令单台服务器工作起来如同 12 个独立的服务器。而在 2002 年，IBM 的 AIX5L V5.2 还首次包括了 IBM 实现的动态逻辑分区（DLPAR）。DLPAR 允许在无需重启系统的情况下，将包括处理器、内存和其他组件在内的系统资源分配给独立的分区。这种在不中断运行的情况下进行资源分配的能力不仅让系统管理变得更加轻松，而且因为能够更好地使用资源而帮助降低

总拥有成本。在应用上，Unix 上的虚拟化也非常成熟，IBM 的客户订购的 System i5 595 中有 82% 具备逻辑分区功能，IBM 客户管理的分区总数超过 45000 个。

3.1.1.1 服务器虚拟化技术

IBM PowerVM 是在基于 IBM POWER 处理器的硬件平台上提供的具有行业领先水平的虚拟化技术家族。它是 IBM Power System 虚拟化技术全新和统一的品牌，包含的技术有逻辑分区、微分区、Hypervisor、虚拟 I/O 服务器、高级 Power 虚拟化（APV）、PowerVM Lx86 和在线迁移灵活性（Live Partition Mobility）。

PowerVM 有三个版本，分别是 PowerVM 快速版（Power VM Express Edition）、PowerVM 标准版（PowerVM Standard Edition）和 PowerVM 企业版（PowerVM Enterprise Edition）。

1. PowerVM 快速版

PowerVM 快速版只能在 IBM Power520、Power550、PS700、PS701、PS702、PS703、PS704、Power710、Power720、Power730、Power740、Power750 等几种服务器上购买。它们的主要功能包括：最多可以创建三个分区，支持微分区虚拟化功能，支持共享分布式容量（Shared Dedicated Capacity），支持 PowerVM Lx86 功能，不支持多共享处理池（Multiple Shared Processor Pools）、在线分区灵活性（Live Partition Mobility）和有效内存共享（Active Memory Sharing）功能，分区创建和管理使用 IVM 工具。

2. PowerVM 标准版

PowerVM 标准版可以在全部的 IBM Power 服务器上购买。它们的主要功能包括：对分区数量没有限制，支持微分区虚拟化功能，支持共享分布式容量、PowerVM Lx86 和多共享处理池功能。不支持在线分区灵活性和有效内存共享功能。分区创建和管理可以使用 IBM 虚拟化管理（IVM）或者硬件管理控制台（HMC）工具。

3. PowerVM 企业版

PowerVM 企业版可以在全部的 IBM Power 服务器上购买。它们的主要功能包括对分区数量没有限制，支持微分区虚拟化功能，支持共享分布式容量支持 PowerVM Lx86 功能，支持多共享处理池功能，在线分区灵活性和有效内存共享功能，分区创建和管理可以使用 IVM 或者 HMC 工具。

共享分布式容量，多共享处理池，在线分区灵活性和有效内存共享功能只有在基于 IBM POWER6/POWER7 处理器的服务器上才可以实现。

3.1.1.2 IBM 存储虚拟化技术

虚拟存储是一种具有智能结构的系统，它将允许以透明有效的方式在磁盘和磁带上存储数据，统一管理磁盘空间，使存储系统能够容纳更多的数据，更多的用户可以共享同一个系统。

IBM 提供的存储虚拟化解决方案是 System Storage SAN 卷控制器（SAN Volume Controller，SVC），SVC 是一个软硬件集成化的产品，它集成了 IBM 服务器、基于 Linux 内核的存储操作系统以及专业的虚拟存储软件。SVC 融合了存储业界突破性技术，通过

创建共享的存储池来聚集不同存储系统提供了的容量，从单一界面管理，实现异构磁盘系统的有效整合与集中管理。SVC 提供了通用的复制功能，适用于所有存储系统，允许在不关闭服务器的情况下动态转移数据。

如图 3-1 所示，SVC 是整个 SAN 网络的控制器，将整个 SAN 网络中的各种存储设备整合成一个巨大的“存储池”，使用户充分利用存储资源并可按需分配存储空间、性能和功能。SVC 的主要功能包括：

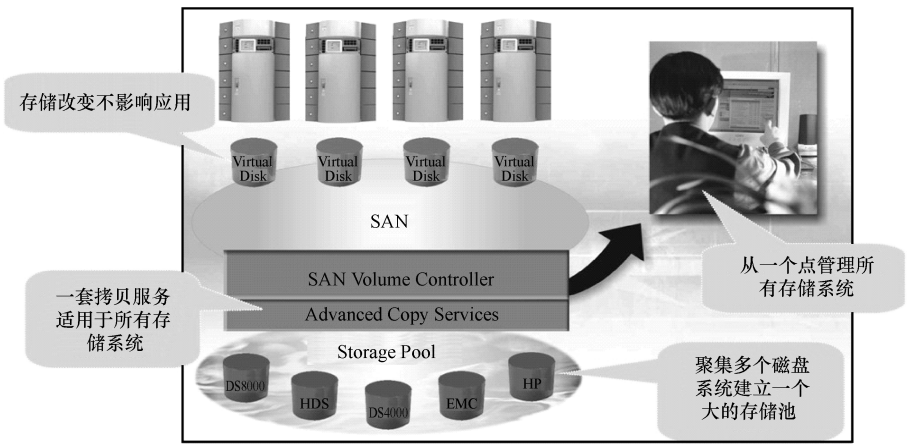


图 3-1　SVC 物理部署

注：Virtual Disk：虚拟盘；SAN：存储区域网络（Storage Area Networking）；SAN Volume Controller：存储虚拟化产品，提供对拉于 SAN 内部的磁盘存储进行快集中和卷管理功能；Advanced Copy Services：高级复制服务；Storage Pool：存储池；DS8000：一款 IBM 存储产品；HDS：日立数据系统，这里泛指 HDS 的存储产品；DS4000：一款 IBM 存储产品；EMC：易安信公司，这里泛指 EMC 的存储产品；HP：惠普公司，这里泛指 HP 的存储产品。

1）构建统一、合理的、高可扩展的存储架构。

2）集中管理存储系统，把多个存储系统整合成单一的存储池，兼容市面上常见的主机系统和存储产品。

3）简化各种复杂的存储系统管理，可通过图形界面（GUI）管理所有存储系统。

4）迁移过程无需停机，支持 7 ×24h 业务不间断运行，迁移过程中主机系统和应用不受影响。

5）提供快照、数据复制等功能，做到跨存储及跨地域的数据保护。

6）采用 SVC 可方便地帮助用户实施分级存储。

1. SVC 的基本概念和工作原理

SVC（SAN Volume Controller SAN 卷控制器，IBM 一款硬件型存储虚拟化产品）采用带内（In-Band）方式进行存储虚拟化。SVC 系统实际上是一个集群（Cluster）系统，由节点（node）组成。一个 SVC 系统至少包含 2 个节点，每 2 个节点组成一个输入/输出组（I/O Group），它用来为主机提供输入/输出（I/O）服务。到现在为止，一个 SVC 系统最多包含 8 个节点，即 4 个输入/输出组。

在一个 SVC 系统中，存储子系统中的一个或多个存储单元被映射为 SVC 内部的

存储单元管理盘（Managed Disk，MDisk），一个或多个 Mdisk 可以被虚拟化为 1 个存储池（MDG），所有的 MDG 对所有的输入/输出组均可见。MDG 是一个存储池，根据一定的分配策略，如条状化（Striped）、镜像（Image）、序列化（Sequential），分配虚拟的存储单元，称为虚拟盘（VDisk）。输入/输出组以 Vdisk 为单位对主机提供逻辑单元掩码（LUN-Masking），也称为逻辑单元映射（LUN-Mapping）服务，使主机可通过主机总线适配器（HBA）访问被提供逻辑单元掩码服务的虚拟盘，如图 3-2 所示。

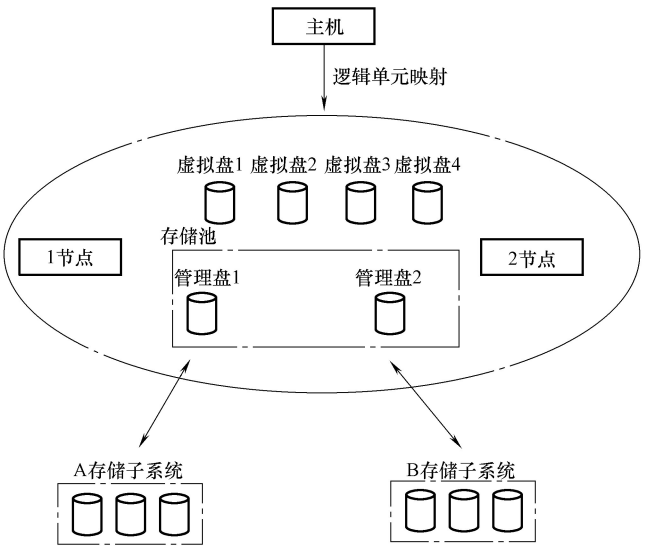


图 3-2　SVC 中的管理盘和存储池以及虚拟盘之间的关系

如图 3-3 所示，在存储子系统与主机之间引入 SVC 后，主机所有的 I/O 必然要经过 SVC，相当于 SVC 要接管从主机过来的所有 I/O。要做到这一点，SVC 内部必须实现一个虚拟层，使得主机仿佛可以直接访问真正的物理存储系统。这个虚拟层的实现依赖于存储虚拟化技术。存储虚拟化的基本概念是将实际的物理存储实体与存储的逻辑表示分离开来，应用服务器只与分配给它们的逻辑卷（或称为虚卷）打交道，而与其数据是在哪个物理存储实体上无关。

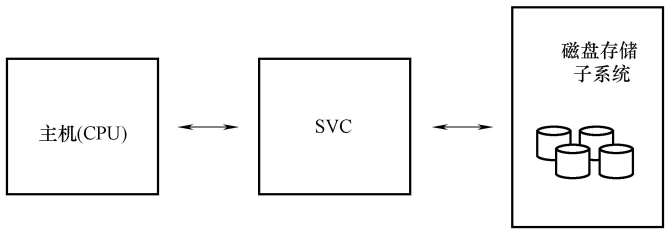


图 3-3　加入 SVC 后加速对存储子系统的访问 I/O

2. 存储子系统级别的虚拟化

存储子系统级别的存储虚拟化如图 3-4 所示，使用主机总线适配器（Host Bus Adapter）例如 1394 主机总线适配器（控制器）连接磁盘柜，通过 1394 控制器驱动，物理磁盘被映射为系统中的 sda、sdb、sdc 等小型计算机系统接口（SCSI）磁盘块设备，块设备上层的虚拟化在原理上和主机级别子系统块设备的虚拟化类似。

3. 网络级别的存储虚拟化

网络级别的存储虚拟化分为两种，即带外（Out of Band）和带内（In Band）。图 3-5 是带外存储虚拟化的一种方式，存储子系统通过 SAN 使得 3 个不同类别的操作系统在元数据服务器（Metadata Server）的锁机制控制下共用存储子系统中的 3 个存储单元。在每个主机上，3 个存储单元被虚拟化为一个条化组（Stripe Group），使得各个主机可以采用统一的条化（Stripe）策略控制各自的 I/O 行为。

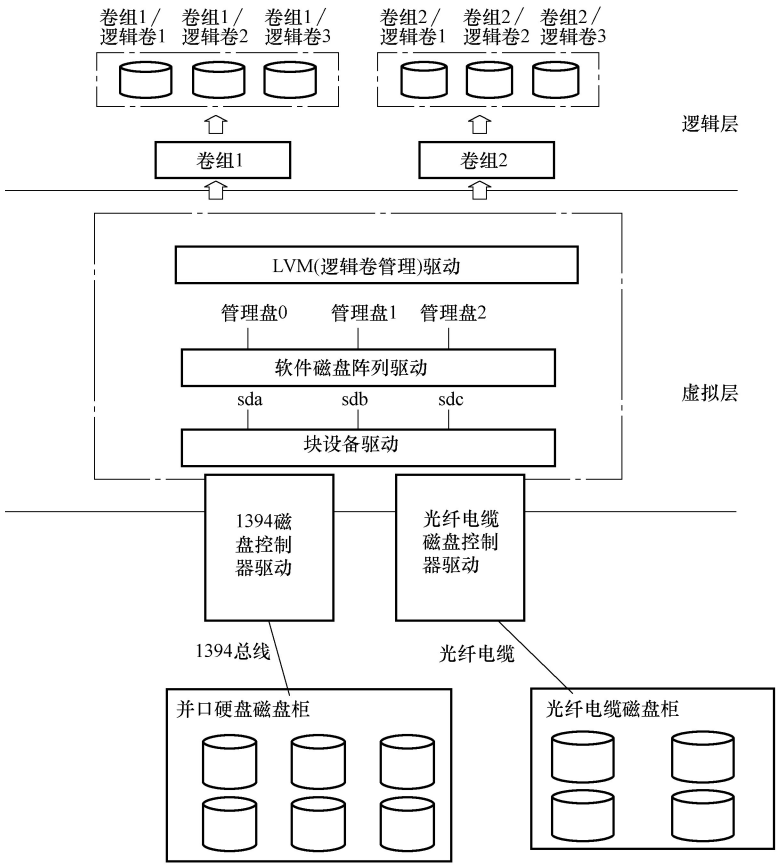


图 3-4　存储子系统级别的存储虚拟化（如 Linux）

注：sd：SCSI（小型计算机系统接口）磁盘块设备。

带内的方式实际上是通过数据通道（Data Path）上的虚拟化软件，把呈现在 SAN 中一个或多个存储子系统的存储单元虚拟化成另一种方式的虚拟存储单元，称为虚拟盘。图 3-6 属于带内存储虚拟化。SVC 使用带内虚拟化方式，即 SVC 把主机级别的虚拟化实现在 SAN 的网络层次上加以实现。

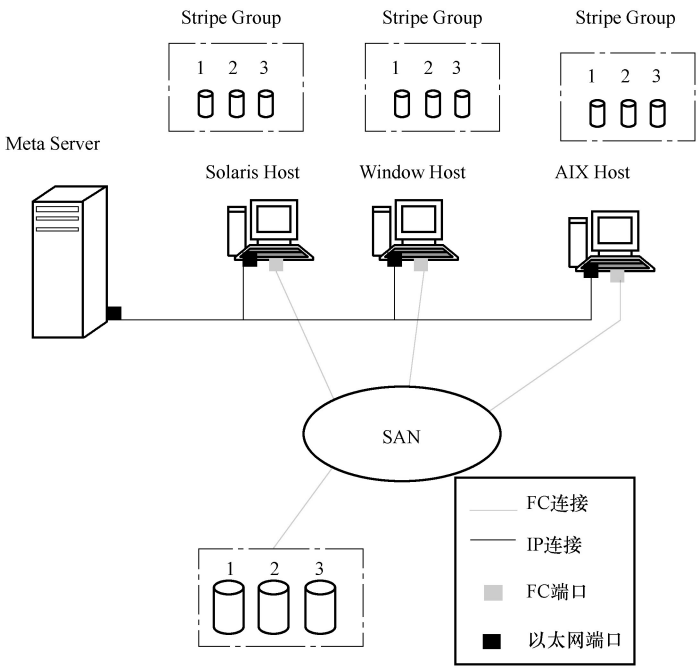


图 3-5　带外的存储虚拟化

注：Stripe Group：独立组织成一到多个分组；Solaris：Sun Microsystems（太阳计算机系统有限公司）研发的计算机操作系统；AIX：IBM 开发的一套类 UNIX 操作系统；SAN：存储区域网（Storage Area Networking）；FC：光纤通道（Fibre Channel）；IP：网际协议（Internet Protocol）。

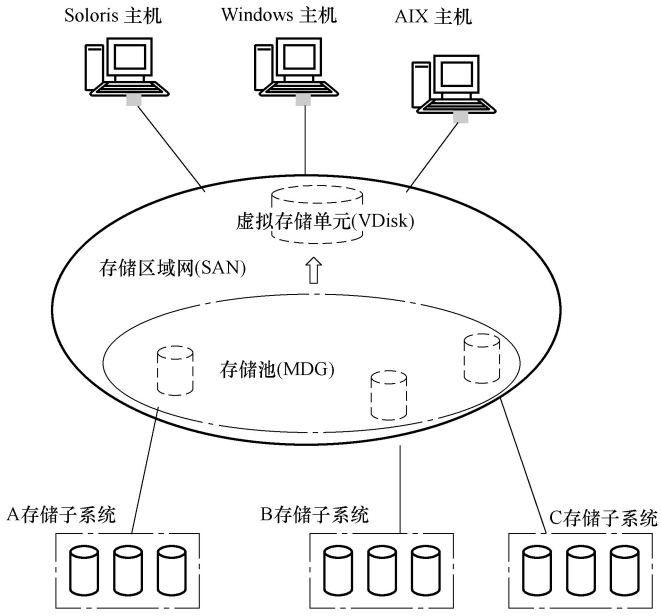


图 3-6　带内存储虚拟化

如图 3-7 所示，传统的 SAN 网络中，每种存储系统都自成一体，就像一个个独立的孤岛，无法构成一片统一的大陆。而 SVC 是存储业界又一次崭新的突破，就像存储历史上的磁盘阵列（Redundant Arrays of Independent Disks，RAID），主机系统的存储管理体系和虚拟磁带技术，这些重要的发明均源自 IBM。SVC 是整个 SAN 网络的控制器，它将整个 SAN 中的各种存储设备整合成一个巨大的存储池，充分利用了存储资源和按需分配存储空间、性能和功能。

SVC 实现了虚拟存储层的功能，将存储智能加入到 SAN 网络中。现在用户可以按照应用不断变化的需求来分配存储，而不再受制于存储子系统设备在功能和性能上的限制。SVC 又是一个 SAN 网络的中心管理控制点，而且它对服务器的操作系统和存储子系统是透明的。

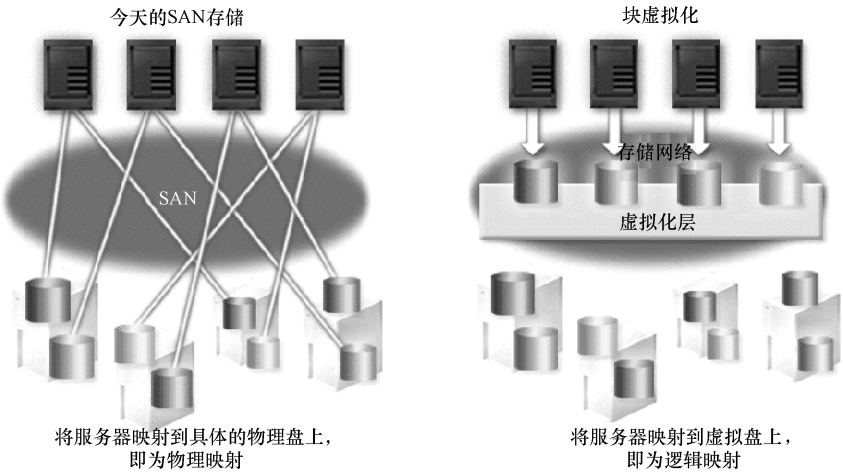


图 3-7　存储池引入 SVC 前后的对比

3.1.1.3　IBM 软件定义解决方案

1. IBM 软件定义存储解决方案

在软件定义环境时代到来的时候，IBM 自然也毫不例外地带着先进的技术和解决方案积极参与着。在软件定义存储的领域，IBM 及时推出了软件定义的“弹性存储”技术和方案，该项技术将主存储、二级存储和全闪存技术完美地结合在一起，通过软件定义提高了云计算过程中数据生命周期管理的效率和存储整体自动化管理水平。

弹性存储技术为企业提供了空前的性能、无限的扩展，并能够通过自动将数据移动到最经济的存储设备上使存储成本降低高达 90%。这项新技术源于 IBM 研究院，它让企业能够利用（而不只是管理）由无数设备、传感器、业务流程和社交网络所产生的各种形式的大量数据。新存储软件更适用于高数据密集型应用，此类应用需要高速存取大量信息，例如地震数据处理、风险管理和金融分析、气象建模和科学研究以及在实时零售中决定下一个最佳行动。IBM 研究院已证明弹性存储能够在仅仅 43min 内利用单个系统成功扫描 100 亿个文件。

在其核心部分，弹性存储以 IBM 的通用文件并行文件系统（GPFS）为基础来提供

在线存储管理、可扩展访问以及能够管理庞大数据量和数十亿文件的集成数据治理工具。例如，相对于标准 SAS（软件公司名字）磁盘，弹性存储还可以利用内置于服务器中的 Flash 存储来实现六倍的性能提升。该功能可识别服务器内的 Flash 存储并自动将其作为高速缓冲存储器来提升性能。

弹性存储将存储虚拟化，允许多个系统和应用共享公用存储池。这有助于实现透明的全球数据存取而无需修改应用，也无需额外的或经常的存储管理变更。由于弹性存储不依靠集中管理来确定文件的位置和布局，所以在发生软件或硬件故障时客户能确保数据存取的连续性和高可用性。

另外一个关键的指标是，其自动和智能地将数据移动到其他可用存储系统上的能力。例如，通过策略驱动的特性和实时分析，弹性存储能够自动将不常使用的数据移动到不太昂贵的低成本磁带驱动机上，而将经常存取的数据存储在高速 Flash 系统上，以便更快地进行存取。这些特性可使企业有效降低存储管理成本。

IBM 弹性存储的其他特性还包括：

1）提供原生加密及安全擦除特性，满足合规性需求。

2）支持 OpenStack 云管理软件，帮助客户能够跨越私有云、公有云和混合云进行数据存储、管理及访问，实现全球数据共享和协作。

3）除了支持 OpenStack Cinder 和 Swift 存取，弹性存储还支持其他开放 API，如 POSIX 和 Hadoop。

2. IBM 软件定义网络解决方案

IBM 的软件定义网络虚拟化技术是 SDN-VE。

1）基于 IBM 的 DOVE 技术。

2）使用现有的 IP 架构和现有的物理网络保持不变。

3）为虚拟化工作负载提供基于服务器的东西向连接。

4）为现有的非虚拟化工作负载提供连接服务。

5）目前发布的版本支持 VMware 和 RHEL KVM 的 Hypervisor。

6）目前发布的版本支持 16000 个虚拟网络和 128000 个终端（VMs）。

IBM SDN（软件定义网络）的优势在于：

1）IBM SDN-VE 基于 VXLAN 标准报文格式。

2）IBM SDN-VE 控制面具有可扩展性。

3）IBM SDN-VE 不需要物理网络的 IP 组播支持。

IBM SDN 流表控制图如图 3-8 所示。从现有主流的 SDN 覆盖（SDN overlay 解决方案厂商来看，IBM overlay 解决方案提供了更加丰富的功能，下面是一个 IBM 和其他厂商的差异化对比。首先，IBM 解决方案不需要特殊的硬件，从而极大地降低了总体拥有成本，而 VMware 和 Cisco 等厂商都需要软件许可证（License）或者特殊硬件。其次，IBM 采用 Openday light 的开源解决方案，所有源代码都是开源的。用户可以按照需要使用并修改源代码，甚至通过将源代码提交到社区来影响 Opendaylight 的架构，然而 Cisco、VMware 等厂商都是非开源的。第三，IBM SDN 解决方案支持多种虚拟化平台，而不是绑定在某一个虚拟化平台之上。

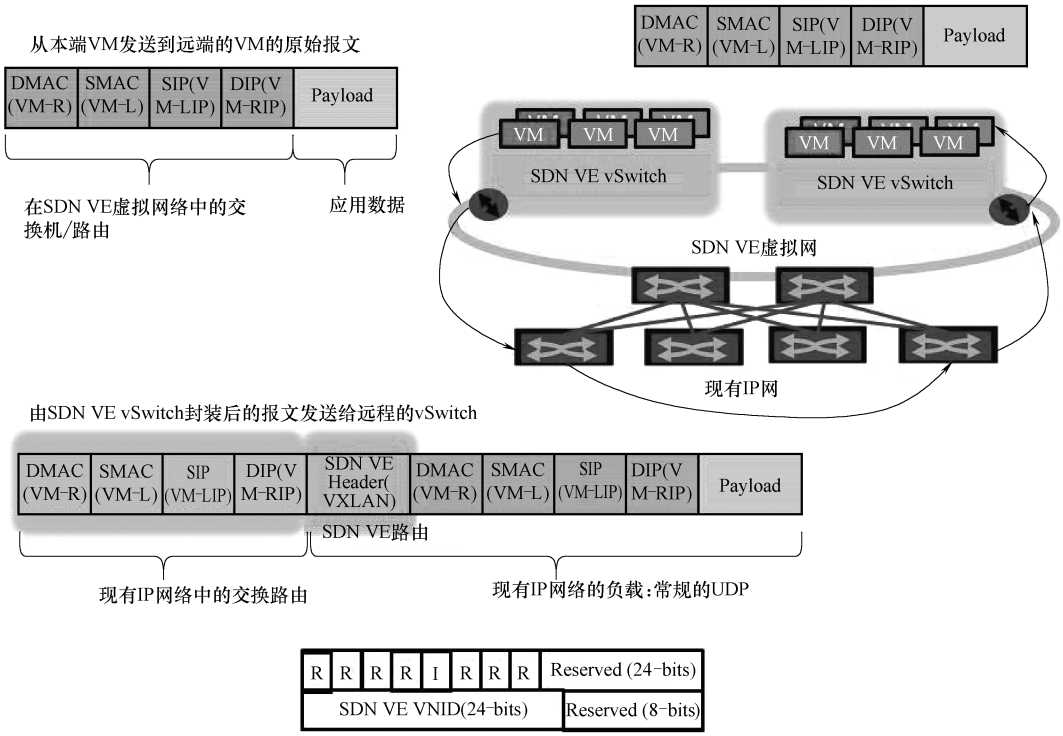


图 3-8　IBM SDN 流表控制图

注：VM：虚拟机（Virtual Machine）；DMAC（VM-R）：直接存储器存取通道（Direct Memory Access Channel）；SIP（VM-LIP）：Session Initiation Protocol，会话初始协议；DIP（VM-RIP）：双列直插式组装（Double In-line Package）；Payload：有效载荷；SDN VE：SDN（Software Defined Networking）软件定义网络，SDN VE 用于虚拟环境的软件定义网络；vSwitch：虚拟交换机；SDN VE vSwitch：用于虚拟环境的软件定义网络虚拟交换机；SDN VE Routing：SDN 虚拟路由；SDN VE Header（VXLAN）：用于虚拟环境的软件定义网络虚拟接头（虚拟可扩展局域网）；SDN VE Virtualized Network：用于虚拟环境的软件定义的虚拟网络；Hypervisor：系统管理程序；Existing IP Network：现在的 IP 网络；UDP packet：用户数据报协议（User Datagram Protocol）包；VXLAN/SDN-VE Header：虚拟可扩展局域网/软件定义网络-虚拟接头；Reserved（24-bits）：保留 24 位；Reserved（8-bits）：保留 8 位。

3.1.2　IBM 云管理技术特性和应用场景

大型企业包括银行在内的私有云环境管理，应该包含从 IaaS、PaaS 到 SaaS 的各个层面。云不是为建而建的，最终在云上将运行各种应用系统，用户通过所谓“云 + 端”的端（即包括网络、移动等各种渠道）来访问的。要保证一个企业云的稳定、安全和健壮，就需要一个强大的管理体系和管理平台来支撑。

IaaS 的管理是指对包括计算、存储、网络在内的基础资源层的管理。PaaS 的管理是指对于各种中间件软件、系统软件的管理，例如：包括数据库、ETL 工具、分析挖掘工具、非结构化处理工具、大数据分析等在内的数据处理和分析类软件，以及包括应用服务器、企业服务总线、业务流程引擎、规则引擎等在内的各种中间件。而 SaaS 的管

理是指对各种最上层应用的管理，包括如何在云的环境下，做到真正满足云特征应用的开发、测试和部署。因为在云计算时代，云应用的上线周期是很短的，尤其是移动类的应用，版本更迭频率极高。由于底层资源保证了随需供给、弹性伸缩，就要求上层应用能够快速开发、快速部署，完全体现思路敏捷性。IBM 银行业务云平台如图 3-9 所示。

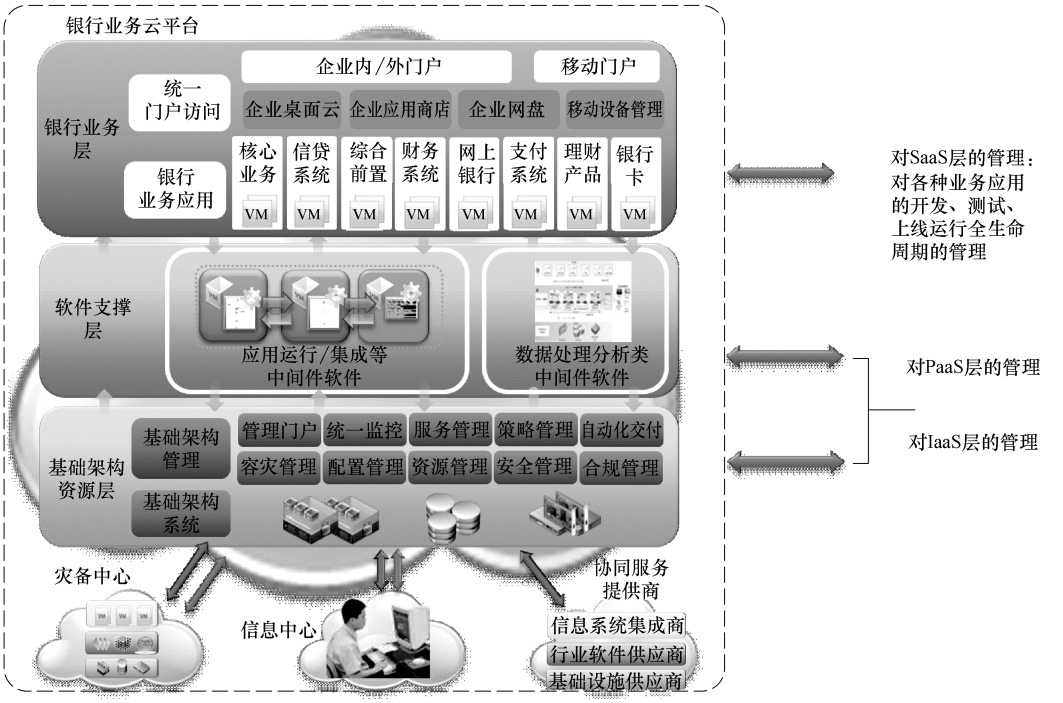


图 3-9　IBM 银行业务云平台

基于上述思路，IBM 的云管理解决方案分为两大部分：一部分是针对资源层和中间件层的管理，解决方案的名称为 SmartCloud Orchestrator（SCO）；另一部分是针对云应用开发、测试、部署的高度自动化的工具 UrbanCode 系列。

1. 资源层和 PaaS 层管理—SCO

SCO 底层利用 OpenStack EE 实现对计算、存储、网络资源的自动部署，以及管理平台和资源池的高扩展性；通过集成自动化流程引擎，实现对部署过程的全流程控制和 SCO 的高可集成性；通过集成模式引擎（Pattern Engine），SCO 能实现对系统和应用模式的设计、管理和部署功能，使得在传统环境需要数天或数周才能完成的应用部署缩短到几个小时。SCO 技术架构如图 3-10 所示。

2. 开放的可扩展的资源池管理

SCO 是一个开放的、可扩展的资源池管理平台，基于业界标准的管理平台 OpenStack 的接口实现各个逻辑组件之间的互连互通。SCO 逻辑架构如图 3-11 所示。

SCO 在 IaaS 层对于不同的虚拟机管理程序的支持，是通过 OpenStack 来实现的，不仅支持 Xen、KVM、Hyper-V、WMware 等基于 X86 体系的虚拟机拟管理程序，更能够支持 PowerVM、ZVM 等小型机的虚拟机管理程序。SCO 同时集成了监控、备份与恢复、

安全控制等模块，为整个云运行环境提供了完整的解决方案。

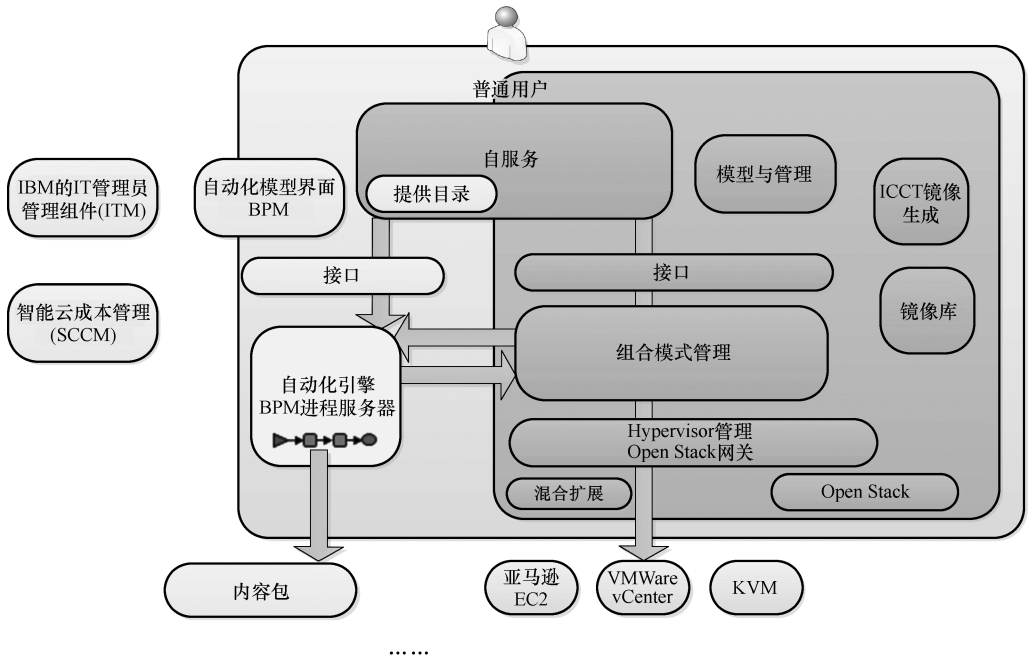


图 3-10　SCO 技术架构

注：BPM—Business Process Management，业务过程管理。

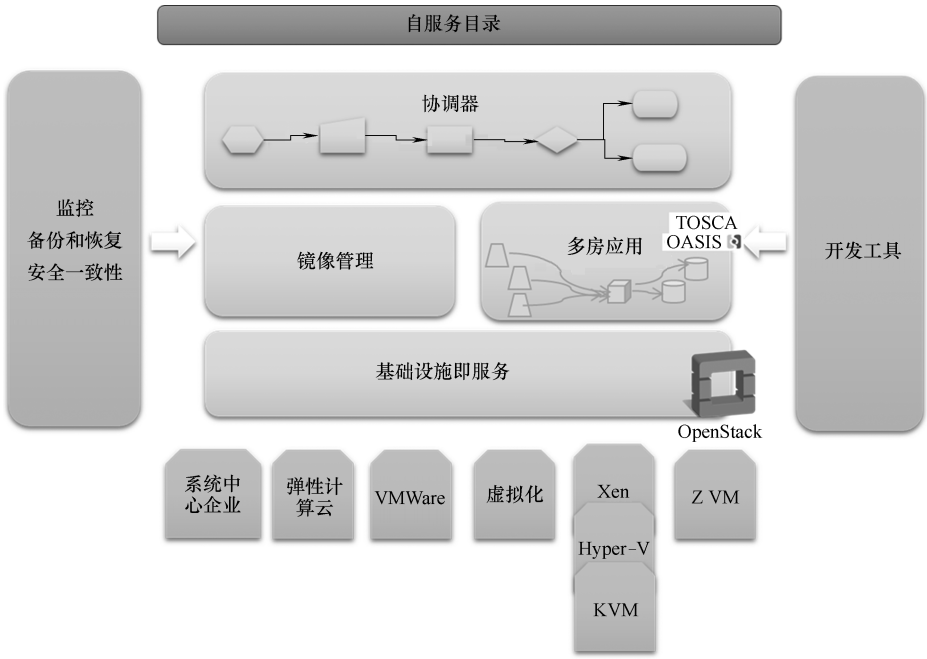


图 3-11　SCO 逻辑架构

注：VMWare：虚拟机软件；Xen：虚拟机管理程序名；Hyper-V：虚拟机；KVM：基于内核的虚拟机（KVirtual Machine）；Z VM：IBM 虚拟机名称。

IBM 作为 OpenStack 的基金会的铂金会员，目标是帮助 OpenStack 平台的进步，维持一个充满活力的生态系统，我们推荐云用户以及云提供商的 IaaS（基础设施即服务）平台首选使用 OpenStack。

3. 通过业务流程管理工作负载

每个域的资源，例如计算域的计算资源、存储域的存储资源、网络域的网络资源都有其生命周期，同时，针对这些资源，需要监控并进行 IT 资产管理、配置管理和运行维护请求的管理，通过业务流程（Orchestration Engine）可以把这些资源以及针对监控、管理、运维等服务进行有机整合。SCO 功能组件如图 3-12 所示。

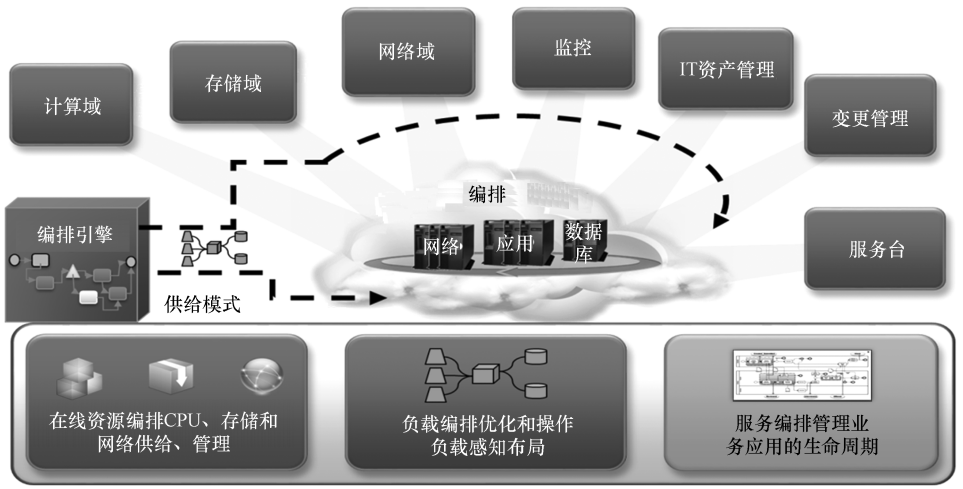


图 3-12　SCO 功能组件

4. 运维管理

SCO 运维平台可以提供 IT 资产管理、配置管理、流程管理（包括事件、问题、变更等）的服务，结合监控，为数据中心的稳定运行提供强有力的保证。SCO 功能示意图如图 3-13 所示。

5. 统一的 IaaS 网关

IBM 基础架构云快速部署解决方案套件 SCO 的 IaaS 网关（IaaS Gateway）模块，管理了各种不同类型的资源池。通过 IaaS 网关可调用不同的资源池内的服务，其原理架构如图 3-14 所示。

SCO IaaS 网关模块对上层应用，提供了统一的遵循 OpenStack 规范的标准接口；对下层资源池的管理则集成了各个不同类型的资源池。SCO IaaS 网关资源池集成原理如图 3-15 所示。

SCO IaaS 网关模块将不同类型的资源池的服务，统一到服务目录（Service Catalog）中并对外集中提供相关服务。SCO IaaS 网关服务提供集成原理如图 3-16 所示。

6. 镜像管理

SCO 镜像管理模块通过 OpenStack 的标准接口 REST 接口与 IaaS 网关互连，接收 IaaS 网关上的指令后，可以管理不同类型的资源池上的本地镜像以及远端镜像，SCO 镜

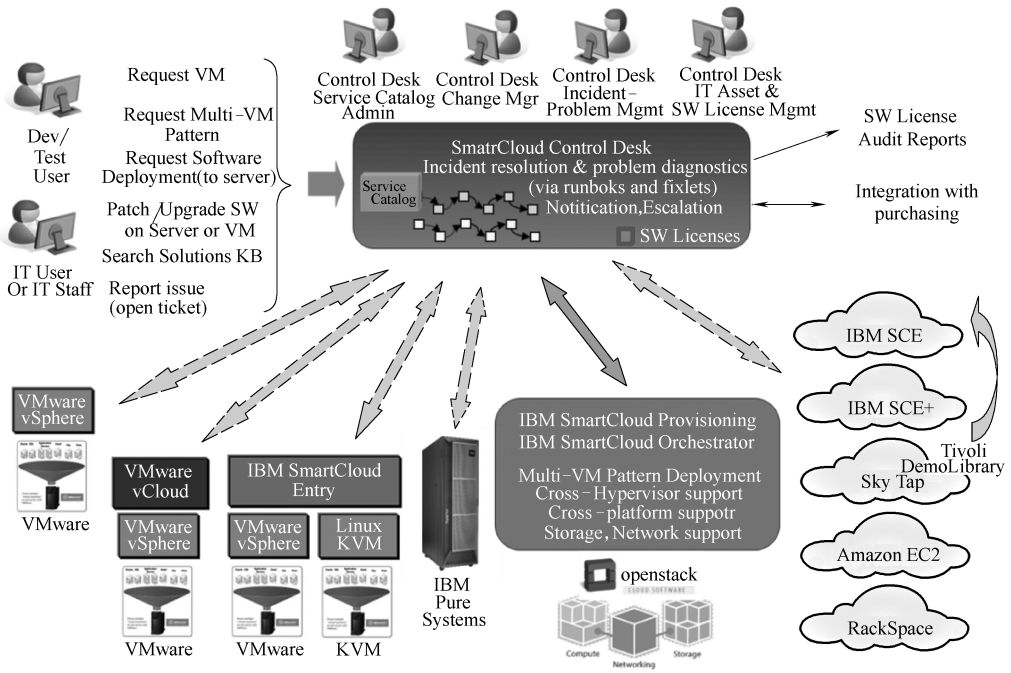


图 3-13　SCO 功能示意图

注：Dev/Test User：开发、测试用户；IT User or IT Staff：IT 用户或工作人员；Request VM：请求虚拟机；Request Multi-VM Pattern：请求多虚拟机模式；Request Software Deployment（to server）：请求软件部署到服务器；Patch/Upgrade SW on Server or VM：服务器或虚拟机上补丁/更新；Search Solutions KB：搜索解决方案 KB；Report issue（open ticket）：报告问题（打开罚单）；Control Desk Service Catalog Admin：控制台服务目录管理员；Control Desk Change Mgr：控制台变更经理；Control Desk Incident-Problem Mgmt：控制台事件－问题经理；Control Desk IT Asset & SW License Mgmt：控制台工厂资产 &SW 认证经理；SmartCloud Control Desk：智能云控制台；Incident resolution & problem diagnostics（via runbooks and fixlets）：事件解决 & 问题诊断（通过手册和解决方案）；Notification，Escalation：提取，扩大；SW License：SW 认证；SW License Audit Reports：SW 认证审计报告；Integration with purchasing：与采购整合；IBM SCE：一种云解决方案；Amazon EC2：Amazon Elastic Compute Cloud 亚马逊弹性计算云；Tivoli Demo Library：演示库；RackSpace：公有云；IBM SmartCloud Provisioning：IBM 智能云供应；IBM SmartCloud Orchestrator：IBM 智能云协调器；Muti-VM Pattern Deployment：多 VM 模式部署；Cross-Hypervisor support：跨 Hypervisor 支持；Cross-platform support：跨平台支持；Storage，Network support：存储，网络支持；Openstack：一个云平台管理的项目；Compute：计算；Networking：网络；Storage：存储；IBM Pure Systems：IBM 纯系统；VMware vSphere：云计算操作系统；VMware vCloud：虚拟云基础架构；KVM：基于内核的虚拟机（K Virtual Machine）；IBM SmartCloud Entry：IBM 智能云条目。

像管理功能原理如图 3-17 所示。

SCO 镜像管理模块提供注册新的扩展镜像服务，并整合所有其他云的镜像注册服务；实现 OpenStack 的 v1 和 v2 的 REST 的 API 接口，提供其他额外属性（如操作系统信息）；除了标准镜像服务的 API 服务，还提供了可使镜像元数据检索的服务。IBM 基础架构云快速部署解决方案套件 SCO 的镜像管理模块接口示意图如图 3-18 所示。

7. 利用 IBM 特有的 Pattern 模式实现 PaaS 功能

SCO PaaS 模块可以部署用户所需要的应用系统平台，SCO 应用系统平台示意图如

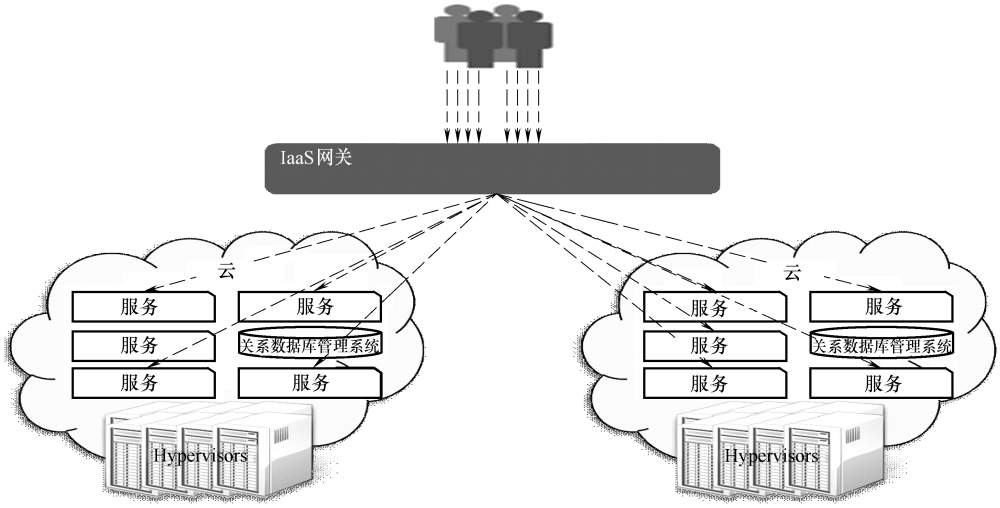


图 3-14　SCO IaaS 网关原理架构

注：Hypervisors：管理程序。

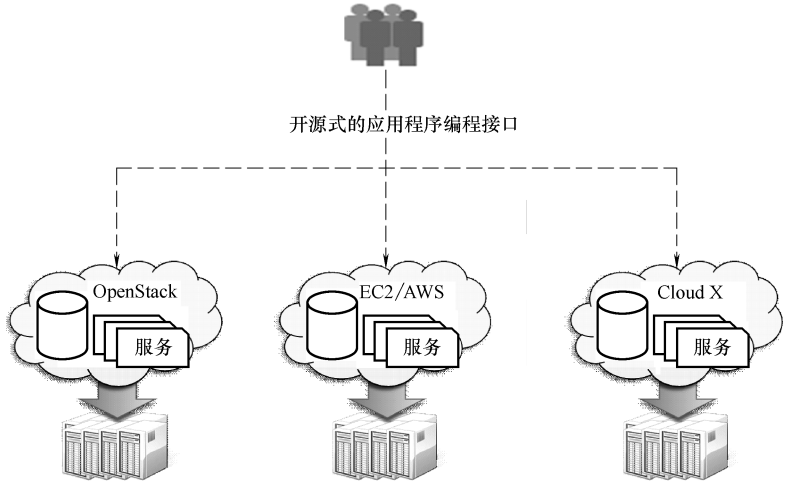


图 3-15　SCO IaaS 网关资源池集成原理

图 3-19 所示。

SCO PaaS 模块定义好部署的虚拟机类型和虚拟机上所需安装的软件堆栈，根据拓扑结构，可以自动部署并得到用户所需要的平台。SCO 应用系统拓扑结构示意图如图 3-20 所示。

8. 云应用管理-Urban Code

我们知道，企业应用（Enterprise Application）是指运行在操作系统和中间件之上，实现特定业务逻辑的软件发布包和业务数据，与之对应，企业应用部署（Enterprise Application Deployment）是把企业应用部署在一台或多台计算机的操作系统或中间件上，从而可以提供一个可供测试、培训和生产的运行环境。

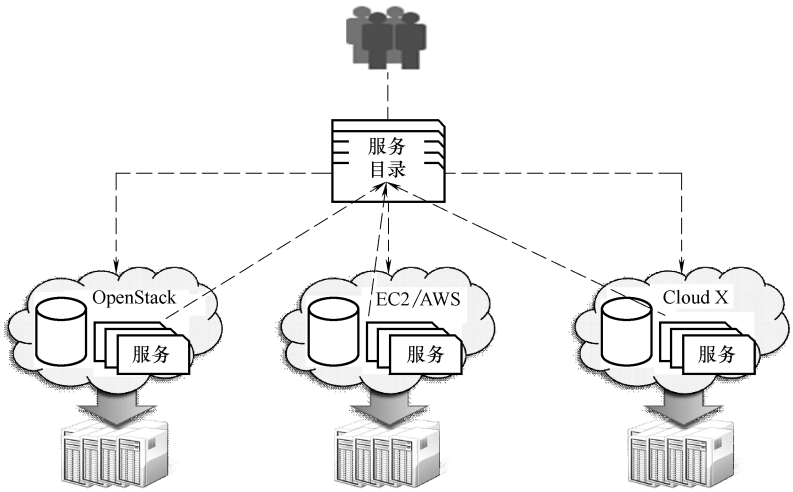


图 3-16　SCO IaaS 网关服务提供集成原理

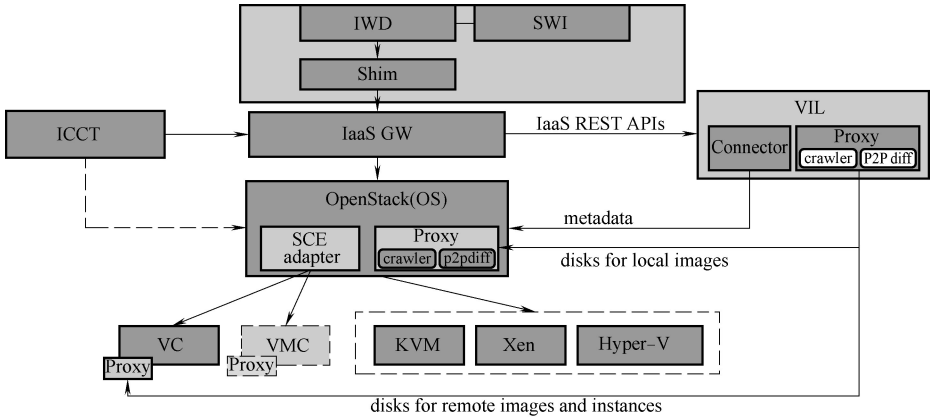


图 3-17　SCO 镜像管理功能原理

注：IWD：工作负载部署软件（IBM Workload Deployer）；SWI：磁敏感加权成像（susceptibility weighted imaging）；Shim：填充层；IaaS GW：IaaS 网关；Open Stack（OS）：开源架构；VIL：垂直注入逻辑（Vertical Injection Logic）；VC：虚拟计算机；VMC：虚存计算机（Virtual Memory Computer）；KVM：基于内核的虚拟机（K Virtual Machine）；Xen：一虚拟机管理程序名；Hyper-v：虚拟机；IaaS REST APIs：IaaS 数据接口；Metadata：元数据；Disk for local images：本地磁盘图像；Connector：连接器；Proxy：代理服务器；Crawler：爬行器；P2P diff：P2P 扩散器；disk for local images：本地磁盘图像；disk for remote images and instances：远程磁盘图像和实例。

当前，企业尤其是银行业，为了应对新技术带来的市场快速变化，需要将业务的流程调整和创新尽快交付到业务部门或终端用户，这就需要提高软件的持续交付能力，包括交付速度、交付成本、交付质量和体验等方面。而企业应用部署是软件持续交付（Continuous Delivery）的重要环节，当所需要交付的企业应用部署架构复杂（比如多机）、部署环境多（比如开发、测试、生产环境）和部署频率高（比如每周一次）时，如何提高企业应用部署的效率和质量，对企业信息化能否快速满足业务需求的快速变化至关重要。

然而，当前很多同行在企业应用部署方法体系上缺少完善的管理信息模型，在工具上采用手工编写和执行部署脚本。这种传统的应用部署方式会导致部署效率不高，对部

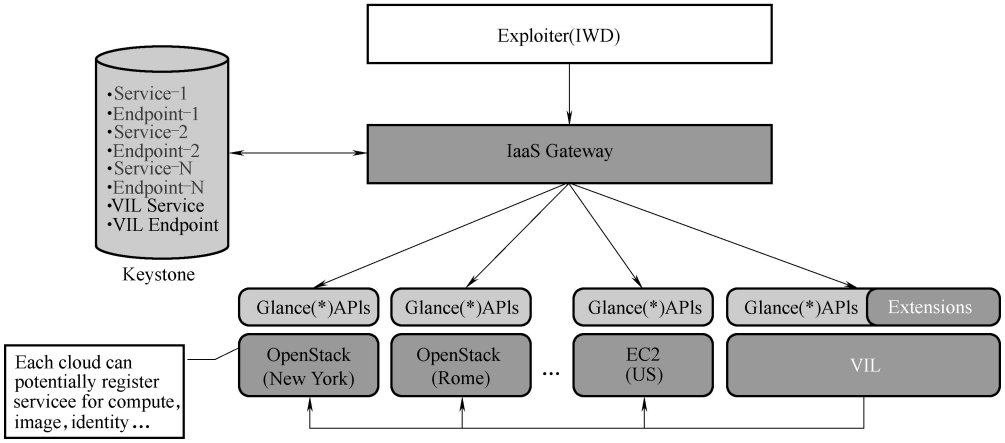


图 3-18　SCO 镜像管理模块接口示意图

注：Exploiter（IWD）：开发者（工作负载部署软件 IBM Workload Deployer）；IaaS Gateway：IaaS 网关；Keystone：基石；Service：服务；Endpoint：终点；VIL Service：VIL 服务；VIL Endpoint：VIL 终点；Extensions：扩展；OpenStack：开源架构；EC2：弹性计算云；New York：纽约；Rome：罗马；US：美国；Each cloud can potentially register services for compute，iamge，identity：每一个云都能潜在的为计算、镜像、身份注册服务。

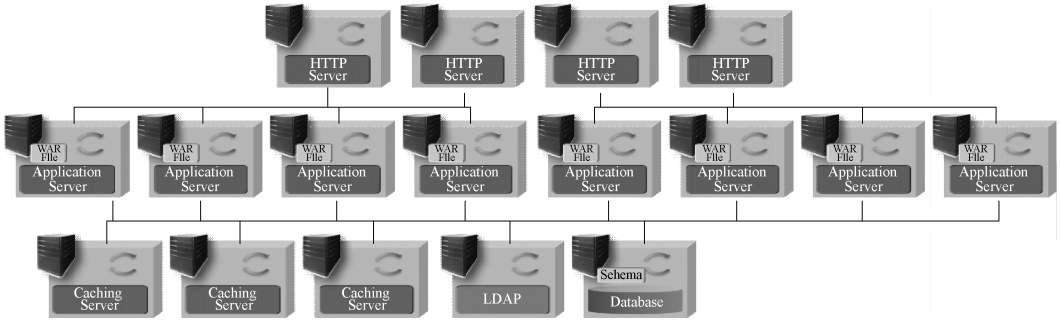


图 3-19　SCO 应用系统平台示意图

注：HTTP Server：HTTP 服务器；WAR File：一个网络应用程序资源或一个实例；Application Server：应用服务器；Caching Server：缓存服务器；LDAP：轻量级目录访问协议；Database：数据库；Schema：架构。

署过程缺乏管控，对部署结果缺乏审计。这将导致版本发布和部署上线成为整个软件交付流程的瓶颈之一。

而建立在云平台上的基础架构使 IT 在基础资源和环境管理层面都具备了足够的灵活性和弹性，但如果无法与业务的持续交付结合起来，那么也就无法真正全面发挥云平台的价值。为了适应云环境下应用开发和交付的特征和需求，并充分发挥云对业务交付的支撑价值，必须将 DevOps 技术运用到云应用管理层面。DevOps 的典型特征是以自动化为基础，打破部门之间的壁垒，进行软件的持续交付和持续创新，并快速获取客户反馈，形成以软件交付为核心的企业竞争力，这样才能真正发挥企业云平台的整体实力。

IBM UrbanCode Deploy（UCD）是 IBM 应用的自动化部署工具。该工具基于一个完善的应用部署自动化管理信息模型，提供可视化的部署逻辑设计手段，并通过远程代理技术，实现对复杂应用在不同环境下的自动化部署。IBM UrbanCode Deploy with Pattern（UCDP）是在 UCD 基础上针对云平台部署的增强功能，它能够帮助企业将云平台上基

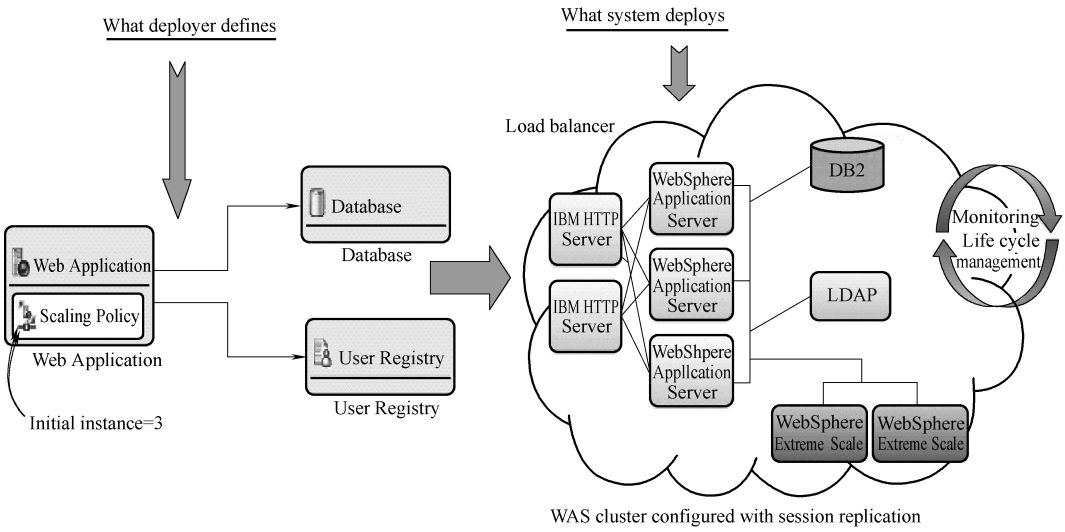


图 3-20　SCO 应用系统拓扑结构示意图

注：What deployer defines：部署者定义什么；What system deploys：系统部署什么；Web Application：Web 应用；Scaling Policy：衡量策略；Database：数据库；User Registry：用户注册；Initial instance = 3：初始化实例等于 3；Load balancer：负载均衡器；IBM HTTP Server：IBM HTTP 服务器；WebSphere Application Server：应用服务器 WebSphere；DB2：二代数据库；LDAP：轻量级目录访问协议；Monitoring Life cycle management：监控生命周期管理；WebSphere Extreme Scale：WebSphere 极限规模；WAS cluster configured with session replication：WAS 集群配置会话复制。

础设施的部署和应用的部署集成起来，实现一键式全栈部署。同时，UCD 还可通过插件方式实现向 PaaS 平台部署服务和应用、向移动终端部署应用。总之，通过 UCD 和 UCDP 可以实现混合云和终端的统一部署管理，帮助企业实现从传统 IT 向基于云和终端转型，也为长期拥有多种类型环境的企业提供统一的应用发布和部署管理平台。UCD 功能示意图如图 3-21 所示。

UCD 不仅是一个实现应用部署自动化的工具，还是一个服务于系统工程师（负责部署环境的系统管理）、发布工程师（负责应用的部署逻辑设计）、部署工程师（负责执行应用部署）和质量工程师（负责应用部署前和部署后的质量审计）等多个角色的管理平台。该平台所提供的功能有：

1）环境管理：实现对被部署机器以及所提供资源（比如，数据库、中间件等）的管理。

2）组件管理：实现对应用部署组件（比如，企业应用的安装包、数据更新脚本等）的维护，以及组件部署逻辑（比如，如何在 Tomcat 上安装一个发布包）的可视化设计。

3）应用管理：实现企业应用的维护、关联组件以及如何编排多个组件的部署逻辑来完成整个应用的部署逻辑。

4）执行管理：实现应用部署环境的定义、资源的映射以及应用部署的执行。

5）仪表盘：实现对应用部署的统计分析，包括成功、失败统计分析以及部署时长统计分析等。

6）部署文件管理：实现对部署文件的集中存储，版本比对等功能。

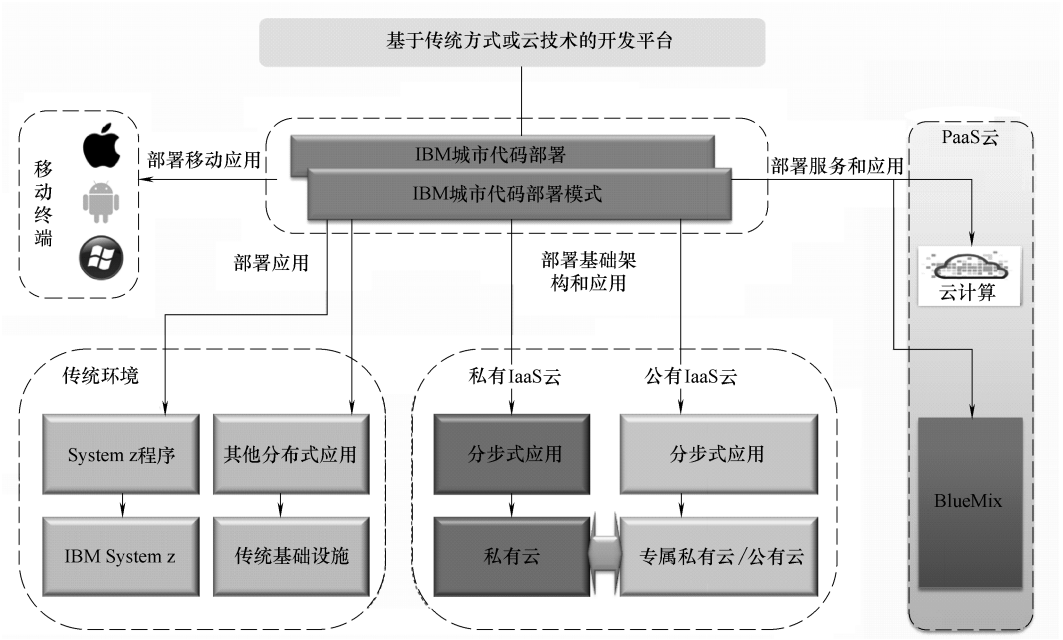


图 3-21　UCD 功能示意图

注：BlueMix：一种开放式标准的云平台，用于构建、运行和管理应用程序与服务；IBM System z：企业级服务器。

7）插件管理：实现对可重用部署步骤的管理，并可自定义可重用的部署步骤。

8）系统管理：实现用户、组、角色、团队的管理，定义完善的权限策略。

在 UCD 的基础上，为了完全适应云计算的环境，IBM 又推出了 UCDP 解决方案，该方案是专门针对云环境而设计的。它的底层支持 OpenStack 等开放标准，是一套完整的云环境下应用的管理与部署方案。

它提供了用于绘制环境蓝图的图形化编辑器和进行代码编写的文本编辑器，用户可以通过“拖拽”→“修改属性”→“编辑代码”的方式在不同的云环境中设计、部署或更新一个“全栈式”的环境，还可以与 UCD 集成实现应用程序到云环境的端到端的持续交付管道。我们可以使用 UCDP 管理程序负载，从而获得从计算资源级到程序级的更高级别的控制；还可以利用改进的协作生命周期管理和可移植云来快速地设计、部署和重用环境。UCDP 通过自动提供云环境加速了应用程序的开发部署。

3.2　VMware 私有云解决方案

3.2.1　VMware 虚拟化技术

3.2.1.1　基于 vSphere 的虚拟数据中心基础架构

基于 VMware vSphere 的虚拟数据中心由基本物理构建块（例如 X86 虚拟化服务器、

存储器网络和阵列、IP 网络、管理服务器和桌面客户端）组成。vSphere 数据中心的物理拓扑如图 3-22 所示。

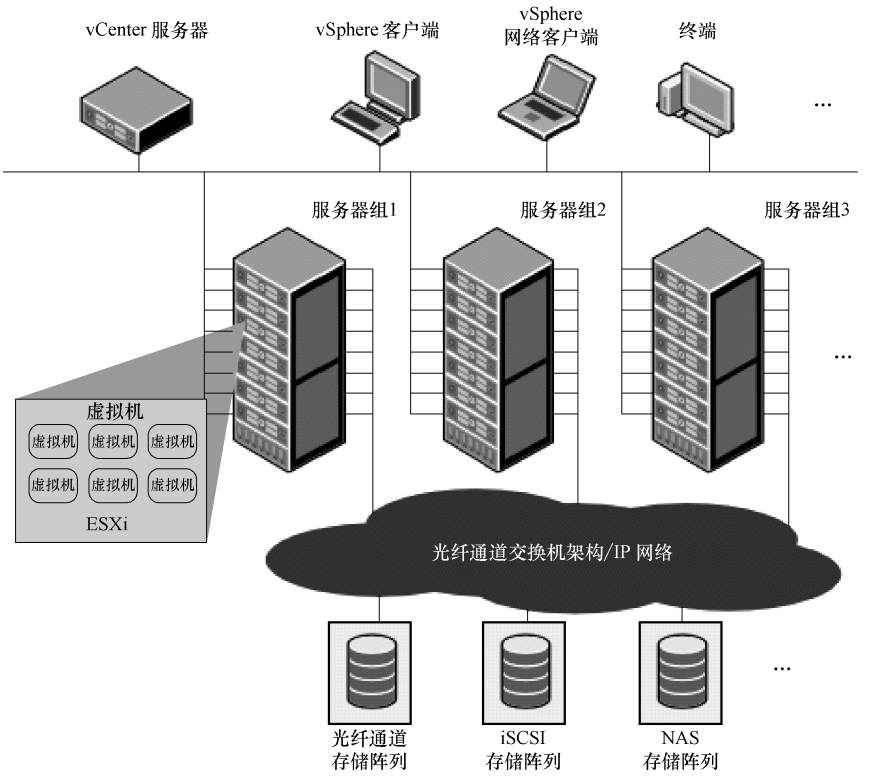


图 3-22　vSphere 数据中心的物理拓扑

vSphere 数据中心拓扑包括下列组件：

1. 计算服务器

在裸机上运行 ESXi 的业界标准 X86 服务器。ESXi 软件为虚拟机提供资源，并运行虚拟机。每台计算服务器在虚拟环境中均称为独立主机。我们可以将许多配置相似的 X86 服务器组合在一起，并与相同的网络和存储子系统进行连接，以便提供虚拟环境中的资源集合（又称为群集）。

2. 存储网络和阵列光纤通道

SAN 阵列、iSCSI SAN 阵列和 NAS 阵列是广泛应用的存储技术，VMware vSphere 支持这些技术以满足不同数据中心的存储需求。存储阵列通过存储区域网络连接到服务器组并在服务器组之间实现共享。此安排可实现存储资源的聚合，并在将这些资源置备给虚拟机时使资源存储更具灵活性。

3. IP 网络

每台计算服务器都可以有多个物理网络适配器，为整个 VMware vSphere 数据中心提供高带宽和可靠的网络连接。

4. vCenter Server

vCenter Server 为数据中心提供一个单一控制点。它提供基本的数据中心服务，如访问控制、性能监控和配置功能。它将各台计算服务器中的资源统一在一起，使这些资源在整个数据中心中的虚拟机之间实现共享。其原理是：根据系统管理员设置的策略，管理虚拟机到计算服务器的分配，以及资源到给定计算服务器内虚拟机的分配。

在 vCenter Server 无法访问（例如，网络断开）的情况下（这种情况极少出现），计算服务器仍能继续工作。服务器可单独管理，并根据上次设置的资源分配继续运行分配给它们的虚拟机。在 vCenter Server 的连接恢复后，它就能重新管理整个数据中心。

5. 管理客户端

VMware vSphere 为数据中心管理和虚拟机访问提供多种界面。这些界面包括 VMware vSphere 客户端（vSphere 客户端）、vSphere Web 客户端（用于通过 Web 浏览器访问）或 vSphere Command-Line Interface（vSphere CLI）。

3.2.1.2 计算资源虚拟化技术

1. CPU 虚拟化

VMware 通过 CPU 虚拟化技术解决了如何在一个操作系统实例中运行多个应用的难题。实现这一任务的困难之处在于每一个应用都与操作系统之间有着密切的依赖关系。一个应用通常只能运行于特定版本的操作系统和中间件之上，这就是 Windows 用户常常提到的“DLL 地狱”。因此，大多数用户只能在一个 Windows 操作系统实例上运行一种应用，操作系统实例独占一台物理服务器。这种状况会导致物理服务器的 CPU 资源被极大地浪费。能够使多个操作系统实例同时运行在一台物理服务器之上，是 VMware 所提供的 CPU 虚拟化技术的价值所在。通过整合服务器充分利用 CPU 资源，可以给用户带来极大的收益。

服务器整合的益处能够得以实现的前提是工作负载并不需要知晓它们正在共享 CPU，虚拟化层必须具备这种能力。这是 CPU 虚拟化与其他虚拟化在形式上所不同之处。

具体实现方式是为每个虚拟机提供一个或者多个虚拟 CPU（vCPU）。多个 vCPU 分时复用物理 CPU。虚拟机监控器（VMM）必须为多个 vCPU 合理分配时间片并维护所有 vCPU 的状态，当一个虚拟机 vCPU 的时间片用完需要切换时，要保存当前 vCPU 的状态，将被调度的 vCPU 的状态载入物理 CPU。

VMkernel 在调度 vCPU 的时候采用“插槽-核心-线程”的拓扑逻辑。“插槽”是指处理器单个封装件，该封装件可以具有一个或多个处理器内核且每个内核具有一个或多个逻辑处理器。

当 vCPU 需要运行时，VMkernel 会将一个 vCPU 映射到处理器调度一个执行线程的能力，它对应于一个 CPU 核心或一个超线程（如果 CPU 支持超线程）。超线程或多核 CPU 提供了两个或多个调度 vCPU 运行的硬件基础。

可以将虚拟机配置为最多具有 64 个 vCPU。主机上许可的 CPU 数量、客户机操作系统支持的 CPU 数量和虚拟机硬件版本决定着可以添加的 vCPU 数量。

与 vCPU 和管理 vCPU 相关的技术和概念有：

（1）处理器管理　从客户操作系统 vCPU 发往 ESXi VMkernel 的指令被 VMM 拦截。在固定时间间隔内，VMKernel 动态地在服务器和不同处理器（或多核处理器的内核）中分配 VM 工作负载。因此，VM 指令根据每个处理器的工作负载从一个处理器（或内核）转移到另一个处理器。

（2）多内核和虚拟化　多核处理器为执行虚拟机多任务的主机提供了很多优势。Intel 和 AMD 均已开发了将两个或两个以上处理器内核组合到单个集成电路（通常称为封装件或插槽）的处理器。

例如，双核处理器通过允许同时执行两个虚拟 CPU，可以提供几乎是单核处理器两倍的性能。同一处理器中的内核通常配备由所有内核使用的最低级别的共享缓存，这可能会减少访问较慢主内存的必要性。如果多个虚拟机运行在同一个逻辑处理器上，并且争用相同内存总线资源导致内存工作负载过大，那么连接到主内存的共享内存总线可能会降低逻辑处理器的性能。

研究显示，使用多内核可以导致可观的耗电下降，并提供良好的性能。虚拟化是最好地利用多内核提供的高性能技术之一，因为 ESXi 能够像管理物理处理器一样管理内核。

（3）对称多处理器　vSphere 虚拟对称多处理技术（Virtual SMP）可以使单个虚拟机同时使用多个物理处理器，并能够在处理器之间均衡负载。必须具有虚拟 SMP，才能打开多个处理器虚拟机电源。一些关键业务，比如数据库类应用（Microsoft SQL、Oracle、IBM DB2、SAP）、商业和科研应用，在开发时就考虑了并行执行任务的需求，具有多个物理处理器的服务器就能利用 SMP 并从中获益。

（4）超线程　超线程是在一个物理处理器或者内核上创建两个逻辑内核实例，从而在核心中并行执行任务，以提高效率。在 vSphere 虚拟机的处理器分配过程中，一个超线程可以对应一个 vCPU。

（5）CPU 虚拟化类型　从架构上看，传统的 X86 平台并不是为支持多操作系统并行而设计的。因此 CPU 厂商如 AMD 和 Intel 都需要重新设计 CPU，增加虚拟化特性，以解决上述问题。当前 X86 虚拟化平台的主要厂商如 VMware 等，也已经开始充分利用芯片厂商在处理器架构中构建的硬件辅助功能，以以提高系统运行效率，降低 Hypervisor 带来的系统开销。

在传统的 X86 运行环境下，操作系统使用保护环提供保护级别以完成不同任务代码的执行。这些保护环以等级排列，从最有特权的（Ring 0）到最少特权的（Ring 3）。在未虚拟化的服务器上，操作系统拥有在 Ring 0 运行设备 I/O 等核心指令的权利，同时应用运行在 Ring 3 上。在虚拟化系统上，Hypervisor 和虚拟机监控器（VMM）需要运行在 Ring 0，因此虚拟机子操作系统必须运行在 Ring 1 上。但由于多数操作系统的设备 I/O 等核心指令必须运行 Ring 0，所以 VMM 工具通过捕获特许指令和模拟 Ring 0 到子虚拟机，使子操作系统以为它运行在 Ring 0 上。这样就产生了延时和开销。

因此，Intel 和 AMD 等芯片厂商在 CPU 内引入了一个新的、具有超级特权和受保护

的 Ring-1 位置来运行虚拟机监控器（VMM），这样 VMM 就可以运行在新的叫作 Ring-1 的环里，它而且 GuestOS 天生就运行在 Ring 0 里。这种 CPU 架构上的虚拟化支持扩展提升了性能。VMM 不再让 GuestOS 以为自己运行在 Ring 0 里，因为 GuestOS 已经能在此操作，并且不会与 VMM 冲突——VMM 已经移动到新的 Ring 1 级别。选择支持这些虚拟化优化扩展的 CPU，可以更好地降低系统开销、提升虚拟化效率。

基于以上设计，Intel 和 AMD 分别推出了 VT-x 和 AMD-v 两种主要的 X86 处理器架构的虚拟化硬件辅助功能。

（6）CPU 负载均衡（见图 3-23） CPU 调度器可以让多个虚拟机复用逻辑处理器（逻辑处理器的单位是一个 CPU 核心或一个超线程），提供给虚拟机类似于传统对称多处理器（SMP）的执行能力，并使它们之间相互独立。例如，配有 2 个 vCPU 的虚拟机可以使虚拟处理器运行在属于相同内核的逻辑处理器上，或运行在不同物理内核的逻辑处理器上。VMkernel 通过智能地管理处理器时间来保障工作负载在处理器内核间进行迁移。每隔 2 ~ 40ms，VMkernel 把 vCPU 从一个逻辑处理器迁移到另一个逻辑处理器来保障负载均衡。如果存在超线程，VMkernel 尽量把相同虚拟机的 vCPU 负载分散到不同内核的线程上来实现性能的优化。

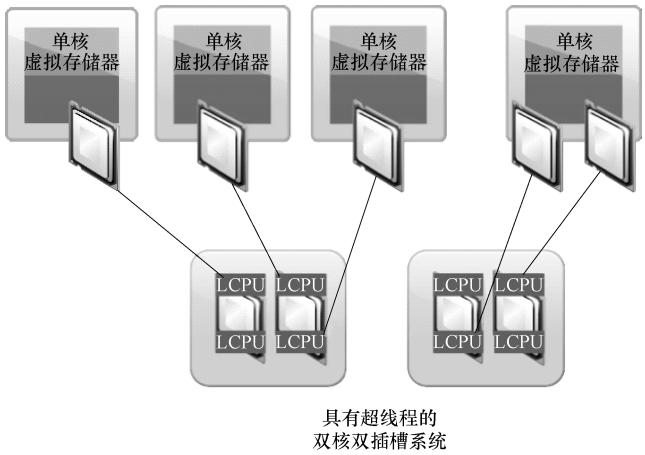


图 3-23 CPU 负载均衡

ESXi CPU 调度程序可以解释处理器拓扑（包括插槽、内核和逻辑处理器之间的关系）。调度程序使用拓扑信息优化虚拟 CPU 在不同插槽上的放置位置，以使总体缓存利用率实现最大化，并通过最小化虚拟 CPU 迁移来改善缓存关联性。

在未过载的系统中，ESXi CPU 调度程序在默认情况下将负载分配到所有插槽中。这样便可通过最大化供正在运行的虚拟 CPU 使用的缓存总量来改善性能。因此，单个 SMP 虚拟机的虚拟 CPU 在多个插槽之间分配（除非每个插槽本身还是 NUMA 节点，在这种情况下，NUMA 调度程序会限制虚拟机的所有虚拟 CPU 都驻留在同一插槽上）。

综上所述，对 CPU 的调度管理功能有如下要求：

1）支持基于 Intel/AMD X86 指令集架构的处理器，支持最新的处理器硬件辅助虚拟化功能。

2）支持处理器多核技术。

3）支持虚拟多路运算，每个虚拟机可以支持多达64个虚拟CPU（vSMP），以满足高负载应用环境的要求。

4）可以灵活分配调度物理服务器上的CPU资源，如可按主频赫兹分配给虚拟机计算时间片。

5）对CPU的调度应能实现虚拟机按需使用，随用随取，不用即释放，使得计算资源能被充分利用。

6）在虚拟机操作系统提供支持的前提下，应能支持虚拟机的CPU热添加技术。

2. 内存虚拟化

当运行一个虚拟机时，vSphere的VMKernel为虚拟机生成一段可编址的连续内存，与普通操作系统提供给上层应用使用的内存具有相同的属性与特征。引入内存虚拟化后，同样的内存地址空间允许VMkernel同时运行多个虚拟机，并保证它们之间在使用内存时具有独立性。

VMware vSphere的三层内存映射结构如图3-24所示。

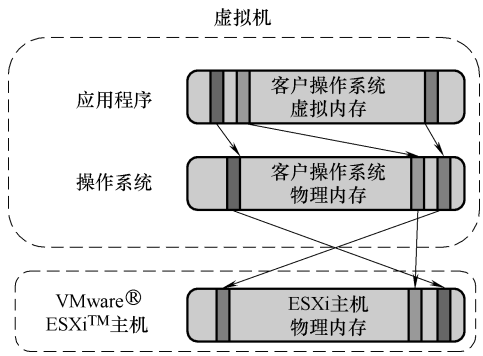


图3-24　VMware vSphere三层的内存映射结构

ESXi裸机体系结构的强大功能主要体现在内存优化方面，这些功能可以提高内存的使用效率。ESXi主机的内存管理支持安全的内存过量分配（见图3-25）。分配给每个虚拟机的内存总和可超过主机上安装的物理内存总和。ESXi主机采用了几种有效方法来支持安全的内存过量分配。例如，过量分配率为2:1时，通常只会对性能产生非常小的影响。

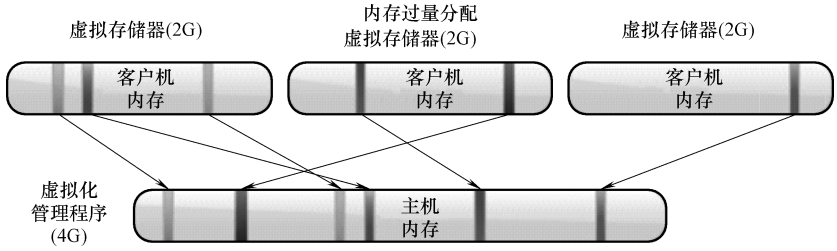


图3-25　内存的过量分配

内存通常是最有限的资源，vSphere VMkernel管理服务器的RAM可进行多种资源的节约操作。通过VMware设计的若干功能，vSphere可支持实现RAM的高效使用和更高的整合率，包括透明页共享、客户机内存回收和内存压缩。实现过量分配的内存管理机制如下：

（1）透明页共享TPS　透明页共享（Transparent Page Sharing，TPS）是VMware独有的一种内存优化方法。VMkernel可检查虚拟机存储的每个内存页面，以便识别相同的页面，并仅存储一个页面副本，如图3-26所示。

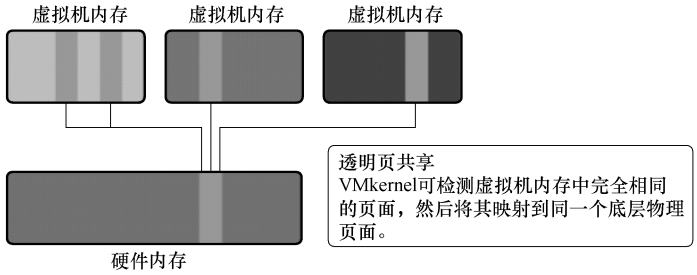


图 3-26　透明页共享

（2）气球内存回收　ESXi 主机使用一种随 VMware Tools 提供的内存释放驱动程序，该程序安装在每个虚拟机中。如果内存不足，则 VMkernel 将选择一个虚拟机并扩充其内存，也就是说，它会通知该虚拟机中的释放驱动程序从客户操作系统要求更多的内存。客户操作系统通过生成内存满足这一需求，然后 VMkernel 会将释放出的页面分配给其他虚拟机，如图 3-27 所示。

（3）内存压缩　当内存过量分配时，内存压缩可以帮助提高虚拟机性能，如图 3-28 所示。默认情况下已启用该功能。因此，当主机内存发生过量分配时，ESXi 会尝试将该页面交换到磁盘前压缩虚拟页面并将其存储在内存中。

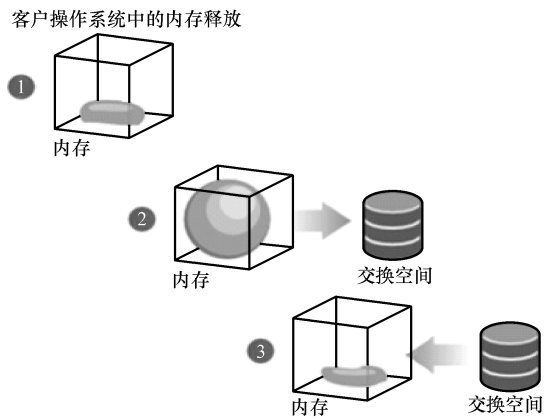


图 3-27　气球内存回收

（4）主机级 SSD 交换文件　每个虚拟机都包含一个 VMkernel 交换文件。如果多个虚拟机需要完全使用分配给它们的内存，则 ESXi 主机将根据为每个虚拟机指定的内存资源设置，按比例将其内存区域交换到本地或网络固态驱动器（SSD）设备中，如图 3-29 所示。

将虚拟机内存分页移到磁盘中，如非必要，系统不会使用 VMkernel 交换空间，因为这种方式的性能很差。

综上所述，对内存的调度管理功能应满足如下需求：

1）单个虚拟机能够支持扩展到最大 1TB 的内存。

2）可以灵活分配调度物理服务器上的内存资源，如可按 MB 大小分配给虚拟机内存资源。

3）对内存的调度应能实现虚拟机按需使用，随用随取，不用即释放，使计算资源能够被充分利用。

4）具有合理的内存调度机制，能够实现内存的过量使用，支持不同虚拟机中内存相同数据部分的页面共享，保障内存资源的充分利用。

5）在虚拟机操作系统提供支持的前提下，应能支持虚拟机的内存热添加技术。

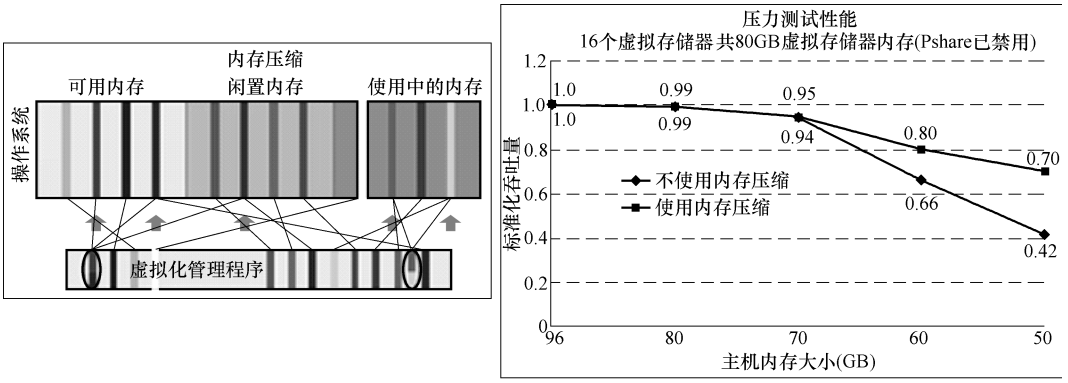


图 3-28　内存压缩

6）支持内存压缩技术，减少虚拟内存在虚拟机高压力内存状态下交换到磁盘上的概率，从而提升性能。

3.2.1.3　虚拟机性能

虚拟机有一个对应的硬件版本的概念，该硬件版本指示虚拟机支持的虚拟硬件特性，如 BIOS 或 EFI、虚拟插槽数、最多 CPU 数、最大内存配置和其他硬件特征。创建虚拟机所用的 vSphere 主机的版本决定了虚拟机的硬件版本。最新版本支持的最大虚拟机能力为：64 个虚拟 CPU、1TB 内存、1000000 IOPS（Inpwt/Output Operations Per Second，每秒进行读写操作的次数）的磁盘读取速度以及 36Gbit/s 以上的网络吞吐量。远远大于典型应用的需求，足以满足关键应用和甚至大数据的需求。这足以满足每天处理 20 亿次交易的大规模数据库的需求，只需一台虚拟机，即可存储 NASDAQ 每天 20 亿笔交易的全部信息。

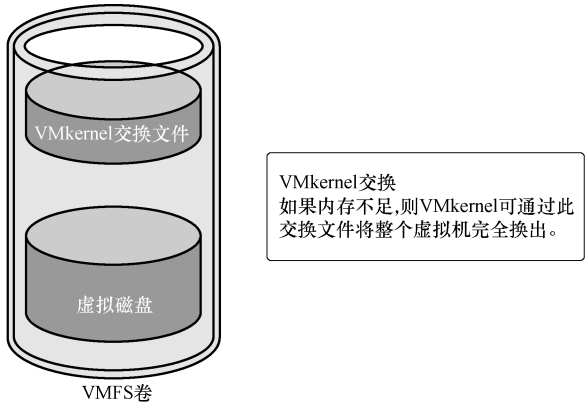


图 3-29　交换文件

3.2.2　VMware 云管理方案

3.2.2.1　运维管理

VMware 的 vCenter Operations Management Suite（vCenter 运营管理组件）产品可使

用户更全面地了解基础设施所有层的情况。它可收集和分析性能数据、关联异常现象，并可识别构成性能问题的根本原因。它提供的容量管理可优化资源使用率，基于策略的配置管理则可确保合规性并消除数量剧增和配置偏差问题。应用发现、依赖关系映射和成本计量功能为基础设施和运维团队带来了更高级别的应用感知和财务责任。

vCenter Operations Management Suite 使 IT 部门可以获得更好的可见性和可操作的智能信息，从而主动确保动态虚拟环境和 vCloud 中的服务级别、资源利用率优化和配置合规性，它具有如下三个基本特征：

（1）自动化　根据 Forrester 的调查，凭借获得专利的自学式分析方法，该产品可实现比传统管理工具高得多的自动化程度，使工作效率提高近 70%，资源消耗减少 30%，还可以带来更多的业务优势。

（2）集成式　本产品采用集成式方法实现性能、容量和配置管理，以集成式套件的方式提供，它聚合了各种管理规程，并将不同基础设施和运维部门的团队统一成一体。

（3）全面性　vCenter Operations Management Suite 以开放且可扩展的操作平台为基础而构建，可提供一整套全面的管理功能，包括性能、容量、变更、配置和合规性管理、应用发现和监控，以及成本计量。

借助 vCenter Operations Management Suite，基础设施和运维团队可获得全面可见性、智能自动化和主动式管理，从而能以尽可能高效率的方式确保服务质量。

vCenter Operations Management Suite 的核心功能如图 3-30 所示。

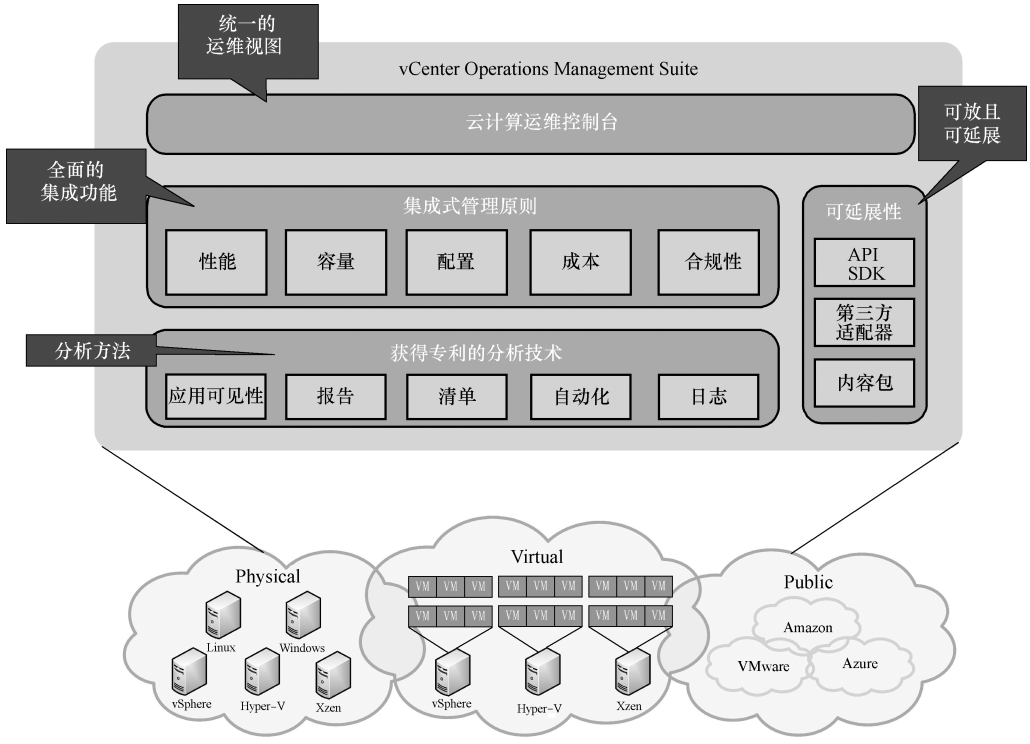


图 3-30　vCenter Operations Management Suite 的核心功能

注：vCenter Operations Management Suite：vCenter 运营管理组件；API：应用编程接口（Application Programming Interface）；SDK：软件开发工具包（Software Development Kit）；Physical：物理设备；Virtual：虚拟设备；Public：公有云。

1）提供了环境状况的统一视图，该视图支持虚拟环境，这是组织的运维团队关注的主要功能。想要获取一个有关虚拟云计算环境状态的统一视图，是很难做到的，而VMware 提供了一个解决方案。

2）提供了一套全面的集成功能。在虚拟环境中，遗留给 IT 运维团队的问题是需要不同的运维团队管理网络资源、存储资源和计算资源。但在虚拟化环境中，所有这些资源一一具备，虚拟团队能够一起管理所有这些资源，vCenter Operations Management Suite 为此提供了全面的集成解决方案。

另外，就管理性能而言，该方案所采用的方法与该领域其他传统供应商所采用的方法有很大不同，即它更加注重分析。VMware 了解什么对环境而言是正常的并将该信息用于智能警报，而不是依赖会导致产生大量误报的某些传统阈值的方法。此核心功能是该产品组合的核心组成部分，该产品组合能够提供大量的优势功能，与虚拟化和 vCloud 管理领域的传统管理供应商和新供应商的功能有很大的差异。

由于客户需要管理虚拟环境，而且管理的虚拟环境是基于物理环境构建的，并且他们需要通过云来进行管理。因此，该方案提供了一套异构功能，不仅能够解决物理和虚拟环境的问题，而且能够满足同时跨私有云和公有云或混合云管理的需求，这套功能对客户非常重要。

3）提供了可延展性和开放式框架，以用于将该解决方案以及将其他解决方案集成到软件定义的数据中心中。

3.2.2.2 服务调配

这里将介绍 vCloud Suite 的服务调配解决方案 vCloud Automation Center 与多租户自助服务门户 vCloud Director。

对于云计算平台而言，服务调配存在大量的挑战。由于自定义服务过多，以及对于谁能访问什么服务、在何处调配服务等方面缺乏控制，服务调配非常耗时，需要大量的人工操作，而且成本高昂。

VMware 通过采用针对基础架构服务，包括计算和桌面两方面，以及应用服务的调配解决方案解决了这些问题。借助 vCloud Automation Center（vCAC），客户可以对通过自助式门户和目录向终端用户提供的预定义基础架构和桌面服务进行自动化调配，从而快速实现价值。这些功能可以促使企业加大创新力度，并且能够提高企业的敏捷性以及降低 IT 成本，同时还能确保其符合行业和公司的法规和策略。此外，vCAC 还能够简化和自动化将任何自定义或打包应用调配到任何已批准的云的过程，从而缩短应用的上市时间。可重复使用的标准化应用组件能够降低成本，并且有助于确保合规性，此外还能够调配到多个云中，因而可以提高业务的敏捷性。

企业异构硬件环境可以通过 vCAC 来进行集中化和标准化的调配和管理。对于管理虚拟机，可以对 VMware vCloud Director 环境、vSphere 环境虚拟机进行管理，并且可以管理 Microsoft Hyper-V、Citrix Xenserver、Redhat KVM 等虚拟化环境。对于物理机，支持管理主流 X86 服务器厂商的服务器，包括 HP（通过 iLO）、DELL（通过 iDRAC）、CISCO（通过 UCSM）和 IBM（需要作一定的定制开发），此外，还可以管理外部公有

云如 VMware vCloud Director 和 Amazon AWS 虚拟环境的云资源。

vCAC 自身不具备虚拟化资源的能力，而是与虚拟化平台协作，提供调配和管理虚拟化平台所创建的虚拟机和产生的虚拟计算资源的能力，如图 3-31 所示。要完成上述调配和管理的功能，vCAC 需要使用包含平台结构在内的代理。类似地，vCAC 并不是直接管理云虚拟机，而是直接与云服务交互，来调配和管理云平台创建的虚拟机。对于管理物理机器，vCAC 直接与每个系统的管理接口通信来执行诸如操作系统安装、重启、重调配等操作。

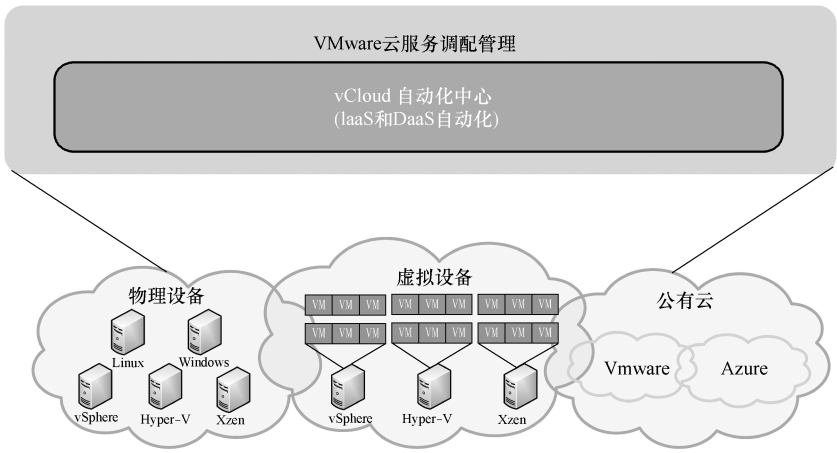


图 3-31　调配和管理的资源

VMware vCloud Automation Center（vCAC）由三大模块组成，即虚拟机资源管理器（VRM）、外部云管理器（ECM）和物理资源管理器（PRM）。这些模块共同为企业提供广泛的管理和调配虚拟资源、云资源和物理资源的能力，同时全面管理机器的生命周期，从用户请求和管理员审批到期满退役和资源回收，有效地提高了资源成本控制和管理。内置的定制化和可扩展功能也使 vCAC 成为一种可以根据需求制定机器配置和与其他关键的企业系统整合来完成机器调配和管理的高度灵活的方式。

VRM 负责在现有的虚拟环境（VMware vSphere，Hyper-V）中配置新的机器。ECM 负责在云基础架构（VMware vCloud Director，Amazon）上配置虚拟机。vCAC 利用两者的底层环境关系来实现虚拟化和云计算的优势。

PRM 负责在物理服务器上部署机器。VRM 和 ECM 的功能是调配虚拟机资源，而 PRM 只能在一个物理机上配置。PRM 借助硬件管理界面来部署新的机器。

不论新机器在 vSphere、Hyper-V、Citrix XenServer、Red Hat KVM、Amazon AWS、Dell 和 HP，还是 Cisco 等平台上部署，vCAC 都能够提供给用户相同的体验。

1）实现业务的灵活性：更快速地响应业务需求，降低了服务交付时间，从几天缩减到几小时甚至几分钟。

2）提高 IT 效率：基于策略的管理消除了过度配置和无效资源的浪费。

3）提供选择：广泛的厂商支持，强大的合作伙伴配合，以及可扩展性设计，提供了一个灵活的云基础架构以满足当前和未来的需求。

4）快速部署：在数天内把现有的基础设施转变为可扩展的云服务，这样就可以快速和经济地响应商业需求。

VMware 服务调配解决方案通过一个自动交付 IT 服务的灵活解决方案，可实现业务所需的敏捷性和 IT 所需的控制力。

1. 云端自助服务和业务管理

借助 VMware 服务调配解决方案中的业务感知管理和控制功能，用户无需更改组织流程或策略即可将自己独有的业务方式应用到云中。借助 VMware 服务调配解决方案，用户还可以根据既定的运营策略请求和管理计算资源，同时将交付 IT 服务所需的时间从数天或数周缩短到几分钟。

1）跨云平台发挥服务调节器的作用，根据业务和 IT 策略调配工作负载。

2）拥有用户感知能力的自助门户可以为用户提供相应的 IT 服务目录。

3）“资源预留”策略可以将资源分配给特定的组使用，并可确保资源不会无意中被其他组使用。

4）“服务级别”策略负责定义特定服务在初始调配期间或进行任何配置变更时能获得的资源数量和类型。

5）蓝本包含各种自动化策略，这些策略定义了构建和重新配置计算资源的流程。

2. 基础架构调配和生命周期管理

VMware 服务调配解决方案是一款已经企业验证通过的解决方案，专为自动交付私有云和混合云服务而构建；它让企业能够快速证明云部署具有更高的业务价值。

1）资源治理策略可避免超额配置，确保每个用户都能获得处于适当服务水平的适量应用，以满足他们履行自身工作职责的需要。

2）资源回收策略和自动化的回收工作流，有助于发现并回收非活动资源及已弃用的资源。

3）借助“回收节约”报告，企业可以了解具体节约了多少成本。

3. 调配和管理应用服务或部署用户 PaaS

VMware 服务调配解决方案可缩短“平台即服务”（PaaS）和应用部署时间。一项主要功能是应用蓝本，它描述了独立于底层基础架构的应用部署拓扑。这可以提供对应用建模一次而将其多次部署到不同环境中的能力。应用蓝本通过预构建、可重用的组件组合而成。它包括以下主要优势：

1）通过简化部署过程和使用可重用组件及蓝本来消除重复工作，可加快 PaaS 和应用部署的速度。这样就可以更快捷地向业务用户交付应用。

2）利用 Cloud Applications Marketplace 提供的即时可用的组件、用于创建任何自定义组件的可延展性框架以及随时可以运行的合作伙伴解决方案，可对任何应用（自定义或打包）灵活建模以加以部署。

3）可通过横向扩展已部署应用的节点或实施应用或代码更改来更新应用部署。

4. 专门构建的灵活性和可延展性

VMware 服务调配解决方案专为与现有基础架构配合使用而构建。它可以支持众多

业务部门的不同需求，并且可与现有 IT 系统和最佳实践集成。利用以下功能，VMware 服务调配解决方案可与现有 IT 基础架构和流程兼容，或者适应现有的 IT 基础架构和流程：

1）利用工作流设计器可以在机器生命周期中的各个状态转变点轻松将活动插入工作流存根。借助这些存根，可以简化在 VMware 服务调配解决方案标准生命周期自动化流程中添加自定义流程的流程方法。

2）与 vCenter Orchestrator 的集成可扩展预构建自动化任务的活动库，这些活动可轻松纳入 vCAC 的现有流程。

3）VMware Solution Exchange 提供了 VMware 和合作伙伴提供的 vCO（vCenter Orchestrator）工作流和插件库，这些工作流和插件可以加快自定义 vCAC 标准功能的速度。

5. 多供应商统一云计算管理

VMware 服务调配解决方案可跨越广泛的多供应商部署技术和管理工具编排基础架构和应用服务的交付。

1）使用简单的拖放式界面来设计应用部署—设计和建模一次，即可在任意地点部署。

2）保护对现有工具和专业技能的投资，并赋予决定未来技术的自主权。

3）通过集成式应用商店，可以轻松获得数百种最佳实践应用组件和蓝本、开发环境以及可提高业务工作效率的应用。

3.2.3 VMware 的 SDDC

在服务器虚拟化之前，管理员部署一台服务器大概需要数周时间并花费上万美金。有了服务器虚拟化后，虽然部署一台虚拟机只需要几分钟且仅花费几百美金，但是，在该虚拟机“周边”部署所需的各种网络安全以及存储等设备仍需数天的时间，同时耗费数千美金。可见，虽然服务器虚拟化可以加快虚拟机的部署，但是与之相连接的网络安全以及存储等组件并没有跟上服务器虚拟化的步伐。

为了彻底解决现有数据中心存在的问题，全面提升数据中心的运营效率，VMware 提出了软件定义的数据中心解决方案（Software Defined Data Center，SDDC）。

软件定义的数据中心就是虚拟化和软件化数据中心的一切资源，专门的软件会代替专门硬件，它们会贯穿到数据中心的方方面面。虚拟化是从服务器虚拟化开始的，它所带来的好处前面已经提到了。而网络、存储是物理性很强的资源，虚拟机虽然带来了一些灵活性，但它没有办法在其他资源上体现。软件定义的数据中心就是把数据中心所有的传统、物理、硬件的资源进行虚拟化和软件化。

VMware 软件定义数据中心在各种底层硬件架构上面加载了一个虚拟的基础设施层，它提取所有硬件资源并将其汇集成资源池，支持安全高效自动地为应用按需分配资源。它可以将虚拟化技术的好处扩展到包括计算、存储、网络与安全以及可

用性在内的数据中心所有领域，从而实现支持灵活、弹性、高效和可靠 IT 服务的计算环境。

SDDC 提供了让数据中心适配新形势和新应用所需的一切，管理了从存储到网络与安全的方方面面。虚拟化一切，底层硬件的任何变化都与上层应用无关，有了这个基础，可伸缩性和性能问题便可迎刃而解。因此，包含大量遗留资产的数据中心可以提高效率、降低成本、实现动态化。

VMware 软件定义数据中心的整体架构如图 3-32 所示。

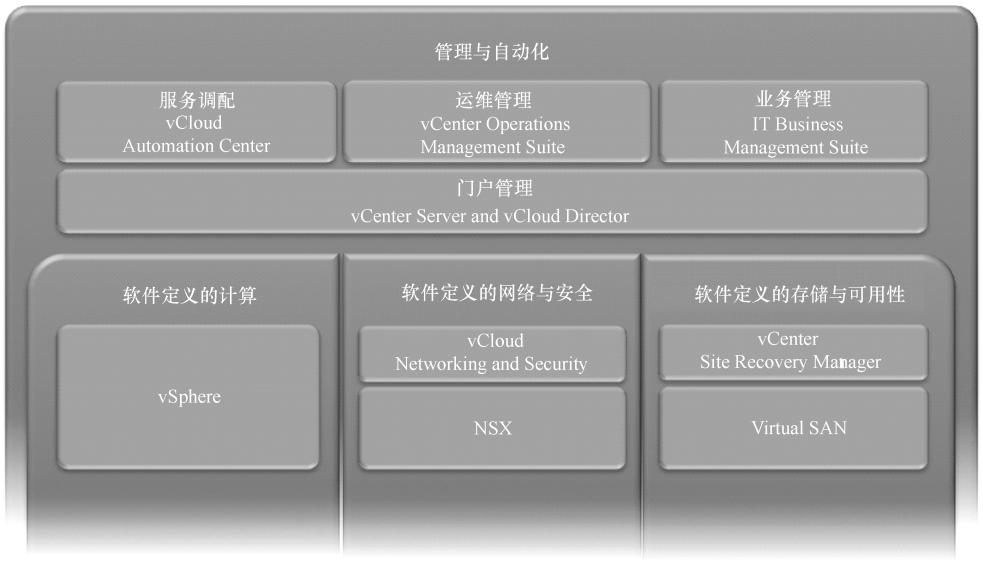


图 3-32　VMware 软件定义数据中心的整体架构

注：vCloud Automation Center：vCloud 自动化中心；vCenter Operations Management Suite：vCenter 运营管理组件；IT Business Management Suite：业务管理组件；vCenter Server and vCloud Director：vCenter 服务器和 vCloud 数据中心；vCloud Networking and Security：vCloud 网络和安全；vCenter Site Recovery Manager：vCenter 站点恢复管理器；Virtual SAN：虚拟 SAN。

3.2.3.1　软件定义的计算

软件定义的计算是针对 X86 系统的虚拟化技术，可以将 X86 系统转变成通用的共享硬件基础架构，由原先多台服务器完成的工作可以整合到少数服务器完成。摆脱了竖井式的结构，服务器物理硬件、操作系统和应用以松耦合的方式联接，虚拟机和上面的操作系统及应用完全独立于底层硬件。

除此之外，软件定义的计算通过把服务器计算资源抽象化、池化和自动化来实现资源的自由调配和充分利用，可以使资源充分利用，并按需调配。当数据中心的服务器需要升级或维护时，通过虚拟机迁移技术可以把服务器上的虚拟机在工作状态迁移到另一个主机上，且始终保持业务的连续性。服务器虚拟化大大增加了数据中心的灵活性和 IT 的敏捷性，减少了管理的复杂程度和 IT 响应时间。这部分功能已经在前文做了介绍，此处不再介绍。

3.2.3.2 软件定义的存储

软件定义的存储可以对存储资源进行抽象化处理，它把应用于服务器的先进技术运用于存储领域，可对异构存储资源进行抽象化处理，以支持存储的池化、复制和按需分发，并以应用为中心进行消费和管理，最终实现基于策略的自动化。该方案使存储层与虚拟化计算层非常相似：都具有聚合、灵活、高效和弹性扩展的特点。它们的优势是：全面降低了存储基础架构的成本和复杂性。

综合来看，软件定义的存储具备如下三个特征：

1）以应用为中心的策略，可实现存储使用自动化。软件定义的存储支持对异构存储池中的所有资源实施一致的策略，使存储的使用像为每个应用或虚拟机指定容量、性能和可用性要求那样简单。这种基于策略的自动化最大限度地利用了底层存储资源，同时将管理开销降至最低。

2）与硬件无关的虚拟化数据服务。数据服务（如快照、克隆和复制）作为虚拟数据服务在软件中交付，并按虚拟机进行调配和管理。独立于底层存储硬件使得这些服务的分配极其敏捷和灵活。

3）通过硬盘和固态磁盘虚拟化确保数据持久性。随着服务器功能的增多，软件定义的存储解决方案可让企业利用廉价的行业标准计算硬件来扩大其存储资源。利用固态磁盘和硬盘作为虚拟机的共享存储，可获得高性能、内置的恢复能力和动态可扩展性，并将存储总体拥有成本降低50%之多。

VMware在软件定义存储方面的计划主要侧重于一系列围绕本地存储、共享存储和存储/数据服务的计划。从本质上来说，VMware希望使vSphere成为一个存储服务平台。软件定义的存储旨在通过主机上与底层硬件集成并对其进行抽象化处理的软件层，实现存储服务和服务级别协议的自动化。

通过软件定义的存储，可以动态满足虚拟机存储要求，而无需重新调整LUN或卷。虚拟机工作负载可能会随着时间的推移有所变化，而底层存储可以随时适应工作负载的变化。

软件定义存储的一个关键因素是基于存储策略的管理（SPBM）。SPBM可以视为新一代VMware vSphere存储配置文件功能。SPBM是VMware实施软件定义存储的一个关键要素。使用SPBM和VMware vSphere API时，底层存储技术会呈现一个抽象化的存储空间池，为vSphere管理员提供用于虚拟机调配的各种功能。这些功能可能与性能、可用性或存储服务（例如VMware vSphere Thin Provisioning）有关。然后，vSphere管理员即可使用虚拟机上运行的应用所需的部分功能创建虚拟机存储策略。在部署时，vSphere管理员可以根据虚拟机的需要选择恰当的虚拟机存储策略。SPBM会将要求向下推送到存储层。这时将启用多种数据存储以供选择，这些数据存储可提供虚拟机存储策略中包括的各种功能。这意味着系统将始终根据虚拟机存储策略中设置的要求，在恰当的底层存储上创建虚拟机实例。如果虚拟机的工作负载随时间推移发生变化，只需将具有能够反映新工作负载的最新要求的策略应用于虚拟机即可。VMware解决方案成员如图3-33所示。

软件定义的存储通过纯软件实现了与存储相关的三个层面的功能：

（1）通过策略自动化消费存储资源　以虚拟机为中心的安置、保护和性能策略。

（2）基于虚拟化的、不依赖于硬件的数据服务　以虚拟机为中心的快照、克隆、复制和备份。

（3）通过虚拟化管理程序提取出存储抽象层　以数据存储和 VMDK 形式使用的异构存储。

贯穿这三个层面，VMware 提供对应于共享存储 SAN/NAS 和分布式存储 DAS 两个维度的解决方案：可扩展 DAS 解决方案 Virtual SAN 和提高存储性能的解决方案。

图 3-33　VMware 解决方案成员

注：SAN/NAS：存储区域网（Storage Area Networking）/网络存储器；DAS：数据收集台（Data Acquisition Station）；VMware Virsto：虚拟化巨头 VMware 收购了存储虚拟机管理器厂商 Virsto；vSphere Flash Read Cache：虚拟环境闪读缓存；Virtual Volumes：虚拟卷；VMware Virtual SAN：虚拟 SAN；SSD：固态硬盘（Solid State Disk）。

3.2.3.3　软件定义的网络

现有的网络体系结构对底层物理硬件有很大的依赖，它们依赖于专用物理设备，因此灵活性很差。除此之外，这种不灵活的体系结构对工作负载和应用的扩展与迁移都产生了很大的限制。在安全方面，传统的安全防护手段价格昂贵，对虚拟化平台不具有感知能力，使用不够灵活，管理难度大，不能很好地满足新架构的需要。

软件定义的网络会创建一个 2～7 层的网络服务，通过创建软件驱动型抽象层将网络连接和安全组件与底层物理网络基础架构完全分离，因此它可以确保硬件具有独立性，使网络连接和安全服务摆脱与硬件绑定的限制。

该方案可以从终端主机的角度忠实地再现物理网络模型：工作负载感觉不到任何差异，因此，软件定义的网络与安全对上层应用是透明的，上面的业务可以不做任何修改而继续使用。

现在，VMware 的软件定义数据中心（SDDC）体系结构扩展虚拟化技术到整个物理数据中心基础设施。VMware 的 NSX 和网络虚拟化平台是 VMware SDDC 架构的关键组成部分。服务器虚拟化以编程方式创建、拍摄快照、删除和恢复基于软件的虚拟机（VM），同样的，NSX 网络虚拟化编程方式创建、拍摄快照、删除和恢复基于软件的虚拟网络（服务器虚拟化与网络虚拟化比照如图 3-34 所示）。其结果是用一个完全革命性的方法来部署网络，不仅使数据中心管理员能够更好更具规模地处理业务订单，同时还大大简化了管理底层物理网络运营模式。NSX 平台可以部署在任何具有 IP 转发能力的网络上，也可以运行在现有传统网络模型或者新型的扁平化 IP 网络上，NSX 具备高兼

容、非破坏性的解决方案。事实上，在现有物理网络基础设施上部署 NSX 平台，可以部署一个软件定义的数据中心。

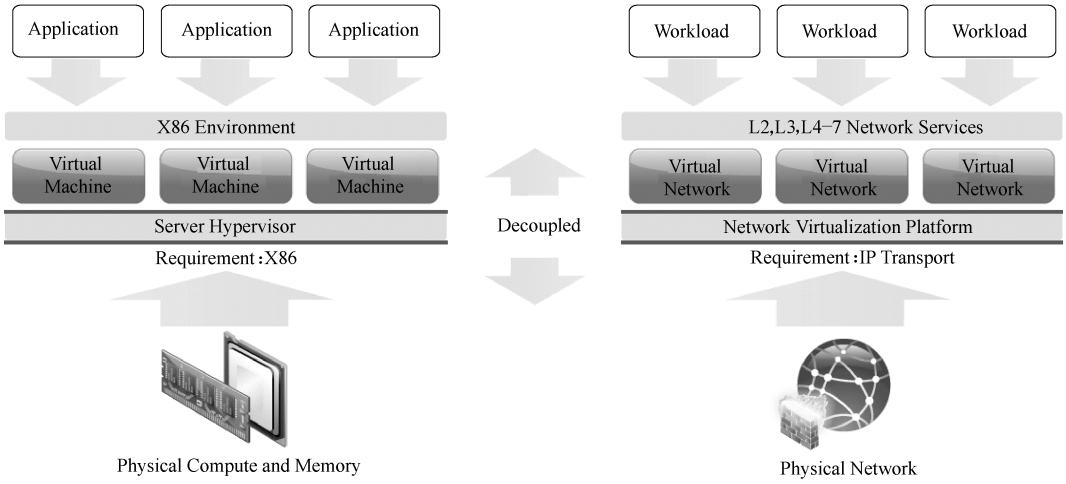


图 3-34　软件定义数据中心

注：Application：应用；Workload：负载；X86 Environment：X86 环境；Virtual Machine：虚拟机；Server Hypervisor：服务器虚拟化；Requirement X86：要求 X86；Physical Compute and Memory：物理计算和内存；Decoupled：减弱；Network Service：网络服务；Network Virtualization Platform：网络虚拟平台；Requirement IP Transport：要求 IP 传输；Physical Network：物理网络。

VMware NSX 为虚拟机提供部署在普通 IP 网络硬件上的、能够提供完整服务的、可编程以及可移动的虚拟网络。NSX 将 Nicira NVP 和 VMware vCloud Network and Security（vCNS）整合成一个统一的平台，让 VMware 能够实现网络虚拟化就如同实现计算虚拟化一样。

NSX 提供了一个简化的逻辑网络元素和服务的完整套件，包括逻辑交换机、路由器、防火墙、负载均衡器、VPN、QoS、监控以及安全。这些服务可以在虚拟网络中通过基于 NSX API 的任何云计算管理平台进行调配，并且可以安排在任何隔离和多租户拓扑中。虚拟网络可以通过任何现有的网络进行无中断部署，并且可以部署在任何虚拟化管理程序上。

NSX 可以提供网络虚拟机运营模式，以转变数据中心的运营模式和经济性。它以编程方式创建、调配、拍摄快照、删除和还原复杂网络，所有这一切均可在软件中完成。VMware NSX 突破了当前物理网络障碍，使数据中心操作员得以将速度、经济性和选择性提高若干个数量级。NSX 以软件的方式提供 27 层的整体网络和安全模式，实现与底层网络硬件的解耦，它可以充分利用企业现有网络基础架构而无需做出任何改变，从而提高服务交付的速度和敏捷性，并降低成本。

NSX 将网络连接和安全服务绑定到每一个虚拟机上，并随虚拟机进行迁移，这样无需人工干预即可大量添加或转移工作负载，因此其可扩展性和移动性都得到了增强。

在运维管理方面，NSX 提供了统一的集中控制点，它可以进行快速的编程式调配

和无中断部署，这使得从网络调配到部署和维护，所有的一切都实现了高度自动化。

除此之外，NSX 还可以在任何通用 IP 网络硬件上同时支持旧版应用和新应用。

3.2.3.4 软件定义运维管理与服务调配

软件定义数据中心的所有计算、存储及网络资源都是松耦合的，可以根据数据中心内各种资源的消耗比例而适当增加或减少某种资源的配置，这样则使数据中心的管理具有较大的灵活性。

VMware 提供的高效敏捷的服务调配方案不仅可以管理基础计算资源，也可以管理桌面资源。它提供了一个可以跨越不同云提供商的，用于管理和调配虚拟机和物理机，并管理它们生命周期的自助式门户。

而运维管理解决方案可使用户更全面地了解基础设施所有层的情况。它可以收集和分析性能数据、关联异常现象，并可识别出构成性能问题的根本原因。它提供的容量管理可以优化资源使用率，基于策略的配置管理则可以确保合规性并消除数量剧增和配置偏差带来的问题。应用发现、依赖关系映射和成本计量功能为基础设施和运维团队带来了更高级别的应用感知和财务责任。

上述这些自动化的管理方案是传统数据中心没有的功能，它可以实现较少工作人员对数据中心的高度智能管理。这种特性一方面能降低数据中心的人工维护成本，另一方面还能够提高管理效率，提升客户体验。

3.3 华为私有云解决方案

3.3.1 整体架构

数据中心虚拟化是基于传统的硬件基础设施，采用虚拟化的软件技术和统一的云管理平台，可以构建与传统数据中心不一样的云数据中心。数据中心虚拟化后，对外提供的能力比传统数据中心更多，管理更加聚焦，而数据中心运行的用户应用不会发生业务逻辑变化，应用系统会平滑迁移到云平台。

华为数据中心虚拟化解决方案的逻辑架构如图 3-35 所示。

华为数据中心虚拟化解决方案从逻辑上分为产品解决方案和专业服务解决方案，其中产品解决方案又分为硬件部分和软件部分，硬件部分是指华为可以提供从数据中心基础层的机房建设、供电、散热方案到数据中心使用的服务器（刀片式和机架式）、存储、网络设备、安全设备等全套硬件产品。当然，华为也可以基于客户提供的机房现有可用硬件设施建设虚拟化数据中心，同时兼容客户指定的业界主流的硬件产品。软件部分则以业界领先的华为虚拟化软件系统 FusionSphere 为主体来构建云平台及管理系统。

采用华为 FusionSphere 构建云平台，在资源的使用和管理上可以为用户带来如下好处：

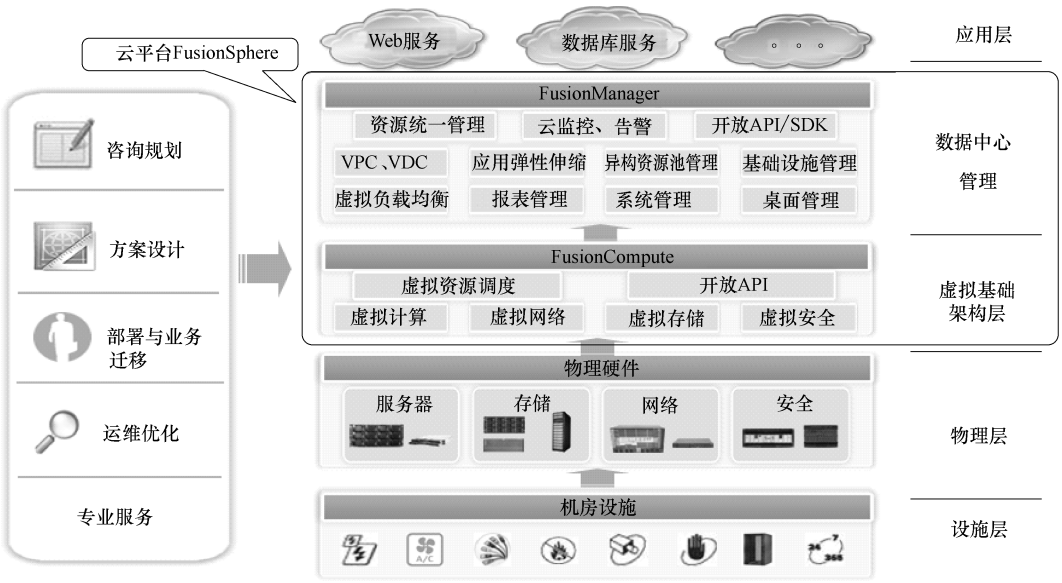


图 3-35　整体逻辑架构

注：FusionSphere：云操作系统；FusionManager：华为公司提供的面向硬件设备、虚拟化资源与应用的管理软件；VPC：虚拟私有云（Virtual Private Cloud）；VDC：虚拟化数据中心（VIRTUAL DATA CENTER）；API/SDK：应用编程接口（Application Programming Interface）/软件开发工具包（Software Development Kit）；FusionCompute：云操作系统基础软件。

1）FusionSphere 提供高可用性（HA）的弹性虚拟机，支持热迁移功能，能够有效减少设备故障时间，确保核心业务的连续性，避免传统 IT 单点故障导致的业务不可用保障业务系统的连续性与虚拟机的安全隔离。

2）易实现物理设备、虚拟设备、应用系统的集中监控、管理维护自动化与动态化。

3）便于快速投放业务，缩短业务上线周期，增强系统的扩展性和灵活性最终提高了管理维护效率。

4）利用云计算技术可自动化并简化资源调配，实现分布式动态资源优化，智能地根据应用负载进行资源的弹性伸缩，从而大大提升系统的运作效率，使 IT 资源与业务优先事务能够更好地协调。

5）在数据存储方面，通过共享的 SAN 存储架构，可以最大化地发挥虚拟架构的优势。

6）提供虚拟机的 HA、虚拟机热迁移、存储热迁移技术提高系统的可靠性；提供虚拟机快照备份技术（HyperDP）等，而且为以后的数据备份与容灾提供扩展性并打下基础。

在产品解决方案之外，华为数据中心虚拟化解决方案还包括专业服务解决方案，主要是在项目建设的各个阶段为用户提供各种专项服务，包括项目启动前的咨询、分析和规划服务、现网业务的性能数据采集和容量分析评估服务、应用迁移云平台的建议、应用迁移方案设计、迁移实施、应用迁移的连续性保障、迁移效果评估、应用部署后的系

统运维优化、云数据中心代维服务等。

3.3.2 虚拟化软件系统

虚拟化层提供基础的虚拟化功能，提供服务器、存储、网络的虚拟化功能，并向上对云管理系统提供接口。每套虚拟化引擎应该由一对主备管理节点组成。一对主备管理节点对应一个物理集群。一个物理集群中可以把多台服务器划分成一个资源集群，一个计算资源池有相同的调度策略，一个物理集群中可以包含多个资源集群。

虚拟化引擎主要负责硬件资源的虚拟化，以及对虚拟资源、业务资源、用户资源的集中管理。它采用虚拟计算、虚拟存储、虚拟网络等技术，完成计算资源、存储资源、网络资源的虚拟化。同时通过统一的接口，对这些虚拟资源进行集中调度和管理，从而降低业务的运行成本，保证系统的安全性和可靠性，构筑安全、绿色、节能的云数据中心能力。

3.3.2.1 虚拟化计算

1. 服务器虚拟化

将服务器物理资源抽象成逻辑资源，使一台服务器变成几台甚至上百台相互隔离的虚拟服务器，不再受限于物理上的界限，而是使 CPU、内存、磁盘、I/O 等硬件变成可以动态管理的“资源池”，从而提高资源的利用率，简化系统管理。同时硬件辅助虚拟化技术可以提升虚拟化效率，并增加虚拟机的安全性。

2. 虚拟机资源管理

客户可以通过自定义方式或基于模板创建虚拟机，并对集群资源进行管理，包括资源自行动态调度（包含负载均衡和动态节能）、虚拟机管理（包含创建、删除、启动、关闭、重启、休眠、唤醒虚拟机）、存储资源管理（包含系统盘、用户盘和共享盘的管理）、虚拟机安全管理［包含自定义虚拟局域网（Virtual Local Area Network，VLAN）或使用安全组］，此外，还可以根据业务负载灵活调整虚拟机的 QoS（包括 CPU QoS 和内存 QoS）。

3. 虚拟机资源动态调整

FusionCompute 支持虚拟机资源动态调整，用户可以根据业务负载动态调整资源的使用情况。虚拟机资源调整包括以下内容：

1）离线或在线调整 VCPU 数目。

2）无论虚拟机处于离线（关机）或在线状态，用户都可以根据需要增加或者减少虚拟机的 VCPU 数目。

3）离线或在线调整内存大小。

4）无论虚拟机处于离线或在线状态，用户都可以根据需要增加或者减少虚拟机的内存容量。

5）离线添加或删除网卡。

6）虚拟机离线状态下，用户可以挂载或卸载虚拟网卡。

7）离线或在线挂载虚拟磁盘。

8）无论虚拟机处于离线或在线状态，用户都可以挂载虚拟磁盘，实现存储资源的灵活使用。

4. 分布式资源调度和电源管理

FusionCompute 提供各种虚拟化资源池，包括计算资源池、存储资源池、虚拟网络和虚拟防火墙。资源调度是指这些虚拟化资源根据不同的负载进行智能调度，达到系统各种资源的负载均衡，在保证整个系统高可靠性、高可用性和良好的用户体验的同时，也有效提高了数据中心资源的利用率。

5. 虚拟机热迁移

FusionCompute 支持在同一共享存储的主机之间自由迁移虚拟机。在虚拟机迁移期间，用户业务不会有任何中断。该功能可避免因服务器维护造成的业务中断，进而起到降低数据中心电能消耗的作用。

3.3.2.2 虚拟化网络

1. 虚拟网卡

虚拟网卡均有自己的 IP 地址、MAC 地址（Medium/Media Access Control），从网络角度来看，虚拟网卡与物理网卡一致。FusionCompute 支持智能网卡，实现多队列、虚拟交换、QoS、上行链路聚合功能，提升虚拟网卡的 I/O 性能。

2. 灵活的虚拟交换机

虚拟交换机为 FusionCompute 的网络模块提供新的虚拟交换模式，具备 VLAN、DHCP（Dynamic Host Configuration Protocol，动态主机配置协议）隔离、带宽限速等基本功能，同时，还具备良好的功能扩展性。通过虚拟交换机，同一主机上的虚拟机可以使用与物理交换机相同的协议相互通信。

3. 网络 I/O 控制

网络 QoS 策略提供发送方向的带宽配置控制能力，包含如下几个方面：

1）基于网络平面的带宽控制。

2）提供基于网络平面的带宽控制功能。

3）基于虚拟网卡的带宽控制。

4）提供基于虚拟网卡的带宽保证（服务器需要配备智能网卡）、上限带宽、带宽优先级控制能力。

5）基于端口组成员接口的带宽控制。

6）基于端口组的每个成员接口提供上限带宽、带宽优先级的控制能力。

4. 分布式虚拟交换机

分布式交换机的功能类似于普通的物理交换机，每台主机都连接到分布式交换机中。分布式交换机的一端是与虚拟机相连的虚拟端口，另一端是与虚拟机所在主机上的物理以太网适配器相连的上行链路。通过它可以连接主机和虚拟机，实现系统网络互通。另外，分布式交换机在所有关联主机之间作为单个虚拟交换机使用。此功能可使虚拟机在跨主机进行迁移时确保其网络配置保持一致。

3.3.2.3 虚拟化存储

1. 虚拟镜像管理系统

虚拟镜像管理系统是一种高性能的集群文件系统。虚拟镜像管理系统使虚拟化技术的应用超出了单个存储系统的限制，针对虚拟服务器环境，可让多个虚拟机共同访问一个整合的集群式存储池，从而显著提高了资源利用率。

2. 虚拟存储管理

存储虚拟化是将存储设备抽象为数据存储，虚拟机在数据存储中作为一组文件存储在自己的目录中。数据存储是逻辑容器，类似于文件系统，它将各个存储设备的特性隐藏起来，并提供一个统一的模型来存储虚拟机文件。存储虚拟化技术可以更好地管理虚拟基础架构的存储资源，使系统大幅提升存储资源利用率和灵活性，提高应用的正常运行时间。

3. 虚拟存储精简置备

虚拟存储精简置备是一种通过灵活地按需分配存储空间来优化存储利用率的方法。精简置备可以为用户虚拟出比实际物理存储更大的虚拟存储空间，只有写入数据的虚拟存储空间才会被真正分配物理存储，未写入数据的虚拟存储空间不占用物理存储资源，从而提高存储利用率。

4. 虚拟机快照

虚拟机快照功能是指把某一时刻的虚拟机状态像照片一样保存下来，在需要时用快照功能把虚拟机恢复到快照时的状态。虚拟机快照保存的内容包括虚拟机所有磁盘的信息。该功能应用于数据备份和容灾场景，可提高系统运行的安全性和可靠性。

5. 存储热迁移

虚拟机正常运行时，管理员可通过手动操作，将虚拟机的磁盘迁移到其他存储单元。存储热迁移可以在存储虚拟化管理下的同一个存储设备内，以及不同存储设备之间进行迁移。热迁移使客户在业务无损的情况下动态调整虚拟机存储资源，以实现设备维护、存储 DRS（Distributed Resource Scheduler，分布式资源调度）等操作。同一时刻，单个主机上最多允许 1 个存储（磁盘）进行迁入和迁出。

3.3.3 虚拟化系统特性

3.3.3.1 兼容性

FusionCompute 支持基于 X86 硬件平台的多种主流服务器、I/O 设备、存储设备、Linux/Windows 操作系统及应用软件，可供企业灵活选择。

3.3.3.2 可用性

1. 虚拟机支持热迁移

FusionCompute 支持在一个计算集群内自由迁移虚拟机。在虚拟机迁移期间，用户业务不会有任何中断。如果迁移失败，目的端的虚拟机将销毁，而用户仍可以使用源端

的虚拟机。该功能可避免因服务器维护造成的业务中断，降低数据中心的电能消耗。

2. 虚拟机支持故障迁移

该功能支持虚拟机故障后自动重启。用户创建虚拟机时，可以选择是否支持故障重启，即是否支持 HA 功能。系统周期性地检测虚拟机的工作状态，当物理服务器故障并引起虚拟机故障时，系统会将虚拟机迁移到其他物理服务器重新启动，保证虚拟机能够快速恢复。目前系统能够检测到的引起虚拟机故障的原因包括物理硬件故障和系统软件故障。

3.3.3.3 安全性

1. 虚拟网络访问控制

通过设置虚拟网卡的 VLAN ID 划分虚拟机所示的网络范围，同时，提供 DHCP 隔离安全功能。

2. 业务安全隔离

用户通过申请占 VPC 及相对应的 VLAN，实现不同用户或不同业务之间的安全隔离。

3. 虚拟机安全隔离

FusionCompute 支持安全组功能，实现外部网络对安全组内虚拟机的访问权限控制。

安全组规则随虚拟机的启动而自动生效，虚拟机的迁移不影响安全组规则。

3.3.4 云管理系统

华为云管理平台（FusionManager）聚焦于数据中心虚拟化资源管理、自动化发放、一体化的快捷自动运维管理，并对企业 IT 管理提供开放的管理接口。华为云管理系统将整个数据中心云化，并将系统中用户可见的资源抽取出来纳入统一的资源池管理，为用户提供一体化的资源管理，实现资源自动发放。为用户提供了一种方便获取资源的途径。用户可以通过服务目录自动化地获取资源并在资源上部署用户需要的应用。

云管理软件从软件层面统一各资源管理。FusionManager 云管理平台负责全系统硬件和软件资源的操作维护管理，以及用户业务的自动化运维。

FusionManager 提供服务管理、服务自动化、资源和服务保证等功能，如图 3-36 所示。

1）从底层接入来看，FusionManager 可以接入物理资源和各种云服务。其中，物理资源包括计算设备、存储设备和网络设备。

2）从上层提供的功能来看，FusionManager 可以为用户提供 ITaaS 的多样化的功能和体验。

3）从 FusionManager 本身提供的功能来看，FusionManager 提供了虚拟和物理资源管理、资源自动化管理及其资源的可用性和安全性保障等。

因此，FusionManager 的定位是以云服务自动化管理和资源智能运维为核心，构筑“敏捷、精简”的云数据中心管理体验。

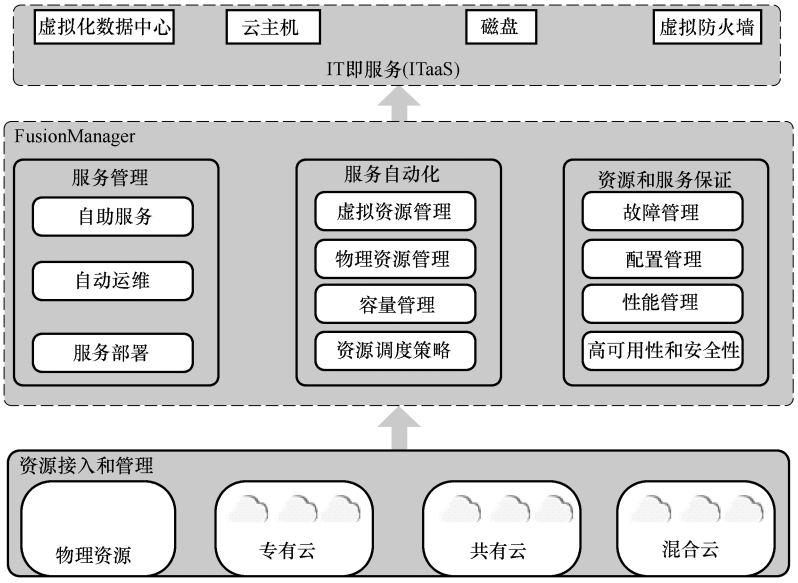


图 3-36　FusionManager 功能示意图

云管理是云数据中心必不可少的功能，主要提供如下功能：

1）资源统一管理：实现对云数据中心中虚拟资源、物理资源的统一管理，包括资源的生命周期管理、资源分配等。

2）云监控和告警：提供物理服务器、虚拟机、存储、交换机、物理集群等各个维度各种性能指标的监控功能；提供各种软件、硬件设备的不同级别的告警界面并呈现、邮件转发、告警短信提示功能；各类物理资源、虚拟资源的拓扑呈现。

3）开放 API/SDK：对外提供开放的 API 接口，并提供 SDK 开发包，方便用户或第三方进行二次开发，对系统进行进一步集成。

4）VPC 和 VDC 功能：为满足企业内部总部和多个分支机构之间或者多个业务部门之间对数据中心资源自主使用、自主管理的需求，VPC 功能可以从网络上对不同分支机构或不同部门的物理资源、虚拟资源进行隔离，保证不同分支机构或不同部门的资源在各自子网内访问；虚拟数据中心（Virtual Data Center，VDC）功能是从组织的角度设置的逻辑概念，可以是一个部门或一个分支机构，每个 VDC 可以被管理员划分一定的物理资源或虚拟资源，VDC 管理员可以管理该 VDC 下的资源，从资源管理、使用的角度进行隔离。

5）应用弹性伸缩：系统按照管理员设置的应用资源使用的变更策略，根据应用负载的大小自动调整应用所需要的虚拟机数量，以达到资源按需使用，弹性伸缩。

6）异构资源池管理：华为云管理平台 FusionManager 不仅能管理华为自己的 FusionCompute 构建的虚拟化资源池，而且可以管理 VMware vSphere 和 Citrix XenServer 构建的虚拟化资源池，从而实现管理流程和操作的完全统一。

7）基础设施管理：主要包括物理服务器、交换机、存储设备的接入、监控、告警，以及物理服务器的上、下电控制。

8）虚拟负载均衡：用户可以在 FusionManager 上申请负载均衡器，将业务虚拟机关联到负载均衡器。负载均衡器根据用户设定的负载均衡策略，将业务请求均匀分发到与之关联的虚拟主机上，使得每个业务虚拟机的负载基本保持均衡，保证业务运行的稳定性和可靠性。

9）报表管理：支持将监控数据导出为报表，便于用户进行进一步分析和管理。

10）系统管理：包括用户管理、系统配置、定时器设置、设置密码规则等功能。

11）桌面管理：FusionManager 中集成了华为桌面云系统的管理入口，可以支持桌面云、云主机的统一管理。

3.4 青云解决方案

3.4.1 虚拟化技术

3.4.1.1 计算

QingCloud 的底层虚拟化技术采用了国际最主流的 KVM（Kernel-based Virtual Machine，KVM 基于内核的虚拟机），是一个开源的系统虚拟化模块，自 Linux 2.6.20 之后集成在 Linux 的各个主要发行版本中。它使用 Linux 自身的调度器进行管理，所以相对于 Xen，其核心源码很少。KVM 目前已成为学术界的主流 VMM（Virtual Machine Manager）之一。KVM 的虚拟化得到硬件厂商的支持（如 Intel 或者 AMD 虚拟化技术），是基于硬件的完全虚拟化，因此其性能超过了其他虚拟化技术，经过青云调优之后的 KVM 更是将这种性能发挥到极致，虚拟资源的能力可以达到 99.98% 的物理资源的能力。

软件定义网络（Software Defined Network，SDN）是 QingCloud 的核心技术之一，而 QingCloud 的私有网络也是 SDN 技术在云计算领域应用的成功范例。目前认为，云计算将替代传统硬件形式实现 IT 产业的真正变革，成为新型 IT 形式。因此，云计算必须满足企业级 IT 的需求，在功能层面传统 IT 能够提供的，云计算也需要提供，例如组网功能，传统 IT 能够实现的性能，云计算也要实现；传统 IT 做不到的，云计算仍然要做到，例如弹性、简便与廉价。

3.4.1.2 存储

QingCloud 采用软件定义存储，针对用户不同类型数据的不同存储需求提供了不同的存储解决方案。存储设计主要考虑容量与性能两个方面，如图 3-37 所示。

运行或在线系统需要高性能存储。用户的业务数据几乎都会存储在数据库里面，需要高性能存储。

离线或备份数据需要高容量，低价格。这部分数据通常量很大，但是对性能要求不

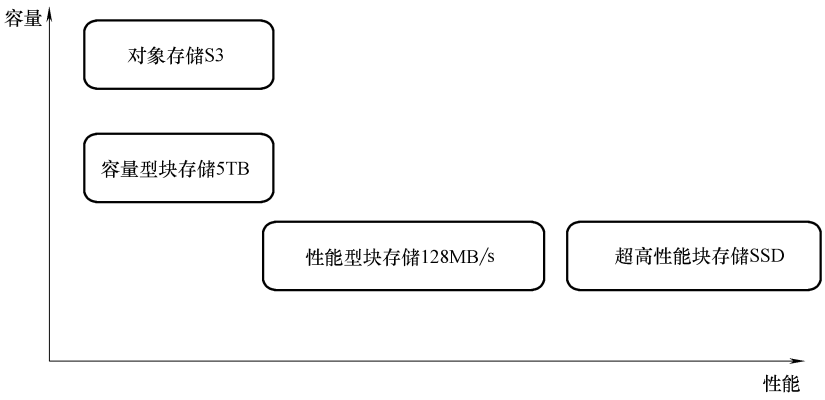


图 3-37　容量与性能

高，对不经常用的东西也不希望负担高额成本。

所有的数据都必须是可靠的，绝对不能丢失。所有严肃的企业用户都会将可靠放在最重要的位置上，因此考验云服务商存储虚拟化技术能力的重点就在于充分保障数据可靠的基础上实现高性能甚至超高性能。同时，还应充分考虑成本与价格因素。

从上面的分析来看，之所以没有银弹方案，是因为用户对存储的需求差异很大，不同的需求需要用不同的方式来解决。鉴于此，QingCloud 推出了一种多维块存储解决方案以满足不同的企业用户各种类型数据存储需求，如图 3-38 所示。

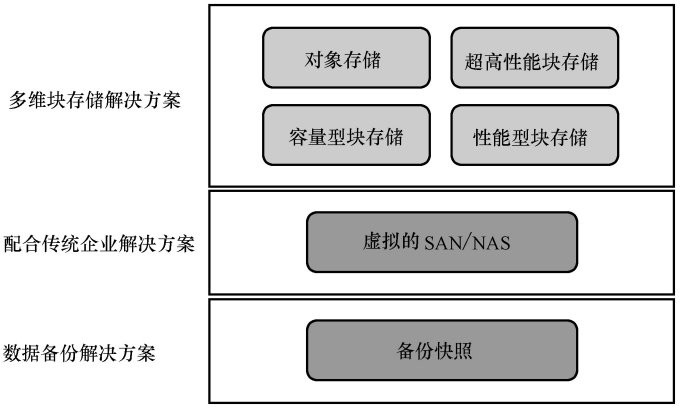


图 3-38　QingCloud 多维块存储解决方案

3.4.1.3　网络

传统网络方案配置复杂，难于扩展，同时也无法支持云上大规模的用户和使用，而 SDN 技术为 IaaS 的网络方面提供了一个非常好的方向。

通常企业用户对网络方面有着如下需求：

1）自由组网。

2）对遗留 IT 环境无侵入。

3）安全性，100% 二层隔离。

4）构建混合云。

5）提升服务体验。

但传统网络方案存在着如下问题：

1）VLAN 4K 限制。

2）部署困难。

3）难于扩展。

4）成本压力。

采用 SDN 技术，QingCloud 很好地解决了用户实际需求，实现了如下目标：

1）物理架构对接传统 IT 环境。

2）控制与转发分离。

3）分布式网络控制。

4）提供开放式 API 支持。

5）网络功能虚拟化，云中网络具有灵活性。

6）利用通用硬件和软件，摆脱专有设备。

对于具体的设计方面，QingCloud 的 SDN 网络分为用户层和物理层两个方面。

（1）用户层　用户层主要考虑功能设计，二层网络（交换机）实现不同用户间 100% 的隔离；三层网络（路由器）连接二层设备，进行端口转发、过滤控制，并支持 VPN、GRE、IPsec 隧道等协议。针对高级用户的需求，QingCloud 提供自管网络，即只提供网络设备，把一切配置的权利交给用户，满足他们的高级需求。

（2）物理层　物理层方面，需要同时考虑物理链路和逻辑链路两个层面，而云中的逻辑链路又包含外部和内部之间的数据流，以及内部之间资源之间的数据流两个方面。

1）外部和内部之间的数据流　需要考虑流量如何进入并到达某个资源；同时还要考虑如何正确返回。因为资源会随时在云内迁移，改变在云中的位置，所以对应的逻辑链路也需要随时变更。

2）内部资源之间的数据流　需要考虑流量如何在内部流动。例如，一个虚拟私有网络有多台主机，每台主机都运行在不同的位置，这个网络会不停地有主机加入和退出，主机的位置也会因为迁移而不停地变化。

由此可见，以上两个方面都是动态的网络流，因此如何在静态的物理链路上处理或者引导动态的逻辑流是 QingCloud 软件定义网络要解决的核心问题。而其中最关键的是：物理链路和逻辑链路和设计和管理需要解藕；动态的逻辑链路管理，需要自行做流控（控制 + 转发）。这其实就是 QingCloud SDN 的核心理念。

在具体的架构设计方面，QingCloud 网络控制节点分散运行于任意物理或虚拟设备中，网关节点用于流控的公共池，代理节点执行于虚拟交换机中。QingCloud SDN 网络架构如图 3-39 所示。

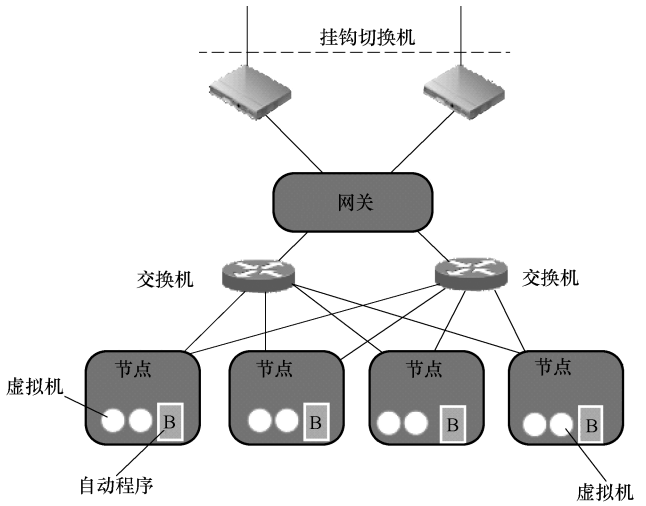


图 3-39　QingCloud SDN 网络架构

3.4.2　云管理方案

QingCloud 虚拟资源的调度管理完全由具备人工智能的软件机器人社区进行掌控。这里面有专门负责计算、存储、网络、安全的机器人，也有负责预警、故障、监控、冗余的机器人，它们会主动上报自己的任务负荷，也会将上层新派发过来的任务进行分解，我们会在网络虚拟化关键技术中详细描述实现的机制。通过机器人社区实现自动任务调度的青云系统没有技术上的资源管理上限，只要网络可达到，云平台操作系统就可以管理成千上万的物理资源。这样的架构带来的另一个好处就是在上线以后的运营与维护之中，大量的脏活累活都完全由不知疲倦的软件机器人承担，这将大幅度减少后期维护成本。

正是因为采用了 KVM 这样全虚拟化技术，青云系统可以为用户提供支持 QoS 策略用于保障虚拟机进行资源分配，不会造成用户之间共享物理资源时相互干扰，这是以前半虚拟化技术所不能达到的。

多点、跨域自动化调度是云平台操作系统另外一个特点，目前已经上线的 www. qingcloud. com 就是一个多 Zone 部署的系统，已经运行超过上千个虚拟服务器，支持 X86 架构服务器的管理数量无设计上限，在北京与广东两地实现统一管理。

机器智能（Machine Intelligence）是 QingCloud 云计算系统资源调度与管理的核心技术之一，其基本理念是通过软件实现智能的资源调度与管理，将基础的运营维护工作交由代码管理，将人力资源从烦琐的基础设施维护中解放出来，专注于代码层面的工作，从而最大限度地提升系统效率，避免人为失误并降低运维投入。

3.4.2.1　资源安置

资源安置（Resource Placement）是云计算领域最基础和核心的问题之一，其解决

的是大量工作负载如何最合理、有效地分配到分布式集群中的不同物理设备中。与大多数云服务商不同的是，QingCloud 的资源安置策略（Resource Placement Policy）除了参考物理设备实时的工作负载情况之外，还将设备的历史状态和信用值（Credit）作为重要参考指标，对三大指标分配不同权重，经过科学计算完成最佳的资源安置。

1. 根据工作负载情况（Workload）

根据均匀化法则，判断当期物理设备工作负载情况，进行任务分配，这是最基础的判断方法。其局限性是当期设备的工作负载情况并不能代表设备的真正工作负载情况。

2. 根据历史状态

所有计算机在运行过程中，不管是物理的还是虚拟的设备，它们在运行过程中，实际上是有大量的历史数据的，每时每刻每分每秒的数据都应该记录下来，予以分析。这样才能够非常清晰地知道，每一个设备，不管是物理的还是虚拟的，它的变化曲线是怎样的，这个是需要抓住的第二个历史因素。

3. 根据每台物理设备的信用值（Credit）

根据机器故障规律、历史运行情况等大量数据计算并判断物理设备健康状态，授予其一定的信用值，该信用值也是系统自动判断和安置资源的重要参照因素。

4. 运行监控系统（Monitoring）

运行监控工作主要是通过 P2P 软件机器人社区来承担，整个过程都是完全自动的，无需人的干涉和参与。监控内容主要包括故障、性能、安全和配置，见表 3-1。

通过运行监控系统，用户可以很方便地了解系统运行情况以及分析和定位问题。监控信息分为不同的粒度，从 1min 到一年的历史数据都有，另外还有 10s 级别的实时数据。

表 3-1　监控内容

监控类型	监控内容
故障监控	• 服务器状态监控，包括 CPU、内存和磁盘状态 • 网络状态监控
性能监控	• CPU 利用率监控 • 内存利用率监控 • 虚拟内存利用率监控 • 磁盘读写监控 • 网络流量监控
安全监控	• TCP 连接数监控
配置监控	• 用户资源使用情况监控，判断是否超出配额限制

3.4.2.2　故障预测

如果说资源安置更多是根据静态的数据进行分析和分配，那么故障预测除了要考虑大量历史数据之外还要更多地基于实时数据流（Real-time Data Stream）进行超快速地实时分析。在多大程度上能够对可能出现的风险和故障进行准确的预测，对大型云平台的稳定运行至关重要，可以最大程度地避免故障的发生以及降低可能出现的故障后果。

3.4.2.3 故障处置

故障处置分为两个阶段，即人为设置并更新规则和机器学习自动创建新规则。在设计上，没有单点故障。无论计算、存储还是网络，所有功能不依赖于任意特定的服务器、硬盘、交换或路由设备，整个系统都是通过一个称为 P2P 机器人社区（Robots Community）进行控制的，故障设备可以快速被取代，负载可以实时转移到其他设备上，以保证故障无害。

第 4 章
开源云计算框架

4.1 云计算领域开源软件概述

开源软件和云计算分别代表着软件开发与 IT 服务模式变革，这两种历史潮流汇合在一起，形成了推动云计算发展的强大动力。开源作为一种无形的力量，对于云计算的普及和应用起着不可忽略的作用，现在的云计算已不再是一个新的尖端技术，它已成为彻底改变我们使用和开发应用软件方式的一种极有价值的重要技术。在云计算的不同服务层次（IaaS、PaaS 和 SaaS）存在着许多优秀的开源软件。

4.1.1 基础设施即服务（IaaS）

1. OpenStack

OpenStack 是目前云计算领域最受欢迎的解决方案。OpenStack 于 2010 年 7 月发布，并且迅速成为标准开源 IaaS 解决方案。OpenStack 是两种云计划的组合，即 Rack-Space Hosting（Cloud Files）和 NASA 的 Nebula platform。OpenStack 是用 Python 语言开发的，并且在 Apache 许可下的开发活动非常活跃。OpenStack 使用一种模块化架构来管理一组计算和存储服务器。可通过三个主要组件来定义 OpenStack，即 Nova（表示计算端）、Swift（表示对象存储）和 Glance（实现映像服务）。OpenStack 支持各种各样的虚拟管理程序，包括 KVM、Linux Containers（LXC）、QEMU、UML、Xen 和 XenServer。OpenStack 现在促进了云基础架构的开放标准，并且因此其采用率也在迅速增加。

2. CloudStack

CloudStack 是一个开源的具有高可用性及扩展性的云计算平台，它可以加速高伸缩性的公共云和私有云的部署、管理、配置。使用 CloudStack 作为基础，数据中心操作者可以快速方便地通过现存基础架构创建云服务。CloudStack 的前身是 Cloud. com，后来

被思杰收购。2011 年 7 月，Citrix 收购 Cloud. com，并将 CloudStack100% 开源。2012 年 4 月 5 日，Citrix 又宣布将其拥有的 CloudStack 开源软件交给 Apache 软件基金会管理，目前 CloudStack 已成为 Apache 基金会最大的顶级项目之一。

CloudStack 支持管理大部分主流的 Hypervisor，如 KVM、XenServer、VMware、Oracle VM、Xen 等。

3. EUCALYPTUS

EUCALYPTUS（Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems），是一种开源的软件基础结构，用来通过计算集群或工作站群实现弹性的、实用的云计算。它最初是美国加利福尼亚大学 Santa Barbara 计算机科学学院的一个研究项目，其独特之处是其接口与 Amazon Elastic Compute Cloud（Amazon EC2——Amazon 的云计算接口）兼容。此外，EUCALYPTUS 还包含了 Walrus，它是一个云存储应用程序，与 Amazon Simple Storage Service（Amazon S3——Amazon 的云存储接口）兼容。

对于虚拟管理程序，EUCALYPTUS 支持 KVM/Linux 和 Xen；对于集群管理，它又包括了 Rocks 集群分发。

4. OpenNebula

OpenNebula 是一个开源云计算基础管理工具，用来方便管理员在数据中心统一部署、创建、分配和管理大量的虚拟机，提供 IaaS。除了支持私有云结构之外，OpenNebula 还支持混合云的概念。混合云允许私有云基础架构与公有云基础架构（比如 Amazon）的集成以提供更高级别的伸缩。

OpenNebula 支持 Xen、KVM/Linux 和 VMware，并且依赖于 libvirt 等元素来进行管理和内省。

5. Nimbus

Nimbus 是一种以科学计算为中心的 IaaS 解决方案。使用 Nimbus，可以借助远程资源（比如由 Amazon EC2 提供的远端资源）并能对它们进行本地管理（配置、部署 VM 和监视等）。Nimbus 由 Workspace Service project（Globus. org 的一部分）演变而来，由于依赖于 Amazon EC2，所以 Nimbus 支持 Xen 和 KVM/Linux。

6. Xen 云平台

Citrix 已经将 Xen 集成到一个 IaaS 平台，Xen 被用作虚拟管理程序，同时又并入了其他的开源功能，比如 Open vSwitch。Xen 解决方案的优势之一是其着重于来自 Kensho 项目的基于标准的管理［包括 OVF、Distributed Management Task Force（DMTF）、Common Information Model（CIM）和 Virtualization Management Initiative（VMAN）］。Xen 管理栈支持 SLA 保障，以及具体的退款标准。

7. OpenQRM

OpenQRM 是一种数据中心管理平台，提供了单个控制台来管理整个虚拟化数据中心，在架构上它允许插入以便集成第三方工具。OpenQRM 集成了对高可用性（通过冗余）的支持，并支持各种各样的虚拟管理程序，包括 KVM/Linux、Xen、VMware 和 Linux VServer。

4.1.2 平台即服务（PaaS）

1. Cloud Foundry

Cloud Foundry 是 VMware 于 2011 年 4 月 12 日推出的业界第一个开源 PaaS 云平台，它支持多种框架、语言、运行环境、云平台及应用服务，使开发人员能够在几秒钟内进行应用程序的部署和扩展，无需担心任何基础架构的问题。同时，它本身是一个基于 Ruby on Rails 的由多个相对独立的子系统通过消息机制组成的分布式系统，使平台在各层级都可水平扩展，既能在大型数据中心运行，也能运行在一台桌面计算机中，两者使用相同的代码库。

作为新一代云应用平台，Cloud Foundry 专为私有云计算环境、企业级数据中心和公有云服务提供商所打造。Cloud Foundry 云平台可以简化现代应用程序的开发、交付和运行过程，在面对多种公有云和私有云的选择、符合业界标准的高效开发框架以及应用基础设施服务时，可以显著提高开发者在云环境中部署和运行应用程序的能力。

2. Openshift

OpenShift 是 RedHat 公司推出的一个 PaaS 云计算应用平台，开发者可以在上面构建、测试、部署和运行应用程序，它支持 Java、Ruby、Node. js、Python、PHP、Perl 等众多语言环境和开发框架，并且支持 MySQL、PostgreSQL、MongoDB 等数据库服务。最重要的是，它的整个体系都是开源的，与 IaaS 开源云计算软件 OpenStack 一样，对于拥有硬件资源而希望部署云的服务提供商来说很有研究价值。

OpenShift Online 服务构建在 Red Hat Enterprise Linux 上。Red Hat Enterprise Linux 提供集成应用程序，运行库和一个配置可伸缩的多用户单实例的操作系统，以满足企业级应用的各种需求。

4.1.3 软件即服务（SaaS）

SaaS 是一种通过 Internet 提供软件的模式。它的目标是用户不用再购买软件，而改用向提供商租用基于 Web 的软件来管理企业经营活动，且无需对软件进行维护。它消除了企业购买、构建和维护基础设施和应用程序的需要，从而降低了成本。可见 SaaS 是提供面向某一业务领域的服务，因此一般由商业公司提供收费服务或者互联网企业提供的免费服务，而不是一个简单的开源软件，在此不对这些 SaaS 进行详细讨论。

4.2 Cloud OS 开源软件

4.2.1 OpenStack 架构

OpenStack 是一个开源的 IaaS 云计算平台，由 Rackspace Cloud 和 National Aeronau-

tics and Space Administration（NASA，美国国家航空和航天局）共同发起。自成立以来，OpenStack 已经获得业界广泛认可，目前，它的支持者数量已超过 100 个，其中包括许多业内最大的组织。它目前的白金会员包括 IBM、AT&T、Canonical、HP、Nebula、Rackspace、Red Hat 和 SUSE。

4.2.1.1 总体架构

如图 4-1 所示，OpenStack 包括 9 个核心模块，与其他开源 IaaS 相比，OpenStack 在架构设计具有松耦合、高可扩展、分布式的特点，采用纯 Python 实现。总体架构设计先进，模块划分、组件交互、消息机制、异常处理和扩展机制考虑周全，有利于扩展、持续优化和规模化部署。OpenStack 在设计上采用统一的框架和标准 API，同时支持多种技术实现和第三方产品。OpenStack 核心模块的名称及描述见表 4-1。

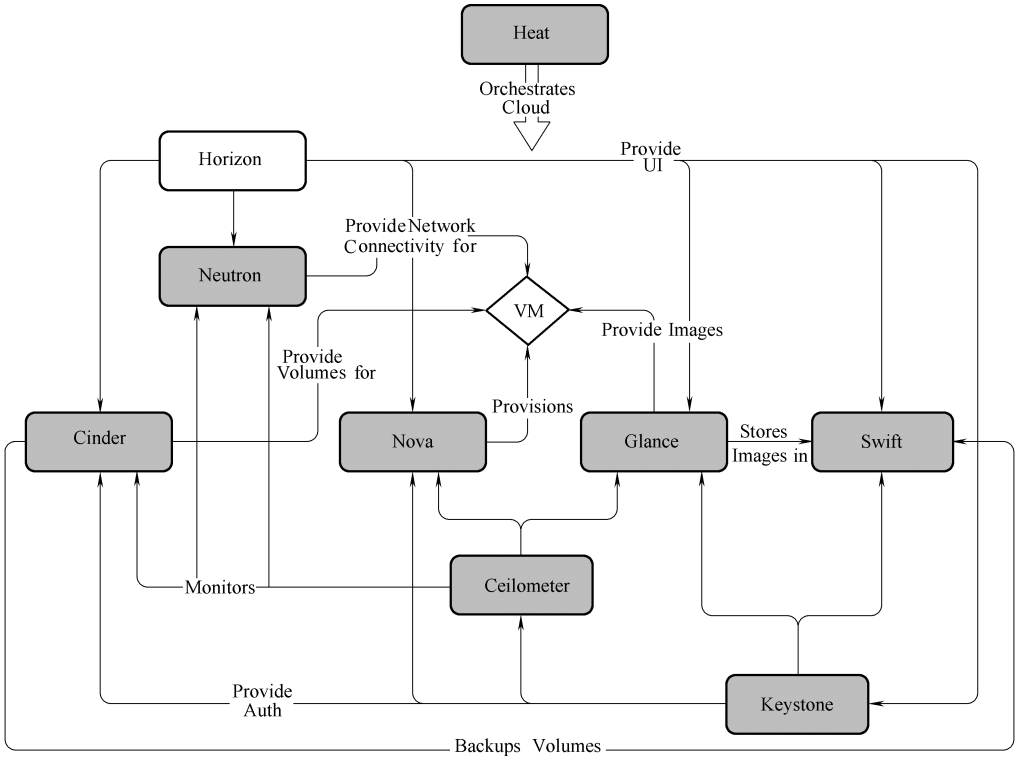


图 4-1 OpenStack 概念架构图

注：Heta：服务编排组件；Horizon：Web 自服务门户；Neutron：网络服务组件；Cinder：存储服务组件；Nova：计算服务组件；Ceilometer：实现监控和计量的组件；Glance：镜像管理服务；VM：虚拟机；Swift：对象存储服务组件；Keystone：身份认证和授权的组件；Orchestrates Cloud：云流程编排；Provide Network Connectivity：提供网络连接；Provide Volumes：提供卷；Provide Images：提供镜像；Stores Images：存储镜像；Monitors：监控；Provide Auth：提供权限；Backups Volumes：备份卷。

表 4-1 OpenStack 核心模块的名称及描述

服务	模块名称	描　　述
Dashboard	Horizon	WEB 自服务门户，用户通过该服务与 OpenStack 的各服务进行交互，如启动虚拟机实例、分配 IP 地址、设置访问控制等

（续）

服务	模块名称	描　述
Compute	Nova	计算服务组件，OpenStack 的核心服务，管理 OpenStack 环境中的虚拟机实例
Networking	Neutron	网络服务组件，把网络连接作为一个服务提供给 OpenStack 其他服务，例如计算服务。允许终端用户创建并添加网络接口，通过一个插件式架构支持大量网络厂商设备及网络技术
存储		
Object Storage	Swift	对象存储服务组件
Block Storage	Cinder	块存储管理组件，向虚拟机提供可用于持久存储的块存储服务。插件式的驱动架构支持各种块存储设备，如本地磁盘、LVM 或各大厂商提供的设备
共用服务		
Identity service	Keystone	身份认证和授权组件
Image Service	Glance	镜像管理服务，本身不负责存储、支持本地存储、NFS、Swift、Sheepdog 和 Ceph。支持多个数据中心的镜像管理
Telemetry	Ceilometer	实现监控和计量组件
更高层服务		
Orchestration	Heat	服务编排组件，使用自带的 HOT 模板或 AWS 的 CloudFormation 模板，通过 OpenStack 中各服务的 REST API，将各组件的资源组织形成云应用

4.2.1.2　计算组件

OpenStack Compute（Nova）控制云计算架构（基础架构服务的核心组件）。它是用 Python 编写的，创建了一个抽象层，使 CPU、内存、网络适配器和硬盘驱动器等商品服务器资源实现虚拟化，并具有提高利用率和自动化的功能。

如图 4-2 所示，Nova 采用无共享的架构，主要组件可以部署在不同的机器上，依靠消息队列实现组件间的异步通信，虚拟机的信息和状态存储在数据库中。它主要由一系列的守护进程组成：调度程序（Nova-scheduler）、计算（Nova-compute）守护进程、网络管理（Nova-network）和卷管理（Nova-volume）。

1）Nova-scheduler 确定为虚拟机请求分配哪个计算主机，只在进行配置时做出此决定，不会重新分配正在运行的实例。

2）Nova-compute 用于管理与虚拟机管理程序和虚拟机的通信，从消息队列中获取任务，并使用虚拟机管理程序的 API 执行虚拟机的创建和删除等任务，负责更新数据库中的状态。

3）Nova-network 负责管理 IP 转发、网桥和虚拟局域网。在 OpenStack 的后期版本中新的网络服务组件 OpenStack Neutron 不仅涵盖了 Nova-network 的功能，而且能提供许多新的特性。

4）Nova-volume 负责处理将块存储卷附加到虚拟机以及从虚拟机分离块存储卷（类似于 Amazon Elastic Block Store）。在 OpenStack 的后期版本中此功能已被提取到 OpenStack Cinder。

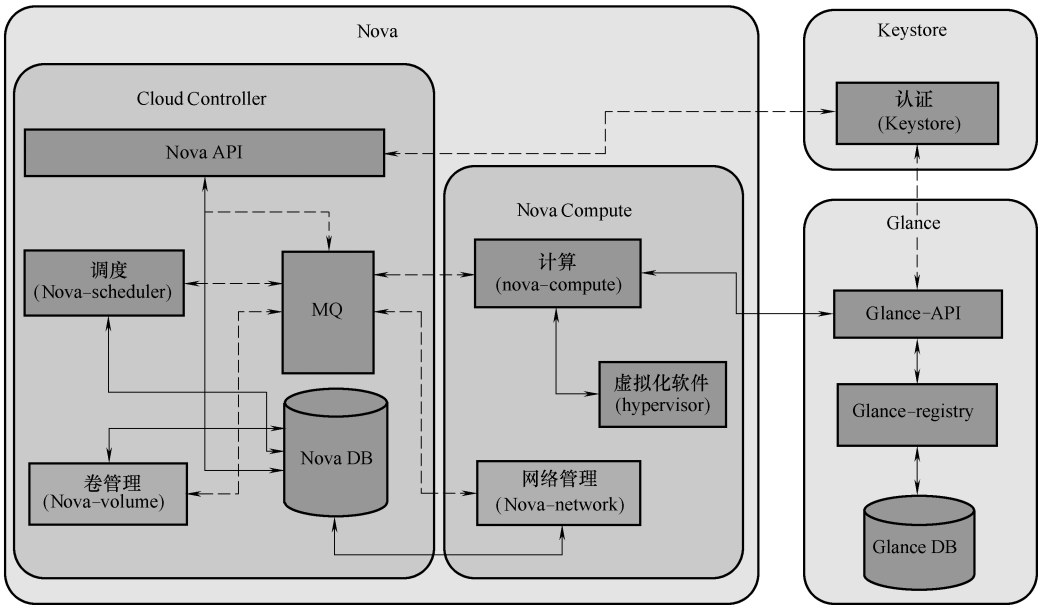


图 4-2　OpenStack Nova 架构

注：Nova：计算服务组件；Cloud Controller：云控制器；Nova API：API（应用程序编程接口）服务；MQ：消息中间件；Nova DB：Nova 数据库；Nova Compute：计算服务；Keystone：身份认证和授权的组件；Glance：镜像管理服务；Glance-API：Glance 应用程序编程接口；Glance-registry：Glance 注册，对数据库进行增删改查；Glance DB：Glance 数据库。

Nova-scheduler 作为一个后台进程运行，它会根据一定的算法从计算资源池中选择一个计算节点用于启动新的 VM 实例，其步骤主要有以下两步：

1）根据配置文件中定义的过滤器（Filter）获得候选主机，Nova-scheduler 过滤器如图 4-3 所示。

2）使用不同的算法计算权值（Weight）获取最终的节点。

① Costsand Weights：主机进行权值的计算，根据策略选择相应的某一台。

② ChanceScheduler：从候选节点中随机选择一个节点。

③ MultiScheduler：包含多个子调度程序，如可以为 Nova-compute 和 Nova-volume 分别指定调度程序。

OpenStack 默认支持多种过滤策略，如 CoreFilter（CPU 数过滤策略）、RamFilter（Ram 值选择策略）、AvailabilityZoneFilter（指定集群内主机策略）、JsonFiliter（Json 串指定规则策略），开发者也可以实现自己的过滤策略，一些常用的过滤策略见表 4-2。

表 4-2　常用的过滤策略

名称	描　　述
AllHostsFilter	不做任何过滤,直接返回所有可用的主机列表
AvailabilityZoneFilter	返回创建虚拟机参数指定的集群内的主机
ComputeFilter	根据创建虚拟机规格属性选择主机
CoreFilter	根据 CPU 数过滤主机

（续）

名称	描　　述
IsolatedHostsFilter	根据“image_isolated”和“host_isolated”标志选择主机
JsonFilter	根据简单的 Json 字符串指定的规则选择主机
RamFilter	根据指定的 RAM 值选择资源足够的主机
SimpleCIDRAffinityFilter	选择在同一 IP 段内的主机
DifferentHostFilter	选择与一组虚拟机不同位置的主机
SameHostFilter	选择与一组虚拟机相同位置的主机

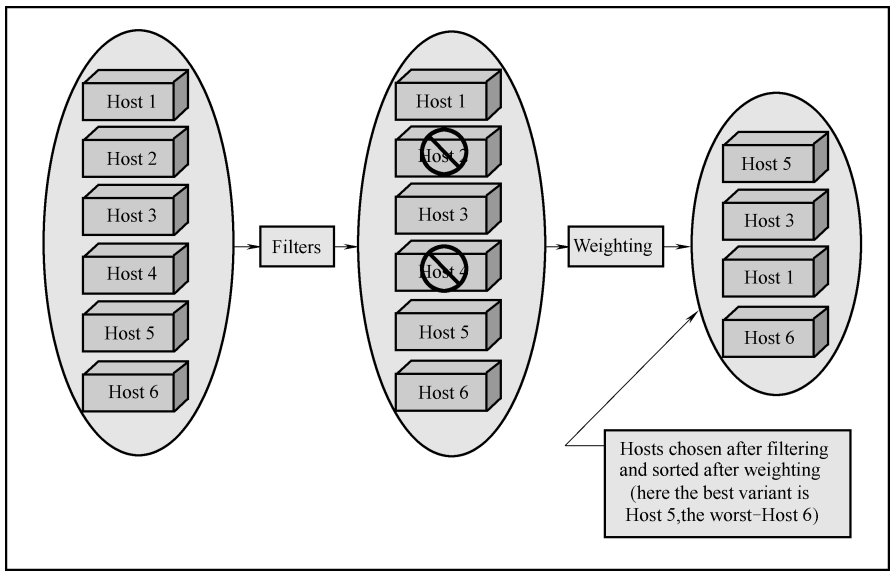


图 4-3　Nova-scheduler 过滤器

注：Host：主机；Filters：过滤器；Weighting：计算权值；Hosts chosen after filtering and sorted after weighting（here the best variant is Host 5，the worst-Host 6）：通过计算权值、分类和过滤，获取主机（最好的选择是主机 5，最差的是主机 6）。

经过主机过滤后，需要对主机进行权值的计算，根据策略选择相应的某一台主机（对于每一个要创建的虚拟机而言）。OpenStack 对权值的计算需要一个或多个（权值，代价函数）的组合，然后对每一个经过过滤的主机调用代价函数进行计算，将得到的值与权值乘积，得到最终的权值。Openstack 将在权值最小的主机上创建一台虚拟机。OpenStack 虚拟机分配示意图如图 4-4 所示。

4.2.1.3　存储组件

OpenStack 包含三个与存储有关的组件：

1. 镜像存储（Glance）

Glance 提供虚拟机镜像（Image）存储和管理。OpenStack 镜像服务支持多种虚拟机镜像格式，包括 VMware（VMDK）、Amazon 镜像（AKI、ARI、AMI）以及 VirtualBox 所支持的各种磁盘格式。

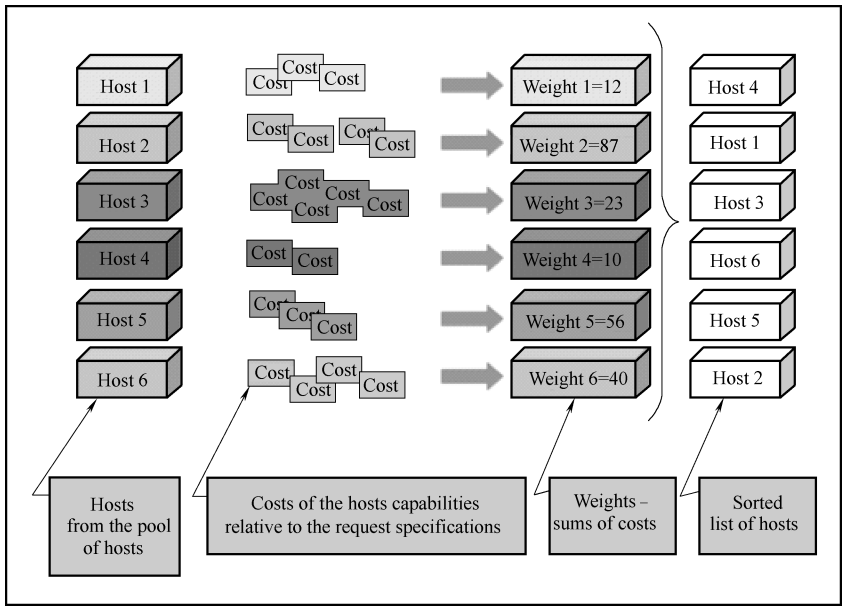


图 4-4　OpenStack 虚拟机分配示意图

注：Host：主机；Cost：消耗；Weight：权值；Hosts from the pool of hosts：从计算资源池中获取主机列表；Costs of the hosts capabilities relative to the request specifications：主机性能消耗需求说明；Weights-sums of costs：权重 – 消耗总值；Sorted list of hosts：主机排序列表。

Glance 提供虚拟机镜像的发现、注册、取得服务。Glance 提供 REST API 查询虚拟机镜像的元数据，并且可以获得镜像。虚拟机镜像可以被存储到多种存储上。Glance 的架构如图 4-5 所示。

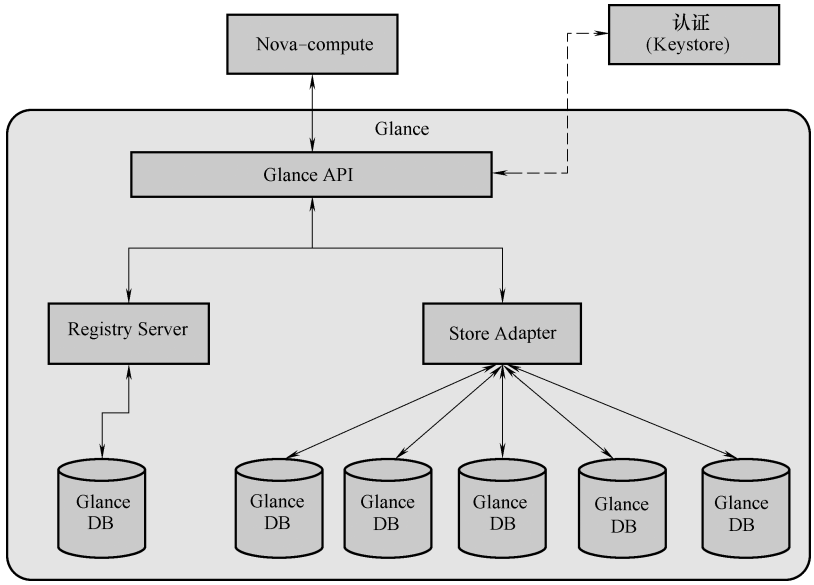


图 4-5　Glance 架构

注：Nova-computer：计算服务；Glance：镜像服务；Glance API：镜像服务应用程序编程接口；Registry Server：注册服务；Glance DB：镜像服务数据库；Store Adapter：存储适配器。

2. 块存储（Cinder）

Cinder 为虚拟机提供了持久块存储服务。它的核心是对卷进行管理，允许对卷、卷的类型、卷的快照进行处理。它适用于可扩展的文件系统、与企业存储服务的集成以及需要访问原生块级存储的应用程序。Cinder 没有实现对块设备的管理和实际服务，而是提供了一个抽象层，为后端不同的存储结构提供了统一的接口，不同的块设备服务厂商在 Cinder 中实现其驱动支持用来与 OpenStack 进行整合。Cinder 架构如图 4-6 所示。

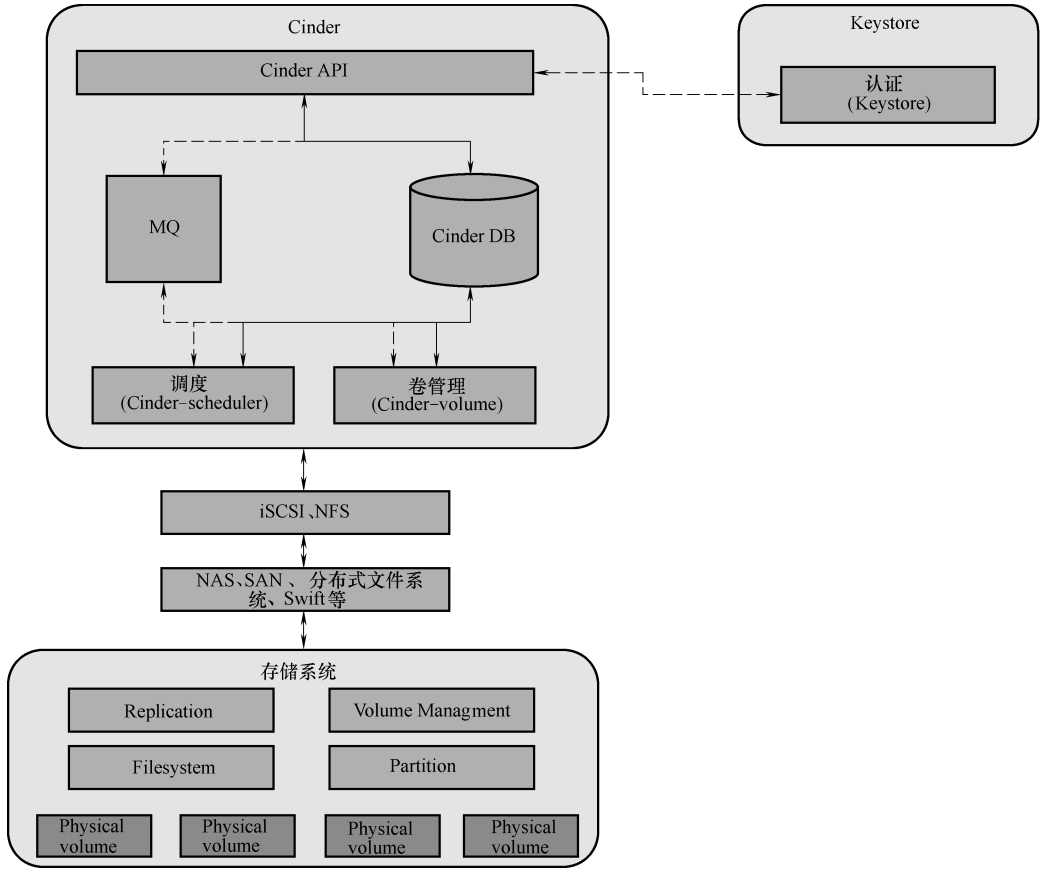


图 4-6　Cinder 架构

注：Cinder：块存储管理组件；Cinder API：块存储管理组件应用程序编程接口；Cinder DB：块存储管理组件数据库；MQ：消息中间件；Cinder-scheduler：调度管理；Cinder-volume：卷管理；iSCSI：Internet 小型计算机系统接口，是一种基于 TCP/IP 的协议，用来建立和管理 IP 存储设备、主机和客户机等之间的相互连接，并创建存储区域网络（SAN）；NFS：网络文件系统；NAS：网络存储器；SAN：存储区域网络；Swift：对象存储服务组件；Replication：复制；Volume Management：卷管理；Filesystem：文件系统；Partition：分区；Physical Volume：物理卷。

Cinder 服务由以下几部分组成：

1）API Service：负责接受和处理 Rest 请求，并将请求放入 MQ 队列。

2）Scheduler Service：处理任务队列的任务，并根据预定策略选择合适的 Volume Service 节点来执行任务。

3）Volume Service：该服务运行在存储节点上，管理存储空间。每个存储节点都有

一个 Volume Service，若干个这样的存储节点联合起来可以构成一个存储资源池。

对于本地存储，Cinder-volume 可以使用 LVM 驱动，该驱动当前的实现需要在主机上事先用 LVM 命令创建一个 Cinder-volumes 的卷组，当该主机接收到创建卷请求时，Cinder-volume 在该卷组上创建一个逻辑卷，并且用 open—iSCSI 将这个卷当作一个 iSCSI 目标给输出。当然还可以将若干主机的本地存储用 sheepdog 虚拟成一个共享存储，然后使用 sheepdog 驱动。EMC、NetApp、IBM 块存储架构如图 4-7 ~ 图 4-9 所示。

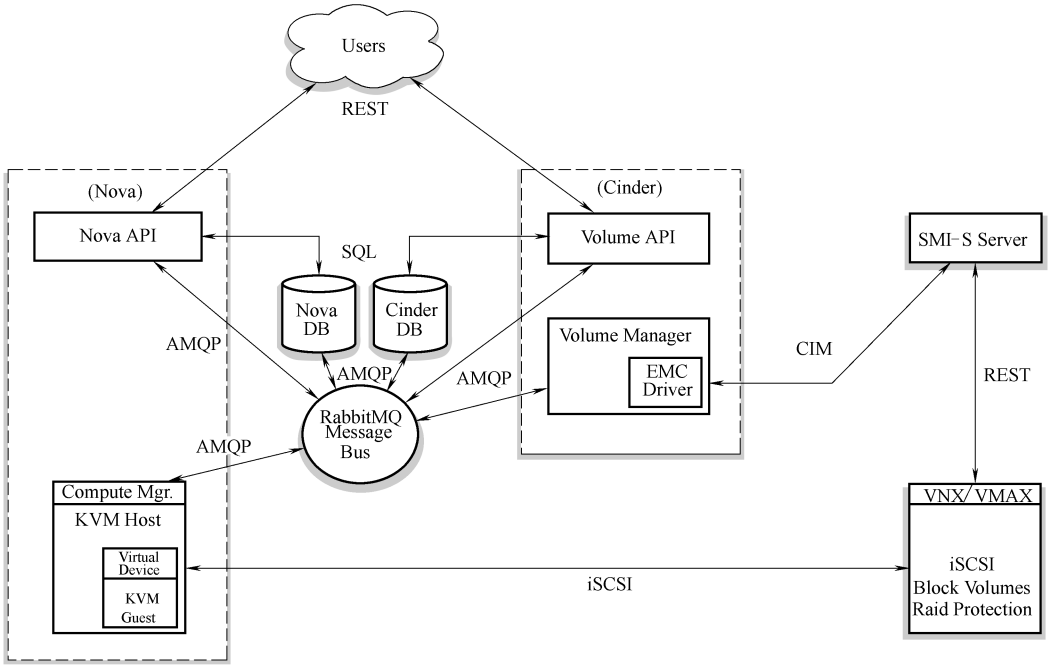


图 4-7　EMC 块存储架构

注：Users：用户；Nova：计算服务组件；API：应用程序编程接口；SQL：Structured Query Language，结构化查询语言；Nova DB：计算服务组件数据库；Cinder：块存储管理组件；Cinder DB：块存储管理组件数据库；AMQP：Advanced Message Queuing Protocol，提供统一消息服务的应用层标准高级消息队列协议；RabbitMQ：一款消息队列软件；Message Bus：消息总线；Volume：卷；Volume Manager：卷管理；EMC driver：EMC 设备驱动；CIM：公共信息模型；SMI-S Server：存储管理主动规范服务器；REST：Representational State Transfer，表述性状态传递；一种针对网络应用的设计和开发方式，可以降低开发的复杂性，提高系统的可伸缩性；Compute Mgr.：计算管理；KVM Host：KVM 宿主机；Virtual Device：虚拟设备；KVM Guest：KVM 客户机；VNX/VMAX：EMC 存储产品品牌；Block：块；Volume：卷；Raid Protection：磁盘阵列保护。

3. 对象存储（Swift）

Swift 是一个高可用分布式对象存储服务，通过配置普通硬盘的标准服务器提供可伸缩的冗余存储集群。Swift 并不代表一个文件系统，它实现的是一个更传统的对象存储系统，可用于主要是静态数据（例如，VM 映像、备份和归档以及较小的文件）的长期存储。

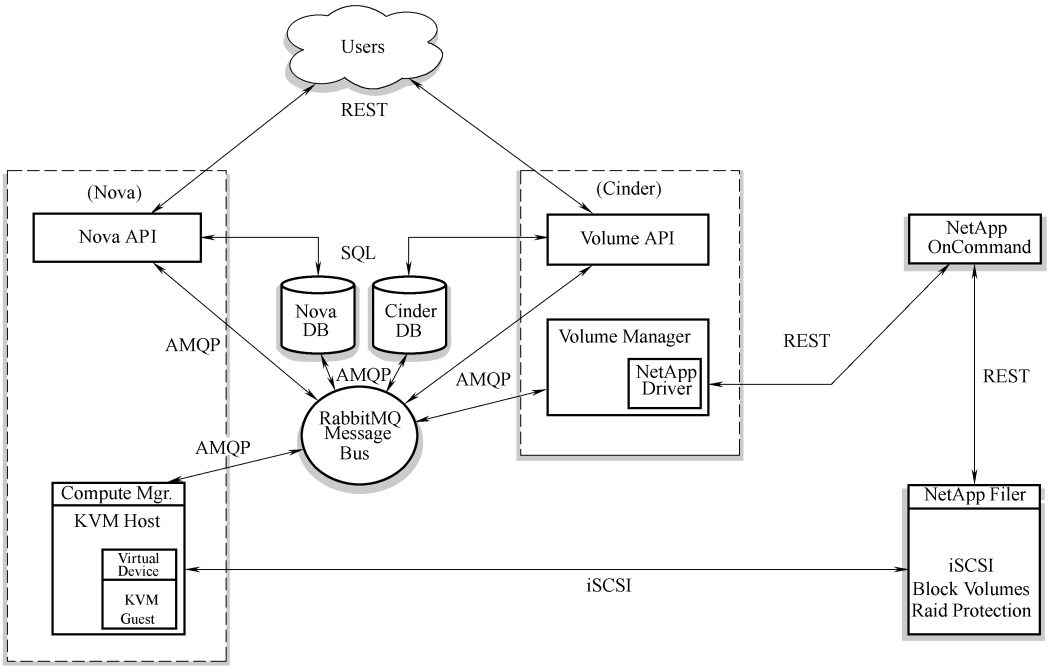


图 4-8　NetApp 块存储架构

注：Users：用户；Nova：计算服务组件；API ：应用程序编程接口；SQL：Structured Query Language，结构化查询语言；Nova DB：计算服务组件数据库；Cinder：块存储管理组件；Cinder DB：块存储管理组件数据库；AMQP：Advanced Message Queuing Protocol，提供统一消息服务的应用层标准高级消息队列协议；RabbitMQ：一款消息队列软件；Message Bus：消息总线；Volume：卷；Volume Manager：卷管理；NetApp driver：NetApp 设备驱动；NetApp Oncommand：NetApp 数据管理软件；REST：Representational State Transfer，表述性状态传递；一种针对网络应用的设计和开发方式，可以降低开发的复杂性，提高系统的可伸缩性；Compute Mgr. ：计算管理；KVM Host：KVM 宿主机；Virtual Device：虚拟设备；KVM Guest：KVM 客户机；NetApp Filer：NetApp 文件管理器；Block：块；Volume：卷；Raid Protection：磁盘阵列保护。

（1）数据模型　Swift 采用层次数据模型，共有三层逻辑结构：Account、Container 和 Object（即账户、容器和对象），每层节点数均没有限制，可以任意扩展。这里的账户和个人账户不是一个概念，可理解为租户，用来做顶层的隔离机制，可以被多个个人账户所共同使用；容器代表封装一组对象，类似文件夹或目录；叶子节点代表对象，由元数据和内容两部分组成。Swift 逻辑结构如图 4-10 所示。

（2）系统架构　Swift 采用完全对称、面向资源的分布式系统架构设计，所有组件都可扩展，避免因单点失效而扩散并影响整个系统运转；通信方式采用非阻塞式 I/O 模式，提高了系统吞吐和响应能力。Swift 系统架构如图 4-11 所示。

1）代理服务（Proxy Server）：Swift 通过代理服务（Proxy Server）向外提供基于 HTTP 的 REST 服务接口，会根据环的信息来查找服务地址并转发用户请求至相应的账户、容器或者对象，进行 CRUD（增、删、改、查）等操作。由于采用无状态的 REST 请求协议，可以进行横向扩展来均衡负载。在访问 Swift 服务之前，需要先通过认证服务获取访问令牌，然后在发送的请求中加入头部信息 X-Auth-Token。代理服务器负责

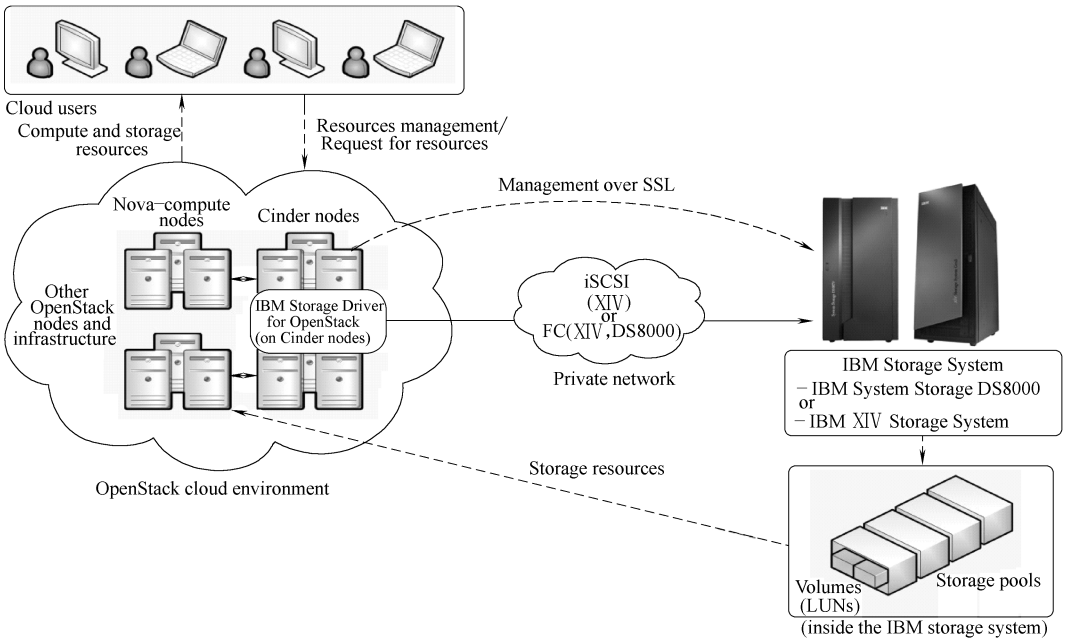


图 4-9　IBM 块存储架构

注：Cloud users：云用户；Compute and storage resources：计算和存储资源；Resources management/Request for resources：资源管理、资源请求；Nodes：节点；Other openstack nodes and infrastructure：其他 openstack 节点和基础设施；Openstack cloud environment：Openstack 云环境；IBM Storage Driver for Openstack（on Cinder nodes）Openstack 的 IBM 存储驱动（部署在 cinder 节点）；Management over SSL：通过 SSL（Secure Sockets Layer，安全套接层）管理；iSCSI：Internet Small Computer System Interface，Internet 小型计算机系统接口；FC：光纤通道协议；XIV，DS8000：IBM 存储产品；Private network：私有网络；Stroage resources：存储资源；IBM Storage System：IBM 存储系统；IBM System Storage DS8000：IBM DS8000 存储系统；IBM XIV Storage System：IBM XIV 存储系统；Volumes（LUNs）：卷（逻辑盘）；Storage pools：存储池；inside the IBM storage system：IBM 存储系统内部。

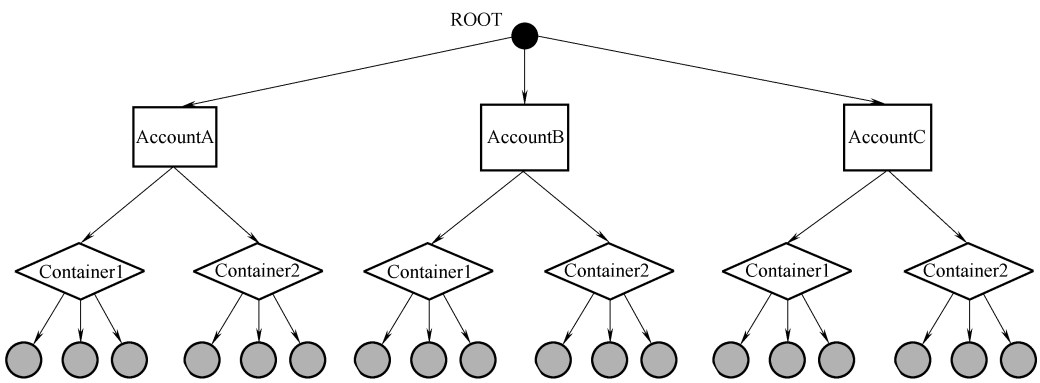


图 4-10　Swift 逻辑结构

注：ROOT：根；Account：账户；Container：容器。

Swift 架构的其余组件间的相互通信。代理服务器也处理大量的失败请求。例如，如果对某个对象 PUT 请求，某个存储节点不可用，它将会查询环中可传送的服务器并转发

请求。对象以流的形式到达（来自）对象服务器，它们直接从代理服务器传送到（来自）用户-代理服务器并不缓冲它们。

2）认证服务（Authentication Server）：验证访问用户的身份信息，并获得一个对象访问令牌（Token），在一定的时间内会一直有效；验证访问令牌的有效性并缓存下来直至过期时间。

3）缓存服务（Cache Server）：缓存的内容包括对象服务令牌，账户和容器的存在信息，但不会缓存对象本身的数据；缓存服务可采用 Memcached 集群，Swift 会使用一致性哈希算法来分配缓存地址。

4）账户服务（Account Server）：提供账户元数据和统计信息，并维护所含容器列表的服务，每个账户的信息被存储在一个 SQLite（一款轻型数据库）数据库中。

5）容器服务（Container Server）：提供容器元数据和统计信息（例如，对象的总数、容器的使用情况等），并维护所含对象列表的服务。容器服务并不知道对象存储在哪儿，只知道指定容器里存储了哪些对象。这些对象信息以 SQLite 数据库文件的形式进行存储，和对象一样在集群上做类似的备份。

6）对象服务（Object Server）：提供对象元数据和内容服务，可以用来存储、检索和删除本地设备上的对象。在文件系统中，对象以二进制文件的形式存储，它的元数据存储在文件系统的扩展属性（xattr）中，建议采用默认支持扩展属性（xattr）的 XFS 文件系统。每个对象使用对象名称的哈希值和操作的时间戳组成的路径来存储。最后一次写操作总可以成功，并确保最新一次的对象版本将会被处理。删除也被视为文件的一个版本（一个以“.ts”结尾的 0 字节文件，ts 表示墓碑）。

7）复制服务（Replicator）：检测本地分区副本和远程副本是否一致，具体是通过对比哈希文件和高级水印来完成，发现不一致时会采用推送（Push）更新远程副本：对于对象的复制，更新只是使用 rsync 同步文件到对等节点。账号和容器的复制通过 HTTP 或 rsync 来推送整个数据库文件上丢失的记录。另外一个任务是确保被标记删除的对象从文件系统中移除：当有一项（对象、容器或者账号）被删除，则一个墓碑文件被设置为该项的最新版本。复制器将会检测到该墓碑文件并确保将它从整个系统中移除。

8）更新服务（Updater）：当对象由于高负载或者系统故障等原因而无法立即更新时，任务将会被序列化到本地文件系统中进行排队，以便服务恢复后进行异步更新；例如，成功创建对象后容器服务器没有及时更新对象列表，这时容器的更新操作就会进入排队中，更新服务会在系统恢复正常后扫描队列并进行相应的更新处理。

9）审计服务（Auditor）：在本地服务器上会反复地爬取来检查对象、容器和账户的完整性，如果发现比特级的错误，文件将被隔离，并复制其他副本以覆盖本地损坏的副本；其他类型的错误（例如，在任何一个容器服务器中都找不到所需的对象列表）会被记录到日志中。

10）账户清理服务（Account Reaper）：移除被标记为删除的账户，删除其所包含的所有容器和对象。删除账号的过程是相当直接的。对于每个账号中的容器，每个对象

先被删除然后容器被删除。任何失败的删除请求将不会阻止整个过程，但是将会导致整个过程最终失败（例如，如果对一个对象的删除过程发生超时，容器将不能被删除，因此账号也不能被删除）。整个处理过程即使遭遇失败也要继续执行，这样它不会因为一个麻烦的问题而中止恢复集群空间。账号收割器将会继续不断地尝试删除账号直到它最终变为空，此时数据库在 db_ replicator 中回收处理，最终移除这个数据库文件。

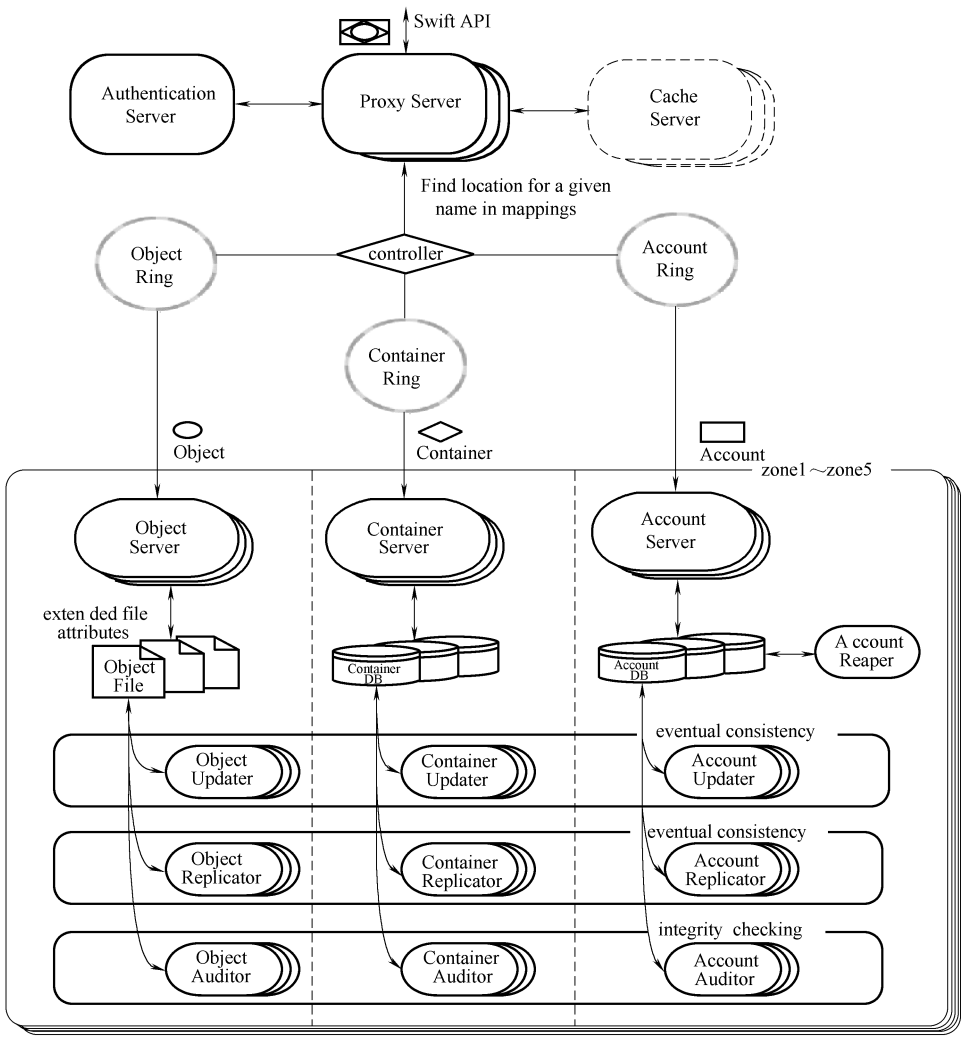


图 4-11　Swift 系统架构

注：Swift API：Swift 应用程序编程接口；Authentication Server ：认证服务器；Proxy Server：代理服务器；Cache Server：缓存服务器；Ring：Swift 中最重要的组件，用于记录存储对象与物理位置间映射关系；Objects：对象；Controller：控制器；Find location for a given name in mappings：通过映射关系找到给定名称所在的位置；Account：账户；Container：容器；Objects Server：对象服务；Account Server：账户服务；Container Server：容器服务；extended file attributes：扩展性文件属性；Object file：对象文件；Container DB：容器数据库；Account DB：账户数据库；Account reaper：账户清理；Object Updater：对象更新；Object Replicator：对象复制；Object Auditor：对象审计；Container Updater：容器更新；Container Replicator：容器复制；Container Auditor：容器审计；Account Updater：账户更新；Account Replicator：账户复制；Account Auditor：账户审计。

4.2.1.4 网络服务

Neutron 是 OpenStack 中服务网络服务的组件，是一个定义良好的网络服务插件框架，用来调用网络 L2 ~ L7 层的不同实现，且对外提供北向应用程序接口（API）来提供虚拟网络服务。Neutron 技术架构如图 4-12 所示。Neutron 包含以下几个组件：

1）Neutron-API：核心 API 实现核心功能（例如，L2 网络、IPAM 等），扩展 API 实现附加的网络服务。例如，L3 路由、负载均衡（Load Balanceing）、防火墙（Firewall）、虚拟专用网络（VPN）。

2）Neutron-Server：后台进程，将用户请求从 OpenStack Networking API 中继到配置的插件。

3）Neutron 代理：

① Neutron-dhcp-agent：向所有租户网络提供动态主机配置协议（DHCP）服务。

② Neutron-l3-agent：执行 L3/网络地址转换（NAT）转发，以支持网络访问租户网络上的 VM。

③ 一个特定于插件的可选代理（Neutron- * -agent）在每个虚拟机管理程序上执行本地虚拟交换机配置。

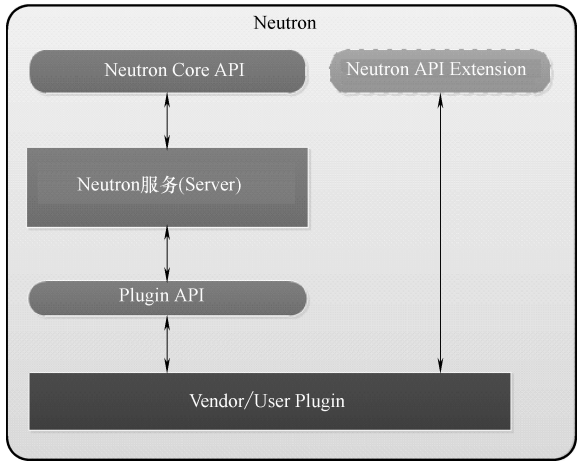


图 4-12 Neutron 技术架构

注：Heat：服务编排组件；Neutron：网络管理组件；Neutron Core API ：网络管理组件核心应用系统编程接口；Neutron API Extension：网络管理组件扩展应用系统编程接口；Neutron 服务（Server）：网络管理组件服务；Plugin API ：插件接口；Vendor/User Plugin ：供应商/用户插件。

最初的 OpenStack Compute 网络实现采用了一种基本模型，通过 Linux® VLAN 和 IP 表执行所有隔离操作。OpenStack Networking 引入了插件的概念，插件是 OpenStack Networking API 的一种后端实现。插件可使用各种不同的技术来实现逻辑 API 请求。Neutron 支持主流的各种网络产品：

① Open vSwitch Plugin。

② Cisco UCS/Nexus Plugin。

③ Linux Bridge Plugin。

④ Modular Layer 2 Plugin。

⑤ Nicira Network Virtualization Platform（NVP）Plugin。

⑥ RyuOpenFlow Controller Plugin。

⑦ NEC OpenFlow Plugin。

⑧ Big Switch Controller Plugin。

⑨ Cloudbase Hyper-V Plugin。

⑩ MidoNet Plugin。

⑪ Brocade Neutron Plugin Brocade Neutron Plugin。

⑫ PLUMgrid Plugin。

⑬ Mellanox Neutron Plugin Mellanox Neutron Plugin。

⑭ Embrane Neutron Plugin。

⑮ IBM SDN-VE Plugin。

⑯ CPLANE NETWORKS CPLANE NETWORKS。

⑰ Nuage Networks Plugin。

⑱ OpenContrail Plugin。

⑲ Extreme Networks Plugin。

⑳ Ruijie Networks Plugin。

㉑ Juniper Networks Neutron Plugin。

4.2.1.5 OpenStack 其他组件

1. Keystone

Keystone 为 OpenStack 框架提供注册服务，包含身份验证、服务规则和服务令牌功能。每类服务需要先通过 keystone 注册其服务的 Endpoint（服务访问的 URL），任何服务之间相互的调用，在通信前需要先经过 Keystone 的身份验证获得目标服务 Endpoint。Keystone 功能如图 4-13 所示，Keystone 认证流程如图 4-14 所示。

Keystone 服务包括：

1）Identity 服务：验证了身份验证凭证，并提供了所有相关的元数据。

2）Token 服务：验证并管理用于验证请求身份的令牌。

3）Catalog 服务：提供可用于端点发现的服务注册表。

4）Policy 服务：暴露一个基于规则的身份验证引擎。

2. Ceilometer

Ceilometer 负责收集 OpenStack 内部发生的事件，为计费和监控以及其他服务提供数据支撑。Ceilometer 处理流程如图 4-15 所示。Ceilometer 包含以下几个重要组件：

1）Compute Agent：该组件用来收集计算节点上的信息，在每一个计算节点上都要运行一个 Compute Agent，该 Agent 通过 Stevedore（python 动态扩展包，用于插件管理）管理一组采集插件，分别用来获取虚拟机的 CPU、Disk I/O、Network I/O、实例等信息。值得一提的是，这些信息大部分是通过调用 Hypervisor 的 API 来获取的。目前，

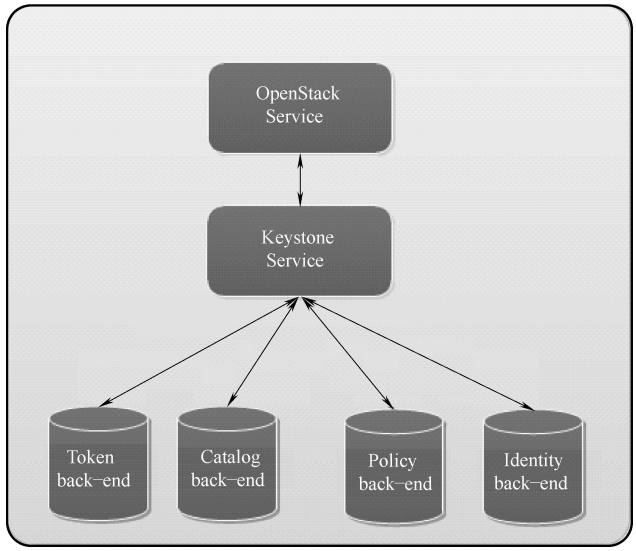


图 4-13　Keystone 功能

注：Keystone：身份认证和授权组件；OpenStack Service：OpenStack 服务；Token back-end：令牌后端；Catalog back-end：目录后端；Policy back-end：策略后端；Identity back-end：身份后端。

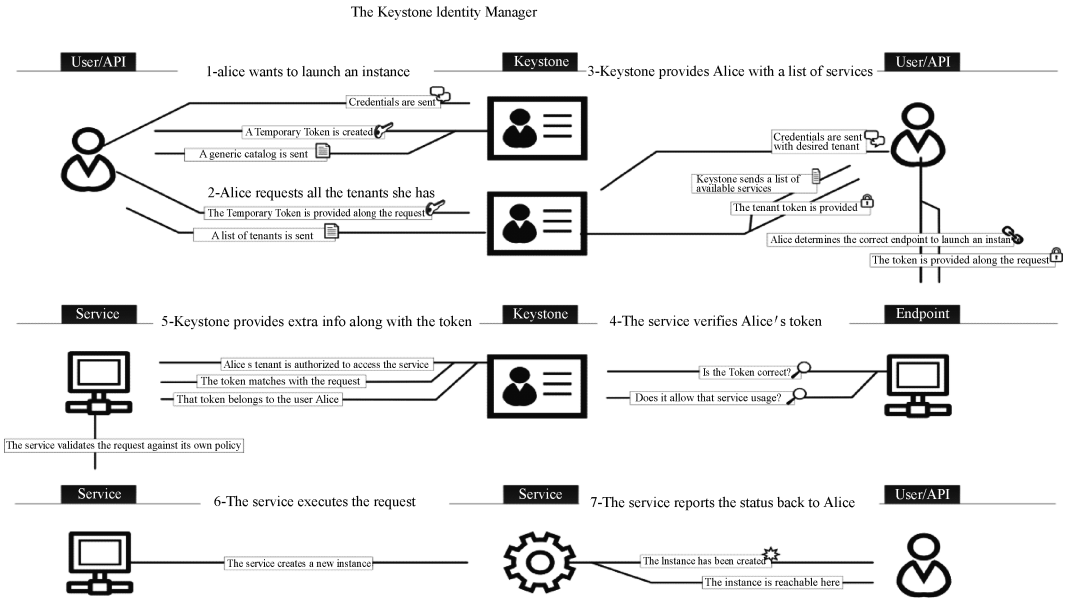


图 4-14　Keystone 认证流程

注：The keystone identity manager：身份认证管理；Alice wants to launch an instance：爱丽丝想启动一个实例；Alice requests all the tenants she has：爱丽丝要求所有她的房客；Keystone provides alice with a list of services：身份认证组件提供爱丽丝服务列表；The service verifles alice's token：该服务核查爱丽丝的令牌；Keystone provides extra info along with the token：身份认证组件提供令牌额外信息；The service executes the request：该服务执行请求；The service reports the status back to alice：该服务报告状态给爱丽丝。

Ceilometer 仅提供了 Libvirt（实现 Linux 虚拟化功能的 API）的 API。

2）Central Agent：Central Agent 运行在控制节点上，它主要收集其他服务（Image、

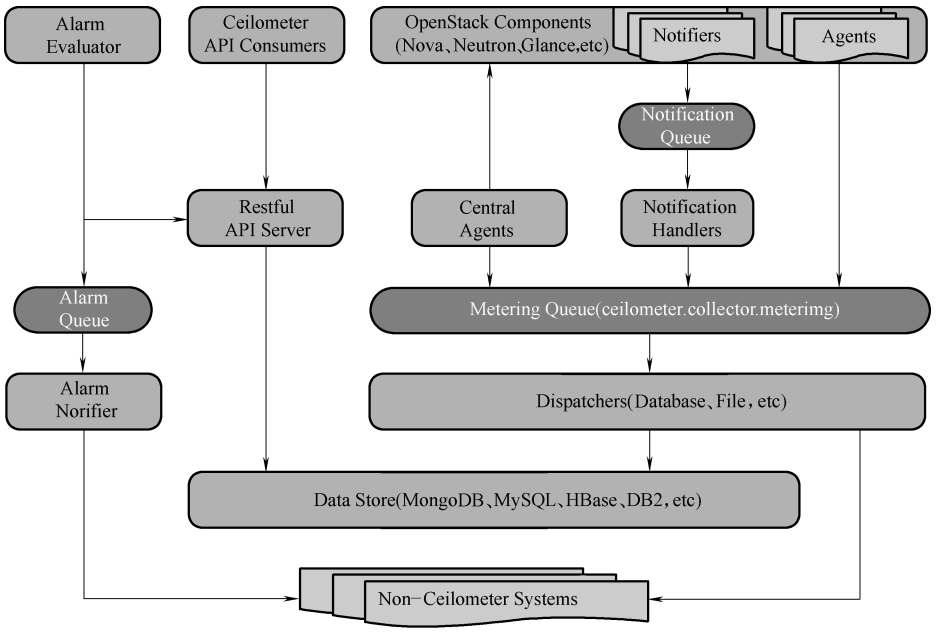


图 4-15　Ceilometer-处理流程

注：Alarm Evaluator：告警者；Ceilometer API Consumers：监控计量组件接口消费者；Openstack Components（Nova、Neutron、Glance，etc）：Openstack 的各个组件；Notifiers：通知者；Agent：代理；Alarm Queue：告警队列；Restful API server ：服务；Central Agents：中央代理；Notification Handlers：通知处理；Metering Queue：计量队列；ceilometer. collector. metering：监听消息队列；Alarm Notifier：报警通知；Dispatchers（Database、File，etc）：调度员（数据库、文件等）；Data store（MongoDB、MySQL、HBase、DB2，etc）：数据存储（MongoDB、MySQL、HBase、DB2 等）；Non-ceilometer Systems：非监控计量组件系统。

Volume、Objects、Network）的信息，实现逻辑和 Compute Agent 类似，通过调用这些服务的 REST API 可获取这些数据。

3）Collector ：是 Ceilometer 最为核心的组件，它的主要作用是监听 Message Bus，将收到的消息以及相应的数据写入到数据库中，它是在核心架构中唯一一个能够对数据库进行写操作的组件。除此之外，它的另一个作用是对收到的其他服务发来的通知消息进行本地化处理，然后再重新发送到消息总线中去，随后再被其收集。

4）Storage 是 Ceilometer 的数据存储服务，数据存储现在支持 MongoDB、MySQL、Postgresql、Hbase、DB2 等。

5）REST API：是 Ceilometer 对外服务接口。

6）Message Bus：是整个数据流的瓶颈，所有的数据都要经过消息总线被 Collector 收集而存储到数据库中，目前消息总线采用 RabbitMQ 实现。

7）Pipeline：它虽然不是其中一个组件，但是也是一个重要的机制，是 Agent 和 Message Bus 以及外界之间的桥梁，Agent 将收集来的数据发送到 Pipeline 中。在 Pipeline 中，先经过一组 transformer 的处理，然后通过 Multi Publisher 发送出去，可以通过消息总线（Message Bus）发送到 Collector，或者是发送到其他地方。Pipeline 是可配置的，Agent poll 数据的时间间隔，以及 Transformers 和 Publishers 都是通 Pipeline. yaml 文件进

行配置的。

4.2.2 CloudStack 架构

CloudStack 是一个开源的具有高可用性及扩展性的云计算平台，由世界最大的开源组织 Apache 基金会管理，它提供了对云计算资源的灵活部署与管理能力。目前，Cloudstack 支持管理大部分主流的 Hypervisor，如 KVM、XenServer、VMware、Oracle VM、Xen 等。使用 CloudStack 作为基础，数据中心操作者可以快速方便地通过现有基础架构创建云服务。现在，国外已经有多个用 CloudStack 部署的大规模云环境，包括英特尔、阿尔卡特-朗讯、Juniper、韩国电信、日本 KDDI、NTT、塔塔通信、迪斯尼、Zynga 等。

4.2.2.1 总体架构

CloudStack 架构设计上采用了典型的分层结构，即客户端、核心引擎和资源层。它面向各类型的客户提供了不同的访问方式，即 Web Console、Command Shell 和 Web Service API。通过它们用户可以管理使用在其底层的计算资源、网络资源和存储资源。CloudStack 概念架构如图 4-16 所示。

CloudStack 云基础架构在内部组织采用层次结构和现实物理环境进行映射。图 4-17 清楚展示了 Cloud 云基础架构的组成。

1. 资源域（Zone）

CloudStack 可包含一个或多个可用资源域（Zone），Zone 往往代表一个数据中心或者机房，是 CloudStack 系统中逻辑上最大范围的组织单元，由一组提供点（Pod）、二级存储（Secondary Storage）以及网络架构配置构成。在一套 CloudStack 系统中可以添加多个资源域，资源域之间可以实现完全物理隔离，而且硬件资源、网络配置、虚拟机也都是独立的。一个资源域在建立时只能选择一种网络架构基本网络（Basic Zone）或高级网络（Advanced Zone），但如果整套系统有多个资源域，每个资源域可以选择不同的网络架构。根据这一特点，也就可以实现 CloudStack 对多个物理机房的统一管理。从业务需求来说，也可以在一个机房内划分出两个独立的资源域，提供给需要完全隔离开的两套系统使用。由于资源域之间是相互独立的，如果需要有通信，只可以在网络设备上配置打通资源域的公共网络。

如图 4-18 所示，一个 Zone 可以包含一个或多个 Pod。每个 Pod 包括一个或多个集群主机或者一个或多个主存储服务器。所有 Zone 中的机架所共享的二级存储。对每一个 Zone，管理员必须决定以下几项：

1）在 Zone 中配置 Pod 的数量。

2）每个 Pod 配置集群的数量。

3）每个集群中放置主机的数量。

4）每个集群中放置主存储服务器（Primary Storage Servers）的数量和存储服务器的总容量。

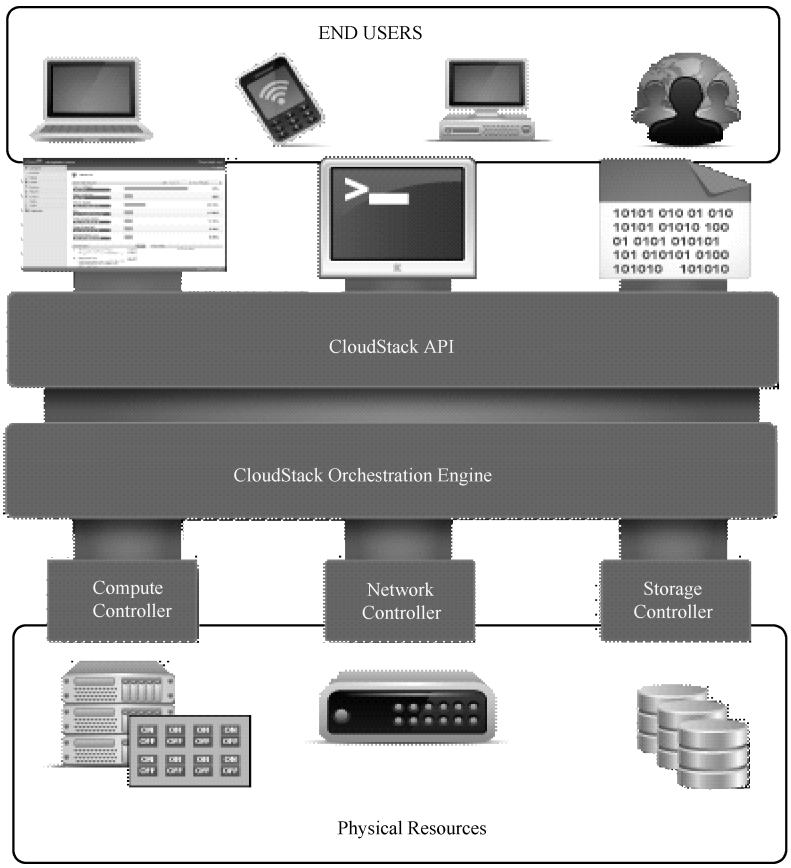


图 4-16　CloudStack 概念架构

注：END USERS：终端用户；CloudStack：一个开源的具有高可用性及扩展性的云计算平台；CloudStack API：CloudStack 应用系统编程接口；CloudStack Orchestration Engine：CloudStack 业务流程引擎；Compute Controller：计算控制器；Network Controller：网络控制器；Storage Controller：存储控制器。

5）每个区域中配置二级存储（Secondary Storage）的数量。

2. 提供点（Pod）

提供点（Pod）是 CloudStack 的资源域内第二级的逻辑组织单元，可以理解为一个物理的机架，包含交换机、服务器、存储的整合。位于同一个 Pod 内的计算服务器、系统虚拟机、客户虚拟机都在同一个子网中。一般来说，Pod 上的服务器连接至同一个或一组 Layer2 交换机上，所以很多实际部署中基本也都是以一个物理机架来进行规划的。一个资源域内可以有多个独立的提供点，在数量上没有限制。一个提供点可以由一个或多个集群构成，包含一个或多个主存储服务器（Primary Storage Servers）。出于对网络灵活扩展的目的，提供点是不可或缺的一个层级。Pod 对终端用户是不可见的。Pod 结构如图 4-19 所示。

3. 集群（Cluster）

集群是 CloudStack 的系统中最小的逻辑组织单元，由一组计算服务器和一个或多个主存储所组成。同一个集群的计算服务器必须使用相同的 Hypervisor 类型，硬件型号也

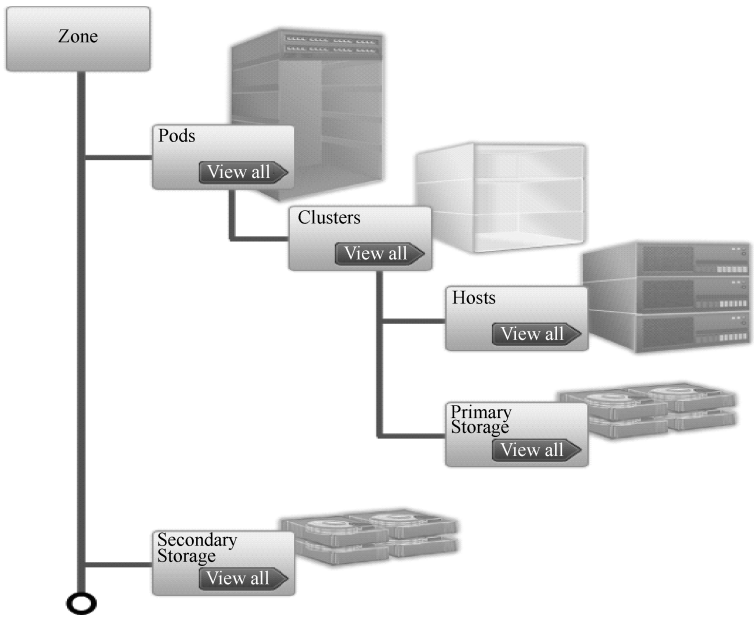


图 4-17　Cloud 云基础架构的组成

注：Zone：资源域；Pods：提供点；View all：查看全部；Clusters：集群；Hosts：主机；Primary Storage：主存储；Secondary Storage：二级存储

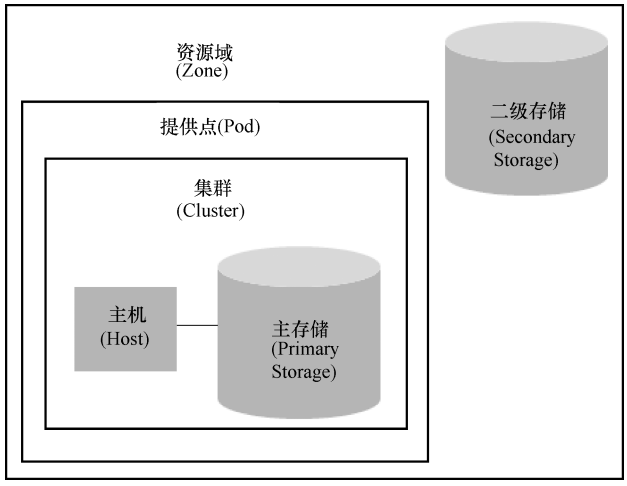


图 4-18　资源域（Zone）结构

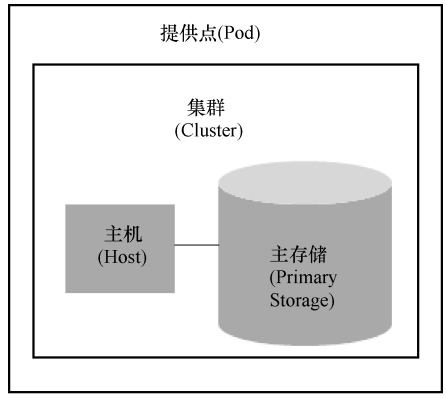


图 4-19　提供点（Pod）结构

必须相同。虚拟机可以在集群内的不同主机之间实现动态迁移（live migrate）。一个 Pod 可以包含多个集群，也可以包含使用不同 Hypervisor 程序的集群。虽然 CloudStack 并不限制集群的数量，但由于提供点所划分的子网范围，提供点内的集群和主机数量不会是完全无限制的。而集群内主机的数量，虽然也没有限制，但实践表明，一个集群内的计算服务器的数量建议不超过 16 台。

集群内可以添加多个作为共享存储所使用的主存储（primary storage），主存储的类

型没有限制，只要可以和计算服务器正常通信即可。在新版本中，CloudStack 可以实现虚拟机的所使用的镜像文件在多个主存储之间进行迁移，但肯定需要关闭虚拟机电源进行迁移。Cluster 结构如图 4-20 所示。

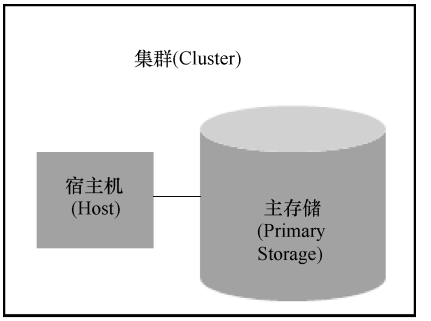


图 4-20　集群（Cluster）结构

4. 宿主机（Host）

宿主机是 CloudStack 系统中最基本的硬件模块之一，用于提供虚拟化能力和计算资源，并运行客户创建的虚拟机，根据系统压力可以进行弹性增减。计算服务器上需要安装 Hypervisor 程序，目前 CloudStack 支持 CitrixXenServer、VMware ESXi、KVM、Oracle VM 的 Hypervisor 等。CloudStack 环境中的宿主机的特点如下：

1）提供虚拟机需要的所有 CPU、内存、存储和网络资源。

2）用高带宽的网络互联互通，并具备 Internet 连接功能。

3）可以位于不同地理位置的不同数据中心。

4）可以拥有不同的规格（CPU、内存），虽然位于一个集群中的主机必须是同质的。

5）CloudStack 可以自动地发现宿主机提供的 CPU 数量和内存资源。

6）在 CloudStack 系统中运行一个宿主机时，必须在宿主机上配置虚拟机管理软件，分配 IP 地址给宿主机，并确保宿主机已经链接到 CloudStack 管理服务器。

5. 主存储（Primary Storage）

主存储一般作为每个集群中多台计算服务器共同使用的共享存储存在，一个集群中可以有一个或者多个不同类型的存储，也可以通过参数进行配置，使用计算节点的本地磁盘作为本地存储使用。主存储用于存储所有虚拟机的镜像文件和数据卷文件。集群中的所有计算节点都可以访问共享存储，用以实现虚拟机的在线迁移和高可用的功能。如果将本地磁盘作为主存储，虚拟机的磁盘读写具有很好的性能，但无法解决主机故障导致虚拟机无法启动而出现的单点故障。

CloudStack 的主存储支持的设备依赖于计算节点所使用的 Hypervisor 程序，如 XenServer 和 vSphere 可以全面的支持 NFS、iSCSI、FC-SAN 等传统存储设备。对于 KVM，支持 NFS 较容易，但需要使用 SheardMountPoint 功能来间接支持 iSCSI、FC-SAN、CLVM，并可以间接支持较新的分布式存储技术架构。

6. 二级存储（Secondary Storage）

二级存储是 CloudStack 根据 IaaS 平台的特点，专门设计出来的存储。每个资源域只需一个二级存储，用于存放创建虚拟机用的 ISO 镜像文件、模板文件，以及对虚拟机做的快照文件。二级存储可以支持 NFS 存储和 Openstack 的组件 Swift 存储。

4.2.2.2 计算架构

CloudStack 使用管理服务器管理云环境下的主机和虚拟机，管理服务器是无界的，可以是物理服务器或者是虚拟机。单个管理服务器可管理五千个主机，如果需要更好的扩展性，可采用多点部署。单个管理服务器可管理多个 Zone。如果虚拟机使用 Xen 和 KVM，需要安装 CloudStack Agent 来支持其与管理服务器的交互。而管理服务器与 Xen Server 交互则是依靠 XAPI，与 vCenter、ESX 交互依靠 HTTP。CloudStack 计算架构如图 4-21所示。

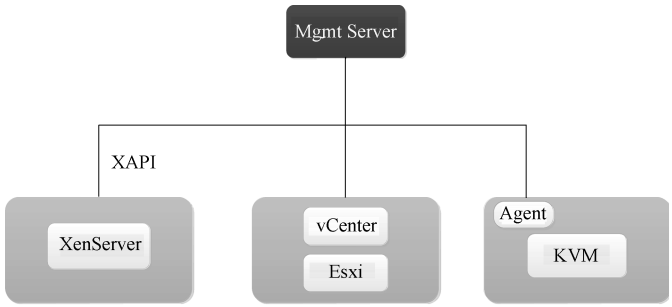


图 4-21 CloudStack 计算架构

注：Mgmt Server：管理服务器；XAPI：Xen 应用编程接口；XenServer：Xen 服务器；vCenter：vmware 出品的 Esxi 管理中心；Esxi：X86 裸机虚拟化技术软件；Agent：代理；KVM：目前已成为学术界的主流 VMM 之一。

4.2.2.3 网络架构

CloudStack 根据不同的数据流量类型设计了公共、管理、客户及存储网络，可以简称为 PMGS（Public、Management、Guest、Storage）网络。

（1）公共网络 当虚拟机访问 Internet 或外部网络时，需要通过公共网络，这说明客户虚拟机必须被分配某种形式的外网 IP，用户可以在 CloudStack 的 UI 上获得一个 IP 作为 NAT 映射，也可以在客户与公共网络之间做负载均衡。所有的 Hypervisor 都需要共享 Public VLAN 以保证虚拟机对外部网络的访问。

（2）管理网络 CloudStack 内部资源相互通信会产生 Management 流量，这些流量包括管理服务器节点与 Hypervisor 集群之间的通信，与系统虚拟机之间的通信或与其他组件之间的通信等；集群规模较小时管理流量只占用很少的带宽。

（3）客户网络 最终用户运行 CloudStack 创建的虚拟机实例时产生 Guest 流量，虚拟机实例之间通过客户网络相互通信。

（4）存储网络 主存储与 Hypervisor 之间互连互通的流量；主存储与二级存储之间也会产生存储流量，例如虚拟机模板和快照的搬移。

CloudStack 支持两种网络架构，即基本网络架构和高级网络架构。基本网络架构类似 AWS 的扁平网络模式，适合大规模扩展，所有 VM 部署于同一子网中，通过安全组进行账户间的隔离（VMware 不支持），虚拟路由（Virtual Router）提供 DHCP 与 DNS

服务。高级网络架构支持 Isolated 与 Shared 两种虚拟来宾网络模式，虚拟机可接入多个网络，通过 VLAN 进行账户间的隔离，但 VLAN 数量受到限制（4096），虚拟网户提供了更多的网络服务，即 DHCP、DNS、NAT、Firewall、VPN、LoadBalancing、PortForwarding 等。其中，管理网络、客户网络、存储网络三种网络对于基本网络及高级网络通用，而 Public 则只是针对高级网络才存在。CloudStack 基本网络与高级网络的功能对比见表 4-3。

表 4-3　CloudStack 基本网络与高级网络的功能对比

网络功能	基本网络	高级网络
网络数量	单一网络	多种网络
防火墙类型	物理	物理和虚拟
	物理	物理和虚拟
隔离类型	三层	二层和三层
	否	是
	物理	物理和虚拟
	物理	物理和虚拟
Source NAT	否	物理和虚拟
	是	是
	在物理路由器上：sFlow / netFlow	Hypervisor 和虚拟路由器
	是	是

4.2.2.4　存储架构

CloudStack 定义了两种存储：主存储和二级存储。主存储可以使用 iSCSI 或 NFS 协议；另外，直接附加存储可被用于主存储。二级存储通常使用 NFS 协议。CloudStack 不支持临时存储，所有节点上的所有卷都是持久存储。

1. 主存储

主存储的速度会直接影响来宾虚拟机的性能。如果可能，为主存储选择容量小、转速高的硬盘或 SSDs。CloudStack 用两种方式使用主存储。

（1）静态　是 CloudStack 管理存储的传统方式。在这个模式下，要给 CloudStack 预先分配几个存储（比如一个 SAN 上的卷），然后 CloudStack 在上面创建若干个卷（可以是 root 和/或者数据盘）。如果使用这种技术，确保存储上没有数据，否则给 CloudStack 添加存储会销毁已存在的所有数据。

（2）动态　是一个比较新的 CloudStack 管理存储的方式。在这个模式中，给 CloudStack 使用的是一个存储系统（但不是预分配的存储）。CloudStack 配合存储一起工作，动态地在存储系统上创建卷并且存储系统上的每个卷都映射到一个 CloudStack 卷，这样非常有利于存储的 QoS，目前数据磁盘（磁盘方案）支持这个特性。CloudStack 存储支持见表 4-4。

表 4-4 CloudStack 存储支持

存储媒介\hypervisor	VMware vSphere	CitrixXenServer	KVM	Hyper-V
磁盘、模板和快照的格式	VMDK	VHD	QCOW2	不支持 VHD 快照
支持 iSCSI	VMFS	集群化的 LVM	支持,通过共享挂载点	否
支持 FC	VMFS	支持,通过已有的 SR	支持,通过共享挂载点	否
支持 NFS	是	是	是	否
支持本地存储	是	是	是	是
存储超配	NFS 和 iSCSI	NFS	NFS	否
SMB/CIFS	否	否	否	是

2. 二级存储

二级存储是和 Zone 关联的，存储的内容如下：

1）模板：可以用来启动虚拟机和包括附加配置信息（例如，已经安装的应用程序）的操作系统镜像。

2）ISO 镜像：包含数据或可引导操作系统媒介的磁盘镜像。

3）磁盘卷快照：可用来进行数据恢复或创建新模板的虚拟机数据的副本。

基于区域的 NFS 二级存储中的元素可以被区域中的所有主机使用。CloudStack 管理将客户虚拟机磁盘分配到特定的主存储设备上存储（所有虚拟机磁盘都存在主存储上）。除了基于区域的 NFS 二级存储之外，还可以添加 OpenStack 项目存储 Swift，以使二级存储中的元素对云中的所有主机都可用。当使用 Swift 时，要对整个 CloudStack 配置 Swift 存储，然后像往常一样配置，为每个 Zone 配置 NFS 二级存储。当所有的模板和其他二级存储转发到 Swift 之前，每个 Zone 中的 NFS 存储作为一个缓存区。这个 Swift 存储像一个位于云端的存储，使在云中的其他 Zone 可以使用模板或者其他数据。在 Swift 存储中没有层次结构，一个 Swift 容器包含一个存储对象。整个云中的任何二级存储在需要时可以从 Swift 拉一个容器，不用再将模板和快照从一个 Zone 复制到另一个 Zone，但是当只使用 Zone NFS 时复制是必需的。

4.3 PaaS 开源软件

4.3.1 Cloud Foundry 架构

Cloud Foundry 是由相对独立的多个模块构成的分布式系统，每个模块单独存在和运行，各模块之间通过消息机制进行通信。Cloud Foundry 各模块本身是基于 Ruby 语言开发的，每个部分可以认为拿来即可运行，不存在编译等过程。Cloud Foundry 云平台

整体逻辑分层架构如图 4-22 所示。

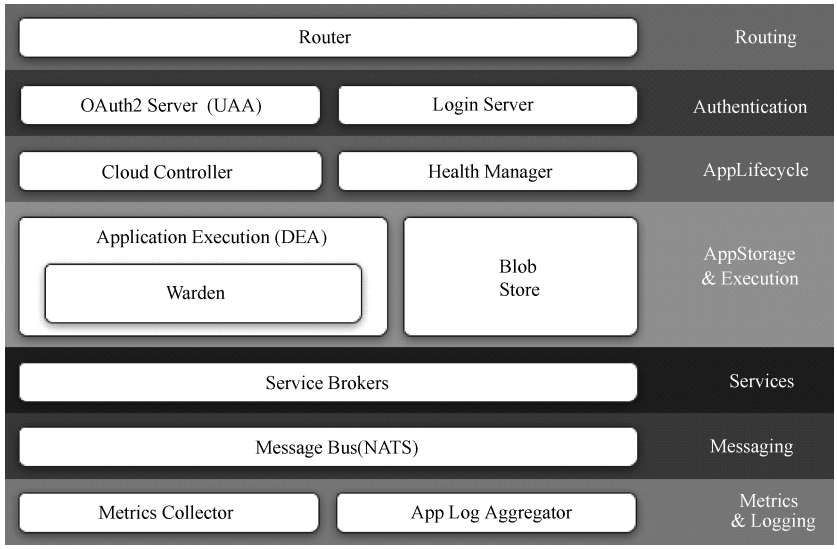


图 4-22　Cloud Foundry 云平台整体逻辑分层架构

注：Router：路由；Routing：按某线路发送；OAuth2 Server（UAA）：授权资源服务器；Login Server：登录服务器；Authentication：身份验证；Application Execution（DEA）：应用程序执行；Blob Store：Blob 存储；App Storage & Execution：应用存储与执行；Service Brokers：服务代理；Services：服务；Message Bus（NATS）：消息总线；Messaging：消息传递；Metrics Collector：度量收集器；App Log Aggregator：应用程序日志聚合；Metrics & Logging：度量和数据记录。

Cloud Foundry 云平台主要由路由器（Router）、认证服务（UAA）、登录服务（Login Server）、云控制器（Cloud Controller）、健康管理（Health Manager）、Droplet 执行代理（DEA）、Blob 存储（Blob Store）、消息总线（NATS）、云控制器数据中心（Cloud Controller Database）以及 Service 等模块组成。这些模块协同合作，通过特定的消息传输机制和 API 接口进行通信，就可以使整个云平台正常运行。由于在集群环境下每个模块都有多个部署节点，从而保证了云平台的可靠性和弹性动态扩展的需求，使得应用程序可以稳定可靠地运行在 Cloud Foundry 云平台上。

（1）路由器（Router）　路由器组件负责对 Cloud Foundry 所有进来的请求进行路由，进入 Cloud Foundry 系统的请求都会经过路由器组件。路由器组件支持扩展，可由多个路由器共同处理进来的请求。管理员可用 DNS 实现负载均衡，也可以部署专用硬件来实现，或者使用 Nginx 进行负载均衡。

（2）云控制器（Cloud Controller）　Cloud Foundry 的云控制器。负责与 VMC 命令行工具和 STS 插件交互的服务器端，它收到指令后发消息到各模块，管理整个云的运行，相当于 Cloud Foundry 的管理器，它的主要工作包括：

1）对应用程序的管理，即增加、修改或者删除。

2）应用程序的启动、停止。

3）把应用程序打包成一个 Droplet。

4）修改应用程序运行环境，包括实例、内存等。

5）管理服务，包括服务与应用的绑定等。

6）管理 Cloud Foundry 环境。

7）管理 Cloud Foundry 的用户信息。

8）查看 Cloud Foundry，以及每一个应用程序的 log 信息。

（3）认证服务（UAA） Cloud Foundry 的权限管理模块。认证服务负责实现用户认证，它可以与企业已有的认证系统进行整合，例如 LDAP、CAS 等。

（4）Droplet 执行代理（DEA） 它是 Droplet Execution Agent 的缩写，是 Cloud Foundry 的应用运行代理模块。一台虚拟机上会运行一个或多个 DEA。一个 DEA 可以启动多个 App（又称为 Droplet）。在 Cloud Foundry 的概念里面，Droplet 是指应用源代码、Cloud Foundry 配套好的运行环境，以及一些管理脚本全部压缩好在一起的 Tar 包。而制作这个 Tar 包并把它存储起来的过程叫做 Staging App，Cloud Foundry 会自动保存 Droplet。当用户启动一个 App 时，一台部署了 DEA 模块的服务器会来拿一个 Droplet 的拷贝去运行。所以如果用户将 App 扩展到 10 个实例时，那么这个 Droplet 就会被复制 10 份，并在 10 个 DEA 服务器上运行。

（5）健康管理（Health Manager） 负责从各个 DEA 获得运行信息，然后进行统计分析、报告、发出告警等。统计数据会与 Cloud Controller 的设定指标进行比对，并提供 Alert 等。在云计算里面，自动化 Health 管理、分析是一个很重要的领域，而这方面可以扩展的地方也很多，结合业务流程引擎可以使云自管理、自预警；而与 BI 方面的技术相结合，可以统计运营情况、合理分配资源等。

（6）服务（Services） 服务应属于 PaaS 的第三层。Cloud Foundry 将服务模块设计成一个独立的、插件式的模块，便于第三方服务提供商方便地将自己的服务整合成 Cloud Foundry 服务。从架构上来说，Cloud Foundry 服务部分使用了模板方法设计模式，可通过重构子方法来实现自己的服务。如果不需要特别逻辑则可以使用默认方法。现实情况是，由于种种原因使有些系统服务难以或不愿意迁移到云端，为此 Cloud Foundry 引入了 Service Broker 模块。Service Broker 可以使部署在 Cloud Foundry 上的应用能访问本地服务。

（7）消息总线（NATS） Cloud Foundry 的架构是基于消息发布和订阅机制。NATS 的组件负责各模块之间的交互。NATS 是由 Cloud Foundry 开发的一个基于事件驱动的、轻量级的消息系统，它基于 EventMachine 实现。Cloud Foundry 各种优秀特性均源于消息通信架构。每台服务器上的各模块会根据当前的行为，向对应主题发布消息，同时也按照需要监听多个主题，彼此以消息进行通信。

Cloud Foundry 能够部署在私有云或公有云环境中。它可以运行在多种 IaaS 之上，与 IaaS 的耦合性很低。目前，Cloud Foundry 支持的应用方式主要有以下三类：

（1）公有云服务平台 公有云服务平台是指目前 VMware 公司运营的一个免费公有云平台及其合作伙伴的 PaaS 云平台。当前 VMware 公司运维着一个公有 PaaS 云平台，该平台为开发者提供了一个简单的途径来试用 Cloud Foundry 云平台，并为新的服务和软件的运维优化提供一个测试平台。

（2）私有云服务平台　私有云服务平台是指在企业内部构建的云服务平台。VMware 提供商业版本的 Cloud Foundry 给要部署 PaaS 云平台的企业，帮助它们在自己的云中构建企业 PaaS 云平台。企业 PaaS 云平台在技术本质上是构建统一的企业 IT 基础架构，也就是将企业 IT 资源整合为服务，以供企业各部门和其他企业共享使用，从而提高企业 IT 资源的使用率。

（3）本地微云平台　本地微云平台 Micro Cloud Foundry 是指可以压缩云到开发者的笔记本上单个虚拟机里运行的测试 Cloud Foundry 云平台环境。通过微云平台，Cloud Foundry 提供了运行在单个虚拟机里的版本，可以帮助开发者在自己的机器上建立和测试他们的应用，确保开发环境和生产环境具有一致性。

Cloud Foundry 同时支持多种开发框架、编程语言、应用服务以及多种云部署环境。

1）开发框架的支持。Cloud Foundry 支持各种框架的灵活选择，这些框架包括 Spring for Java、. NET、Ruby on Rails、Node. js、Grails、Scala on Lift 等。

2）应用服务的支持。Cloud Foundry 云平台将应用和应用依赖的服务彼此分开，通过在部署时将应用和应用依赖的服务相绑定的机制使应用和应用服务相对对立，增加了在 PaaS 平台上部署应用的灵活性。这些应用服务包括 PostgreSQL、MySQL、SQL Server、MongoDB、Redis 以及更多来自第三方和开源社区的应用服务。

3）IaaS 的支持。Cloud Foundry 支持多种云基础设施，用户需要在不同的云服务器之间切换，而不用担心被厂商锁定的问题。Cloud Foundry 可以灵活地部署在公有云、私有云或者混合云之上，如 vSphere/vCloud、AWS、OpenStack、Rackspace 等多种云环境中。

4.3.2　OpenShift 架构

OpenShift 是红帽公司推出的一个云计算服务平台。用户可以创建、部署、管理云端应用，其云环境具体提供了磁盘空间、CPU 计算资源、内存资源、网络连接以及应用服务器。根据不同应用类型（数据库、编程语言等），OpenShift 会提供不同的文件系统布局（例如，PHP、Python、Ruby、Java）来创建不同的运行环境。此外，OpenShift 也提供了一定程度的 DNS（域别名）。

4.3.2.1　基本功能单元

Openshift 中包括了两个基本功能单元，如图 4-23 所示。

1. Broker（提供接口）

Broker 是所有应用管理活动的入口。它主要负责管理用户登录、DNS、应用状态以及应用服务编排（服务分发）。用户和 Broker 交互主要是通过 Web 管理控制台、CLI（Command line interface，命令行界面）工具、JBoss 工具或者 REST API。

2. Cartridges（提供应用框架）

Cartridges 为应用程序运行提供环境的插件，每个 Cartridge 只能提供一种运行环境，比如 Python 或者 Mysql，不能同时提供多种。Cartridge 分为两种：Framework Cartridge 和 Embedded Cartridge，前者提供 Web 能力的服务，后者提供数据库、数据库 Web 接口的

图 4-23　OpenShift 基本功能单元

注：User Interaction：用户交互界面；Broker：是所有应用管理活动的入口。它主要负责管理用户登录、DNS、应用状态以及应用服务编排；Cartridges：为应用程序运行提供环境的插件，每个 Cartridge 只能提供一种运行环境。

服务。一个应用程序显然需要至少一个 Framework Cartridge。Cartridges 提供了多种编程语言支持（PHP、Python、Java 等），还提供了不同数据库支持（PostgreSQL、MySQL、MongoDB）。

4.3.2.2　系统资源与应用容器

OpenShift 内部是通过 Gears、Nodes 以及 Districts 来管理系统资源与应用容器的，其组织结构如图 4-24 所示。

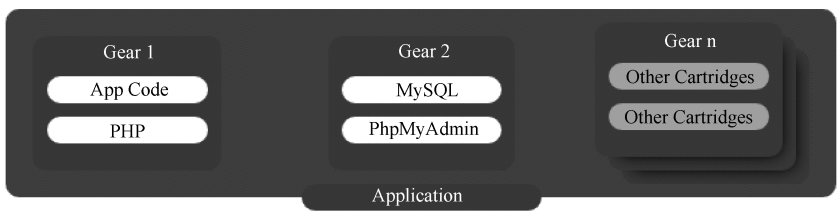


图 4-24　OpenShift 组织结构

注：Gear：拥有一系列软硬资源的容器；App Code：应用程序代码；MySQL：一个关系型数据库管理系统；Other Cartridges：其他运行环境；PHP：脚本语言；PhpMyAdmin ：MySQL 的 PHP 编写的管理器；Application：应用。

1. Gears

拥有一系列软硬资源的容器（沙箱/沙盒），提供了 Cartridges 运行的容器，一个或多个 Cartridges 可在其中运行，Gear 为每个 Cartridge 提供有限的内存与磁盘空间。

2. Nodes

一台物理机或虚拟机，其中包含多个 Gears。因为某些 Gear 并不是时时刻刻处于运行中，因此，一个 Node 通常会处于超配额状态，即放入了超过限额个数的 Gears。

3. Districts

定义了一些 nodes，其中的 Gears 可以方便地进行 Node 负载均衡。因此，即使是某个 Gear 负载很高时也不会发生 Node 运行超载。Districts 负载均衡如图 4-25 所示。

4.3.2.3　OpenShift 应用

OpenShift 应用管理有三种管理途径，即 Web、Command Line 和 Ide。Web 方式可以

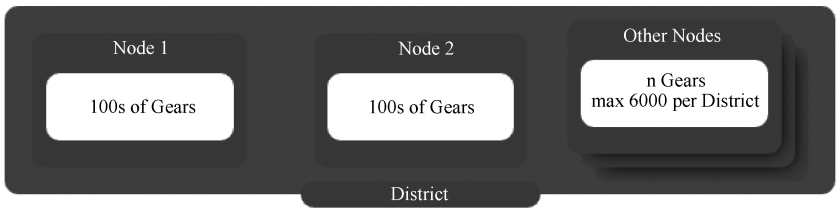


图 4-25　Districts 负载均衡

注：Node：一台物理机或虚拟机；Gears：拥有一系列软硬资源的容器；100s of Gears：100 个 Gears；Max 6000per District ：每个分区最大 6000 个 Gears；District：分区。

在浏览器上快速创建、运行应用。Command Line 方式拥有完全的控制权和代码管理。Ide 方式可以无缝地整合进 Eclipse 的开发环境中，如图 4-26 所示。

OpenShift 应用是由一系列部分构成的。

1）Namespace（用户空间）：每个用户只有一个 Namespace。

2）App Name（应用名）：每个 App 有一个 Unique（唯一）的名字。

3）Aliases（别名）：可以为一个 App 提供一个别名，也就是另外一个 URL。

4）App Dependencies（应用独立性）：App 依赖于哪些 Catridges。

5）App Git Repository（开源分布式版本控制系统）：用户把代码 Push（推送）上去的地方。

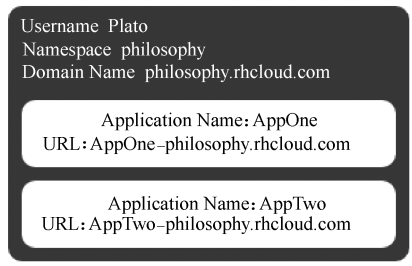


图 4-26　OpenShift 应用管理

注：Username ：用户名；Namespace：名字空间；Domain name：域名；Application Name：应用程序名称；URL：网址。

1. 伸缩性

OpenShift 上的应用可以分为两类：可伸缩应用与不可伸缩应用。

1）可伸缩应用：可伸缩应用按需获取系统资源。一个可伸缩应用至少会使用两个 Gears，一个用于应用本身，另一个用于高可用代理（HAProxy）实现负载均衡。

2）不可伸缩应用：只能使用一个 Gear。

Web 请求到达负载均衡代理 HAProxy 后，它将请求转发给 Gear 中的应用。当 HA-Proxy 探测到请求过载时，OpenShift 将复制一份已经存在的 Web Cartridge 到一个单独的 Gear 里，与已有 Gears 一起处理请求，这相当于提升了两倍请求处理能力。OpenShift 部署架构如图 4-27 所示。

2. 用户交互

通过 OpenShift 创建应用的过程如图 4-28 所示。

可以通过 git push 命令进行部署，也可以使用 Jenkins（Cartridge）进行持续集成，如图 4-29 所示。

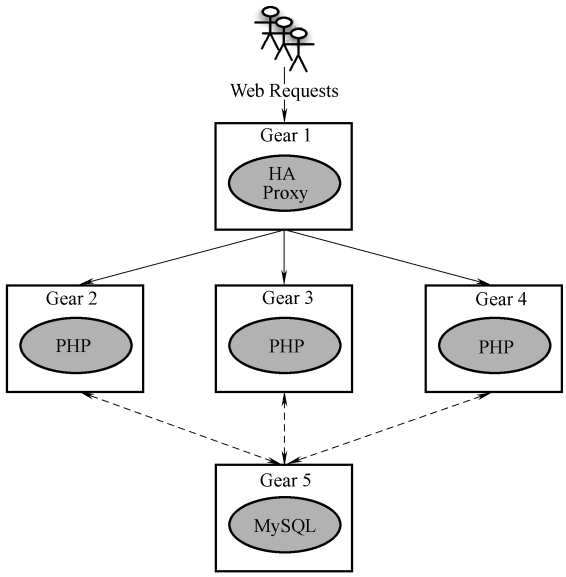


图 4-27　OpenShift 部署架构

注：Web requests ：Web 请求；HA Proxy：高可用性代理；Gear：拥有一系列软硬资源的容器；PHP：脚本语言；MySQL ：一种关系性数据库。

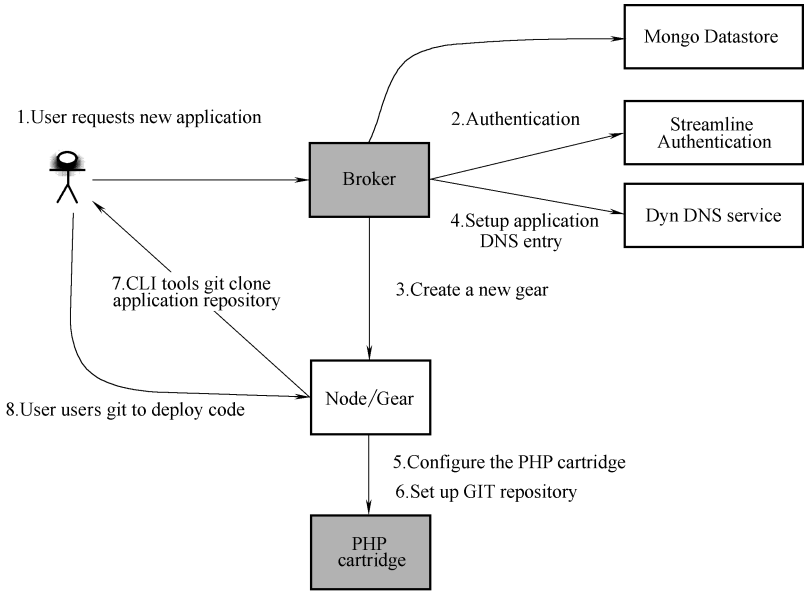


图 4-28　OpenShift 交互式创建应用过程

注：User requests new application：用户请求新的应用程序；Mongo：一种非关系型数据库；Mongo Datastore：mongo 数据存储；Broker：是所有应用管理活动的入口。它主要负责管理用户登录、DNS、应用状态以及应用服务编排（服务分发）；Authentication：认证；Streamline authentication：简化认证；DNS：Domain Name System，域名系统；Setup application DNS entry：安装应用程序 DNS 条目；Dyn DNS service ：动态 DNS 服务；CLI：command-line interface ，命令行界面；GIT：一个开源的分布式版本控制系统；CLI tools GIT clone application repository：CLI 工具的 GIT 克隆应用程序库；Create a new gear：创建一个新的容器；User users GIT to deploy code：使用用户的 GIT 部署代码；Configure the PHP cartridge：配置 PHP 运行环境；Set up GIT repository ：设置 GIT 仓库；PHP cartridge：PHP 运行环境。

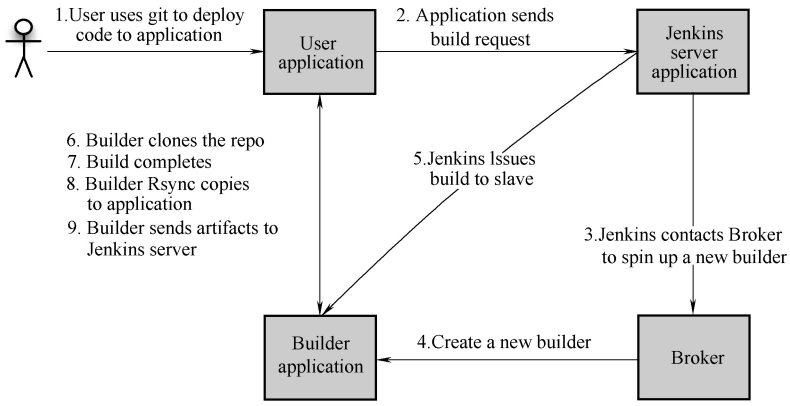


图 4-29　OpenShift 自动化创建应用过程

注：User uses git to deploy code to application：用户使用 git 部署代码；User application ：用户应用；Application sends build request：应用程序发送构建请求；Jenkins：一个开源软件项目，旨在提供一个开放易用的软件平台，使软件的持续集成变成可能；Jenkins server application：Jenkins 服务器应用程序；repo：管理多 git 的工具；Builder clones the repo：构建 repo 克隆；Build completes：构建完成；Builder Rsync copies to application：构建器同步应用程序副本；Builder sends artifacts to Jenkins server：构建器发送构件给 Jenkins 服务器；Jenkins issues build to slave：Jenkins 将应用构建至 slave；Broker：是所有应用管理活动的入口。它主要负责管理用户登录、DNS、应用状态以及应用服务编排（服务分发）；Jenkins contacts Broker to spin up a new bulder：Jenkins 联系 Broker 来启用新构建器；Builder application：构建应用程序:；Create a new builder：创建一个新构建器。

4.4　大数据技术

随着“按需服务”理念云计算的高速发展，“数据即资源”的大数据时代已经来临。大数据利用对数据处理的实时性、有效性提出了更高的要求，需要根据大数据的特点对传统的数据处理技术进行变革，以形成适用于大数据收集、存储、管理、分析、共享和可视化的技术。

大数据技术从大的方面可以分为两类，以 Hadoop 为代表的批处理，以及以 Storm、Spark 为代表的实时处理。

4.4.1　Hadoop

Hadoop 是一个分布式系统基础架构，由 Apache 基金会开发。用户可以在不了解分布式底层细节的情况下，开发分布式程序。充分利用集群的威力高速运算和存储。简单地说，Hadoop 是一个可以更容易开发和运行处理大规模数据的软件平台。

Hadoop 实现了一个分布式文件系统（Hadoop Distributed File System，HDFS）。HDFS 具有高容错性的特点，并且设计部署在低廉的硬件上。而且它提供高传输率来访

问应用程序的数据，适合超大数据集的应用程序。HDFS 放宽了 POSIX 的要求，这样可以以流的形式访问文件系统中的数据。下面列举了 Hadoop 主要的一些特点：

（1）扩容能力（Scalable） 能可靠地存储和处理千兆字节（PB）数据。

（2）成本低（Economical） 可以通过普通机器组成的服务器群来分发以及处理数据，这些服务器群总计可达数千个节点。

（3）高效率（Efficient） 通过分发数据，Hadoop 可以在数据所在的节点上并行处理它们，这使得处理非常的快速。

（4）可靠性（Reliable） Hadoop 能自动地维护数据的多份复制，并且在任务失败后能自动地重新部署计算任务。Hadoop 运行在商用独立的服务群集上。可以随时添加或删除 Hadoop 群集中的服务器。Hadoop 系统会检测和补偿任何服务器上出现的硬件或系统问题。换句话说，Hadoop 是一个自愈系统。在出现系统变化或故障时，它仍可以运行大规模的高性能处理任务，并提供数据。

虽然 Hadoop 提供了数据存储和并行处理平台，但其真正的价值来自于这项技术的添加件、交叉集成和定制实现。为此，Hadoop 还提供了向这一平台增加功能性和新能力的子项目，具体内容如下：

（1）Hadoop Common 支持其他 Hadoop 子项目的通用工具。

（2）Chukwa 管理大型分布式系统的数据采集系统。

（3）HBase 支持大型表格结构化数据存储的可伸缩、分布式数据库。

（4）HDFS 向应用数据提供高吞吐量访问的分布式文件系统。

（5）Hive 提供数据汇总和随机查询的数据仓库基础设施。

（6）MapReduce 用于对计算群集上的大型数据集合进行分布式处理的软件框架。

（7）Pig 用于并行计算的高级数据流语言和执行框架。

（8）ZooKeeper 用于分布式应用的高性能协调服务。

Hadoop 平台的多数实现至少包括其中的一些子项目，因为这些子项目常常在利用“大数据”时是所不可或缺的。例如，大多数机构会选择使用 HDFS 作为主分布式文件系统，选择可以保存几十亿行数据的 HBase（分布式开源数据库）作为数据库。而使用 MapReduce 则更是非常肯定的事情，因为其引擎赋予了 Hadoop 平台一定的速度和灵活性。

4.4.2 MapR

MapR 是 MapR Technologies Inc 的一个产品，号称下一代 Hadoop，使 Hadoop 变为一个速度更快、可靠性更高、更易于管理、使用更加方便的分布式计算服务和存储平台，同时性能也不断提升。它将极大地扩大了 Hadoop 的使用范围和方式。它包含了开源社区的许多流行的工具和功能，例如 Hbase、HIVE。它与 Apache Hadoopd 的 API 完全兼容。它能够为客户节约近 1/2 的硬件资源消耗，使更多的组织能够利用海量数据分析的力量提高竞争优势。

与 Hadoop 相比，MapR 有以下两个优势：

1. Direct Access（直接访问）NFS 技术

顾名思义，用户可以直接在远程通过 NFS 客户端把 MapR HDFS 装载到其本地，像操作本地文件一样来进行操作。同时，Direct Access NFS 支持文件的随机读写，大大地扩展了 MapRHadoop 的应用范围。

2. Snapshot、Mirror 等企业应用特性

Snapshot（快照）、Mirror（镜像）等企业应用特性是企业 IT 管理人员必不可少的工具，也是处理复杂需求的得力助手。通过支持 Volume，MapR Hadoop 方便地支持了这些功能，使得 Hadoop 不再只是开发人员的专宠，数据科学家、IT 管理人员也能够方便地访问功能强大的大数据分析平台 Hadoop。

利用 MapReduce，开发人员可以开发跨处理器分布式群集或独立计算机的、并行处理海量非结构化数据的程序。MapReduce 框架可以划分为两个功能区：其中 Map 具备将工作分配给分布式群集中不同节点的功能；Reduce 则负责核对工作，将工作结果转化为单一值。

MapReduce 的主要优势之一是容错性。MapReduce 是通过监测群集中的每个节点来实现容错性的。每个节点定期向 MapReduce 报告和返回完成的工作与状态更新。如果某个节点的静默时间超出了预期值，主节点就会发出通知，并把工作重新分配给其他节点。

4.4.3 Storm

Storm 是一个分布式的、容错的实时计算系统，目前已经发展到 0.9.3 版本，并且集成了消息中间件 Kaf（Kafka MQ，高吞吐量分布式消息系统）。kaStorm 可以方便地在一个计算机集群中编写与扩展复杂的实时计算，Storm 对应于实时处理，就好比 Hadoop 对应于批处理。Storm 保证每个消息都会得到处理，而且它的速度很快——在一个小集群中，每秒可以处理数以百万计的消息。它可以使用任意编程语言来开发。Storm 主要有以下优点：

1）简单的编程模型。类似于 MapReduce 降低了并行批处理复杂性，Storm 降低了进行实时处理的复杂性。

2）服务化。一个服务框架支持热部署，即时上线或下线 App。

3）可以使用各种编程语言。可以在 Storm 之上使用各种编程语言，默认支持 Clojure、Java、Ruby 和 Python。要增加对其他语言的支持，只需实现一个简单的 Storm 通信协议即可。

4）容错性。Storm 会管理工作进程和节点的故障。

5）水平扩展。计算是在多个线程、进程和服务器之间并行进行的。

6）可靠的消息处理。Storm 保证每个消息至少能够得到一次完整处理。任务失败时，它会负责从消息源重试消息。

7）快速。系统的设计保证了消息能够得到快速的处理，使用 ZeroMQ（异步消息队列）作为其底层消息队列。

8）本地模式。Storm 有一个“本地模式”，可以在处理过程中完全模拟 Storm 集群，从而可以快速进行开发和单元测试。

4.5 其他开源软件解决方案

4.5.1 Docker（开源应用容器引擎）

Docker 是 dotCloud 公司开源的一个基于 LXC 技术之上构建的 Container 容器引擎，基于 Google 公司推出的 Go（2009 年发布第二款开源编程语言）语言开发，后来加入了 Linux 基金会。Docker 让开发者可以打包他们的应用并放置到一个可移植的容器中，发布到任何流行的 Linux 机器上，实现一种轻量级的虚拟化方案。Docker 容器完全使用沙箱机制，相互之间不会有任何接口，几乎没有性能开销，不依赖于任何语言、框架或包装系统，可以很容易地在机器和数据中心中运行。

Docker 在 2014 年 6 月召开 DockerConf 2014 技术大会吸引了 IBM、Google、RedHat 等业界知名公司的关注和技术支持，无论是从 GitHub 上的代码活跃度，还是 Redhat 宣布在 RHEL7 中正式支持 Docker，都向业界传达了一个信号，这是一项创新型的技术解决方案。就连 Google 公司的 Compute Engine 也支持 Docker 在其之上运行，国内百度应用引擎（Baidu App Engine，BAE）平台就是以 Docker 作为其 PaaS 云基础。

如图 4-30 所示，Docker 容器的执行不需要额外的虚拟化支持，它是操作系统核心层级的虚拟化，因此可以实现更高的效能和效率。Docker 容器的启动可以在秒级实现，这相比传统的虚拟机方式要快得多。其次，Docker 对系统资源的使用率很高，一台主

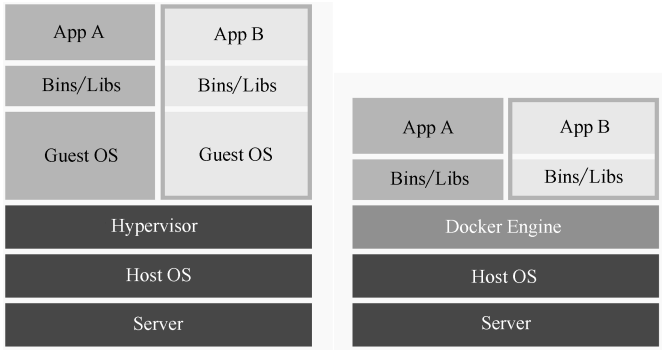


图 4-30　Docker VS VM

注：Docker：一个开源的应用容器引擎；Docker Engine：Docker 引擎；Hypervisor：虚拟机管理程序；Bins：二进制包；Libs：库；App ：应用；Guest OS：客机操作系统；Host OS：宿主机操作系统；Server：物理服务器。

机上可以同时执行数千个 Docker 容器。容器除了执行其中应用外，基本不消耗额外的系统资源，使得应用的效能很高，同时系统资源消耗更少。传统虚拟机方式执行 10 个不同的应用就要启动 10 个虚拟机，而 Docker 只需要启动 10 个隔离的应用即可。

Docker 可以简化部署多种应用实例工作，比如 Web 应用、后台应用、数据库应用、大数据应用等，例如，Hadoop 集群、消息队列等软件都可以打包成一个（镜像文件）进行部署。Docker 容器几乎可以在任意的平台上执行，包括实体机器、虚拟机、公有云、私有云、个人计算机、服务器等。这种兼容性可以让用户把一个应用程序从一个平台直接迁移到另外一个平台。

Docker 采用了客户端/服务端（Client/Server，C/S）架构，包括客户端和服务端。Docker Daemons（守护进程）是运行在宿主机上的 Docker 守护进程，用户通过 Docker Client 与 Docker Daemon 交互。Docker Daemons 作为服务端接收来自客户的请求，并处理这些请求（创建、运行、分发容器）。Docker Client 是 Docker 命令行工具，是用户使用 Docker 的主要方式，Docker Client 与 Docker Daemon 交互并将结果返回给用户，Docker Client 也可以通过 Socket（程度的双向通信连接的一端）或者 RESTful API 访问远程的 Docker Daemon。Docker 的总体架构如图 4-31 所示。

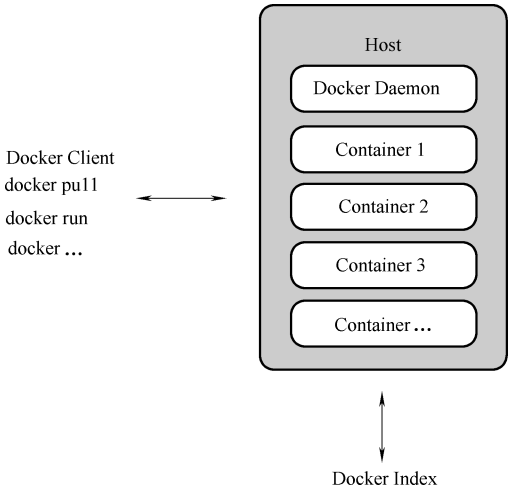


图 4-31　Docker 的总体架构

注：Docker：一个开源的应用容器引擎；Client：客户端；Host：主机；Docker daemon：Docker 后台进程；Container：容器；docker pull：Docker 命令，用于拉取镜像；docker run：Docker 命令，用于运行镜像；Docker Index：主要提供 Docker 镜像索引以及用户认证的功能。

Docker 内部主要由三部分组成。

1）镜像（Docker Images）：镜像相当于一个模板，用于创建 Docker 容器。Image 中可以包含 Linux 操作系统、应用程序等，用户可以下载已经创建好的 Docker Image，也可以创建 Docker Image 给其他用户使用。镜像在容器中运行起来后，可以继续在里面安装部署应用及服务，就像是在一台独立的虚拟机中的操作一样，所有的应用安装及配置变化都可以保存或者打包生成一个新的定制化 Image。每个 Image 都是由很多层组成的，Docker 通过 Union File Systems（Union 文件系统）将这些层绑定在一个 Image 中。每个

Image 都以一个初级 Image 作为基础，然后通过操作指令在这些初级 Image 上添加新层，操作指令可以是运行的命令、添加文件或目录、创建可用操作环境等。这些操作指令都被保存在“Dockerfile”文件中。

2）仓库（Docker Registries）：Docker Registries 是集中存放镜像的地方，分为公用仓库和私用仓库两种。公用的 Docker Registry 就是 Docker Hub（源代码管理集成），是 Docker 官方维护的，其中包含大量的可以直接使用的镜像；对于企业内部使用而言，出于安全、规范化的目的，可以在本地建立一个私有的仓库，供内部使用。

3）容器（Docker Containers）：容器使用镜像建立的执行实例，可以被启动、开始、停止或者删除。每个容器都是相互隔离的安全应用平台，可以把容器看作是一个简易版的 Linux 环境。

Docker 底层的核心技术包括 Linux 上的命名空间、控制组（Control Groups）、Union 文件系统和容器格式（Container Format）。传统的虚拟机通过在宿主主机中执行 Hypervisor 来模拟一整套完整的硬件环境提供给虚拟机使用。虚拟机系统看到的环境是可限制的，也是彼此隔离的，这种直接的做法实现了对资源最完整的封装，相对而言 Docker 的隔离性相比传统虚拟化方案还是有些欠缺，所有容器公用一部分运行库，容器随着用户进程的停止而销毁，容器中的用户数据也不便收集。但是 Docker 是面向应用的，其终极目标是构建 PaaS 平台；而虚拟机主要目的是提供一个灵活的计算资源池，是面向架构的，其终极目标是构建一个 IaaS 平台。因此，Docker 目的不是也不能替代传统虚拟化解决方案。目前 Docker 在容器没有提供完善的自管理、运维管理功能，需要借助一些第三方实现如 DockerUI、Dockland、Shipyard 等。

4.5.2 Solum

Solum 是一个开源项目，它的设计理念是缓冲公有云和私有云之间的可移植性。它是专为 OpenStack（以 Apache 许可证授权的自由软件和开放源代码项目）云计算而设计的，其研发采用了诸多 OpenStack 项目的技术，包括 Heat（负责编排计划）、Keystone（统一验证方式）、Nova（计算组织控制器）以及 Trove（数据库服务）等。Solum 会提供一个模块化的“语言包”解决方案，以此来支持多语言运行时间环境。用户可以自行选择语言环境来运行相关的应用程序。Solum 包括两个逻辑子系统，即控制面板和数据面板，其架构如图 4-32 所示。

Solum 通过 OpenStack 插件 Heat 的业务流程工具，跨开发者、跨测试、跨生产环境地简化应用程序生命周期管理；集成 Git 支持、持续集成以及集成各种开发环境，如 Eclipse、IntelliJ、Komodo 等。

4.5.3 Libcloud

Libcloud 是一个开源的访问云计算服务的统一接口，该项目已经成为 Apache 组织

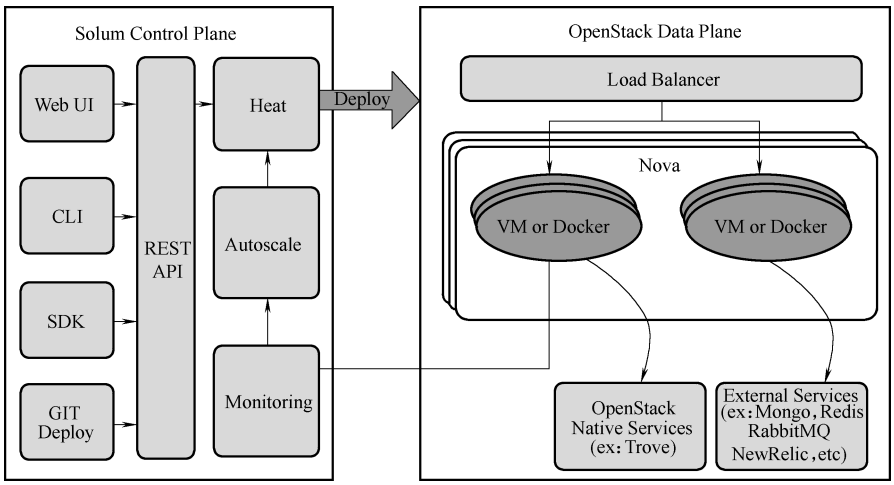


图 4-32　Solum 架构

注：Solum：一个开源项目，它设计理念是缓冲公有云和私有云之间的可移植性；Solum Control Plane：Solum 控制平面；Web UI：Web 用户界面；SDK：软件开发工具包；Git deploy：Git 部署；Heat：服务编排组件；Load Balancer：负载均衡；Autoscale：自动定标；OpenStack Data Plane：OpenStack 数据面板；OpenStack Native Services：OpenStack 本地服务；VM：虚拟机；Docker：一个开源的应用容器引擎。

的顶级项目。它的设计目标是打造一个厂商中立的、针对云供应商 API 的接口。Apache Libcloud 目前的版本已经为 20 多个领先的云供应商提供了后端驱动，包括 Amazon EC2、Rackspace Cloud、GoGrid 和 Linode。

4.5.4　Jclouds

Jclouds 是一个开源的 Java 类库，云计算开发包用于进行云计算应用开发，并可重用已有的 Java 和 Clojure（Lisp 语言）技能。该 API 提供云计算环境的可移植抽象层以及云规范特性，支持 Amazon、VMware、Azure 和 Rackspace 等云计算平台。Jclouds 接口简单，运行时有可迁移性，可以处理基于网络计算引入的一些问题，如瞬时失败和重新定向。Jclouds 提供了 Stub Connection（短连接）来模拟一个云而无需创建网络连接。通过此方法，可以编写单元测试而不再有模拟的复杂性或远程连接的脆弱性。

设计篇

第 5 章　商业银行私有云整体架构

第 6 章　云基础设施设计

第 7 章　私有云服务设计

第 8 章　云管理平台设计

第 9 章　私有云安全设计

第 5 章
商业银行私有云整体架构

5.1 私有云技术路线的选择

云计算是分布式计算、并行计算、网络存储、虚拟化等传统计算机和网络技术发展融合的产物。随着 IT 技术的不断发展和演进，云计算和传统 IT 技术也在不断地融合发展，其技术范畴和应用领域也在不断拓展，业内对云计算也没有一个统一的定义。云计算的核心技术和发展的趋势总结如下：

5.1.1 虚拟化技术

虚拟化并不是云，是建立和管理云的一项支撑技术。虚拟化平台会将计算、存储、网络资源都进行抽象并进行一定程度的融合。这使企业的数据中心在享受虚拟化带来的便捷的同时，还要面对增加了一个虚拟化抽象层面的问题。建设云平台时应重点关注虚拟化管理和自动化配置方面的问题。虚拟化涉及服务器虚拟化、网络虚拟化、存储虚拟化以及服务虚拟化，并使用虚拟化管理工具对其进行管理。数据中心虚拟化技术分类如图 5-1 所示。

1. 服务器虚拟化

服务器虚拟化技术是指将服务器物理资源抽象成逻辑资源，CPU、内存、磁盘、I/O 等硬件均变成可以动态管理的“资源池”，一台服务器变成多台相互隔离的虚拟服务器，可提高资源的利用率，简化系统管理，实现服务器资源池化的整合，提升计算资源对业务变化的适应力。

2. 存储虚拟化

存储虚拟化技术是将底层存储设备进行抽象化统一管理，向服务器层屏蔽存储设备硬件的特殊性，只保留其统一的逻辑特性，从而实现存储系统的集中、统一、方便的管

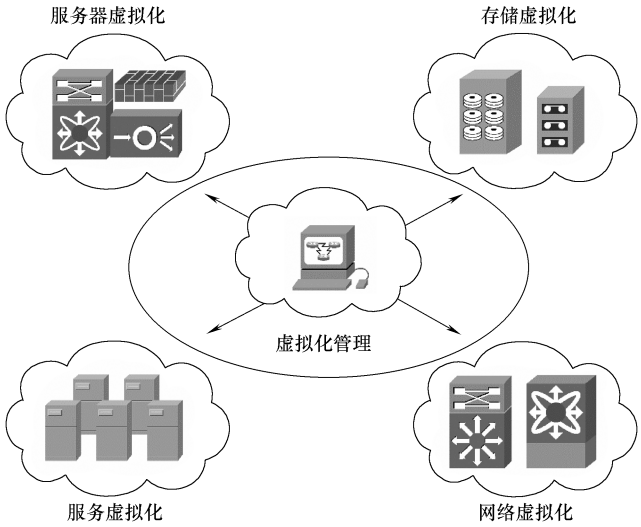


图 5-1　数据中心虚拟化技术分类

理。对于存储的自动化管理，也是存储虚拟化的一大研究课题。

3. 网络虚拟化

在云计算环境下，网络虚拟化技术与服务器虚拟化技术结合使用。物理服务器虚拟化之后，一台物理机上运行着多个不同 VLAN 的虚拟机，虚拟机之间通过虚拟交换机与外界进行数据通信。通过虚拟化软件，在虚拟交换机上部署虚拟防火墙，快速建立新的安全网络。对 IP 地址和 VLAN 的管理，是目前网络虚拟化中最普遍的应用。

4. 服务虚拟化

服务虚拟化是指虚拟接口以服务的方式对外公开。如数据中心常用的 F5 负载均衡设备，可将多个应用作为单一实例来进行管理，与允许用户直接连接到每个应用服务器相比，这种做法可以提供更安全、更健壮的拓扑。这是一种一对多的虚拟化表现方式。对外表现为一台服务器，实际隐藏着反向代理设备后的多台服务器的可用性。

5. 虚拟化管理

虚拟化管理是指协调虚拟资源的配置和协调，以及对资源池和虚拟实例进行运行时加以协调。该特性包括虚拟资源到物理资源的静态映射和动态映射，还包含整体的管理功能，例如容量管理、分析、收费以及 SLA。

5.1.2　软件定义数据中心

软件定义数据中心的概念是对数据中心虚拟化技术的进一步发展，凸显了未来管理软件和管理系统对于数据中心的重要性。云平台设计，也需要吸收相关理论的适用部分。

1. VMware 的软件定义数据中心（Software Defined Data Center SDDC）

VMware 于 2012 年提出了“软件定义数据中心（SDDC）”的概念，该概念重点关注于基础设施即服务（IaaS）。VMware IaaS 产品通过自动化封装和基于虚拟化的云互联来创建一个基于软件的数据中心。

SDDC 的定义如下：

1）所有资源，包括存储、网络、计算，管理等全部虚拟化，并以服务方式提供给用户；数据中心的控制全部由软件来完成。

2）硬件成为哑的“执行层”，即数据中心是“可编程的”。

3）资源提供速度以分钟、秒计算。

2. IBM 的软件定义环境（Software Defined Environment，SDE）

IBM 公司的云计算数据中心提出了类似的概念：软件定义环境（Software Defined Environment，SDE）。其定义为：优化整个基础设施架构以便满足实际需求，实现应用对资源的优化配置、调配和使用，实现上层应用定制策略、底层架构提供 API，核心是自动化和工作流。

5.1.3 IT 运维管理技术

如图 5-2 所示，Gartner 发布的 2013 年 IT 运维管理的技术成熟曲线模型，包含了 Gartner 对 IT 运维管理发展周期的 39 项技术和管理的预测与判断，这些技术的发展与云计算数据中心息息相关，是对云数据中心运维管理涉及的技术和工具的全面总结。

Gartner 的技术趋势分析对云计算数据中心的建设有着重要的指导意义。根据分析，当前 IaaS、SaaS 等已经相对比较成熟，需要重点关注 2～5 年并逐步成熟的工具和技术如下：

1. 开发运维衔接（DevOps）

DevOps 的思想是提升 IT 服务的敏捷性。在业务最初规划设计时，就在多个云服务之间进行比较沟通，选择最佳技术方案，并同时利用自动化工具尽量减少业务生命周期内的人工干预程度。

2. 应用自动发布（Application Release Automation）

在整个业务流程中，应用自动发布系统为应用程序、应用配置、应用数据提供了自动化工具，该工具为应用发布、业务环境和工作流管理平台提供了相关的自动化接口，是 DevOps 在维护大规模系统中使用的关键工具。

3. 云管理平台（Cloud Management Platforms）

云管理平台实现集中控制数据中心的目的。该平台将访问管理层、服务管理层和服务优化层三层紧密地结合起来。访问管理层是用户的入口、包括自服务接口、外部程序接口、用户授权等功能；服务管理层是云管理平台的核心，包括服务目录、服务模型、服务配置等功能；服务优化层包括服务抽象优化、服务接口管理的功能。

Figure 1.Hype Cycle for IT Operations Management，2013

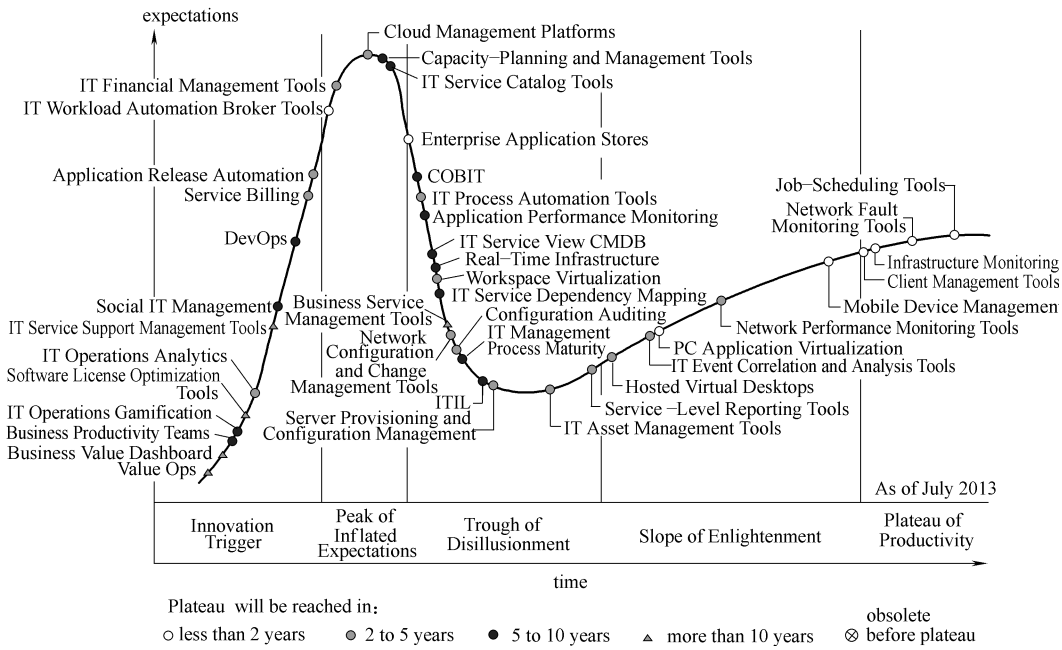


图 5-2　Gartner 2013 IT 运维管理的技术成熟曲线模型

注：Hype Cycle for IT Operations Management：IT 运维管理技术成熟度曲线；expectations：花销；IT Financial Management Tools：IT 财务管理工具；IT Workload Automation Broker Tools：IT 作业自动化代理工具；Application Release Automation：应用发布自动化；Service Billing：服务计费；DevOps：开发运维；Social IT Management：社交 IT 管理；IT Service Support Management Tools：IT 服务支持管理工具；IT Operations Analytics：IT 运维分析；Software License Optimization Tools：软件证书优化工具；IT Operations Gamification：IT 运营游戏化；Business Productivity Teams：企业生产力团队；Business value Dashboard：商业价值仪表盘；ValueOps：价值导向；Cloud Management Platforms：云管理平台；Capacity-Planning and Management Tools：容量规划管理工具；IT Service Catalog Tools：IT 服务目录工具；Enterprise Application Stores：企业应用商店；COBIT：信息系统审计标准；IT Process Automation Tools：IT 流程自动工具；IT Service View CMDB：IT 服务试图 配置管理数据库；Real-Time Infrastructure：实时基础设施；Workspace Virtualization：工作空间虚拟化；IT Service Dependency Mapping：IT 服务依赖映射；Business Service Management Tools：业务服务管理工具；Network Configuration and Change Management Tools：网络配置管理工具；Configuration Auditing：配置审计；IT Management Process Maturity：IT 管理过程成熟度；ITIL：IT 基础架构库；Server Provisioning and Configuration Management：服务器配置管理；IT Asset Management Tools：IT 资产管理；Service-Level Reporting Tools：服务级别报告工具；Hosted Virtual Desktops：托管的虚拟桌面；IT Event Correlation and Analysis Tools：IT 事件关联分析工具；PC Application Virtualization：PC 应用程序虚拟化；Network Performance Monitoring Tools：网络性能监控工具；Mobile Device Management：移动设备管理；Client Management Tools：客户管理工具；Infrastructure Monitoring：基础设施监控；Network Fault Monitoring Tools：网络故障监控工具；Job-Scheduling Tools：作业调度工具；Innovation Trigger：科技诞生的促动期；Peak of Inflated Expectations：过高的期望峰值；Trough of Disillusionment：泡沫化的低谷期；Slope of Enlightenment：稳步爬升的光明期；Plateau of Productivity：实质生产的高峰期；Plateau will be reached in：稳定水平将达到；less than 2 years：不到 2 年；2 to 5 years：2 到 5 年；5to 10 years：5 到 10 年；more than 10 years：超过 10 年；obsolete before plateau：稳定前已淘汰。

4. 企业级应用仓库（Enterprise Application Stores）

该仓库存储了所有应用当前和历史版本，同时为发布自动化系统提供接口，一起帮助用户存储分发应用代码和应用配置。

5. IT 流程自动化工具（IT Process Automation Tools）

该工具包括三个关键的元素，即工作流、自动化引擎和集成框架。工作流代表了 ITIL 系统，集成框架可以嵌入自动化代码，自动化引擎实现两者联动。IT 流程自动化工具减轻了 IT 人员的重复劳动。

6. 配置管理数据库（IT Service View CMDB）

CMDB 不仅是一个配置管理数据库，还是整个业务的配置管理中心，提供 IT 服务模型映射图、业务的集成和耦合关系表、信息和数据的同步策略。在发布应用时就需要自动读取应用在 CMDB 中的相关配置。

7. 网络配置和变更管理工具（Network Configuration and Change Management Tools）

此工具用以管理网络设备的配置、审计变更、保存历史记录，同时生成相关文档。在需要时可以随时利用工具将配置回滚到指定的历史版本。

8. 服务器硬件配置管理（Server Provisioning and Configuration Management）

该工具提供了对各个厂商的设备的硬件配置的操作管理功能，提高对于服务器配置的自动化，对于大规模的云计算数据中心尤其重要。

9. 服务编排（Job-Scheduling Tools）

由于各个业务之间存在复杂的逻辑关系，每个工作流在启动时有时间限制和依赖关系。该工具需要对各个工作流进行分析，并为其提供基于日期和时间的细粒度调度策略，实现工作流更灵活的组合。

5.1.4 私有云建设技术的选择

私有云是指在企业私有网络上使用云计算产品，金融行业对数据的安全性、个人隐私、系统可靠性、自主可控、监管要求等方面的考虑，当前和今后相当长一段时间，金融行业的云计算数据中心普遍采用私有云。对于商业银行而言，建设私有云既要考虑技术方面的因素，也要从行业特点、应用特点、组织架构、现有技术继承等多方面因素综合考虑。

1. 虚拟化技术的选择

虚拟化技术是建设云平台的基础。在目前使用最广泛的服务器虚拟化领域就存在多种虚拟化的技术方案和产品，既有开源产品也有商业产品，如 XenServer、KVM、vSphere 和 FusionSphere 等。对于私有云，成本、稳定性、可靠性、可维护性都是选择的因素，相对商业产品在功能稳定性、完备性方面和技术支持方面都有一定的优势，目前大多数企业都选择商业产品。而互联网企业基本都选择开源产品，这其中有成本因素考虑，但更重要的是技术积累和掌握的问题，相对于一般企业，互联网公司汇聚了大量的技术专家，对开源产品不仅仅是拿过来使用，而是具备了代码级的修改及运维能力，

而且不少公司都直接给开源产品贡献代码。对于商业银行而言，稳定、可靠始终是首要的考虑因素，在对开源产品缺少足够的技术积累前，商业产品还是第一选择。但是从长远来看，自主可控和国产化是未来的技术主线。因此在建设私有云，进行总体架构规划设计时要考虑对虚拟化技术的松耦合，有条件的企业可以考虑走商业产品和开源产品并行的路线，在核心的关键应用领域使用商业产品，在非关键业务或者内部办公应用领域使用开源产品。这样既积累了开源产品的技术和使用经验也避免了单一商业产品绑定。

2. 云平台建设方式的选择

目前国内外不少公司都推出了自己的私有云产品。长期以来金融行业出于风险控制的考虑，大部分基础产品都使用商业产品。云计算技术的核心是理念、流程、技术的深度结合，从技术本身而言不存在太高的技术壁垒和风险，在云计算发源的互联网和电信行业，基本都采取自主研发路线。当前云计算还在快速发展中，厂商产品的成熟度还不够，往往很难覆盖全部需求，而且使用厂商产品还要面临整合的问题。因此，在充分调研和反复论证的基础上，吸取最新研究成果和先进厂商的成熟产品优点，走自主研发的道路，量身定制符合企业需求的私有云平台应当是重点考虑的建设方式。自主研发的云平台能够充分满足并体现的商业银行的业务特点和实际需要，也能够扭转技术路线被厂商驱动的被动局面。当然，自主研发要抓关键点，要避免“重复发明轮子”，应采取博采众家所长的策略，对需求分析、系统设计的完全自主化；边缘的系统集成、非核心的功能开发，可采取厂商产品或合作公司产品。

3. 工具和流程一体化

在云计算数据中心，资源的池化和规模化降低成本的要求使得服务器的数量远超过了传统数据中心的规模，而运维管理人员的数量则相对非常有限。在一些大规模的云计算数据中心，管理员和服务器的比率甚至达到了 1:1500，如果不借助高效自动化的管理工具，依靠人工的模式根本无法保障云计算数据中心的正常运行。利用自动化实现集中统一的自动化管理，强制执行最佳实践和自动化服务，是云计算数据中心运行管理的关键。和传统数据中心的自动化不同的是，云数据中心的自动化不仅是工具的自动化，还是实现流程的自动化。通过管理工具和管理流程的同步建设和深度结合，实现运维管理、运维流程、运维操作的全面自动化，解决管理工具和流程之间的信息孤岛。

1）运维管理自动化：云管理平台实现了流程编排、资源部署、资源管理、资源监控等的自动化。

2）运维流程自动化：云管理平台与服务管理流程平台进行信息交互，实现了服务申请、服务审批、服务实施、服务交付等环节的关联与互通，达到了运维流程的自动化。

3）运维操作自动化：云管理平台实现了服务启停、自动巡检、合规检查、配置比对、系统部署、软件分发、版本及补丁管理等运维操作的自动化，大大提升了日常运维的工作效率，减少了因人为误操作引起的风险。

5.2 私有云总体架构设计

5.2.1 设计思路

商业银行私有云架构应基于支持虚拟化与云管理的自动化、监管控一体化、开发与运维流程化、深化决策分析、面向 SOA 的组件及服务管理的运维管理模式等思路设计。

1）云环境部署和管理：随着虚拟化和云环境发展，基础环境与以往环境有了很大的变化，必须充分认识理解虚拟化环境和云环境的管理特性，结合发展趋势，采用先进技术平台架构和管理理念进行建设。

2）监管控一体化：总结前期运维经验，在架构设计中关注监控、流程、自动化的综合协作管理，实现监控发现（眼）、流程管控（脑）、自动化修复及核查（手）的问题处理全流程协作，将监管控三个不同功能域整合在一起，相互驱动，减少线下操作，提升操作标准化、规范化和自动化程度。

3）开发与运维衔接：借助业界一些方法论如 DevOps，根据各商业银行的特点，将开发到运维环节衔接起来，纳入统一管控，既要支撑高效率的应用发布工作，也要避免频繁发布中一些管控环节不到位导致的业务系统故障，从源头开始管控，有效支撑业务发展。

4）面向 SOA 的组件及服务管理：在面向服务的架构环境中，应从以往关注 IT 组件层面的监控管理提升到端到端面向业务的管理，实现对应用层面、业务流程层面、用户层面全方位深入的监控和管理；同时应结合新架构的技术特点，关注 IT 服务层运行状态，深化 IT 内部运行机制的管控，IT 服务层是承接业务与 IT 组件的关键环节，SOA 的架构更加强化规范了 IT 服务层，IT 服务层对上支撑错综复杂的业务运行，对下协作各 IT 组件的工作，是整个新一代系统运行的重中之重。

5）开发测试生产环境标准化管理：实现从开发测试生产环境一条线，一体化管理，除确保生产环境保障外，应将开发测试环境也纳入统筹管理，在依托平台组件的基础上将开发测试环境管理任务标准化、流程化。

6）深化决策分析：现有平台数据分析及展现分散在多个平台环境中，一方面操作不便，使用方法和表达方式都不统一；另一方面也形成了数据孤岛，缺乏横向联合分析手段，使用者需要跨平台操作，采用集中化的管理门户和报表系统将各类数据源进行统一展示和分析，使用者在个性化的视图中对多个平台数据进行浏览和处理。

5.2.2 架构概览

如图 5-3 所示，商业银行私有云架构应包括云服务、云管理、资源池等部分，同时还需要与配置管理、监控管理、流程管理、容量管理等相结合，共同实现云管理的相关

功能。

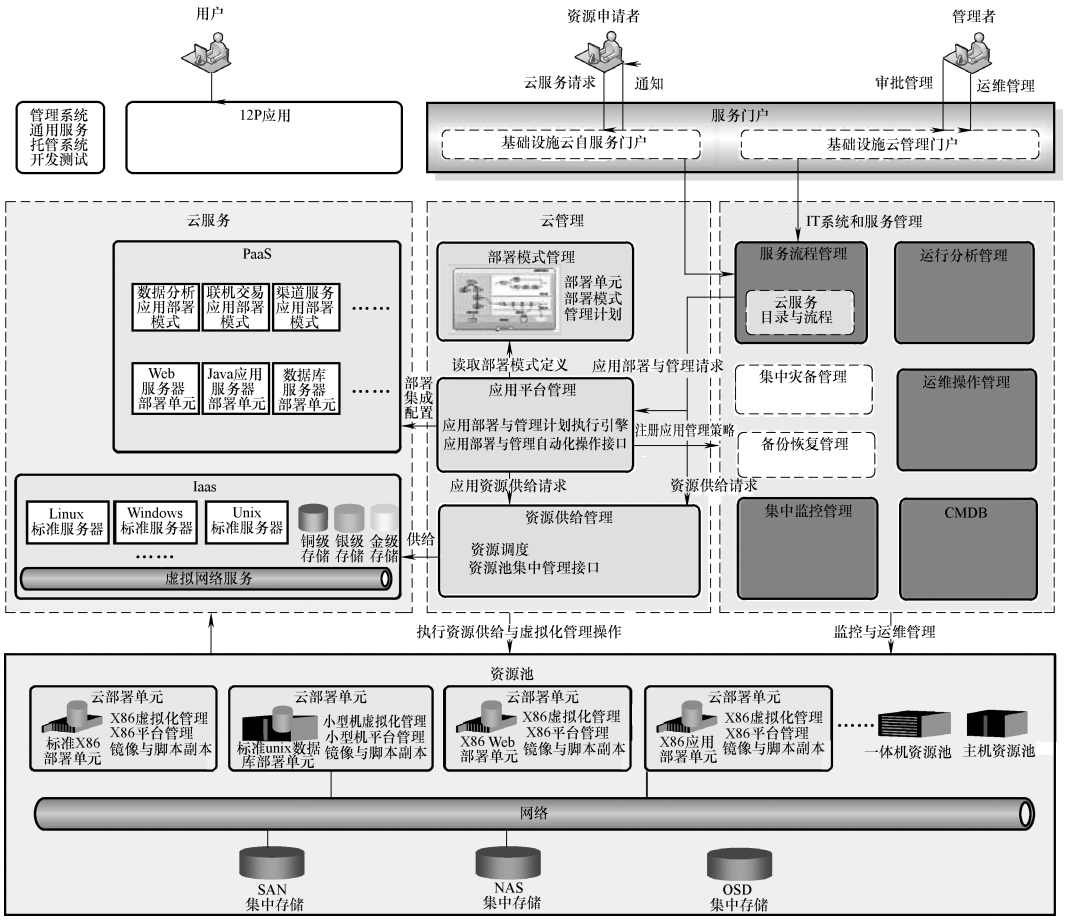


图 5-3　商业银行私有云架构

注：PaaS：平台即服务；IaaS：基础设施即服务；Web：万维网；Java：一种可以撰写跨平台应用程序的面向对象的程序设计语言；Linux：一套免费使用和自由传播的类 Unix 操作系统；Windows：美国微软公司研发的一套操作系统；Unix：是一个强大的多用户、多任务操作系统；CMDB：配置管理数据库；X86：对基于 intel 处理器的系统的标准缩写；SAN：Storage Area Network，存储区域网络，是一种高速网络或子网络，提供在计算机与存储系统之间的数据传输；NAS 网络附属存储（Network Attached Storage）；OSD：基于对象的存储设备（Object-based Storage Device）

第 6 章
云基础设施设计

6.1 机房环境资源规划与设计

6.1.1 云计算对机房环境资源的要求

云计算是一种资源交付和使用模式，通过网络可获得应用所需的资源（硬件、平台、软件），是随着处理器技术、虚拟化技术、分布式存储技术、宽带互联网技术和自动化管理技术的发展而产生的。而所有的应用所需要的计算能力、存储、带宽、电力都由数据中心提供。因此，云计算环境下的数据中心机房规划显得尤为重要。

1. 超大规模

“云”一般具有相当的规模，亚马逊云计算已经拥有上百万台服务器，微软、阿里云等的“云”均拥有几十万台服务器。“云”能赋予用户前所未有的计算能力。因此，云计算数据中心机房的面积也非常大。

2. 高密度

云计算是一种集中化的部署方式，要在有限空间内支持高负载、刀片式服务器等高密度设备是必然选择。

3. 灵活快速扩展

“云”的规模可以动态伸缩，满足应用和用户规模增长的需要。其数据中心必须具有良好的伸缩性，同时，为了节省投资，最好能边成长边投资。

4. 降低运维成本

云计算可以对承载的应用进行计量、计费服务，这也是原来 IT 部门是成本部门的重大转变。通过容量分析，实现资源动态、弹性伸缩，将会大大降低资源使用率，降低运维成本。

5. 自动化资源监控和测量

云计算数据中心应是 24h ×7 无人值守的、可远程管理的，这种管理涉及整个数据中心的自动化运营，它不仅监测与修复设备的硬件故障，而且要实现从机房风火水电环境、服务器和存储系统到应用的端到端的基础设施统一管理。

6. 高可靠性

云计算要求其提供的云服务连续不中断，“云”使用了数据多副本容错、计算节点同构可互换等措施来保障服务的高可靠性，使用云计算比使用本地计算机更可靠。同样，在机房环境设施方面，也提出了对机房环境高可靠性的要求。

6.1.2 云计算机房规划设计要点

本节主要介绍的是因云计算环境特点而需考虑的机房规划设计要点。而机房常规设计要素，如：环境要求、建筑与结构、空气调节、电气技术、机房布线、排水灌水、消防安防等，不在本节介绍范围内。

6.1.2.1 机房资源信息管理

机房资源信息是机房运维管理，以及资源规划和分配的基础，因此在进行机房规划和设计时，必须考虑机房资源信息的获取及管理。机房资源信息应支持按机房功能区域、配电柜、机柜等不同颗粒度或需求进行统计展示。机房资源信息主要包括的要素有：

1）设备：设备厂商、类型、型号、序列号、物理尺寸、物理位置、功耗、电源数量、网口数量和类型、光纤口数量和类型等。

2）机柜：机柜类型、物理尺寸、已占用空间、已占用位置、可用空间、可用位置等。

3）网络及存储信息点：信息点数量、信息点类型、已用信息点、可用信息点等。

4）供电：电流、电压、功率、可用电量等。

5）温湿度：温度、湿度等。

6.1.2.2 设备管理

由于云计算设备数量巨大，而且经常存在上架安装、退库下线、位置迁移等变更操作，所以管理好设备实物，使得与机房资源信息系统中的记录“账实相符”就显得尤为重要。射频识别（RFID）是一种无线通信技术，可以通过无线电信号识别特定目标并读写相关数据，而无需识别系统与特定目标之间建立机械或者光学接触。无线电信号是通过调成无线电频率的电磁场，把数据从附着在物品上的标签上传送出去，以自动辨识与追踪该物品。某些标签在识别时从识别器发出的电磁场中就可以得到能量，并不需要电池；也有标签本身拥有电源，并可以主动发出无线电波（调成无线电频率的电磁场）。标签中包含了电子存储的信息，数米之内都可以识别。与条形码不同的是，射频标签不需要在识别器视线之内，也可以嵌入被追踪物体之内。

6.1.2.3 机房制冷

云计算相比传统环境，大量应用高密设备，单机柜内设备数量、功率密度、发热密度都有巨大提高。因此，在机房规划设计时需重点考虑高密设备的制冷、散热问题。具体需考虑的因素主要包括机柜负载、气流组织形式、制冷方式、制冷设备、节能因素、服务器散热方式等。

随着液冷技术的开发和兴起，HP、DELL、SGI、思科、富士通、超微、联想、华为、曙光等厂商均推出了液冷服务器，其工作方式大致可以分为间接的冷板方式和直接的浸没方式两种。用液体来制冷，其热容和导热方面，比空气要好 1000 倍以上，因此，液冷服务器可以实现更高密度、节能省电、绿色环保等目标，是云计算一种不错的选择。如果选择使用液冷服务器，机房在规划设计时，还需考虑液体管路的相关设计。

6.1.2.4 模块化数据中心

由于云计算数据中心需要良好的扩展性，所以可考虑采用模块化设计。模块化数据中心，是指将电力、制冷、通信电缆以及相关的环境监控等都预先部署在一个框架上，预先完成测试，然后将这个框架直接部署到数据中心，从而形成一个完整的数据中心。所以，云计算可以根据规模和需求定制模块化数据中心，并随着业务发展需求，逐步扩建数据中心规模，实现边成长边投资的目标。此外，由于模块化数据中心采用的是规格统一、标准化的预制件，所以，其除了扩展灵活简便以外，还具备施工部署快速高效、易于维护管理、节能环保等优点。

6.1.2.5 动环监控

动环监控是针对各类机房中的动力设备及环境变量进行集中监控，一套完善的动环监控可以实现机房的无人值守，以及电源、空调的集中监控维护管理，提高供电系统的可靠性和通信设备的安全性，为机房管理的自动化、运行智能化和决策科学化提供有力的技术支持。动环监控的对象一般包括 UPS、配电柜、加湿器、空调、温湿度仪等设备的运行状态，以及机房温度、湿度、漏水等环境要素。

6.2 计算资源规划设计

6.2.1 资源池组成

资源池是基础设施云的重要组成部分。在基础设施云架构下，计算资源、存储资源、网络资源在统一的云管理平台下被封装整合为资源池，以云服务的方式提供给服务使用者。

按照技术平台类型分为 ×86 平台、AIX 和 HPUX 三种技术平台。

1） ×86 平台资源池又分为 VMware 虚拟化平台架构和 ×86 物理服务器组成的大数

据类应用集群架构。其中，VMware 虚拟化平台架构资源池主要满足 Web、中间件，以及其他以标准 Linux、Windows 为系统平台的应用的需求；大数据资源池满足 Hadoop 和 MPP（大规模并行处理平台）等大数据分析类应用需求。

2）AIX 平台资源池全部采用 PowerVM 虚拟化技术，主要满足数据库类应用，同时满足部分应用服务器的部署需求。

3）HPUX 平台资源池，不采用虚拟化技术，部署的业务类型和 AIX 平台资源池相同。

综上，资源池的类型见表 6-1。

表 6-1　资源池的类型

序号	平台类型	资源池类型名称	主要服务属性	次要服务属性	其他服务属性
1	×86	×86 虚拟化资源池 A	Web	—	—
2	×86	×86 虚拟化资源池 B	应用	前置	Web
3	×86	×86 虚拟化资源池 C	应用	Web	—
4	×86	大数据资源池 A	Hadoop	—	—
5	×86	大数据资源池 B	MPP	—	—
6	AIX	AIX 虚拟化资源池 A	数据库	应用	—
7	AIX	AIX 虚拟化资源池 B	数据库	应用	—
8	HPUX	HPUX 资源池 A	应用	数据库	—
9	HPUX	HPUX 资源池 B	应用	数据库	—
10	HPUX	HPUX 资源池 C	数据库	应用	—
11	HPUX	HPUX 资源池 D	数据库	应用	—

6.2.2　资源池分区

在银行数据中心的基础设施环境中，为了保证风险可控，对安全性要求较高，为此，从网络上划分了多个安全区域。为了满足应用上线的需求并同时符合网络安全访问控制的策略，基础架构网络中需要部署资源池的有以下几个区域：

1）开放服务区：部署各类核心业务系统。

2）运行管理区：部署各类管理服务器。

3）互联网 DMZ 服务区：实现互联网接入及互联网应用部署。

4）外联 DMZ 服务区：实现与外部机构互联及外联网应用部署。

6.2.3　资源池部署规划

由于要满足网络安全访问控制的要求，所以同一个标准资源池不能跨网络区域部署。在同一个网络区域内，按照不同的硬件平台（如 AIX、HP、×86）划分为多个不同类型的资源池。

为了满足应用系统上线的需求，在相关区域中（开放服务区、运行管理区、互联网 DMZ 服务区、外联 DMZ 服务区）选择以下标准资源池进行部署。部署区域与资源池类型的关系见表 6-2。

表 6-2　部署区域与资源池类型的关系

序号	部署区域	平台	资源池类型
1	互联网 DMZ 服务区	X86 资源池	X86 虚拟化资源池 A
2	外联 DMZ 服务区	X86 资源池	X86 虚拟化资源池 B
3	开放服务区	X86 资源池	X86 虚拟化资源池 A
		X86 资源池	X86 虚拟化资源池 B
		AIX 资源池	AIX 虚拟化资源池 A
			AIX 虚拟化资源池 B
3	开放服务区	X86 资源池	大数据资源池 A
		X86 资源池	大数据资源池 B
		HP 资源池	HPUX 资源池 A
			HPUX 资源池 B
			HPUX 资源池 C
			HPUX 资源池 D
4	运行管理服务区	X86 资源池	X86 虚拟化资源池 C
		AIX 资源池	AIX 虚拟化资源池 A
			AIX 虚拟化资源池 B

6.2.4　部署单元规划

资源池构建过程为：由计算、存储、网络等领域部署单元（TDU）组成构建单元，由一个或多个构建单元按照一定的构造策略组成云部署单元（CDP），满足一类应用对安全、高可用、可扩展等要求。在相应的网络区域部署所需类型和数量的云部署单元构成提供某种服务的资源池。云部署单元构建流程如图 6-1 所示。

1. 领域部署单元

领域部署单元是提供某一领域功能的产品或能力，领域部署单元包括计算领域部署单元、存储领域部署单元和网络领域部署单元。

（1）计算领域部署单元　计算领域资源按照不同的技术平台类型、厂商、性能等指标进行分类组织，形成能够提供计算功能的不同档次的部署单元，该部署单元和存储领域部署单元、网络领域部署单元结合形成能够提供不同类型服务的构建单元，由一类构建单元按照一定架构策略组织起来形成有一定可用性、安全性、可管理性的集合云部署单元（CDP），由云管理平台统一进行调度和资源分配，从而实现自动化、弹性的基础架构。

（2）存储领域部署单元　存储领域资源按照不同的技术类型、配置、性能等指标

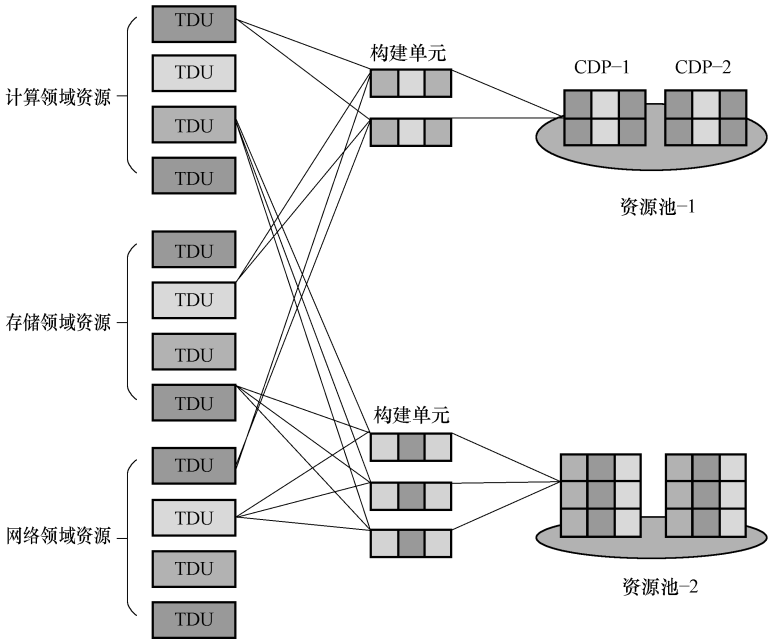


图 6-1　云部署单元构建流程

进行分类组织，形成能够提供数据存储功能的不同档次的部署单元（TDU），该部署单元和计算领域部署单元、网络领域部署单元结合形成能够提供不同类型服务的构建单元，由一类构建单元按照一定架构策略组织起来形成有一定可用性、安全性、可管理性的集合云部署单元（CDP），由云管理平台统一进行调度和资源分配，从而实现自动化、弹性的基础架构。

（3）网络领域部署单元　按照实际需求，网络领域资源按照服务提供点（POD）描述网络接入资源分为四类，即千兆接入 POD、普通万兆接入 POD、大数据万兆接入 POD 和 NAS 万兆接入 POD。

2. 构建单元

构建单元是由一个或多个领域部署单元组成的，能够形成服务供给能力的最小单位。由一类构建单元按照一定架构策略组织起来形成有一定可用性、安全性、可管理性的集合称为云部署单元（CDP）。

6.3　网络资源规划设计

6.3.1　网络虚拟化关键技术

6.3.1.1　网络设备虚拟化

目前的虚拟交换机技术有两种，一种是多对一虚拟化；另一种是一对多虚拟化。

1）多对一虚拟化是指将多台物理交换机的控制平面整合起来，使其表现为一个逻辑交换机，主流技术有思科公司的 VSS（Virtual Switching System，虚拟交换系统）和 H3C 的 IRF（Intelligent Resilient Framwork，智能弹性架构）。通过多对一虚拟化，可以带来以下好处：实现二层无环的网络架构，避免维护复杂的生成树协议，网络中无阻塞端口，提高物理线路的有效利用率；无需采用传统的虚拟路由冗余协议（Virtual Router Redundancy Protocol，VRPP），两台交换机均可实现三层转发，提高设备利用率；多对一虚拟化减少了网络节点，从而使网络管理得以简化。

2）一对多虚拟化是指将一台物理交换机划分为多台虚拟交换机，各个虚拟交换机间的管理和数据转发功能都独立，通过一对多虚拟交换机技术，可以把对安全要求高、原来必须运行在独立网络里的业务运行整合到统一的网络资源池上来运行，从而实现网络资源的灵活调度，以及数据安全和节能减排。虚拟交换机的网络整合主要放在数据中心网络的汇聚层和核心层，整合方式包括水平整合、垂直整合等。

6.3.1.2 虚拟机支持

随着服务器虚拟化技术的成熟，数据中心部署的虚拟化服务器的数量越来越多。虚拟机的出现使数据中心服务器网络接入层出现了一个被称为 VEB（Virtual Ethernet Bridge，虚拟以太网桥接）的网络层。在服务器上采用纯软件方式实现的 VEB 就是通常所说的“VSwitch”（Virtual Switch，虚拟交换机）。虽然 VSwitch 的实现方式简单，而且技术兼容性好，但也面临着诸多问题，例如 VSwitch 占用 CPU 资源导致虚拟机性能下降、虚拟机流量监管问题、虚拟机的网络策略实施问题以及 VSwitch 管理可扩展性问题。

为此，IEEE（Institute of Electrical and Electronics Engineers，电气和电子工程师协会）DCB（Data Center Bridging，数据中心桥接）IEEE DCB（Data Center Bridging）任务组（DCB 任务组是 IEEE 802.1 工作组的一个组成部分）正在制定两个新标准：802.1Qbg 和 802.1Gbh。

目前这两种标准都在发展过程中，思科设备目前已经支持 802.1Qbh，HP 主导的 802.1Qbg 在编写文档时还没有成熟的产品。具体如何选择，还需要跟进厂商和市场的发展情况进行考虑。

6.3.1.3 自动路由感知

面向云计算的数据中心，虚拟服务器可以在不同的数据中心间自由迁移。传统情况下，一个 IP（Internet Protocol，网际协议）地址为 A 的虚拟服务器从数据中心甲迁到数据中心乙时，客户端访问 A 服务器的数据包仍要先抵达数据中心甲，然后通过甲、乙之间的链路发送到服务器上，这样数据包在网络中经过了多次转发，不但增加了延时，而且浪费了宝贵的网络资源。在新一代的数据中心中，采用了新的路由技术 LISP（Locator/ID Separation Protocol）来解决这个问题。在 LISP 协议的设计中，我们在原有的 IP 包前又封装了一个 IPv4（Internet Protocol Version4，网际协议第 4 版）或 IPv6（Internet Protocol Version6，网际协议第 6 版）包头作为数据中心的标识，也就是说当一个服务器发生迁移时，IP 地址未发生改变，但位置标识做出改变，客户端可以感知位置标识的

改变，直接把数据发给数据中心，避免了上述问题。

6.3.1.4 自动资源供给

通过在虚拟机软件 VMware vSphere5.0 中安装负载均衡设备管理插件实现负载均衡设备和虚拟机的协作管理功能，通过负载均衡设备管理插件，可以添加指定的负载均衡设备，并能登录负载均衡设备，添加负载均衡组员、管理负载均衡组员状态。

通过在虚拟机软件 VMware vSphere5.0 中安装负载均衡设备 Icontrol 脚本实现自动化运行，简化网络部署、管理和维护，实现无需中断的按需扩容，进行服务器虚拟化整合，使服务器使用率提升。

6.3.2 资源池网络设计

6.3.2.1 X86 虚拟资源池

逻辑上 X86—CDPA 标准 CDP（Cloud Deploy Point，云部署节点）是由三个集群组成的，每个集群是由相邻机柜内的 16 台 2 路 X86 服务器、一台 10TB 用容量 NAS 设备、两台千兆架顶交换机、两台万兆架顶交换机两台万兆列中交换机组成的。网络交换机硬件配置见表 6-3。

表 6-3 网络交换机硬件配置

类型	端口数	端口类型	功能
千兆架顶交换机	48	电口	管理接入
万兆架顶交换机	32	电口	生产接入
万兆列中交换机	48	光口	汇接网络

1. 机柜部署

X86-CDP 由三个集群构成，每个集群包括 16 台 2U 或 4U 高的 PC 服务器和一台双控制器 NAS 设备。

一个集群内 16 台 PC 服务器部署在两个 42U 高的标准服务器机柜中，每台服务器机柜内安装一台万兆架顶交换机和一台千兆架顶交换机。两台架顶交换机均通过冗余万兆光纤上联口连接到两台列中交换机。一台 NAS 设备部署在一台独立的机柜中，NAS 设备的两个控制器分别通过万兆光纤连接到所属集群 ESXi 服务器上联的列中交换机。

2. 服务器连接

构成 X86—CDP 一个集群的两个机柜内的 ESXi 共用架顶交换机和列中交换机，每台 ESXi 的第一个万兆口连接第一个机柜的万兆架顶交换机，第二个万兆口连接第二个机柜的万兆架顶交换机；每台 ESXi 的第一个千兆口连接第一个机柜的千兆架顶交换机，第二个千兆口连接第二个机柜内的千兆架顶交换机。对于双口网卡，使用第一个端口。万兆架顶交换机和千兆架顶交换机均通过冗余的上联链路，分别连接到两台列中交换机。

3. NAS 设备连接

X86—CDP 每个集群配置一台双控制器 NAS 设备，每个控制器均通过 4 条万兆光纤链路分别连接两台列中交换机，每个控制器到每台交换机提供两条千兆链路。NAS 设备的管理接口，分别连接到所属集群内两台安装服务器机柜内的千兆架顶交换机。

6.3.2.2 AIX 虚拟资源池

1. 机柜部署

逻辑上 AIX—CDP 标准 CDP 是由两个集群组成的，每个集群由两台 P750 或 P780 组成。网络交换机硬件配置见表 6-4。

表 6-4 网络交换机硬件配置

类型	端口数	端口类型	功能
千兆架顶交换机	48	电口	HMC 连接
万兆架顶交换机	32	光口	生产和管理网络接入

1）每个机柜安装两台 P750 服务器或 P780 服务器。

2）每个机柜安装一台 HMC（Hardware Management Console，硬件管理台）。

3）每个机柜顶部安装一台万兆交换机。

4）每个机柜顶部安装一台千兆交换机。

2. 服务器连接

网络线缆连接描述：

1）构成 AIX—CDP 标准 CDP 的 4 台服务器共用两个机柜中的两台万兆交换机。

2）按照从上到下、从左到右的顺序，每种功能的第一块网卡连接到 A 机柜万兆交换机，每种功能的第二块网卡连接到 B 机柜万兆交换机。

3）服务器上 HMC 的第一个网络接口连接到第一台千兆交换机，第二个网络接口连接到第二台千兆交换机。AIX P750 有两个 FSP（Flexible Serive Processor，灵活服务处理器）接口，AIX P780 有四个 FSP 接口。

HMC 的两个网络接口连接到同一机柜的千兆交换机，一个网络接口用于远程管理，另一个接口用于本地管理 AIX P750 或 AIX P780 服务器。

6.3.2.3 HP 物理资源池

1. 机柜部署

HP 物理资源池—CDP 由一个集群构成，每个集群包括两台 HP Rx9900 10U 高刀片服务器。

按照位于低密区机柜电量、重量和散热的要求，结合 HP Rx9900（64 个 CPU，512GB 磁盘空间）满配电量、重量和散热等物理指标。制定机柜的低密区设计规范如下：

1）一个 CDP 设备中的两个 C7000 机箱分别位于两个相邻部署的机柜中，这样便于维护及网络接入。

2）每个物理机柜放置一个满配 Rx9900，包括一个 C7000 机箱和两个 BL890c i4。

3）每个物理机柜放置一个 48 口（及以上）千兆电口交换机，用于本机柜和同一 CDP 设备中另一机柜的生产、心跳网络连接，以及 OA（Onboard Administrator，板载管理）接口的连接。

4）每个物理机柜放置一个 16 口（及以上）万兆光口交换机，用于带外管理网络连接。

2. 服务器背板连接

HP 物理机资源池—CDP 包括两台 Rx9900i4、两台万兆列中交换机、两台万兆架顶交换机和两台千兆架顶交换机组成。

一台 HP 小型机服务器接口分为生产、心跳和管理三大类，每类接口有 16 个，其中管理类接口采用万兆光口网卡，心跳、管理采用千兆电口网卡。此外，每台 HP 小型机还有两个 OA 接口，采用千兆电口网卡。因此一个 CDP HP 小型机（2 台）共使用 32 个万兆光口，68 个千兆电口（64 个生产、心跳接口，4 个 OA 管理接口）。

6.3.3 数据中心网络设计

6.3.3.1 设计原则

数据中心是银行基础设施的最重要部分，同时数据中心网络接入组件是数据中心所有系统服务得以互联互通和对外发布的重要支撑。为了能够满足长期科技规划和网络规划的要求及数据中心网络接入组件的设计需求，在数据中心网络接入组件设计时应遵从以下设计原则：

1）先进性：能够满足未来业务开展的需求，网络结构具备较强的弹性。

2）灵活性：满足应用系统灵活多变的部署需求。

3）高可用性：不因为任何一个网络模块发生故障而影响全局网络的畅通。

4）可管理性：网络简单、健壮，易于管理和维护，保障安全运行。

5）规范性：遵循各类安全、网络规范。

6）标准的开放性：支持国际上通用标准的网络协议，有利于保证与其他数据中心之间的平滑连接互通，以及将来网络的扩展。

7）简单实用性：采用先进、合理、实用的技术方案，满足网络的简单化和标准化管理。

在数据中心内，由于管理和应用的要求，所以需要划分多个区域。这与未来云计算的需求有所矛盾，因此区域划分需要平衡这种矛盾，在数据中心内需要重点考虑如何实现逻辑区域和物理区域的隔离，即非一一对应的关系。依据应用系统的要求，数据中心网络逻辑区域划分需要考虑如下原则：

1）根据安全架构，不同安全等级的网络区域归属不同的逻辑区域。

2）不同功能的网络区域归属不同的逻辑区域。

3）承载不同应用架构的网络区域归属不同的逻辑区域。

4）区域总量不宜过多，各区域之间保持松耦合。

6.3.3.2 网络区域划分

根据以上原则，网络的逻辑区域划分为外网区和内网区。

1）外网区：外网区根据功能的不同可划分为互联网（Internet）和外联网（Extranet）两个区域。这两个区域部署对外服务的应用系统，互联网提供互联网客户的访问，部署网银、网站、电商、学习环境等互联网业务；外联网提供第三方机构及大客户的访问，部署外联、托管等业务。

2）内网区：内网区根据功能的不同划分为网络功能区、服务器接入区和带外管理区。

① 网络功能区：无服务器部署，根据功能不同划分为核心区和广域网区两个子区域。核心区提供各个模块间的高速转发功能，广域网区负责数据中心与全行网络的互联互通。

② 服务器接入区：负责银行应用服务器的部署，根据功能的不同，可分为主机接入区和开放服务区两个子区域。主机接入区提供封闭式大型机的接入环境。开放服务区提供开放服务器的接入环境。在开放服务区，根据不同的应用架构进行区域细分，即普通开放服务区、网管安全区、存储区和语音区。

③ 带外管理区：这是个特殊功能区域，负责服务器和网络设备的带外组网和KVM/CONSOLE/HMC的组网。

数据中心网络逻辑架构如图6-2所示。

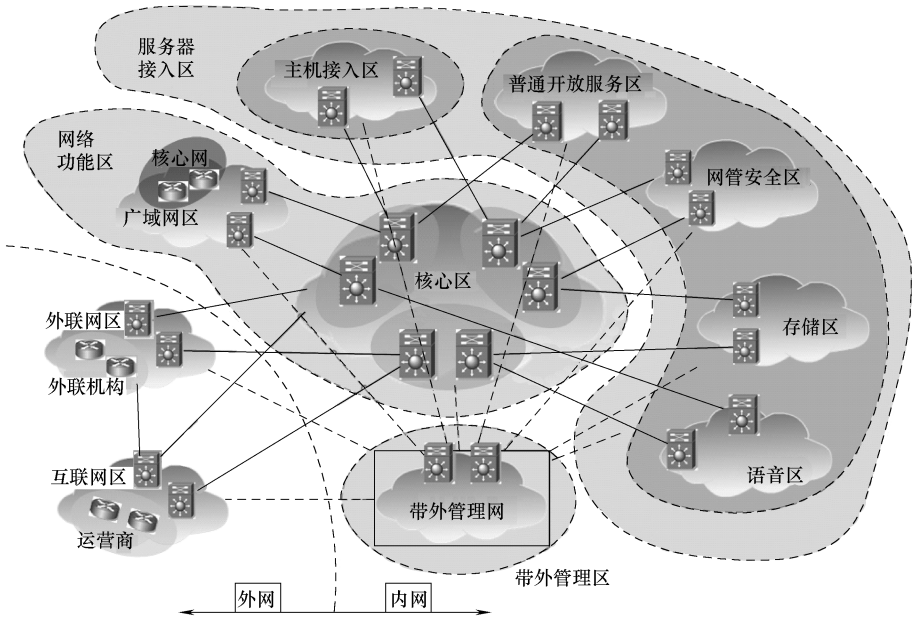


图6-2 数据中心网络逻辑架构

1. 核心区

大楼核心区提供楼内各汇聚间的转发功能。大楼核心的设计考虑高性能、高冗余，

并需要考虑网络运维的松耦合。

在数据中心，考虑到如下两个主要因素，对每栋大楼应设计独立的核心交换机。

1）每栋大楼采用独立的核心交换机，架构清晰，故障能有效隔离，运维松耦合，且端口资源充足，扩展能力高，性能得以保证。对于未来的迁移工作比较容易进行，每个以建筑物为单位的改造、搬迁等动作对其他建筑物没有影响，而且从系统上线步骤来看，初期不会全部部署，分步骤开启不同的建筑，对于节约能源是合理的选择。

2）假如三栋大楼中有一个核心，从冗余角度考虑，核心至少分布在两个不同建筑物中。在布线上，则需要三栋大楼作为一个整体的 TIA-942 模型来考虑。汇聚交换机到核心交换机互联需要大量的光纤资源，由于汇聚交换机到核心交换机之间至少为万兆链路，长距离的万兆链路需要单模模块和光纤支持，扩展能力受到限制。如果将来汇聚到核心交换机有需求，高速率单模模块的成本会很高。

因此设计每栋大楼有独立的大楼核心交换机，大楼核心由四台交换机组成，每个汇聚交换机均与大楼的每个核心交换机互联。在这样的设计下，每个汇聚交换机与其他汇聚交换机之间有多条等价转发路径，路径数量等于核心设备的数量。单台交换机的维护、宕机均不会对网络造成中断影响。

2. 广域网区

广域网区负责将数据中心与数据中心之间互联，数据中心通过核心网与各分行、各业务中心之间互联互通。

广域网区采用核心，接入两层设计，广域网的核心即为大楼核心，接入两台高端交换机。广域网的列中交换机是与核心网对接的 CE（Customer Edge，用户网络边缘）设备，运行 BGP（Border Gateway Protocol，边界网关协议），并和核心交换机之间运行 OSPF（Open Shortest Path First，开放式最短路径优先）路由协议，执行路由的重分发策略。

此外，广域网的列中交换机还负责数据中心之间的互联，数据中心的三栋大楼互联是通过三栋大楼的广域网区列中交换机形成一个环形网络来实现的。

3. 外联网区

外联网区负责完成所有外部机构的接入和服务器托管功能。外联网区汇聚与核心的互联采用开放服务区一致的方式，汇聚层采用 SIA（Service Insertion Architecture，服务插入架构）。外联区内部网络保持现有的结构。

在分行外联区与三个数据中心的外联区新建“核心网接入区”；此区内新增两台交换机，上联核心网。在核心网内建立“外联网 VPN”，用于分行外联区与总行外联区互通。本 VPN 只有分行外联区和三个数据中心的外联区的路由。分行外联单位若要访问总行外联区，必须经过网络地址转换后才能进入核心网。外联区防火墙负责安全级别以及区域间的访问控制。

4. 互联网区

数据中心互联网区逻辑结构计划分为四个区域，分别为互联网接入区域、核心区域、汇聚区域和服务器接入区域，如图 6-3 所示。

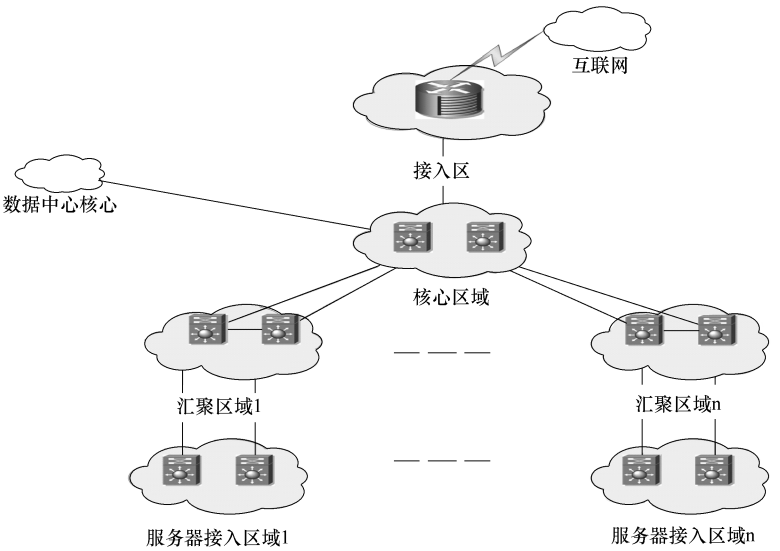


图 6-3　数据中心网络互联网区逻辑架构

各区域的作用如下：

1）核心区域：是整个互联网区的核心；作为核心节点对各区域进行连通，并对数据进行高速转发。

2）互联网接入区域：实现互联网区域与运营商的互联互通，是互联网用户访问中国建设银行的入口，执行域名解析、路由选择、流量清洗等职能；它是阻止互联网攻击的第一道防线。

3）汇聚区域：连接核心区域与多个服务器接入区域之间的中间区域；根据制定的策略对区域间互访的流量进行控制。

4）服务器接入区域：连接服务器的区域，负责对用户提供相关的访问资源。

5. 开放服务区（见图 6-4）

开放服务区负责开放系统的接入和网络服务功能。开放服务区采用三层设计，核心为大楼的核心，汇聚层采用两台高端交换机，接入设备应根据布线方式或服务器密度进行选择。

开放服务区是数据中心服务器接入数量最大的区域，设计时可按物理机房划分汇聚，而且支持全部服务器接入。该区域设计时考虑了集中实现基础设施云的理念，可以实现动态、自动化的资源供给。网络设计既要具备二层扩展能力和虚拟机感知能力，也要兼顾考虑机房物理属性的特征，完成物理视图向逻辑视图的转变，实现物理视图的标准化程度提高和逻辑视图的清晰明确。

开放服务区的综合布线和接入设备可根据机房类型进行选择，如高密、普密机房，应采用架顶模式部署，采用支持板卡延伸技术或多对一虚拟技术的 1U 交换机。如低密机房，则采用列头、列中模式部署，采用高密度的模块化交换机。对于接入设备来说，需要支持多对一虚拟化或板卡延伸简化管理、虚拟机感知（支持全部或

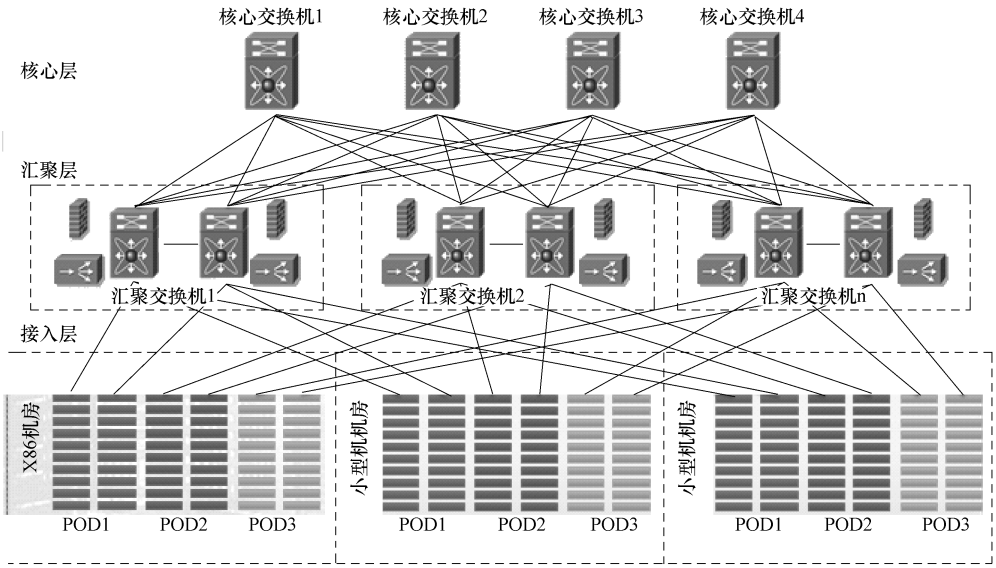


图 6-4　开放服务区

至少两种如下协议：802.1Q、802.1Qbg 和 802.1Qbh）、具备较低的上行收敛比或有较好的扩容能力。接入设备负责服务器的生产、带外、NAS 接入，与汇聚设备通过互联网中继模式互联。

POD 接入通常采用几种模型接入方式，根据需求不同使用不同的模型：

1）每台机柜使用架顶交换机连接到列中柜列中交换机，再通过列中交换机连接到汇聚交换机最终连接到大楼核心交换机。以 NEXUS 系列交换机为例，架顶采用 NEXUS2000，列中交换机采用 NEXUS5000，汇聚交换机和核心交换机采用 NEXUS7000 交换机。

2）几台机柜共用架顶交换机连接到列中交换机，再通过列中交换机连接到汇聚交换机，最终连接到大楼核心交换机。

3）架顶交换机直接使用 NEXUS5000 交换机，再通过列中交换机连接到汇聚交换机最终连接到大楼核心交换机。

6. 存储区

服务器连接 NAS 时可采用两种模式：其一是对 NAS 流量较小的服务器，采用与生产公用网卡接入网络；其二是对 NAS 流量较大的服务器，为避免 NAS 流量对生产流量产生影响，采用单独的 NAS 网卡接入网络，为 NAS 网卡分配单独的 IP 地址，与 NAS 机头三层路由互通。

由于 NAS 需要为所有的业务系统提供服务，所以 NAS 建议采用分布式部署，即在每个大楼均部署 NAS 区，每个楼的 NAS 区为本楼的业务系统提供服务。在网络上，每栋大楼提供单独的 NAS 网络汇聚区，NAS 汇聚区直接与核心交换机采用多个万兆甚至

四万兆或千万兆链路互联，与其他汇聚区通过核心区提供的 ECMP（Equal-Cost MultiPath，等价多路径）互联互通，实现高吞吐率的需求。在 NAS 的部署上应考虑物理布局因素，尽量避免 NAS 跨楼为服务器提供存储服务，降低跨大楼核心的 NAS 传输量。

7. 语音区

语音区可提供呼叫中心系统语音呼入、呼出业务功能。该区域采用开放服务区一致的网络架构，由于语音流对延时敏感，设计该区域的防火墙仅对语音控制流进行安全策略的控制。

1）机房 1 层作为语音区运营商线路接入。

2）机房 2 层部署语音区运营商线路接入语音设备、语音区网络列中交换机、语音区汇聚交换机。

3）机房 3 层部署语音区语音设备、语音区网络列中交换机。

① 语音区设备采用高可靠性配置，多条线路多资源备份，双电源双引擎网络设备，实现稳定运行。

② 语音区均分为控制网络（私有网段）与生产网络（生产网段），分别接入网络交换机，实现物理隔离，以提高语音设备接入的可靠性。语音区设备通过语音区列中交换机，上联至楼层汇聚交换机，至大楼核心交换机，再与数据中心相关系统、各分行远端媒体网关、座席中心 IP 话机互联互通。

灾备切换模式需要与应用一并考虑，语音区在中继集中接入的前提下，可以通过运营商路由策略实现客户进线在数据中心间互转。

8. 网管安管区

网管安管区用于部署网络管理、安全管理、系统管理应用，提供 IT 管理的工具和平台，独立成区。网管安管区的网络架构与开放服务区的网络架构一致。

9. 带外管理区

数据中心的带外管理网包含服务器和网络设备的传统带外管理和各类其他设备管理接口接入功能。整体上带外管理网与生产网独立，采用独立的网络核心连接各个汇聚区；同时采用独立的骨干网络连接到其他等地数据中心实现远程管理。

为确保在服务器带外管理网出现故障时，KVM（Keyboard、Video、Mouse，键盘、显示器、鼠标）管理网络能正常应急。在设计中，尽可能地提高服务器带外管理网和 KVM 管理网络的独立性。考虑分属不同的二层网络，通过路由互通，网络层面松耦合。带外管理区架构如图 6-5 所示。

带外的接入网络分为三类设计：

1）服务器带外网卡接入：服务器的带外网卡和生产网卡实现统一接入，减少接入网的组网数量。服务器的带外在生产的汇聚上终结，通过划分 VRF（Virtual Routing and Forwarding，虚拟路由转发）方式确保与生产网络相隔离。汇聚上的 VRF 连接带外核心交换机。

2）网络设备带外 MGT 接入：在每个生产的汇聚交换机机柜内安放网络设备带外的列中交换机实现本汇聚网络设备的带外管理接入。根据规模，可以考虑一栋楼只设置一

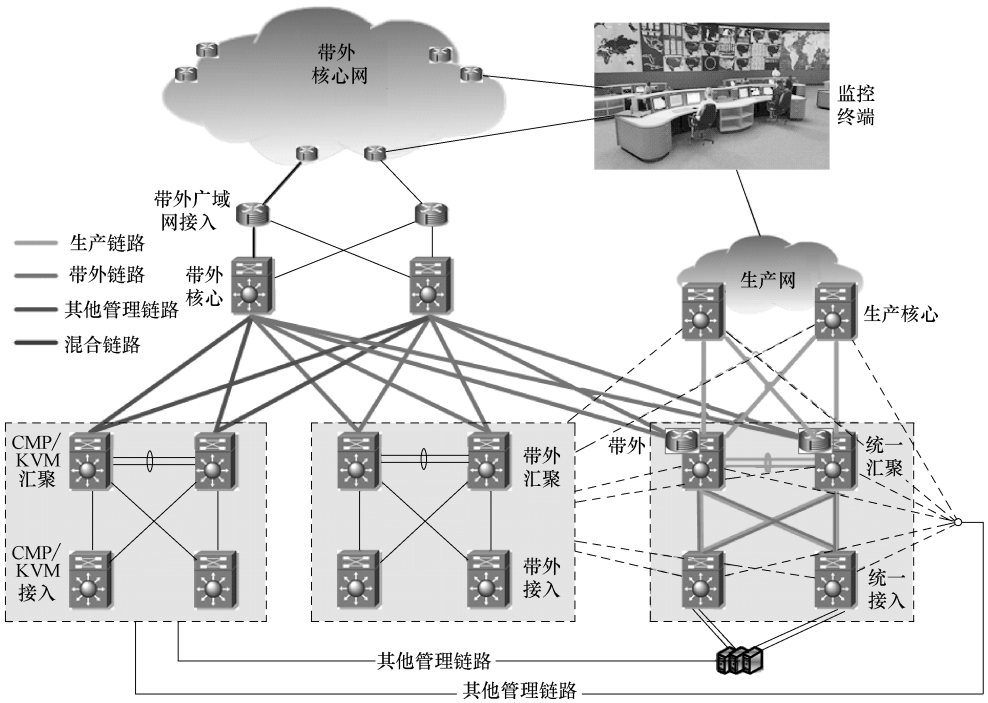


图 6-5　带外管理区架构

个网络设备的带外汇聚，在综合布线的 MDA、IDA、HDA 区均放置 1 ~ 2 个盒式交换机负责网络设备的接入。

3）各类其他设备接入：在新的数据中心，在每个 POD 放置服务器的各类其他设备列中交换机，同时也承担网络设备的 CMP/KVM 接入。如采用 CMP 接入的数量规模较大，可考虑每个楼层设置一个汇聚。

各类其他设备与带外共享核心交换机，但采用独立的汇聚，三层隔离避免了二层潜在的环路或广播风暴风险对彼此的影响，提高了管理通道的高可靠性。同时，也可以根据需要进行安全策略的部署。

采用 CMP 接入需要综合布线进行统一考虑，建议在每列采用列头柜汇集本列的 CMP 接入，列头柜采用铜缆布线到各个机柜，在列头柜部署 1 ~ 2 台盒式交换机提供 CMP 的接入。

6.4　存储资源规划设计

6.4.1　存储服务级别设计

1. SAN（Storage Area Network，存储区域网络）服务级别协议

SAN 存储服务级别分为 SAN 白金级服务、SAN 金级服务和 SAN 银级服务，见表 6-5。

表 6-5　SAN 存储服务级别协议

部署单元	SAN 白金级 （高可用性、高 + 性能）	SAN 金级 （高可用性、高性能）	SAN 银级 （中可用性、中性能）
可用性	99.999%（5.3 分钟/年）	99.999%（5.3 分钟/年）	99.99%（53 分钟/年）
性能	响应时间 < 5ms	响应时间 < 8ms	响应时间 < 10ms
	IOPS 5000 +	IOPS 3500 +	IOPS 2000 +
	吞吐量 2000MB/s +	吞吐量 2000MB/s +	吞吐量 1500MB/s +
可扩展性	扩展方式：整台存储扩容 扩容能力：最大 290TB 裸容量	扩展方式：整台存储扩容 扩容能力：最大 290TB 裸容量	扩展方式：整台存储扩容 扩容能力：最大 240TB 裸容量

2. NAS（Network Attached storage，网络附加存储）服务级别协议

NAS 存储服务级别分为 NAS 白金级服务、NAS 银级服务和 NAS 铜级服务，见表 6-6。

表 6-6　NAS 服务级别协议

部署单元	NAS 白金级 （高可用性、高 + 性能）	NAS 银级 （中可用性、中性能）	NAS 铜级 （中可用性、低性能）
可用性	99.999%（5.3 分钟/年）	99.99%（53 分钟/年）	99.95%（4.4 小时/年）
性能	响应时间 < 5ms	响应时间 < 10ms	响应时间 < 12ms
	IOPS 5000 +	IOPS 2000 +	IOPS 1000 +
	吞吐量 2000MB/s +	吞吐量 1500MB/s +	吞吐量 1000MB/s +
可扩展性	扩展方式：整台存储扩容 扩容能力：最大 290TB 裸容量	扩展方式：横向扩容 扩容能力：最大 3000TB（600GB 磁盘）裸容量	扩展方式：横向扩容 扩容能力：最大 10000TB（2TB 磁盘）裸容量

6.4.2　存储资源池设计

根据服务目录设计，资源池将采用与服务级别一一对应的设计，共分为 6 个资源池，其中 NAS 资源池有 3 个：NAS-白金资源池、NAS-银资源池和 NAS-铜资源池；SAN 资源池有 3 个：SAN-白金资源池、SAN-金资源池和 SAN-银资源池。

1. SAN 资源池

（1）SAN 资源池的构建方式（见表 6-7）

表 6-7　SAN 资源池的构建方式

部署单元	SAN 白金级 （高可用性、高 + 性能）	SAN 金级 （高可用性、高性能）	SAN 银级 （中可用性、中性能）
资源池	两个构建单元之间做镜像，作为基本单元，资源池由多个基本单元构成	两个构建单元之间做镜像，作为基本单元，资源池由多个基本单元构成	一台存储作为基本单元，资源池由多台存储构成

（2）SAN 资源池的详细定义（见表 6-8）

表 6-8　SAN 资源池的详细定义

SAN 资源池	资源池分类	定义
SAN 白金级	SAN-Platinum-A	两台设备之间做数据保护，资源池由多个基本单元构成
	SAN-Platinum-B	
SAN 金级	SAN-Gold-A	两台设备之间做数据保护，资源池由多个基本单元构成
	SAN-Gold-B	
SAN 银级	SAN-Silver-A	单台设备独立使用
	SAN-Silver-B	单台设备独立使用

2. NAS 资源池

（1）NAS 资源池的构建方式（见表 6-9）

表 6-9　NAS 资源池的构建方式

部署单元	NAS 白金级 （高可用性、高 + 性能）	NAS 银级 （中可用性、中性能）	NAS 铜级 （中可用性、低性能）
资源池	NAS 内部实现本地数据保护，作为基本单元，资源池由多个基本单元构成	1 台存储作为基本单元，资源池由多台存储器构成	1 台存储作为基本单元，资源池由多台存储器构成

（2）NAS 资源池的详细定义（见表 6-10）

表 6-10　NAS 资源池的详细定义

NAS 资源池	资源池分类	定义
NAS 白金级	NAS-Platinum-A	内部数据保护
	NAS-Platinum-B	内部数据保护
NAS 金级	NAS-Silver-A 金级	集群模式
	NAS-Silver-B	集群模式
NAS 银级	NAS-Copper-A 银级	集群模式
	NAS-Copper-B	集群模式

6.4.3　存储服务级别使用

6.4.3.1　存储服务级别决策

存储服务决策流程是依据存储服务目录和资源池配置规范，为应用数据选择合适的存储服务级别。存储服务决策流程分为数据可用性决策流程、数据性能决策流程和服务级别决策流程。

1. 数据可用性决策流程

应用按照重要级别进行划分，每个级别内不同数据的可用性要求有很大差异。本文将参考应用级别，把数据可用性作为存储服务的输入条件，详细的数据可用性分析见表 6-11。

表 6-11　数据可用性分析

数据类型 / 数据可用性 / 应用级别	生产数据	应急数据	历史数据	流水、日志数据	归档数据
A +	高	中	中	中	低
A	高	中	中	中	低
B	高	中	中	中	低
C	中	—	—	中	低

2. 数据性能决策流程

应用对 IO 的性能需求可分为四个级别，即高 +、高、中和低。响应时间（Response Time）、IOPS、MBPS（Million Bytes Per Second，兆比特每秒）取最高值，作为性能级别。例如，A 应用数据对 I/O 的响应时间要求为 5ms，IOPS 和 MBPS 要求都很高，性能级别应判断为高 +；B 应用数据对 I/O 的响应时间要求为 10ms，IOPS 要求为 2000 次，MBPS 要求为 1500MB/s，性能级别应判断为中。数据性能要求见表 6-12。

表 6-12　数据性能要求

性能级别	高 +	高	中	低
Response Time/ms	<5	<8	<10	<12
IOPS/次	5000 +	3500 ~ 5000	2000 ~ 5000	1000 ~ 2000
MBPS/次	2000 +	2000 +	1500 +	1000 +

3. 存储服务级别决策流程

应用数据使用方式可分为数据库和文件两种。数据库的特点是数据以块的形式存在，I/O 块小、随机访问、IOPS 高、MBPS 低等，适合 SAN 存储；文件的特点是数据以文件形式存在，多台服务器共享相同的文件数，I/O 块大、顺序访问、IOPS 低、MBPS 高等，适合 NAS。

根据数据访问形式的不同，存储服务目录采用的 NAS 和 SAN 两类技术分别对应数据库和文件。可用性级别和性能级别组成与服务目录的对应关系见表 6-13。

表 6-13　可用性级别和性能级别组成与服务目录的对应关系

服务级别（数据库）	可用性级别	性能级别	服务级别（文件）	可用性级别	性能级别
白金-SAN	高	高 +	白金-NAS	高	高 +
金-SAN	高	高	银-NAS	中	中
银-SAN	中	中	铜-NAS	低	低

6.4.3.2　性能规划

应用数据选择了合适的服务级别后，下一步要规划数据如何在存储上摆放，才能有

效利用存储空间、发挥更好的性能。

1. 资源池性能规划原则

应急存储与生产存储应保持相当的性能水平，比如同样为 HDS 公司的 VSP 或 EMC 公司的 VMAX。应急存储上可以同时部署多个应急库，应用的数据应尽量分散到所有磁盘上，应急库同时运行的性能指标可以高于存储的最大能力。例如，当应急存储部署 3 个应用，且同时运行时，响应时间可以达到小于 5ms 的级别，考虑大多数应急库平时处于空闲状态，这台存储可以再部署 3 个应用。

有明显增长周期的应用，应为应用预留相应的存储空间，满足其扩容需求，比如柜面业务集中处理系统存在明确的推广周期，在存储规划时就应该预留全行推广的容量和性能需要。

2. 存储设备性能基线

根据存储设备自身的性能指标数据，结合生产运行的实践数据，建行形成了一套可操作的性能指标体系，分为相对值指标和绝对值指标。规划时，要确保应用部署到存储后，相对性指标运行在“可接受值”范围内。

采用中位数性能指标作为性能判断标准。存储性能判断标准见表 6-14。

表 6-14　存储性能判断标准

指标类型	指标参数	参数描述	良好值	可接受值	非正常值	适应存储类型
相对值指标	CHP Busy(%)	前端端口处理器利用率	<50	50~70	>70	高端
	Controller Utilization(%)	控制器处理器利用率	<50	50~70	>70	中端
	Cache Write Pending(%)	待写 I/O 占 Cache 的比例	<30	30~40	>40	中、高端
	Read Hit Rate(%)	内存中读命中率	>95	90~95	<90	中、高端
	BED utilization(%)	后端控制器利用率	<50	50~70	>70	高端
	RAID Group Utilization(%)	后端磁盘组的利用率	<50	50~70	>70	中、高端
绝对值指标	LDEV Response Time/ms	逻辑设备响应时间	<5	5~8	>8	高端 SSD
	LDEV Response Time/ms	逻辑设备响应时间	<8	8~10	>10	高端 15kB
	LDEV Response Time/ms	逻辑设备响应时间	<10	10~15	>15	高端 10kB
	CHA IOPS/次	前端 IOPS	<5000	5000~8000	>8000	高端
	控制器 IOPS/次	控制器 IOPS	<10000	10000~20000	>20000	中端
	CHA MBPS/(MB/s)	前端 MBPS	<200	200~280	>280	高端、中端 4GB
	CHA MBPS/(MB/s)	前端 MBPS	<400	400~560	>560	高端、中端 8GB

在表6-14中，只有绝对值指标保持在“良好值”范围内，才能确保相对值指标运行在“良好值”范围，因此，在性能规划时，要对绝对值指标进行详细测算。性能取决于应用的读写比例、I/O块大小、I/O特性（随机还是顺序）、存储的缓存大小、读命中率、磁盘性能等多重因素。

1）LDEV Response Time（逻辑设备响应时间）。存储的整体运行水平需要保持在“良好值”范围内，才能按照服务级别提供的响应时间。

2）CHA（前端通道卡）IOPS。服务目录里定义了每个级别标配的前端端口数量，在性能规划时，还需要对IOPS进行详细测算，进而评估端口数量是否能够满足应用需求。比如应用A的IOPS需求为20000次，服务级别决策为金级，标配为2个前端端口，但是CHA IOPS应保持在5000次以下，需要4个前端端口才能满足性能需要。

3）控制器IOPS。中端存储由2个控制器提供前后端接入，每个控制器的IOPS小于10000次时，处于良好运行状态，应用的IOPS的需求总量不能超过10000次。

4）CHA MBPS。前端端口利用率为50%，最大的吞吐量=前端端口速率×50%。

第 7 章
私有云服务设计

7.1 云服务设计思路

7.1.1 云服务定义及设计原则

云服务是由云计算平台提供者将 IT 能力以面向用户的服务形式来进行包装和集成，并通过云管理平台和 Internet 或者 Intranet 渠道向云服务消费者（用户）来提供的一种服务。

云服务所提供的 IT 能力包含了服务功能和服务质量两个方面。云服务是云管理平台的核心内容，同时是云计算技术实现和业务应用的结合点。开发好的云服务需要发布注册至云管理平台，云管理平台需要将所有的云服务面向消费者（用户）实现交付和管理。一个云服务由云服务开发者定义和创建并发布至云管理平台。当云服务消费者提交服务请求并被批准后，云服务提供者将创建一个云服务实例，向云服务消费者提供云服务。

云服务实例是云服务在生产环境下存在的表现形式。一个云服务可以集成和使用其他的云服务，通常一个高阶的云服务需要使用几个低阶的云服务进行底层构建。

云服务可以分为 IaaS、PaaS 和 SaaS 类型，也可以服务消费者的不同来区分（例如，公有云、私有云和混合云）。

进行云服务设计时，要考虑使用一套成熟的、可行的方法，根据不同的平台、不同的功能划分不同层次的组件，由组件组装成不同的云服务。云服务设计时主要遵从以下几个原则：

（1）通用性原则　云服务具有通用性，适用于较多场景，而不是极少被使用到。

（2）可复用性原则　云服务采用标准、统一的开发方法，开发过程中各类文档、代码、脚本等尽量多地可用于其他云服务的开发，避免重复开发与减少工作量。

（3）可维护性原则　开发出的云服务在云管理平台上能够容易配置、使用和修改。

（4）覆盖全生命周期原则　云服务要覆盖到所管理资源对象的全生命周期，从资源的准备、供给、维护、扩容直到最终的资源回收。

7.1.2　云服务与云管理平台的关系

在一个云服务生命周期中，主要经过开发、运行和回收三种状态，如图 7-1 所示。在服务的开发阶段，完成云服务的业务定义、结构模型和操作模型的设计；在运行阶段形成服务实例并向外提供云服务能力；在回收阶段将服务下架，终止提供该云服务。

云服务的业务定义要通过云服务管理平台发布至服务目录中让使用者可见。

云服务的结构模型和操作模型需要由云服务管理平台调用并维护其与该云服务之间的关系，确保云服务使用者申请使用云服务时，可以通过结构模型和操作模型进行供给并形成服务实例以提供云服务能力。

对于准备终止和回收的云服务，需要云服务管理平台将该云服务从服务目录中下线，并调度停止服务实例，回收资源。

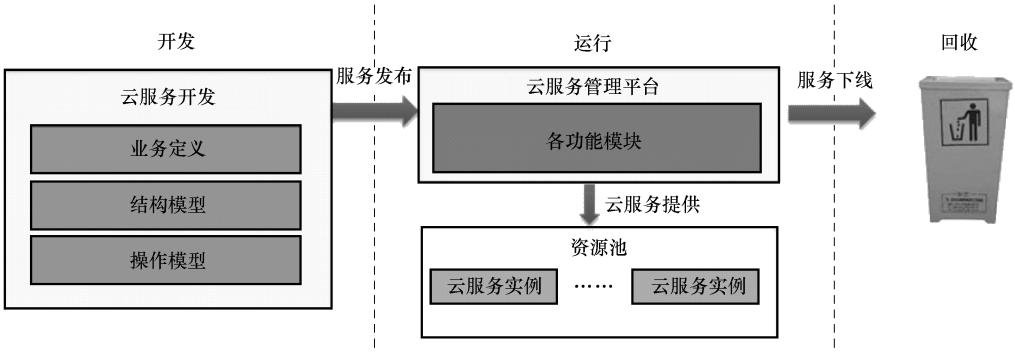


图 7-1　云服务生命周期

7.1.3　云服务描述模型

云服务的模型可以分为三部分，即业务定义、结构模型和操作模型，如图 7-2 所示。

7.1.3.1　业务定义

云服务的业务定义，是从业务视角来描述云服务的服务能力，包括可以提供的功能和服务质量等，使云服务用户能够更容易理解云服务可以做什么，从而判断出云服务是否能够满足其需求，并做出是否需要申请和使用该服务的决定。

7.1.3.2　结构模型

云服务结构模型包括一个或多个服务单元 DU（Deployment Unit），其物理存在形式

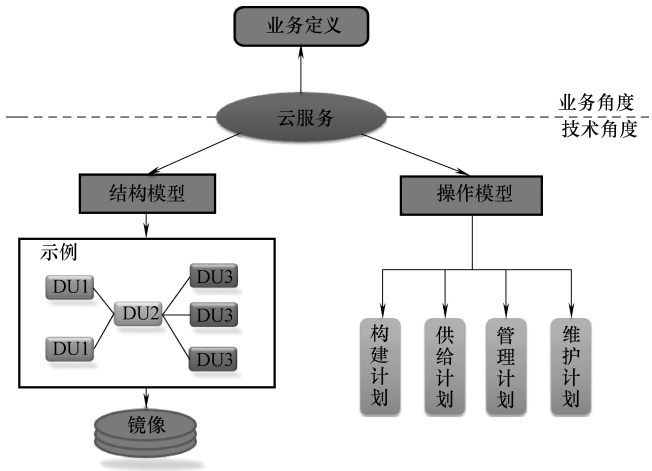


图 7-2　云服务模型

是镜像文件，可以从结构模型中看出该云服务供给之后的组织形式。

1. 服务单元 DU

服务单元 DU 是在虚拟层云服务中部署单元的表述，服务单元 DU 是云服务的最小组成单位，根据云服务需求，一个或一个以上的服务单元 DU 可以组成一个云服务的部署模式，相当于组成云服务的“零部件”，其物理存在形式是 Image 镜像文件。一个或一个以上的服务单元 DU 通过编排、集成和测试，实现一个云服务，服务单元 DU 与云服务的开发流程如图 7-3 所示。

图 7-3　服务单元 DU 与云服务开发流程

可以举个例子，制作一个云服务，就像汽车生产商从各个零部件生产商采购汽车零部件（比如方向盘、发动机、座椅、大灯、轮胎等），通过编排、集成和测试，最后形成不同物理形态（如颜色不同、外形不同）但是结构类似的汽车的过程，如图 7-4 所示。

2. 镜像文件（Image）

镜像是服务单元 DU 的物理存在形式，服务单元 DU 包含的内容通过镜像封装。镜像和 DU 是一对多的关系，因此，镜像的开发包括了镜像文件的制作，以及镜像与对应的服务单元 DU 之间的映射关系的建立。

从云服务中，镜像与服务单元 DU 的对应关系，可以看出需要开发的服务单元 DU

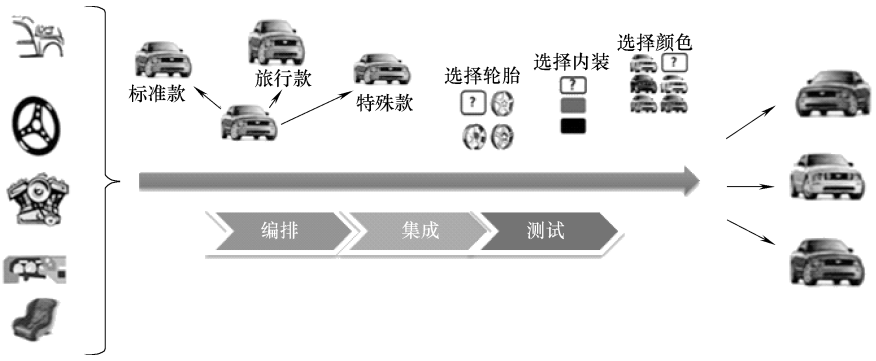


图 7-4　汽车构建流程

的数目。镜像在开发过程中，会开发多个 DU，如果有镜像可以满足，那么直接建立映射关系即可；如果没有，则需要新开发、测试和发布新的镜像，镜像开发过程如图 7-5 所示。

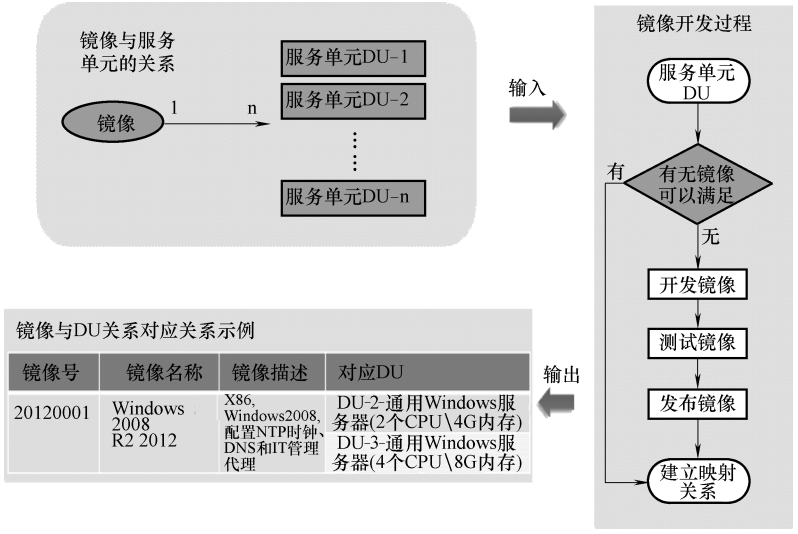


图 7-5　镜像开发过程

7.1.3.3　操作模型

操作模型分为构建计划、供给计划、管理计划和维护计划。

1. 构建计划

构建计划是创建一个镜像的过程，构建计划的同时记录镜像的版本号、配置文件等关系。

2. 供给计划

供给计划是申请取得虚拟环境，并将按照云服务结构模型的部署模式将镜像部署在分配好的虚拟环境的过程。

3. 管理计划

管理计划是当一个云服务被分配，成为虚拟的运行环境后需要处理的流程模型。分

为配置脚本和管理注册两个部分。配置脚本包括网络配置、安全配置、管理配置等；管理注册包括在管理平台的配置和虚拟环境的配置。

4. 维护计划

维护计划处理的是在服务部署完成后，服务运行生命周期中需要执行的各种运维操作的流程模型。例如服务启停、扩容缩容、备份恢复、灾备切换、服务下线等。

7.1.4 云服务发布

根据云服务和云管理平台之间的关系，当一个云服务在开发完成后，至少向云管理平台应该发布如下内容。云管理平台在接受相关发布内容后，必须确保相关内容之间的关系的保持，并基于云管理平台的技术实现方案，完成进一步满足云服务自动供给的要求，例如流程的自动化实现等，以确保云服务提供的自动化。

按照云服务的模型，云服务发布内容见表 7-1。

表 7-1 云服务发布内容

项目类别	项目内容	备 注
业务定义	该云服务业务定义	
结构模型	该云服务使用服务单元 DU 清单	
	该云服务结构模型设计	
	该云服务新增的服务单元 DU 清单(可选)	只针对在服务开发过程中,现有 DU 清单不能满足而新增的情况
	该云服务使用镜像清单	
	该云服务新增的镜像清单(可选)	只针对在服务开发过程中,现有镜像清单不能满足而新增的情况
操作模型	该云服务的构建计划设计(可选)	只针对在服务开发过程中,现有镜像清单不能满足而新增的情况
	该云服务的供给计划设计	
	该云服务的管理计划设计	
	该云服务的维护计划设计	

7.2 云服务开发过程

7.2.1 开发过程中的角色定义

云服务开发过程中主要的参与角色包括云服务设计人员、云服务测试人员、服务单元设计人员、镜像制作人员、专业领域技术人员和云平台管理人员。

（1）云服务设计人员　作为该云服务的负责人，在整体云服务开发过程中发挥作用。

（2）云服务测试人员　测试整个集成后云服务的人员。

（3）服务单元设计人员　服务单元 DU 的设计者。

（4）镜像制作人员　制作云服务镜像的人员，具有一定的专业领域知识。

（5）专业领域技术人员　对云服务所涉及的领域有较多经验的技术人员，涉及该领域的自动化程序、脚本等的开发者。

（6）云平台管理人员　云平台的配置、维护人员。

云服务开发过程中所涉及的人员角色关系如图 7-6 所示。

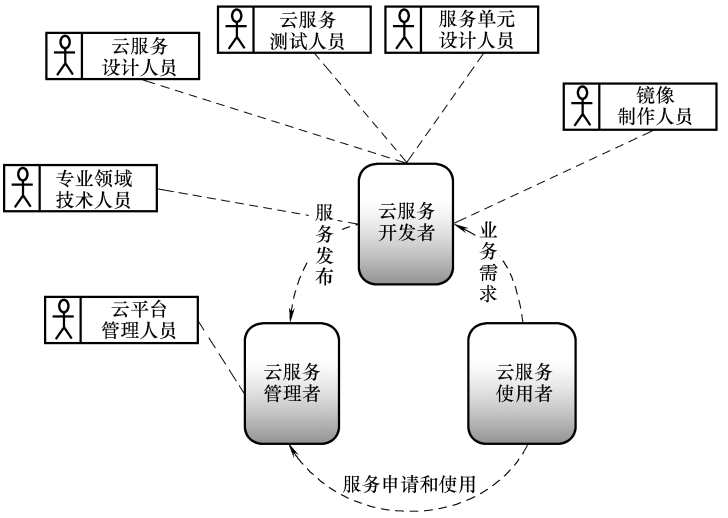


图 7-6　云服务相关人员角色

7.2.2　云服务开发过程

云服务的开发过程分为云服务业务定义设计、云服务结构模型开发、云服务操作模型开发、集成测试、服务发布等过程，如图 7-7 所示。

7.2.2.1　业务定义设计

此阶段分为业务需求分析和云服务业务定义两个子任务，见表 7-2。

表 7-2　业务定义设计阶段的任务

名　称	描　述	负责人员	输　入	输　出
业务需求分析	通过对需求的分析和理解，选择并决定是否应开发一个新的云服务，并将所要实现的云服务从功能性和非功能性进行分析，给出需求分析说明书，说明需要实现的功能和质量	云服务设计人员	需求说明	×××云服务需求分析说明书
云服务业务定义	给出此云服务标准的业务定义，包含服务的能力和服务的质量	云服务设计人员	需求分析说明书	×××云服务业务定义说明

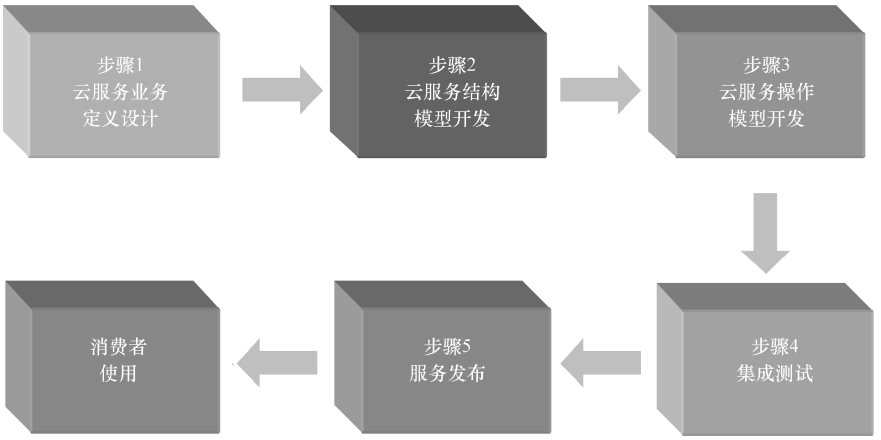


图 7-7　云服务开发过程

7.2.2.2　结构模型开发

此阶段分为选择并分析备选服务单元 DU、设计云服务部署模式和新建（封装）一个新的服务单元 DU 等子任务，见表 7-3。

表 7-3　结构模型开发阶段的任务

名　称	描　述	负责人员	输　入	输　出
选择并分析备选服务单元 DU	根据云服务定义和需求分析，选择支持该服务的备选服务单元 DU，给出选择的 DU 列表。如果需要新增服务单元 DU，则进行任务“新建（封装）一个新的服务单元 DU”	服务单元设计人员	云服务业务定义说明	×××云服务 DU 目录清单
设计云服务部署模式	根据需求分析，结合已选服务单元 DU 清单，设计该服务的部署模式，形成云服务结构模型	服务单元设计人员	服务单元 DU 清单 需求分析说明书	×××云服务结构模型设计
新建（封装）一个新的服务单元 DU	如若现有 DU 不能满足该云服务需求，则需执行此任务，根据需求，结合各领域服务，设计新增服务单元 DU	服务单元设计人员	云服务业务定义说明	×××云服务新增服务单元设计

7.2.2.3　操作模型开发

此阶段分为制定及设计构建计划（包括开发镜像和测试镜像）、开发设计供给计划、开发设计管理计划和开发设计维护计划等子任务，见表 7-4。

表 7-4　操作模型开发阶段的任务

名　称	描　述	负责人员	输　入	输　出
制定及设计构建计划	根据结构模型设计和服务 DU 目录，选择支持该服务单元 DU 的备选镜像，如果现有镜像可以满足，则此任务可以跳过，若不能满足，则需要设计新增镜像	镜像制作人员	云服务结构模型设计、镜像目录清单	×××云服务新增镜像说明

（续）

名　称	描　述	负责人员	输　入	输　出
开发镜像	对镜像封装范围和封装内容进行设计，并在具体部署环境中实验，设计镜像及建立DU对应关系	镜像制作人员	新增的服务单元设计	×××云服务新增镜像说明
测试镜像	在服务单元的DU测试环境中进行镜像的测试，保证镜像通过部署在指定的计算资源环境上能够建立服务单元DU	镜像制作人员	×××云服务新增镜像说明	×××云服务新增镜像测试报告
开发设计供给计划	根据结构模型，结合各领域操作，设计供给计划	云服务设计人员、专业领域技术人员	云服务结构模型设计、各领域操作汇总	供给计划设计
开发设计管理计划	根据结构模型，结合各领域操作，设计管理计划	云服务设计人员、专业领域技术人员	云服务结构模型设计、各领域操作汇总	管理计划设计
开发设计维护计划	根据结构模型，结合各领域操作，设计维护计划	云服务设计人员、专业领域技术人员	云服务结构模型设计、各领域操作汇总	维护计划设计

7.2.2.4　云服务测试

此阶段分为制定测试用例、制定测试计划、申请并搭建测试环境、进行集成测试和形成测试报告并评审等子任务，见表7-5。

表7-5　云服务测试阶段的任务

名　称	描　述	负责人员	输　入	输　出
制定测试用例	编写云服务集成测试用例	云服务测试人员	结构模型，操作模型设计结果	测试用例文档
制定测试计划	制订云服务测试计划	云服务测试人员	测试用例文档	测试计划文档
申请并搭建测试环境	根据测试用例和计划，申请和搭建有关的测试环境	云服务测试人员	测试用例文档、测试计划文档	测试环境
进行集成测试	进行测试	云服务测试人员	测试用例文档、测试计划文档	集成测试报告（评审前）
形成测试报告并评审	形成云服务总体测试报告，并进行评审，确保云服务功能，质量达标	云服务测试人员、评审人员	集成测试报告（评审前）	××云服务测试报告（评审后）

7.2.2.5　云服务发布

此阶段分为云服务业务定义发布、云服务结构模型发布和云服务操作模型发布等子任务，见表7-6。

表 7-6　云服务发布阶段的任务

名　称	描　述	负责人员	输　入	输　出
云服务业务定义发布	发布云服务业务定义,使用户可以了解云服务业务内容	云平台管理人员	云服务业务定义说明	服务目录
云服务结构模型发布	发布云服务结构模型,并与相关云服务对应	云平台管理人员	结构模型设计结果、测试报告(评审后)、×××云服务新增服务单元设计、×××云服务新增镜像说明	镜像目录 服务单元 DU 目录、×××云服务结构模型
云服务操作模型发布	发布云服务操作模型,并与结构模型建立相关关系	云平台管理人员	结构模型设计结果、操作模型设计结果、测试报告(评审后)	×××云服务操作模型

第 8 章
云管理平台设计

8.1 设计原则及思路

8.1.1 设计原则

云管理平台实现云环境下基础设施环境自动化管理，包括资源的自动发现、资源池纳管、资源的分配、资源部署等，同时还可对系统、网络、应用环境自动配置，实现应用自动化发布及变更自动化管理等功能。进行云管理平台设计时主要遵循以下几个原则：

1. 安全性原则

商业银行的数据中心承载着各种复杂和关键的业务，云管理平台处于数据中心的核心节点，最容易受到攻击，安全保证至关重要。云管理平台采用虚拟化层安全加固技术保证虚拟化平台的安全，加密的数据传输保证数据在网络链路中的安全，分权分域管理保证特定用户对系统的可控，提高安全性。

2. 先进性原则

一体化云管理平台利用云计算等先进技术和理念，突出云计算的价值，采用虚拟化的资源自动调配等先进技术，实现 IT 资源的联动，实现自动化管理、运维的新理念、新技术。

3. 成熟性原则

一体化云管理平台承载着运维管理的关键任务，在进行系统建设时要充分考虑系统的成熟和稳定。系统的软件设计和硬件采用的服务器、存储、网络等设备经过充分测试和验证，保证各子系统的稳定和可靠。

4. 可靠性原则

一体化云管理平台采用冗余、高可用集群、与底层松耦合等特性从硬件设备、链路

等方面充分保证系统的整体可靠性，降低系统对单独设备可靠性的要求。

5. 可扩展性原则

一体化云管理平台具备模块化扩展能力，实现简单、快速部署和上线。同时，与相关系统之间是松耦合，可保证一体化云管理平台的最大灵活性和可扩展性。

8.1.2 建设思路

以中国建设银行股份有限公司（简称建行）为例，介绍银行云管理平台的设计思路。

建行的云数据中心台研究和建设思路可归纳为：以最新技术趋势为蓝图，以新一代成果为依托，以 ITIL 实践为指引，以目标需求为导向，以自动化为手段，以一体化管理为目标，以云管理平台为纽带，以自主开发为保障。

1. 借鉴

充分学习和吸收业界有关云计算的理论和技术，把握云管理平台和云数据中心的技术趋势，确保建行的云管理平台的技术路线的正确性和先进性。

2. 衔接

1）衔接建行“新一代”系统和技术架构关于云计算的研究成果，确保云管理平台能有效满足未来业务的需求。

2）衔接建行数据中心多年来在流程、制度、IT 服务管理方面的积累，确保云管理平台能满足金融行业的要求。

3. 自主创新

云计算和云管理是非常前沿的理念和技术，尤其是金融行业在云管理方面目前还缺少非常成熟的产品和方案。在此情况下，不能被动地接受厂商方案，必须坚持自主和创新原则，以需求和目标为导向，寻找最佳的解决方案。

4. 整合

云计算和云管理需要在最佳实践的指导下，对多种技术和理念的整合与集成，实现一体化管理。建行在数据中心建设上的大量投资，不能完全推倒或遗弃，而是要在新的体系框架下集成适用的模块，这样既加快了项目的进度，又保护了原有的投资。

5. 持续改进

云管理平台是一个持续改进，不断演化的系统。在考虑长期的业务需求下，以服务为导向，构建一个可持续的架构。持续服务改进，不断地释放与改进系统功能。

8.1.3 服务质量

8.1.3.1 可靠性

1. 持续运行概率

1）服务时间为 7 × 24h。

2）可支持在线维护。

3）可忍受的停止服务时间为30min，不允许数据丢失。

2. 可用概率

除正常的事务处理时间等待外无其他等待情况，可用概率接近100%。

3. 备份与恢复

可提供完整的备份方案，包括软件备份和数据备份。备份恢复所需要的时间小于30min。

4. 灾难备份

需要具有高级别的灾备能力，提供完善的灾难备份策略。

8.1.3.2 应用适应性

1. 部署指标

云管理平台组件采用Web应用部署方式对外提供服务，支持多种应用服务器，采用参数配置的方式适应不同的部署环境。

2. 应用生命周期管理能力

对外提供两种接口：产品化的功能集成模块和对外的Web Services API接口；同时通过集成、联邦和调和特性集成其他的流程或者数据源。

8.1.3.3 可管理性

1. 状态发布能力

部署在应用服务器中，利用应用服务器提供的工具可以检查和查看组件的部署状态、调整组件部署参数、启动或者停止组件。提供自管理功能，用于检查和查看组件核心服务的基本运行状况。

2. 自动化管理

需提供自动化管理能力，出现故障后可自动修复。

8.1.3.4 安全性

1. 完整性保障能力

平台内部的业务操作，支持中断和退回机制，保证业务的完整性。所有的数据操作都有严格的事务控制。

2. 审计保障能力

对所有的关键操作、安全威胁和系统管理操作都记录详细的操作日志，供问题追溯和审计使用。

8.2 架构设计

8.2.1 架构描述

云管理平台总体架构包括云管理、配置管理数据库和云服务三个模块，如图8-1所示。

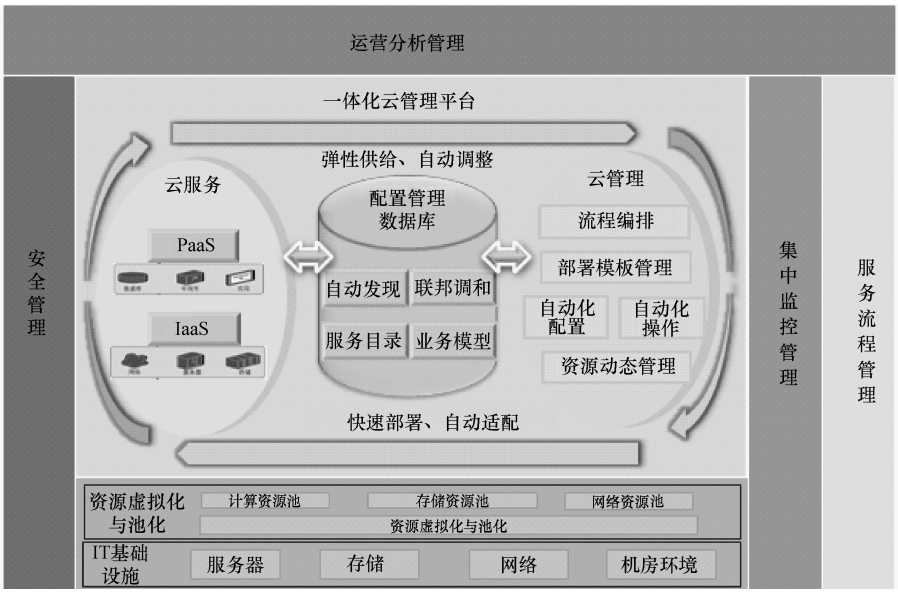


图 8-1　云管理平台总体架构

云管理模块包括流程编排、部署模板管理、自动化配置、自动化操作和资源动态管理；配置管理数据库包括自动发现、联邦调和、服务目录和业务模型；云服务模块包括 IaaS（计算资源、存储资源和网络资源）和 PaaS（数据库、中间件和相关应用）。

云管理平台以配置管理数据库为桥梁和纽带，实现云管理与云服务的协同与融合，通过云管理平台实现系统的快速部署和 IT 基础设施资源的主动适配，通过云服务平台提供 IT 资源的弹性供给和自动调整。

8.2.2　逻辑架构设计

云管理平台逻辑架构如图 8-2 所示。

云管理平台组件设计核心：

1）云管理平台融合现已存在的 ITIL 流程，将云管理纳入建行现有管理框架之中。

2）以 CMDB（Configuration Managment Database，配置管理数库）为核心追踪记录资源变化，实时反映云环境资源与配置状态，构建计费模型，提供准确计费依据。

3）以自动化调度和部署为基础，构建标准、快速、灵活的资源自动分配与部署平台，同时构建长效的持续合规与安全审计机制。

4）以智能监控为手段，探测与预警性能与故障事件，确保云服务满足用户期望与服务级别。

云管理平台依据云管理模型的框架思路按照层次化、松耦合的理念进行产品架构设计，根据功能需求，纵向层次设计和横向功能设计如下：

（1）纵向层次设计

1）资源服务层：实现用户访问系统的前端，支持自助服务的模式，将资源请求以

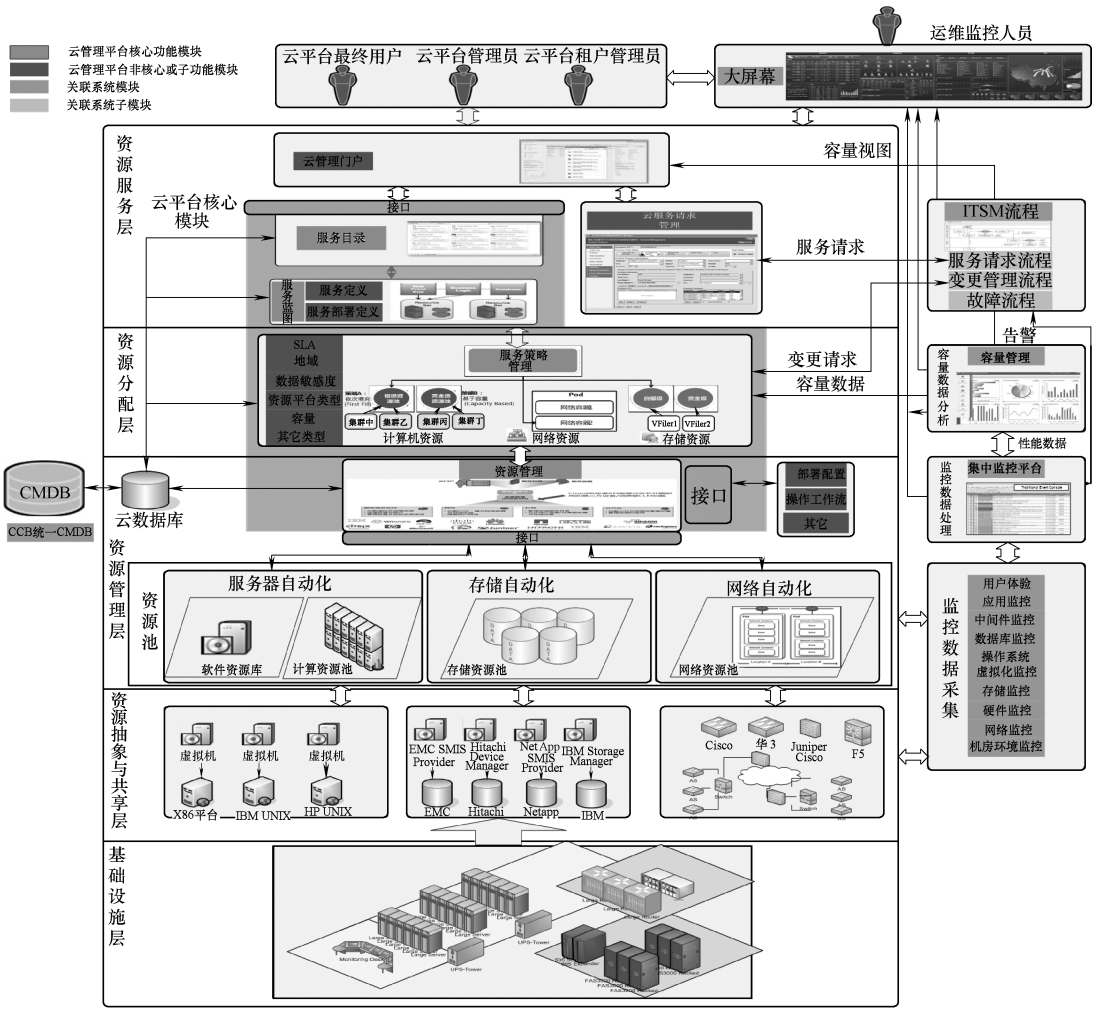


图 8-2　云管理平台逻辑架构

注：EMC SMIS Provider：EMC 存储管理接口；Hitachi Device Manager：日立存储设备管理；NetApp SMIS Provider：Netapp 存储管接口；IBM Storage Manager：IBM 存储管理。

一种标准化和约定的方式进行描述以减少沟通成本并为后续自动化节点提供必要的参数，同时根据最佳实践和用户需求构建审批及变更流程，以契合用户个性，并提供持续调整和优化的能力。

2）资源分配层：根据用户的请求，按照既定策略对资源进行分配，分配资源包括网络资源、存储资源、计算资源等。

3）资源管理层：实现了对具体资源的自动化配置操作，通过统一的平台将资源对象的操作逻辑固化和参数化，并通过配置管理系统和流程系统的整合实现所需的自动化，有效支撑资源环境部署、变更和回收。

4）资源抽象与共享层：第三方资源管理平台，如虚拟机环境、存储环境相关管理平台，通过该层实现对异构环境的统一管理。

5）基础设施层：该层主要包括用户 IT 环境中实际物理设备，包括服务器、网络、存储等资源。

（2）横向功能设计

1）监控管理：提供实时监控实现对云部署环境的可用性以及性能进行监控，对云资源的整体使用情况进行分析。

2）流程审批管理：实现对云服务请求的审批管理，该部分将与现有 ITSM 管理平台整合，实现审批管理流程的统一化。

3）配置信息管理：实现云平台配置信息的统一管理，云平台的相关配置信息将与当前 CMDB 平台同步，将云环境的相关配置信息导入到云平台中，实现配置信息的统一管理、统一展现、统一应用。

4）容量管理：提供云管理平台各资源运行环境和业务相关性能数据，对云管理平台进行统一容量分析、容量预测及规划和成本分析。

8.2.3 功能架构设计

云管理平台的功能层次自下而上包括：资源管理层、平台管理层、应用管理层和云管理层，如图 8-3 所示。

图 8-3　云管理平台功能架构

1）资源管理层：主要实现对计算资源、网络资源、存储资源的纳管、分配及调整。

2）平台管理层：主要实现对计算资源操作系统环境，HA 环境，监控、自动等运维工具代理、备份恢复等安装、配置。

3）应用管理层：实现数据库、中间件、应用软件等的部署、配置及管理。

4）云管理层：实现云管理平台的核心功能，包括门户管理、服务目录、部署模式、服务策略管理、资源管理、配置管理、服务请求管理、资源监控及性能分析等。

此外，基础设施云技术标准及规范是云管理平台建设的支撑。建行信息技术管理部统一安排编写了相关的技术标准规范，为云管理平台的实施奠定了基础。

8.2.4 部署架构设计

依据集中管理原则，云管理平台采用集群方式部署以保证系统性能与可用性。云管理平台部署架构如图 8-4 所示。

1）云管理平台将通过位于主生产中心的系统集中管理所有资源池，包括主生产中心、同城和异地灾备中心以及分行的资源池。

2）每个资源池除了硬件设备之外，还包含适应于该类型资源池的平台管理和虚拟化管理系统。此外，为了提高部署性能，在每个资源池中保存适应该类资源池的镜像和脚本文件副本。

3）为了考虑两地三中心灾备模式，实现灾备，在同城和异地灾备中心也必须部署完整的云管理平台和数据并与主生产中心的系统保持同步。

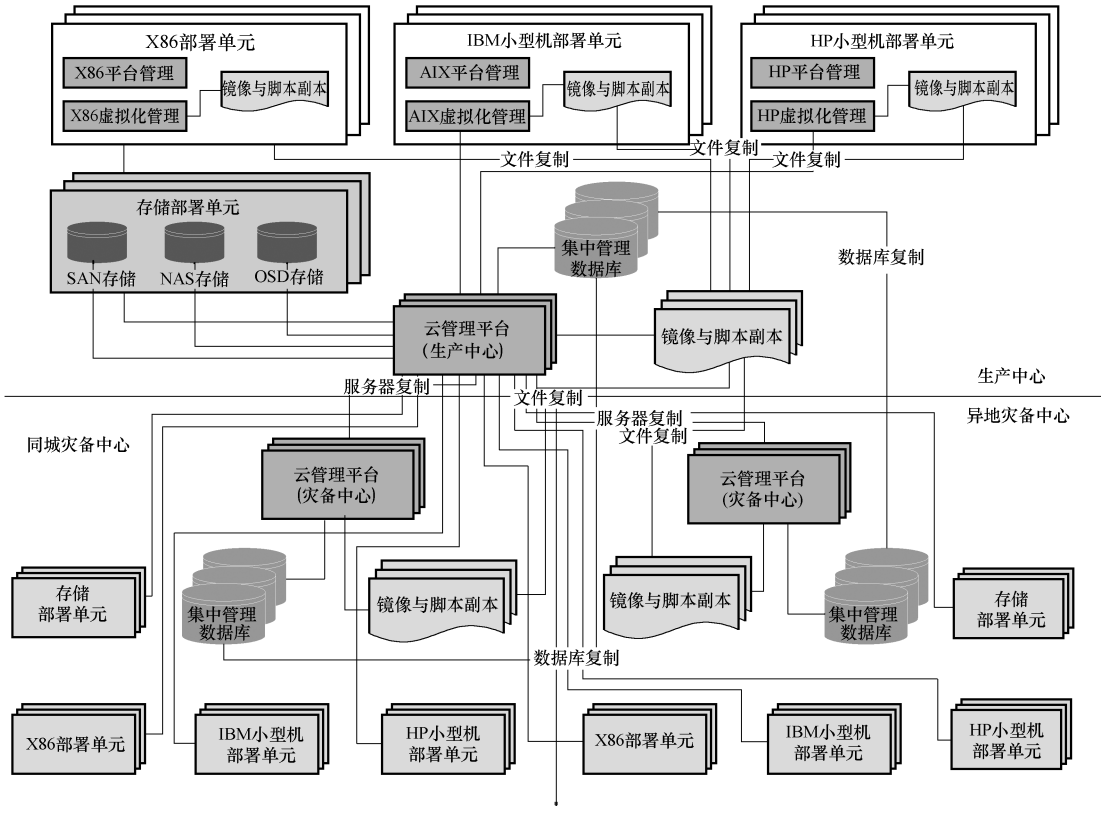


图 8-4　云管理平台部署架构

8.3 功能描述

从服务提供者和服务消费者的视角分析云管理平台，其功能主要分为云资源管理、云服务管理、配置管理、流程管理、监控管理及容量管理。

8.3.1 云服务管理

1）自助服务门户：提供用户访问接口，实现云资源的展示、请求、管理等用户操作接口。

2）服务请求管理：支撑云服务请求的流程管理；涉及审批流程与审批角色，与现有服务请求和变更流程接轨。

3）服务目录：云管理平台对最终用户提供的服务清单及选项。

4）部署模式：云管理平台各服务项目具体定义和部署设计。

8.3.1.1 自助服务门户

自助服务门户是整个云管理平台的前端，是各类角色用户与系统交互的接口，该门户将提供云服务请求、管理、定义、权限分配等全生命周期的各类操作活动接口，在服务请求阶段，该门户对外提供服务目录以供用户进行选择。

8.3.1.2 服务请求管理

负责用户服务请求提交后的审批管理，在本次方案中用户请求管理将由云管理平台解决方案中的服务请求模块实现，但相关审批过程需要与建行当前 ITSM 管理平台服务请求流程对接集成。IT 服务管理平台审批完成后将结果返回给云管理平台并由云管理平台进行后继处理。

8.3.1.3 服务目录

服务目录是云管理平台对外提供的各类服务项目。

1）服务目录的主要用途包括：

① 容纳和管理服务产品。

② 服务的用户语言描述。

③ 通过自助服务门户向用户展现。

2）服务目录描述的主要内容包括：

① 描述服务产品，以业务语言描述和关联费用。

② 向用户展现可消费的服务产品。

③ 提供各种不同服务创建的灵活性，创建服务器、部署应用、加载配置等。

3）服务目录的定义分为三个层次，其主要内容包括：

① 服务的名称及服务内容的描述。

② 服务的分类：从部署模式等角度分拆服务的类型，细化差异化服务内容，以满足同一服务内容的不同级别或要求的差异化需求，并描述其计较方式。该部分每类服务的定义又分为服务交付和请求交付两部分，其中请求交付为展现给用户请求者的信息定义，服务交付则定义该服务的具体内容，服务交付将与部署模式中的服务定义和服务部署定义之间关联。

③ 服务的选项：描述该服务可选择的项目和计价方式。

8.3.1.4 部署模式

1. 部署模式主要目的：

1）定义服务实现相关的基础架构组件、软件包及连接关系。

2）以部署模型描述一种服务如何以不同的规格在基础架构中部署实现。

2. 部署模式内容

部署模式将与服务目录第二个层次中的服务交付相对应，从而实现服务目录到部署模式的映射，将用户看到服务的描述与服务部署框架对应起来。部署模式包括以下两个部分：

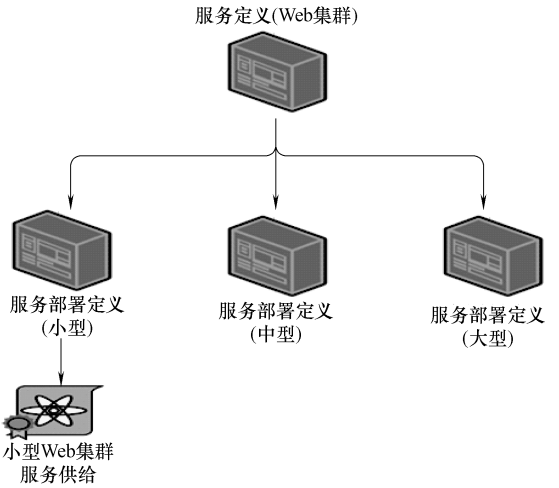


图 8-5 服务定义

（1）服务定义（见图 8-5）

1）提供服务的功能定义（包括单台服务器类型的服务和多层应用环境平台类型的服务）。

2）为服务表明操作系统、软件及连接（组成服务的各功能组件及相互关联关系）。

3）使用介质库作为集中的软件定义库。

（2）服务部署定义

1）提供服务的部署态视图（见图 8-6）：定义部署服务的一种或多种方式（如虚拟部署形态、物理部署形态等）。

2）说明服务运行所需要的资源。

3）由服务器对象、存储对象和网络对象（含负载均衡/防火墙规则）组成。

在部署模式中，管理员可针对资源的构成，如虚拟机模板、硬件规格、网卡个数及所在 VLAN、安装软件及部署方式等诸多资源组成要素进行定制。

3. 部署模式示例

平台以可视化方式呈现定制结果，如图 8-7 所示。

部署模式与服务目录、虚拟机模板和自动化模块的关系如图 8-8 所示。

云管理平台通过统一调度模块衔接上层部署模式和底层自动化操作。当用户提交服务申请后，云管理平台的调度模块根据用户的请求，完成静态 IP 地址获取、虚拟机克隆、软件安装与设定、更新配置数据库等操作，实现端到端的服务创建。

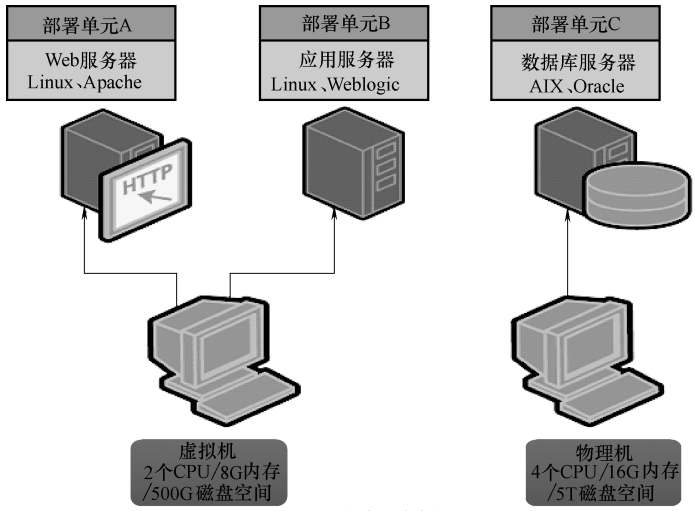


图 8-6　服务部署态视图

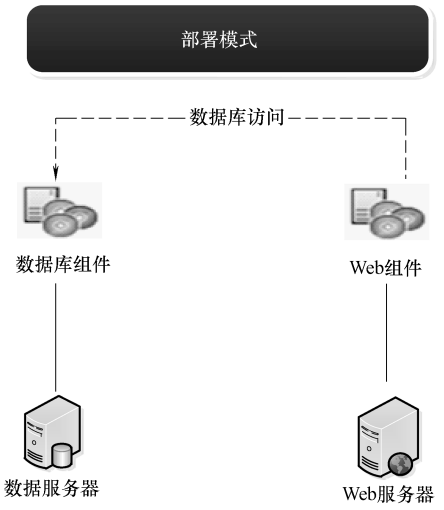


图 8-7　服务定制结果

8.3.2　云资源管理

资源管理层主要对虚拟化或物理资源进行资源纳管和池化，主要是通过服务器自动化、网络自动化、存储自动等管理工具将纳管后的资源管理抽象为统一的接口，并通过该接口实现系统各类资源的部署。该层将承担承上启下的作用，自下而上对各类资源进行纳管，自上而下对资源进行实例化部署和操作。

8.3.2.1　资源池化及纳管

所有资源需要在云管理平台进行登记和纳管才能实现资源的池化。云管理平台主要有三类资源需要统一管理，即计算资源、存储资源和网络资源。云管理平台资源对象如图 8-9 所示。

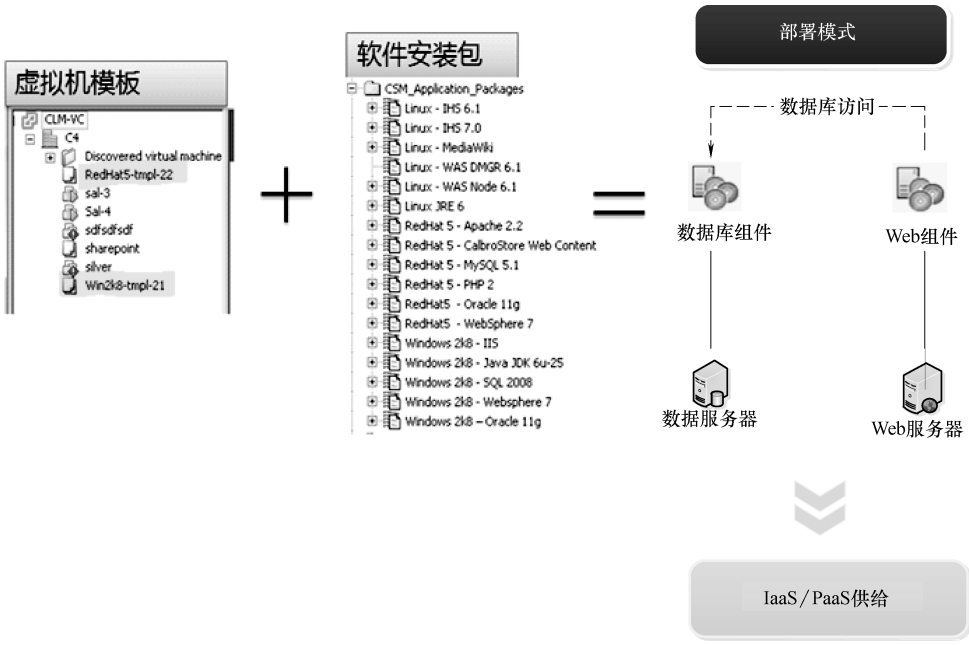


图 8-8　部署模式构成

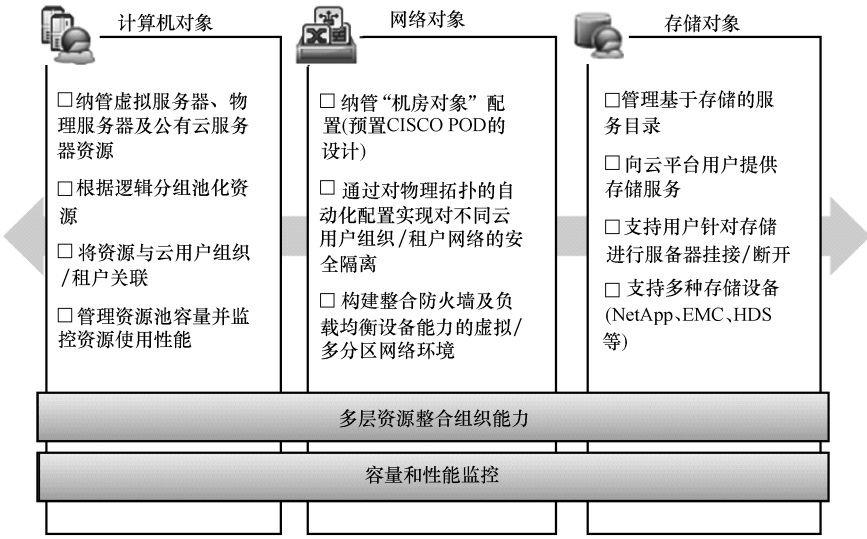


图 8-9　云管理平台资源对象

注：CISCO POD：思科网络发布单元；NetApp：美国网域存储技术有限公司；EMC：易安信公司；HDS：日立数据。

1. 计算资源池化

当前计算资源主要包括分为物理资源和虚拟资源，通过服务器自动化工具完成计算资源池化。

（1）物理资源　自动化工具支持对 X86 服务器、IBM 小型机、HP 小型机服务器的管理。

1）X86 服务器资源：在物理服务器加电启动后通过 PXE（Preboot execute environ-

ment，预启动执行环境）获取其 MAC 地址，并以 MAC 地址作为资源标识进行纳管。

2）IBM、HP Unix 服务器资源：使用服务器厂商自身的分区技术进行资源纳管。

（2）虚拟资源　自动化工具支持以下虚拟环境管理：

1）VMware。

2）Microsoft Hyper-V。

3）Solaris Global Zone。

4）IBM PowerVM。

5）CitrixXen。

6）Redhat KVM。

2. 网络资源池化

网络资源的池化主要通过网络自动化工具完成，支持目前市场上主推的网络设备。运维操作组件对网络设备的管理主要有以下三种类型，见表 8-1。

表 8-1　网络设备管理类型

类　型	描　　述	连 接 方 式
访问	读取各网络设备运行信息，并执行各种指令操作，包括配置、巡检、备份等	通过各网络设备管理接口进行，支持的网络设备管理接口包括 Telnet、SSH、HTTP 以及 HTTPS
配置	配置备份、恢复、上载等	支持 TFTP、FTP 以及 SCP
介质	系统安装、补丁安装等	支持 TFTP、FTP 以及 SCP

相关网络资源纳入管控后将资源池化后网络资源被分配为以下几种类型：

1）提供点（Pod）：代表物理上接近的一个网络环境，与地域相关。

2）网络容器（Network Container）：代表网络二层意义上的一个分段，通常这样划分的主要目的是为特定用户使用或负载的原因。

3）区域（Zone）：代表网络环境更细粒度的划分，通常是从负载或安全角度进行考虑。

4）网络（Network）：VIAN。

3. 存储资源池化

根据云服务规划，存储资源池主要包括 SAN 存储资源、NAS 存储资源和 OSD 存储资源。

存储资源的纳管可根据存储类型（如：NetApp、HDS、EMC）建立不同的适配器，并与存储资源建立管理关系。

（1）HDS 存储　可通过和 HDS 存储管理工具集成的方式实现对 HDS 存储的管理，如图 8-10 所示。

（2）EMC 存储　通过调用存储标准接口方式完成对 EMC 存储的操作，如图 8-11 所示。

（3）NetApp 存储　与 EMC 存储类似，采用与标准接口集成的方式实现对 NetApp 的管理操作。同时，要求云管理平台对存储管理还需要支持与以下存储管理软件集成：

1）IBM Tivoli Storage Manager：IBM Tivoli 存储管理。

2）HP Storage Essentials：HP 存储管理平台。

3）NetApp Provisioning Manager：NetApp 存储管理软件。

4）EMC Clariion：EMC 中端存储系列。

5）Hitachi Storage：日立存储。

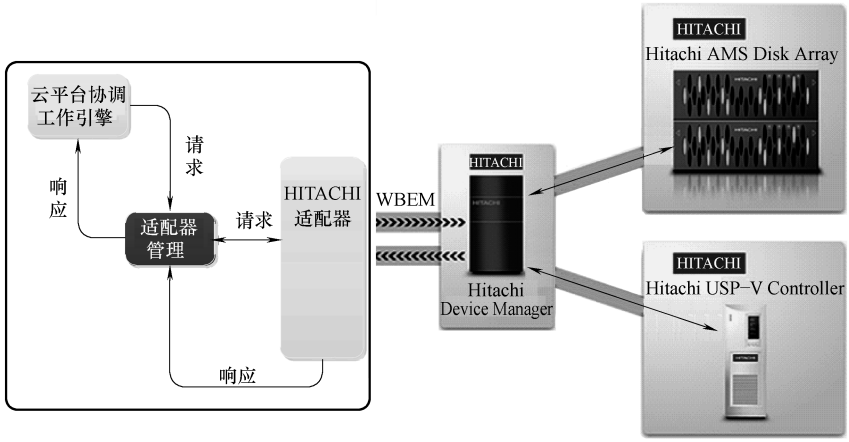


图 8-10　HDS 存储管理技术示意图

注：HITACHI：日立；WBEM：基于 WEB 的企业管理；Hitachi Device Manager：日立设备管理；Hitachi AMS Disk Array：日立 AMS 磁盘阵列；Hitachi USP-V Controller：日立 USP-V 控制器。

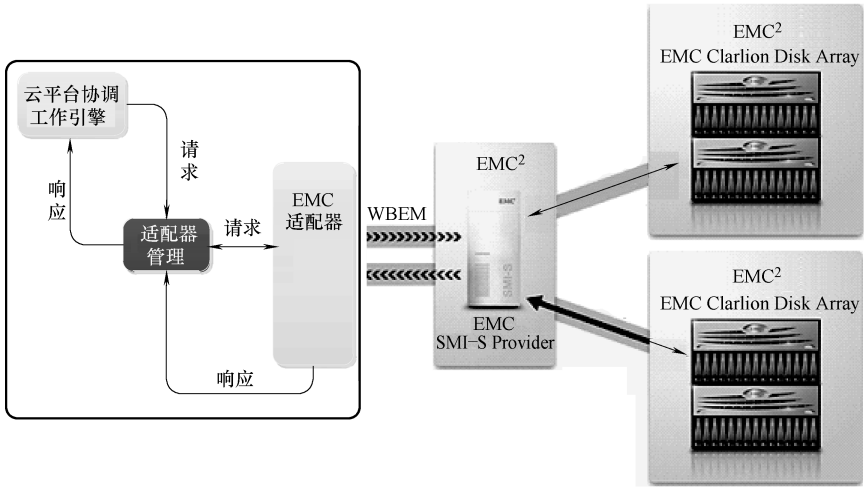


图 8-11　EMC 存储管理技术示意图

注：EMC：易安信；WBEM：基于 Web 的企业管理；EMC SMI-S Provider：EMC SMI-S（Storage Management Initiative Specification，存储管理主动规范）协议提供者；EMC Clariion Disk Array：EMC Clariion 磁盘阵列。

8.3.2.2　资源部署

资源部署是指一个特定服务请求的实例化过程，该实例化工作将由资源管理层，根

据请求服务的部署定义结合服务策略分配的目标资源对象，操作池化资源的接口进行实际的部署工作。该环节将主要涉及三大资源池的实例化、软件部署和整体环境的协同调度。

1. 部署调度

三大资源的实例化，软件部署的协同调度将由云管理平台的核心模块根据部署模式的定义要求，驱动各类池化资源协同完成。

2. 网络资源部署

网络资源的配置将主要通过网络自动化组件来执行，通过与被管理设备的控制接口进行在线配置和管理。

3. 服务器资源部署

服务器资源的部署包括物理环境部署和虚拟环境部署，将主要通过服务器自动化工具执行，支持以下环境的部署。

1）Microsoft Windows。

2）Linux。

3）VMware ESX and WMWare ESXi。

4）Oracle Solaris。

5）IBM AIX。

6）HP Unix。

7）Citrix XenServer。

（1）物理环境部署　对于 X86 平台系统，服务器自动化组件将支持网络启动的机器进行网络操作系统静默安装，对于 Windows 及 Linux 两类 X86 平台都支持，安装介质在打包的同时需要安装执行代理，服务器在完成操作系统安装的同时，安装其他运维工具代理程序，后续其他软件的安装及配置将由执行代理来操作。

（2）虚拟环境部署　虚拟环境部署分为虚拟资源划分和操作系统部署两部分。

1）虚拟资源划分。虚拟机资源的分配和创建将通过服务器自动化组件执行，该组件将通过集成接口直接调用各类虚拟化资源的管理软件接口进行分配并初始化资源。

2）操作系统部署。

① X86 平台：在虚拟机虚拟机环境划分后，其部署由服务器自动化组件驱动虚拟机预定模板进行克隆安装，虚拟机预定模板应包含执行代理，以便支持后续安装部署。

② IBM AIX 服务器：在 HMC 划分 VIOC 后，服务器自动化组件将驱动 NIM server 安装部署操作系统。

③ HP Unix 服务器：在上一步划分 Vpar 和 nPar 后，服务器自动化组件驱动 HP 的 Ignite 系统安装模式进行操作系统部署。

④ IP 地址池管理：服务器的 IP 地址的管理和分配统一由云管理平台负责，在部署操作系统的过程中，云管理平台会将 IP 地址的配置作为参数一同放置在系统安装调度流程中，当操作系统安装后将自动更新 IP 地址。

以上几种操作系统部署方式，均需要在制作安装介质的同时将执行代理打包，系统在完

成操作系统静默安装的同时，安装执行代理，后续软件部署的执行将使用执行代理操作。

4. 存储资源部署

存储资源的部署在纳管阶段即由存储自动化组件将存储的各类操作进行了抽象化，并对存储底层接口进行封装，形成上层服务定义模块提供的标准操作接口，在进行存储资源分配时，云管理平台直接通过该类接口进行管理，实现存储资源的划分和创建。

5. 软件部署

在服务器部署（物理或虚拟机）后，系统将自动安装执行代理。执行代理将根据部署模式的定义，定位该服务所需要安装的各类软件组件，包括操作系统、数据库、中间件、应用等，并按照执行计划自动完成软件部署执行计划。

8.3.2.3 资源分配层

资源分配层的核心是服务策略管理模块，它的主要用途是根据用户的请求和部署模式以及预置的资源分配策略对池化资源进行自动化分配及动态分配。

资源可按照各种维度进行划分，并确定划分策略，在部署时通过云管理员分配的各类策略进行动态匹配，从而映射出下一步部署过程中所需的具体资源，实现资源池化到虚拟化或物理资源的映射。

1. 资源动态策略分配

资源的动态分配将根据资源划分策略以及资源的容量使用情况进行动态分配，分配过程有两步。

1）第一步：在多个候选资源池中确定唯一资源池（如：多个相同配置 ESX 组成的计算机组）。

① 用户/组织←→资源池（网络池 + 计算机池 + 存储池）。

② 部署方式←→资源池（网络池 + 计算机池 + 存储池）。

③ 将用户/组织、部署方式与资源池关联起来，可在运行时确定合适的资源目标。

④ 可以自定义多种特性进行策略制定，如服务级别、地域位置、数据安全性、技术分类条件等。

2）第二步：在目标资源池中确定唯一资源。

① 通过基于性能监控的容量管理技术确定最优容量的资源。

② 平台收集并分析及预测资源的性能变化情况与趋势，并对资源目标选择请求调用并给出返回。

2. 资源分配变更管理

资源分配对象确定后系统将自动对资源的变更提出变更申请，该环节将与 ITSM 领域的变更管理集成，由变更管理流程进行审批，审批结果将返回云管理平台，云管理平台将根据审批结果继续服务的部署或者终止。

8.3.3 配置管理

云平台配置管理由自带的配置管理数据完成，是存储云服务对象相关功能运行信息

的数据库，其立足点是支撑云管理平台的正常运行，关注云服务对象全生命周期的管理需求和功能支撑。该配置管理数据库所描述的对象和属性范围为 CMDB 的逻辑子集，是 CMDB 的提供者之一。

云平台配置管理数据库中的数据必须是结构性的，而且可以将结构化的数据提供给 CMDB。在数据提供过程中，可以根据 CMDB 的数据要求对结构化数据进行必要的取舍。云平台配置管理数据库内部的数据模型应该满足 CDB（Common Data Model）模型要求，并且数据模型可以根据管理的需要进行调整。云平台配置管理数据库可实现云资源数据（如计算资源、网络资源、存储资源）和云管理平台的管理数据（如服务目录、合同信息、云管理对象信息等）的统一存放和管理。

8.3.4 流程管理

云管理平台的主要相关管理流程包括服务请求管理、变更管理以及故障管理，其相关定义如下：

1）服务请求管理：云服务目录项目的服务请求管理，包括各类 IaaS、PaaS 以及非云项目的服务请求管理。

2）变更管理：云环境相关的变更管理，包括云环境资源的创建、分配、调整以及各类 CMDB 项目的变更。

3）故障管理：云平台监控管理相结合实现云环境运维监控故障管理。

上述几项流程可通过标准接口实现与流程平台的集成，在现有流程管理规范下制定符合云管理要求的审批路径及相关要求，审批处理结果反馈给云管理平台，云管理平台依据反馈结果进行下一步处理。

8.3.5 监控管理

云环境下的监控保障体系非常重要，以确保云服务的正常高效运行，支持云环境中资源的最大化使用，更好地发挥云技术的优势，主要包括三个方面。集中监控数据流如图 8-12 所示。

1）要保障整个云可以提供优质的服务。这里主要考虑针对客户的需求监控如何保障云的服务供给。监控需要有一种手段主动发现现在的服务水平是否能够满足客户需求，无论是从用户端还是从服务端都应该可以看到用户对于服务的感受是什么样的。同时还需要有一些手段能够保障云环境所提供的能力是能够满足业务需求发展的，并且监控应该赋予云预测的能力，在服务出现问题之前，就能够通知用户，而不是等服务真正出现问题后才告诉用户现在服务不能进行了。

2）支撑容量规划的能力。监控可以根据现在的容量和状况分析云环境是否能够支撑未来的发展。缩短容量评估的周期，适应业务的快速发展。

3）支持弹性伸缩的能力。根据性能、容量数据实时分析系统的资源使用情况，分

析总结业务模式，并结合业务发展我需求，预测资源需求，合理弹性分配云环境中现有的资源。

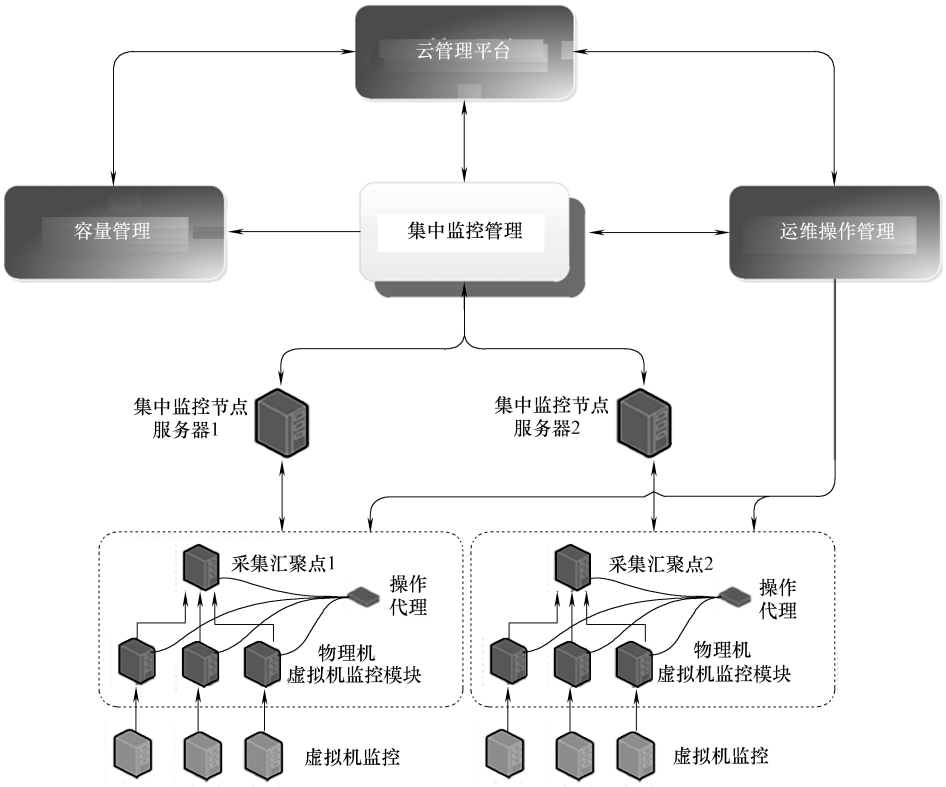


图 8-12　集中监控数据流

（1）监控架构说明

1）数据展现层：监控数据的大屏幕，门户展现，以及进行报表分析。

2）数据处理层：数据处理主要部署的是事件处理服务器，事件处理服务器进行告警处理以及业务影响模型展示分析。在该层可根据系统的负载情况分级部署，即子服务器和父服务器，各子服务器负责制定区域的性能数据采集，动态基线处理以及告警事件预处理；父服务器集中汇聚来自子服务器的告警事件然后进行统一集中处理。

3）数据汇聚层：数据汇聚层负责多个采集节点的数据汇聚以及跨区域或防火墙传输，该层主要部署集成服务，由集成服务和各监控代理连接收集并采集相关数据。

4）数据采集层：数据采集层主要是指云环境下监控对象及指标的数据获取。根据监控对象及监控模式的划分，对云环境的监控分为两个层面。

① 虚拟环境监控：如图 8-12 所示，监控代理通过与虚拟机管理端集成进行监控。该部分内容的监控主要关注虚拟环境的性能及资源使用情况。

② 物理实体机及虚拟机内部监控：通过自动化操作组件部署监控代理进行监控数据采集。

（2）自动化集成

1）数据采集层各监控代理的安装部署，以及监控实例配置将统一由执行代理来实现。

2）监控服务器的补丁安装和日常巡检将统一由执行代理来操作。

3）监控服务器产生的各类告警可触发执行代理并进行相关的故障修复。

4）监控服务器故障分析数据可通过触发执行代理获取相关性能及故障信息。

8.3.6 容量管理

1. 云平台容量管理的主要功能：

1）统计各类关键性能指标，负责长期的历史性能数据分析，方便运维人员及时掌握系统运行情况及健康状态，对整体资源进行及时有效的调控。

2）提供基于实际性能指标的预测分析功能，该功能可根据当前指标的变化情况，预测未来负载的变化趋势，帮助运维人员及时调整资源池容量，满足业务增长的需要，避免对业务需求增长的滞后满足。

3）可帮助分析系统若干关键指标之间的影响关系，如响应时间和内存之间是否存在线性依赖关系，如果存在该依赖关系，系统可通过调整内存来直接优化响应时间，通过该分析手段可为运维人员提供对现有系统的优化分析手段，精准定位系统瓶颈。

4）可以出具各类中长期数据分析统计报表，为各级运维管理人员日常工作提供决策依据。

5）将测量的容量情况智能反馈给云管理平台核心模块，指导云管理平台核心模块在部署 IaaS 和 PaaS 时的资源调配，避免出现资源池资源不足的情况。

2. 云平台容量管理集成接口

1）云平台容量管理通过集中监控平台获取性能数据，容量管理平台不需要对该类数据进行二次采集；对于其他非监控数据源，可定制数据导入接口，实现数据的录入。

2）容量管理平台与云管理平台核心模块集成，将容量分析数据反馈给云管理平台资源分配层，支持资源分配决策应用。

8.3.7 用户设计

从用户权限划分角度看，对于商业银行私有云来说，目前云平台可大致分为以下两种用户角色，见表 8-2。用户角色的划分也指导着未来云管理平台的应用。

表 8-2 云平台用户权限划分

角 色	云 管 理 员	云最终用户
角色描述	云平台的管理者，负责平台日常维护、权限管理、资源配置、服务目录、部署模式定义等	资源请求及使用
服务请求		1. 请求新的服务计算、应用、网络及存储 2. 选择服务项目 3. 理解价格及部署实例 4. 管理请求状态

（续）

角　色	云 管 理 员	云最终用户
服务实例操作		1. 启动/停止服务相关的服务器 2. 释放 3. 扩展 4. 回顾价格及选择可选项目 5. 回顾功能组件
服务器操作		1. 开启、停止、恢复和暂停 2. 修改 CPU 和内存 3. 增加自定义操作，如增加合规、配置及备份等 4. 查看性能图表（CPU 和内存） 5. 查看服务器配置
存储操作		1. 查看存储资源 2. 释放存储 3. 将服务器连接存储实例 4. 将服务器从存储实例中断开
网络操作		1. 管理部署服务器的防火墙规则 2. 在虚拟负载均衡资源池中增加或删除服务器
服务控制台		1. 服务的状态 2. 服务器的数量 3. 存储资源的总数
服务概要	面向所有用户 1. 查看服务请求的状态 2. 查看所有服务实例的状态 3. 服务器的数量 4. 云架构中部署服务的位置	
云资源视图	1. 云环境位置的可视化 2. 快速连接云资源的快速连接	
设定	明确设定角色基于部署模式和标识	
服务目录	1. 创建可提供的服务 2. 明确服务提供的可选项 3. 定义岗位的部署作业 4. 启用变更审批 5. 创建权利的条目 6. 映射部署模式	
部署模式	1. 创建部署模式 2. 定义应用的功能与组织并关联 3. 明确部署的服务的可选参数 4. 分配资源等级 5. 明确网卡、防火墙与负载均衡的设定 6. 标识主动部署的部署模式 7. 导入或导出部署模式 8. 查看定义的部署模式图表	

（续）

角　色	云 管 理 员	云最终用户
资源管理	• 计算机资源 1. 虚拟和物理云架构资源 2. 资源池：集群、资源池、数据存储、物理设备 3. 映射池与网络容器 4. 标识资源池的分配策略 • 网络资源 1. 注册 Pod 2. 创建网络容器 3. 管理区域、防火墙、负载均衡 4. 映射网络容器到角色 5. 标记容器的放置策略 • 存储资源 1. 管理存储服务目录 2. 映射服务提供与存储资源的映射	

8.3.8　接口设计

云管理平台对外需要提供三类接口：

（1）管理服务接口　提供云平台各类信息及资源对外访问的接口，用户可利用该接口定制应用门户及展示界面。

（2）资源供给接口　云管理架构中的各类资源在池化后，将统一不同异构平台的各类管理及操作接口，实现抽象的统一操作管理接口。

（3）触发外部调用接口　云管理平台触发外部调用的接口，用户可加入各类触发程序调用。

8.3.8.1　管理服务接口

通过管理服务接口可实现对云管理平台内置对象的查询、创建、删除等操作，也可作为云平台门户的集成接口。

8.3.8.2　资源供给接口

资源供给接口处于资源管理层，可实现对下层资源管理模块的调用，如服务器自动化模块、网络自动化模块、操作自动化调度模块。对于第三方的资源管理工具，可通过实现该接口来完成与云管理平台的整合。

8.3.8.3　触发外部调用接口

调用外部接口是云管理平台触发外部调用的接口，用户可加入各类触发程序调用，用于实现对云管理平台功能的定制化，如在 VM 创建时自动运行一个用户指定的外部工作流（AO flow）、执行脚本等，并将调用结果返回给云平台执行相应动作。

第 9 章
私有云安全设计

9.1 安全架构设计

9.1.1 安全分析

信息安全问题在云计算问题出现之前就早已有之，但随着云计算的大力发展和广泛应用，私有云环境下信息安全显得尤为突出。

无论是私有云还是传统信息网络都面临着各种安全威胁，主要包括传统的拒绝服务攻击、应用程序编程接口安全问题、内部员工的恶意破坏、非法用户对数据库的非法访问、授权用户的越权访问、信息共享技术导致的数据泄露、正常网络服务被劫持、密钥管理机制的不健全导致的账号泄露、用户安全防范意识薄弱导致的信息泄露、网络安全基础设施不健全导致的黑客入侵等一系列问题。

除此之外，在私有云部署实施后，信息安全还面临着一些新的挑战。

1. 流量模型的转变

传统的企业流量模型相对比较简单，各种应用的基准流量及突发流量都是有规律可循的，即使对较大型的数据中心仍然可以根据 WEB 应用服务器的重要程度进行有针对性的防护，对安全设备的处理能力没有过高的要求。

而在云计算环境下，同类型存储服务器的规模以千/万为单位进行扩展，而且基于统一基础架构的网络进行承载，无法实现分层规划、分而治之，因此对安全设备提出了更高的性能要求。

2. 动态的安全边界

在传统安全防护中有一个很重要的原则，就是基于边界的安全隔离和访问控制，并且强调针对不同安全区域设置差异化的安全防护策略，在各区域之间有明显的边界

划分。

而在云计算环境下，存储和计算资源高度整合，基础网络架构统一化，安全设备的部署边界已经消失，这也意味着安全设备的部署方式将不再类似于传统的安全建设模型，云计算环境下的安全需要寻找新的部署模式。

传统数据中心安全防护独立、分层部署、分而自治，其优点是日常安全运维、安全管理工作可以灵活分开进行，自定义功能比较强，但也存在工作量大、整体安全管理不全面的先天不足。而由于云计算数据中心数据流集中、安全边界消失等特性，安全集中管理是必然要经历的过程。

3. 虚拟环境的网络管理

在数据中心，诸多区域由防火墙来划分，过渡到私有云环境之后，原有网络的一些边界被打破了，原来可通过网络设备控制彼此之间的流量在私有云环境下却无法深入分析了。

因为在虚拟管理层里面可以配置的虚拟网络，甚至修改整个云内设备的拓扑环境，在没有由防火墙来划分区域的环境下，如何去做好网络安全的保障，如何去分割管理的权限，这将会是一个新的挑战。

4. 接口统一标准化

对于企业私有云而言，企业未来必然会走向私有云与公有云通信甚至融合的趋势。因此，标准化工作是必须要考虑的问题，特别是未来当企业逐渐走向混合云的阶段，其流程、应用程序、业务接口必然会做出改变。

标准化对于安全而言，通常意味着便于管理和监控；反之，混乱无序的接口也意味着风险控制的难度加大。

5. 弹性空间和可扩展性

云计算实质上是帮助用户实现即需即用、灵活高效地使用 IT 资源，与此同时提高 IT 资源的利用率，缩减企业的 IT 成本。这对于云计算平台来说，就必须具备非常大的弹性空间和良好的可扩展性。

当前采用传统架构的产品无法具备良好的扩展性，而像 X86 服务器、集群存储产品都具有高度的可扩展性，能够很好地满足私有云对扩展空间的弹性需求。因此，实现按需增减 IT 资源、架构灵活多变是私有云环境的一个重要标志，对于安全而言，这些弹性的和可扩展的空间也是风险陡增的空间，必须对这些外延的、多变的功能进行逐一的风险识别。

私有云安全问题的本质和传统的网络信息安全问题没有实质区别，它只是在传统网络信息安全的基础上有所发展。它们皆面临着上文所述的问题，本文在考虑如上问题及私有云架构设计上，也做出了一些更合理的安全架构调整。

9.1.2 安全架构

云作为一种新型计算模型，它将计算任务分布在由大量计算机构成的资源池上，使

各种应用系统控制模型根据需要获取计算能力、存储空间和软件服务，使得云下的安全控制模型比传统的信息安全更加复杂。因此，传统的安全控制模型已不能满足私有云的设计，本文根据私有云当前的需求和安全分析，设计出一个更合理的架构设计，如图9-1 所示。

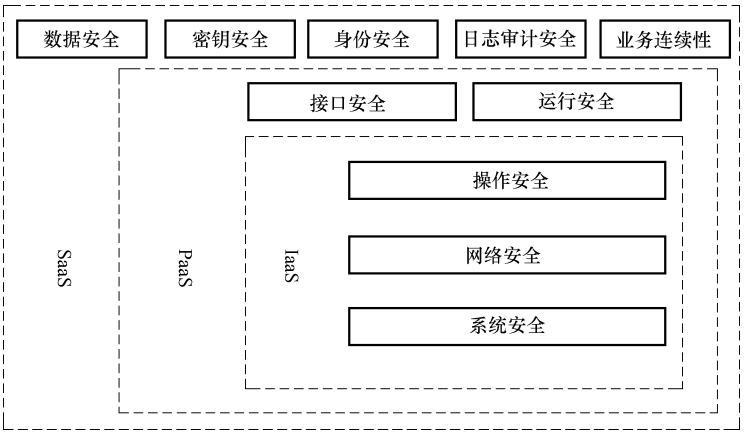


图 9-1　安全控制模型

在图 9-1 的安全控制模型中，云模型分为 IaaS、PaaS 和 SaaS 三层。

IaaS 包括硬件底层设备、虚拟中间层和接口；PaaS 包括中间层、可编程开发接口等；SaaS 包括程序、数据、应用平台等。

在此安全控制模型中，需要针对物理硬件、计算和存储、可信计算的软硬件平台、计算机网络通信、信息处理以及应用程序做好应有的安全防范措施。

根据上述的云安全模型设计，理解云计算模式之间的关系和依赖性，对于理解私有云计算的安全风险非常关键。IaaS 是所有云服务的基础，PaaS 建立在 IaaS 之上，而 SaaS 又建立在 PaaS 之上。如同云服务能力是继承的那样，信息安全风险和问题也是可继承的。然而，云参考模型对于将真实服务和某个架构框架联系在一起，进而理解需要进行安全分析的资源和服务是非常重要的。

以下是三个层面做好信息安全防护的措施。

1. IaaS 层安全

IaaS 层主要包括计算机网络基础设施、主机、网络设备、服务器等所有的计算机硬件平台。在该层中，首先是将硬件资源抽象起来，然后将这些硬件资源纳入整个基础设施的逻辑节点中，然后向用户提供一个可统一编程调用的应用程序接口，使用户通过应用程序对应用程序编程接口进行调用，以完成物理设备的交互使用。

在该层中，主要关注的安全问题包括网络基础设施的物理安全、环境安全、主机安全、主要网络连接设备安全、系统的虚拟化安全等。IaaS 的服务提供商需要对该环境提供一些基础的公共安全保障，需要对用户的数据安全或应用安全提供一定程度的安全保证等。

2. PaaS 层安全

PaaS 层主要提供一个可安全运行的平台以及可以和用户交互的编程接口。它是在

IaaS 基础上增加了一个可用于开发的应用程序接口层面，来完成将数据库、数据等集成在一起完成设备间信息传递和进程间通信的一个平台。PaaS 层的安全主要包括接口安全、运行安全等。

3. SaaS 层安全

SaaS 层主要是为用户提供应用程序的运行环境。在这个运行环境中，用户能够充分利用云服务所提供的资源和软件服务，体验到云服务便捷、高效的服务，并且不必关心应用程序的运行过程和底层硬件的工作原理。

这一层次的安全问题主要表现为软件的应用环境安全，包括信息保密、数据加密方法、密钥管理机制、身份验证、安全审计、访问控制、安全事件处理、业务连续性等。在云计算的安全事件中，多数的安全事件都发生在 SaaS 层。

三个层次对应的信息保护措施见表 9-1。

表 9-1　信息保护措施

服　务　层	安 全 问 题	对 应 措 施
IaaS	系统安全	加强公共基础设施保护能力、服务商对云服务的基础设施提供一定的质量保证
	网络安全	
	操作安全	
PaaS	接口安全	统一的用户编程接口
	运行安全	加强硬件系统和软件系统的稳定性
SaaS	数据安全	加强人员管理和安全行为审计
	密钥安全	先进的加密机制和严格的密钥管理体制
	身份认证	访问控制策略
	日志安全审计	审计策略
	业务连续性	链路负载均衡、自动灾难恢复机制

为更有效地保障云计算服务的安全性，除了上述提到的一些措施外，还应该结合云计算的特点，在数据的完整性、可用性和高可靠性方面进一步做好信息的安全保密工作，在网络身份认证、加密算法研究、入侵检测、虚拟专用网络远程安全接入、数据存储等方面加大研究和投入，构建全面的安全防范体系。

9.2　安全评估

9.2.1　安全管理价值

私有云的应用使得商业银行在系统的高可用性、提供灵活弹性的服务、提高业务交付效率、优化 IT 成本等方面都有了很大的提升。由于金融行业的特殊性，无论基于传

统技术架构还是引入私有云，注重安全一直都是商业银行信息系统的主题，而云平台由于技术架构的重大变革给信息安全领域带来巨大挑战：

1）在云平台中运行的各类云用户没有固定不变的基础设施、没有固定不变的安全边界，难以实现用户数据安全和隐私保护。

2）云服务所涉及的资源多个管理者所有，存在利益冲突，无法统一规划部署安全防护措施。

3）云平台中数据与计算高度集中，安全措施必须满足海量信息处理的需求。

私有云在应用的过程中，不尽如人意的安全事件也频有发生，比如亚马逊（Amazon）的云计算平台多次出现服务中断，微软的云计算平台也有报道出现屏蔽超过 20h 的事故等。因此，如何保障私有云安全、持续、有效运作对商业银行是一个巨大的挑战。

对于商业银行来讲，应用私有云的第一步是要建立被企业内部认可的云安全管理框架，确立组织基本有效的安全管理策略和控制措施，并明确组织中人员的安全责任与义务。

在建立符合商业银行自身实际的云安全管理体系时，不仅可参考传统商业银行信息安全管理理论、IT 服务管理 ITIL V3、COBIT 等理论，还可借鉴 CSA、ENISA 云安全管理模型。CSA 和 ENISA 的云安全管理框架模型关注点均包括了技术和管理两类，其中 CSA《云安全管理指南》的内容更加丰富，在策略指导和实施建议方面更具影响力。

通过实施云安全管理体系，建立严格的安全管理策略和控制措施，严格按照规程操作，私有云的安全性和传统相比可能在以下方面更具保障。

1）减少数据丢失。据统计，全球机场每年都会丢失超过 12000 台便携式计算机，而又有多少台便携式计算机采取了真正强大的安全措施，比如整个磁盘进行数据加密？含有重要数据的便携式计算机一旦丢失，则会给个人或企业甚至国家带来巨大的损失。而通过在云上维护数据，搭配强大的访问控制，云计算可以限制或减少可能丢失的信息量。

2）即时换位。如果企业存储在云中的数据不幸受到危害，在数据中心模式下，首先需要花费时间向高级管理层解释系统由于意外事件而停止运行，然后再去花费若干小时的时间尝试复制数据或者修复损坏。在云中，则可以在执行调查的同时把数据转移到另外一台机器上，这对于用户而言是透明的。

云安全不是一个简单的问题，和云计算本身一样面临着各种现有技术的融合与创新。因此，在评估私有云的风险、寻找应对措施的过程中，需要不断地借鉴过去在安全领域积累下来的有效技术手段，并根据云计算与传统的差异予以创新，从而摸索出一条适合云环境的安全之路。

9.2.2 安全评估方法

总的来说，云安全评估与传统安全评估思路并不冲突，在很多方面遵循了传统的安

全风险评估方法论。这一方面体现了安全的本质和适度安全的根本理念；从另一个方面也说明，采用云计算技术来建设或改造系统，传统安全理念和防护手段在新的环境下同样有效。

云安全评估是指对于云服务进行的审计，或是通过云提供的基于业界标准的方案进行的评估。对于基础设施、应用的传统安全评估以及合规审计，业界已有完备的定义和多个标准的支持。安全评估具备相对成熟的工具集，一些工具已通过 SecaaS（Security as a Service，安全即服务）的交付模式实现。在 SecaaS 交付模式下，用户可以获得云计算变体的典型好处，即弹性扩展、几乎忽略不计的安装部署时间、较低的管理开销、按使用付费以及较少的初始投资。使用这些工具审计云计算环境带来了额外的挑战，虽然这不是这些工具原本关注的焦点。包括 CSA 在内的多个组织，已经在致力于一些指南定义方面的工作来帮助组织和理解这些额外的挑战。

私有云的安全主要涉及数据计算安全、存储安全、传输安全和安全风险控制策略等四个方面，可以从这四个方面考虑云计算的安全评估办法。

1）数据计算安全。对于利用统一平台实现的计算，要保证平台的安全。

2）数据存储安全。在云环境下，数据存储是建立在网络上的高效分布式存储。如何确保数据不发生泄露以及对托管数据进行备份和恢复等都是需要考虑的。

3）数据传输安全。对于数据传输，一是如何确保数据在网络传输过程中不被窃取；二是是否有网络接入设备的备份，以满足计算和存储的需要等。

4）安全风险控制策略。安全风险控制策略是用来规避云计算风险的方法，包括技术方法和管理方法。由于云框架的不同，所以其安全管制策略也各自不同。

对私有云进行安全评估，可基于对其安全问题的分析，借鉴传统的信息安全风险评估方法，结合私有云的特点，从资产、脆弱性、威胁、安全措施、风险、残余风险、安全需求、标准化等要素进行探讨。

1）资产：是指私有云中的各种资源。一般来说，资产价值越高，业务战略要求越高，风险越大。

2）脆弱性：是指私有云中每一个资产存在的漏洞。

3）威胁：是指私有云中因结构、技术、管理等原因而引起安全问题的可能。

4）安全措施：是指针对引起私有云安全事故的原因而采取的技术、管理手段。

5）风险：是指私有云存在安全事故的可能性。

6）残余风险：是指采取安全措施后，私有云存在安全事故的可能性。

7）安全需求：为达到业务战略的目标，而对私有云安全提出的要求（保护等级）。

8）标准化：当前私有云缺乏统一的计算标准，各个云产品在互操作上难以兼容。为了更方便、安全、快捷地在不同云之间实现切换和数据迁移，同时避免服务提供商对用户的锁定，以及在出现安全事件时进行证据采集和责任划分，有必要对云计算服务实现标准化。

以上要素之间的关系是：首先资产拥有者根据其重要程度和标准化的规定，要求某资产应达到的保护等级，根据资产因其脆弱性受到威胁的风险，评估风险级别，并制定

相应安全措施，以对资产的脆弱性进行弥补，降低威胁利用资产脆弱性发生安全事件的可能。如果残余风险未达到保护等级的要求，则继续评估并修改安全措施，直到满足对资产的保护等级要求。

9.3 安全防护

私有云一般部署在企业网内部，主要采用大量的虚拟化资源替代传统的物理资源，达到实现资源共享的目的。结合前文阐述的安全架构设计，私有云计算数据中心所需管理与使用到的技术资源安全，需要重点关注如下四个方面的内容，分别是网络安全、系统安全、操作安全和安全事件监控和处置。

9.3.1 网络安全

传统模型下的网络安全是指网络系统的硬件、软件及其系统中的数据受到保护，不因偶然的或者恶意的原因而遭受破坏、更改、泄露，系统能够连续、可靠、正常地运行，网络服务不发生中断。而私有云下的网络安全结合了虚拟化的概念，增添了新的风险变数，下文将从安全域管理、访问控制及安全防护三个方面来考虑私有云下的网络安全防护措施。

1. 安全域管理

在传统网络安全管理中，网络安全架构通过分层规划、分级建设，安全区域清晰明了，安全管理方式可以按照业务需求对不同类型的系统方便定义安全级别；然而在云安全域管理过程中，由于业务资源聚集、基础网络架构一体化，安全边界消失，传统网络安全设备（如防火墙）很难从实体上实现隔离，无法满足云安全域管理。在网络架构一体化的云网络环境，需要考虑使用新型的虚拟交换技术、虚拟防火墙技术。

建议措施：

（1）域分级　即按照每个区域部署的主机功能及涉密等级划分类别，不同等级间必须采取相应的安全防护策略。

（2）域间控制　安全等级不同的域互访，必须部署有效的安全策略和防护措施。

（3）最小授权原则　安全区域间的防护尽可能按照安全最小授权原则。

（4）整合系统接入　在保证网络系统的各种互联需求的有效提供的前提下对安全域的边界进行合理的整合，对系统接入进行有效的整理和归并，减少接入数量，规避边界不清、连接混乱等问题。

2. 访问控制

网络访问控制是以某种途径批准系统用户的访问能力及访问范围，主要作用是防止非法的主体进入受保护的网络资源，允许合法用户访问受保护的网络资源。防止合法的用户对受保护的网络资源进行非授权的访问。

在私有云环境下，边界的模糊使得传统的访问控制手段需要加以调整，因此，建议重新规划和严格执行细粒度的访问控制策略。

建议措施：

（1）就近防护原则　当通信系统之间在物理上或逻辑上可经过多层防火墙时，应优先选择在提供网络服务的最近端控制点上部署访问策略。

（2）策略从严原则　具有不同等级安全访问控制策略的双方进行通信或合并部署时，安全策略应遵循控制相对严格一方的标准。

（3）缺省禁止原则　防火墙缺省基本访问控制策略是拒绝所有访问，在此基础上逐步根据合理的应用需要开放最小化的地址和端口。

（4）特例审慎审批原则　对于业务需求确实无法满足安全控制要求的情况，开通访问需求需要经特别案例分析后经主管部门审批执行，并落实整改计划或采用其他技术控制手段作为补偿手段。

3. 安全防护

网络安全防护系统由外向内分层保护商业银行信息资产，为体现纵深防御思想，应为互联网安全提供 DDOS（Distribated Denial of service，分布式拒绝服务攻击）防护、入侵检测系统（Intrusion Detection Systems，IDS）防护、防火墙、内网漏洞扫描、安全加固等服务。

建议措施：

（1）部署入侵检测系统　入侵检测系统应部署在流入数据的最前端，并能检测到所有访问私有云的流量旁路方式接入网络环境，优先选择流量镜像方式进行检测。

（2）部署预防分布式拒绝服务攻击系统　预防分布式拒绝服务攻击系统应部署在被保护系统数据链路的最前端，备用旁路部署于数据链路上，以流量镜像方式对数据包进行分析。

（3）定期执行安全分析　应根据内网系统的重要性和遭受攻击的可能性进行资产排序，定期对高风险资产进行有效的内网漏洞扫描、外网渗透测试及安全加固等操作，并及时跟踪相应的风险完成及时的整改。

（4）引入虚拟化防火墙产品　在私有云环境下，应结合虚拟化防火墙等新兴产品将安全防护的触角深入到虚拟化网络内网，使网络边界模糊导致流量不可见再次清晰起来，为安全防护提供助力。

9.3.2　系统安全

虚拟化技术是实现云计算的主要手段，是云计算框架的基础，通过虚拟化技术，在云环境下实现资源的有效利用，虚拟化环境的系统安全仍是首要关注点。虚拟资源的系统安全主要从 Hypervisor 安全、虚拟机镜像安全、vCenter 安全三个方面予以考虑。

1. Hypervisor 安全

Hypervisor 是虚拟环境中的“元”操作系统，由于其可以控制在服务器上所运行虚

拟机的方方面面，所以成为需要保护的对象。其安全性的高低直接影响到虚拟机安全，因此在虚拟机管理方面应从 Hypervisor 层出发，实现纵深式的安全防护。

建议措施：

1）依照传统模式下固有的口令安全、用户安全、权限安全等原则对 Hypervisor 层落实相应的安全要求。

2）制定 Hypervisor 层特有的安全参数配置要求，并定期对其进行核查和加固。

3）使用集中式虚拟化管理系统进行管理，同时建立虚拟化管理流程来防范误操作、错误配置等安全问题。

4）对虚拟机进行跨 Hypervisor 迁移时要进行适时监控、审计和告警，防止违规行为发生。

2. 虚拟机镜像安全

虚拟机镜像是为客户提供的“机器”实体。当虚拟机处于休眠状态或关机状态时，很容易被篡改或修改，对于这个问题，需要对虚拟机镜像进行加密，同时与管理手段、审计跟踪手段相结合，以防止正在运行的虚拟机镜像“逃逸”，避免攻击者访问到虚拟机快照中的数据造成恶意损坏而未被及时发现。

每次部署新镜像时需要创建自定义镜像使终端用户可以自己安装所需要的组件。而对于私有云而言，多种用途的虚拟机镜像是安全着力的关键点，如何做到虚拟机镜像的安全保护，很大程度决定了是否能从源头控制虚拟化环境下虚拟机的风险。

建议措施：

1）虚拟机镜像依照传统模式下各类操作系统的安全配置要求进行加固。

2）制定虚拟化层的独有的安全参数配置要求，并定期对其进行核查和加固。

3）对虚拟机镜像文件进行加密和完整性校验，防止静置的镜像文件遭到窃取导致信息泄露，也防止遭到恶意修改。

4）虚拟机的安装严格按照事先进行安全加固和检查的镜像执行标准化安装。

5）数据迁移后应对虚拟机数据进行销毁，必要时采取磁盘报废等最佳实践措施。

3. vCenter 安全

诸如 VMware vCenter Server 之类的云管理平台是业界现有的成型的商业化虚拟化集中管理解决方案。私有云管理方借助其高级功能，可以对虚拟环境进行最精确的了解、主动管理和扩展；也可以轻松地对其进行扩展以实现与物理环境的端到端集成，并且可以在此基础上构建自由的私有云集中管理架构。

无论是利用成型的商业云管理平台，还是开发自有的特色云管理平台，如何在最大化地利用集中的自动化部署管理的同时进行精细的权限划分并得到权限控制是安全的一大挑战。

建议措施：

1）明确细化的访问控制。通过可配置的分层组定义和精确控制的权限，确保云环境安全，避免遭受权限滥用风险，从而对必要的核心权限进行多次授权控制。

2）保护 vCenter　Server 数据库。vCenter 数据库是存放与配置数据和其他数据的地

方，其中包括角色、许可权、事件、任务、性能数据、数据中心信息、集群信息、资源池信息以及其他更多信息。

3）记录审核信息。保留重大配置的更改以及发起这些更改的管理员记录，事后导出报告以进行事件跟踪。

9.3.3 操作安全

云计算数据中心操作管理是云服务正常提供的一个基础保障措施，大资产、大数据对云计算中心有序运营提出了更高的要求，从操作角度提出的云计算数据中心需要重点关注以下几个方面。

1. 人员安全管理

运维人员是数据中心运维管理体系的基础。一个良好的数据中心管理体系，应有合适的技术和管理人员。伴随数据中心规模的扩张，相应的人员安全问题越来越多，比如私自泄露敏感数据、盗取用户数据等人员管理带来的安全问题，所以云服务管理需要关注云计算数据中心内部人员安全管理。

建议措施：

1）对核心岗位新入职员工进行细致的岗前培训考核、风险意识培训、签署保密协议。

2）对核心岗位在职员工进行在岗评估、岗位轮动，及时发现存在的操作风险问题。

3）对核心岗位离职员工即刻清除所有信息资产的管理权限。

2. 身份与权限管理

在传统模型下，企业或多或少已经建立起了较为完善的员工身份和访问权限管理的控制系统，然而伴随着云计算模式的发展，原有模式下的系统或多或少也出现了不适配的情况，如何结合云计算特性优化现有身份及权限管理系统是操作安全把控的重要一环。

建议措施：

1）应执行实名制用户身份管理，出现相关的操作风险可以及时定位到人，及时处置。

2）应减少用户接触 Hypervisor、云管理平台、虚拟机管理口令的机会，结合传统口令集中管理模式，探索集中管理私有云下各层次管理口令的技术手段。

3）应遵循最小化权限原则进行细致的权限授予和审批，对于云管理平台及 Hypervisor 层的关键账户权限实施严格的职责分离、进行审慎发放。

3. 操作识别与管控

在私有云环境下，对于云管理平台或者 Hypervisor 层的不当操作可能影响几十甚至上百台虚拟机，在这样的衍生风险下，应着力推动精细化的操作识别和相应的管控系统。

建议措施：

1）在设计阶段辨识高风险操作指令或步骤，逐级区分禁止操作、授权操作、常规操作等多层次类别。

2）通过技术手段对非紧急场景下的禁止操作予以阻断和中止，避免操作性失误带来的衍生风险。

3）通过技术手段对授权操作指令或步骤执行时，引入多次授权管控的方式和控制不当操作。

4. 审计追溯

在日常运维中，可能出现各种违规操作或恶意操作，这类操作在云环境下的破坏性将演化扩大，不断完善审计日志的记录粒度和检阅频度，有助于在出现操作风险后及时定位相关人员和确定恢复动作。

建议措施：

1）根据5W（5个W分别指When、Where、Who、How和What）原则记录各类用户操作行为。

2）明确审计日志的留存期限和轮转机制，并做到关键日志的异地存放，防止数据覆盖、丢失而出现的监控真空。

3）明确审计日志的调阅频度、检查范围和违规惩处措施，对违规或恶意操作行为起到警示作用。

9.3.4 安全事件监控和处置

相对于传统数据中心安全事件响应来说，云计算的本质使得当发生安全事件、数据破坏或其他需要调查和采取行动该找谁时显得更难以确定。为了满足相同汇报责任的需求的变化，需要修改标准安全事件响应机制。与此同时，标准化部署的IaaS、PaaS、SaaS也给安全事件的定位和处置提供了便利。因此，建立一个良好的安全事件响应流程和机制对于云计算环境来说显得格外重要。

建议措施：

1）结合舆情监控、客服中心投诉等多途径，配合网络安全、系统安全、操作安全建立纵向的安全事件监控体系。

① 利用网络层面部署的入侵检测系统、分布式拒绝服务攻击防护系统、防火墙筑起外层第一道防线，对各类系统监控的指标出现异动，迅速联合网络运维团队定位和修复问题。

② 利用系统层面运行的漏洞扫描工作、安全配置检查筑起第二道防线，对漏扫发现目标虚拟机漏洞和出现的不当配置，从虚拟机镜像为源头出发整改和加固，并从Hypervisor、云管理平台和虚拟机三个层面进行核实和整改跟进。

③ 利用操作层面的运维风险管理系统筑起第三道防线，对于运维操作层面的违规操作及时发现、警示和拦截，将内部操作风险降到最低，并结合风险意识培训不断强化

运维人员的风险意识。

2）联合私有云运维团队，建立自动化的检测与恢复机制。利用私有云环境下虚拟机部署的一致性，能够建立起统一的问题定位、取证、分析机制，在出现安全事件时能够与私有云运维团队进行快速的联动，通过自动化脚本收集到一致的日志信息供安全防护团队分析，也能够利用同样的技术手段对发现的各类缺陷进行快速地修复。

3）抽取事件经验和技术发展态势，反向推动网络安全、系统安全、操作安全等策略和技术手段向前发展，减少系统面临的各类风险。

应用与探索篇

第 10 章
中国建设银行私有云建设实例

10.1 资源全生命周期管理

10.1.1 资源管理生命周期

云服务在本质上是以服务的方式为用户提供各种能力，包括计算能力、存储能力和网络能力，而这些能力的建设又依托于资源，包括计算资源、存储资源、网络资源以及环境资源。建行云管理平台的资源全生命周期管理是以资源作为管理对象，从系统的整体目标出发，在满足安全、效能的前提下追求资产全生命周期成本最优，使得资产在整个生命周期中得到高效、充分、合理的利用。

建行云管理平台的全生命周期管理包括规划、纳管、申请、分配、运维以及回收六大环节，如图 10-1 所示。

1. 资源规划

数据中心会定期组织专家对资源使用情况进行分析和规划，资源规划的内容包括应用资源、计算资源、存储资源、网络资源和云服务。应用资源指的是数据中心所运维的应用系统及其对外的服务能力，所有应用系统在开发之前均需审核其非功能性需求，在上线之前，均需根据其非功能性需求评估所需要的计算资源、存储资源以及网络资源，而评估所需资源均需通过数据中心专家团队的审核后才能申请，这些被审核通过后的所需资源也会及时更新至云管理平台中，数据中心资源管理员可以通过目前的资源拥有情况和资源使用情况规划资源采购计划，而云服务管理员则根据资源所需供给模式来规划、设计和开发云服务。

2. 资源纳管

资源被采购后，通过登记入库、上架、上电、入池等环节被云管理平台纳管，而只

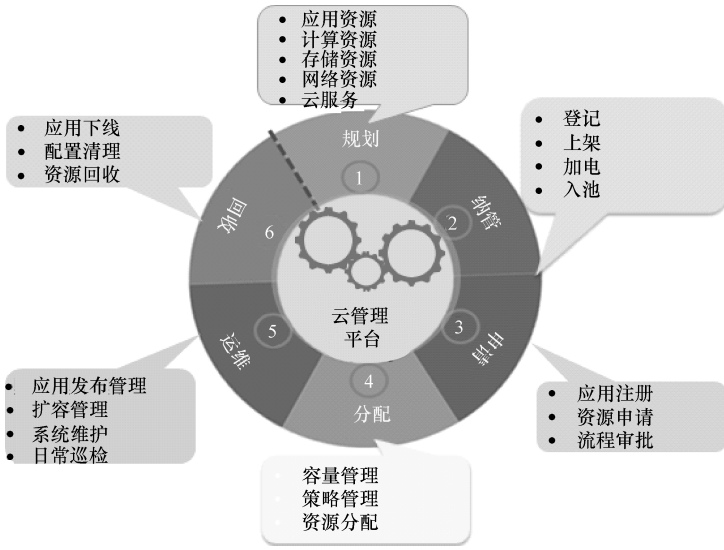


图 10-1　资源管理生命周期

有被纳管的资源才能被分配。

3. 资源申请

用户在申请资源时，需要注册应用系统下物理部署单元的相关部署信息，如该部署单元是 WEB、AP 还是 DB，是在哪个网络区域，需要部署在哪个数据中心，平台类型是 X86、AIX 还是 HPUX 等。注册完部署单元后，即可通过云服务目录选择相应的云服务申请资源，当云服务申请被审批通过后即可进入分配环节。

4. 资源分配

建行私有云的资源是动态分配的，即根据目前私有云内的资源整体容量使用情况、用户的资源需求以及资源分配策略动态地选择分配资源；比如，用户需要一套 AIX_ORACLE_RAC 服务器，云管理平台会自动地在 AIX 资源池中选择两台资源使用率最低的 AIX 物理机，在这两台 AIX 服务器上按照用户资源需求分别安装 Power 虚拟机和 Oracle RAC 应用形成一套 AIX_ORACLE_RAC 服务器提供给用户。云管理平台除了支持自动化地动态分配资源，还要支持资源管理员手工分配，以满足部分用户的特殊需求。

5. 日常运维

资源被分配后，即进入日常运维管理，包括应用发布管理、日常巡检、系统维护和扩容管理等；而在日常运维中，资源的相关配置信息和运行日志都会得到管理。

6. 资源回收

当应用系统需要下线时，用户需提交应用下线申请，云管理平台会自动化地清除应用信息、清理配置信息并将资源回收至资源池中。当资源出现故障或者过了维保期需要下线时，资源管理员也可通过资源下线流程完成资源回收、下电、下架及退库等操作。

10.1.2　资源信息库

资源信息库是整个云计算平台的核心数据，用于存放所有资源信息和配置信息，以

及资源在全生命周期管理过程中的状态信息等。资产信息库的管辖范围为数据中心的环境信息、硬件设备信息、软件信息以及软资产信息等，资源信息库支持可动态添加类及属性，便于日后的数据扩展。

10.1.3 资源管理流程

传统资源管理失败的一个很大原因在于把资源管理视为一个资源记录的工具，大量的资源信息完全靠人员自觉录入，而且这些录入工作是十分繁琐的，所以信息经常得不到及时更新，导致数据准确性不高，运维人员越来越不愿意从资源信息库中获取资源信息，运维人员也失去了更新资源信息库的动力，从而进入一种恶性循环之中，最终导致资源信息库的数据完全失真。

建行云管理平台在设计之初就意识到资源信息如果不能被使用和及时更新，就无法保证资源信息的准确性，所以建行云管理平台对资源信息进行了一一分析，可以通过自动化采集或更新数据来保证其准确性，必须通过人工录入的数据则通过流程来保证资源信息的准确性。

资源管理全生命周期流程图如图 10-2 所示。

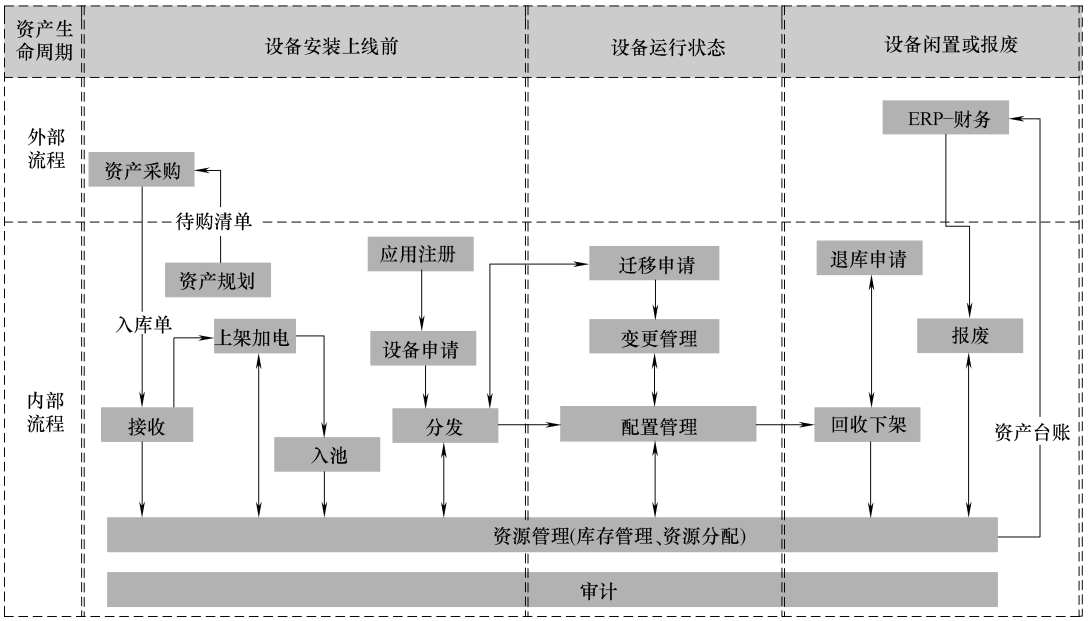


图 10-2 资源管理全生命周期流程图

1）资产采购：资产的采购通过 ERP（Enterprise Resource Planning，企业资源计划）系统的采购流程实现，采购完成后即进入资产接收流程。

2）接收：接收厂商送货，进行资源验收并入库。

3）上架加电：资源入库后，即可进行上架加电，资源进入资源池。

4）入池：资源上架加电后，需要先纳管至资源池后才可被分配。

5）设备申请：资源分发的触发流程为设备申请服务请求。

6）分发：用户通过服务请求申请资源，云管理平台自动分发资源。

7）迁移申请：物理迁移的触发流程为迁移申请服务请求，该请求进一步经过变更管理审批执行后，更新资源环境信息。

8）变更管理：日常资源使用变更操作管理流程。

9）配置管理：资源进入运行状态后的信息维护主要通过配置管理流程实现，配置管理流程对资源信息库的维护是通过变更管理流程来执行的。

10）回收下架：用户通过服务请求申请资源回收，如无需下架，则资源回收至资源池；如需下架，则移出机房，返回库存。

11）退库申请：资源回收下架的触发流程为退库申请服务请求。

12）报废：根据 ERP 资产报废流程要求，处置资源，更新资源状态信息。

13）ERP：资产报废及财务资产盘点的触发流程为 ERP 系统。

14）资源管理：在资源管理的角度对库存及使用中的资源进行管理。

10.1.4 采集库

为了提升资源信息库的自动化率，设计采集库作为云管理平台的重要组成部分，负责各种信息的自动采集，例如服务器、存储、网络等各方面的信息采集，以保证资源信息库的数据准确性。

采集库主要采集的信息有：

1）逻辑服务器主要信息，例如序列号、IP 地址、CPU 情况、内存情况、磁盘信息、物理卷信息、卷组信息、逻辑卷信息、网卡信息、运行情况等基本信息。

2）系统部署的软件版本及补丁、硬件板卡、存储、实例、应用情况等扩充性信息。

3）操作系统基本信息、系统服务状态、系统核心参数、文件系统、服务端口、用户信息等。

4）数据库实例信息、表空间信息、数据文件信息等。

采集库的整个流程设计大致可明显地划分为采集、入库以及信息同步三步。

1. 采集

云管理平台将采集脚本下发至被管理服务器，然后通过自动化运维工具定时执行采集脚本，将采集的信息文本发送给 SFTP 服务器，采集库即可对采集文件进行解析入库。

2. 入库

考虑到异常信息处理，入库过程大致可分为三个步骤，首先是入库文件检查，其次是信息入库，最后是信息提交与入库文件处理，其整体流程如图 10-3 所示。

（1）入库文件检查　主要核查文件信息的合规性，入库文件的文件名是否符合要求、是否存在重复主键问题，字段格式、文本格式是否符合要求，对于不符合要求的需记录相关异常信息。对于正常文本，应记录采集日期、采集文件名称、设备名称、是否

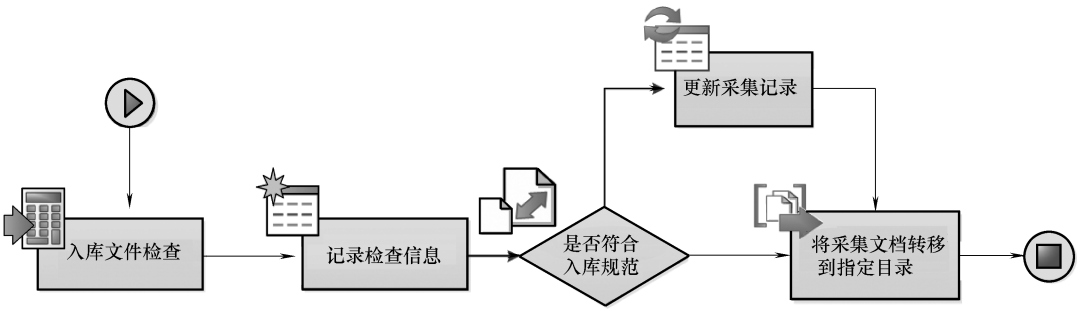


图 10-3　采集信息入库整体流程

更新等字段信息。

（2）信息入库　对于新老设备，分别采取不同的更新方式。对于新设备，首先需要检验文件的 MD5 码，与上一次的采集文件进行对比，如果发生变化，则进行修改；对于老设备，首先删除掉以前的信息，然后再入库，作为一张中间表的存在去更新。

（3）入库文件处理　每次入库完成，对于异常未入库成功的文件，将文件转移到错误数据备份目录，不更改名称；对于正常入库文件，将入库文件转移到正常数据备份目录。

3. 信息同步

由于采集库的着眼点主要是进行信息的收集，在逻辑关系上并不明确，所以需要资产信息库向采集库提供相关的逻辑关系表，然后再建立符合资产信息库要求的视图，最后再返回相应的采集信息。资源信息同步流程如图 10-4 所示。

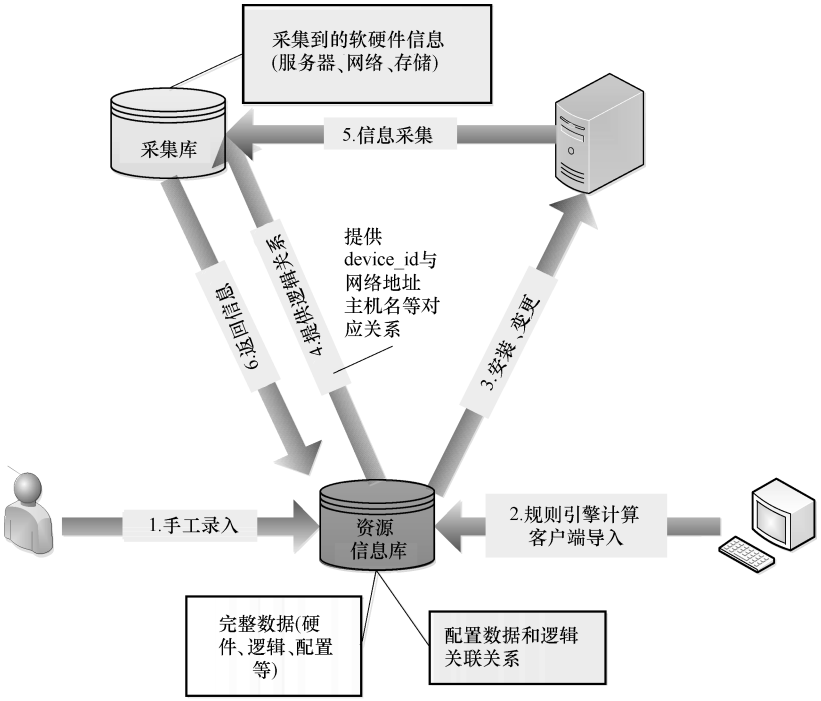


图 10-4　资源信息同步流程

随着需求的增多，采集项也会增加，目前的逻辑关系主要来自资源信息库，但是不排除从其他渠道导入逻辑关系到采集库中，最终实现采集数据的结构化以及可视化，从而达到信息利用价值的最大化。

10.2 弹性伸缩

10.2.1 云服务管理

云服务管理用于将云服务开发的成果注册到云管理平台中，云管理控制模块根据注册的云服务信息结合自动化工具实现云服务的自动部署，形成服务实例以提供云服务能力。

云服务将服务目录相对应并发布到用户自助服务门户中，从而实现服务目录到云服务的映射。云服务管理将用户看到服务的描述与镜像、云服务部署实现对应起来。云服务管理的功能模型如图 10-5 所示。

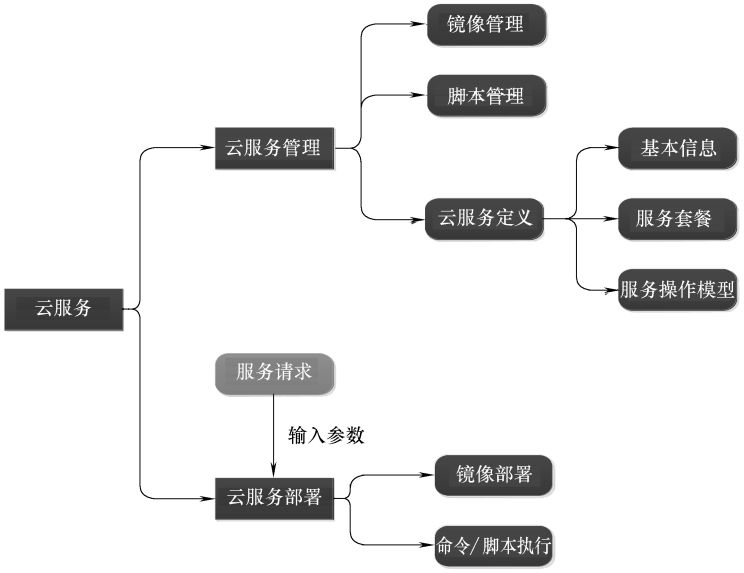


图 10-5 云服务管理的功能模型

10.2.1.1 镜像管理

云服务是由一个或一个以上的云服务部署单元通过编排、集成和测试实现的。在开发云服务部署单元时，需要首先确定云服务部署单元的功能需求，再通过对领域部署单元的封装，形成包含不同封装内容（粒度）和可以产生不同云服务类型（IaaS、PaaS 或 SaaS）的云服务部署单元。保留硬件技术相关性（例如是否为 Power CPU），并保留除各领域必须指定的硬件配置相关性（如 2 个网卡）外，将部署单元剥离硬件配置相

关性后形成的静态文件即为镜像。镜像文件是云服务部署单元的物理存在形式。

镜像管理功能用于集中管理云管理平台中的镜像，包括镜像基本信息和镜像服务器地址的维护，但是不涉及镜像的制作。按照云服务的设计，首先通过手工构建形成系统镜像包，然后将镜像包上传到相应的镜像服务器，最后在云管理平台进行登记注册。

一个完整的镜像信息包含 8 个方面的内容。

1）镜像 ID：标识镜像表的主键，以时间顺序命名。

2）镜像名称：用于在云服务目录展现镜像名称。

3）管理员：该镜像的维护者，只有镜像管理员可以维护镜像信息，默认为创建者。

4）镜像描述：用于在云服务目录展现镜像的具体描述。

5）存放路径：镜像在镜像服务器上的存放路径，用于云服务供给时的镜像复制。

6）镜像服务器信息：用于标识该镜像在不同数据中心所在的镜像服务器，不同虚拟化厂商软件的镜像放置在不同的镜像服务器上，例如 VMware Vcenter 的镜像放置在 Linux 服务器上，而 IBM 的 Power Vioc 镜像则放置在 NIM 服务器上；由于云管理平台是两地三中心一体化管理，所以在不同数据中心会有不同的镜像服务器，以便本地化安装。

7）软件配置信息：标识该镜像中已经预安装的软件或者已有的软件介质，便于用户选择所需要的云服务，包括应用软件、通用技术软件、ITSM 管理软件、中间件、操作系统等信息。

8）系统用户信息：标识该镜像中已经创建的系统用户，用于在资源供给时进行系统权限分配。

10.2.1.2 脚本管理

在云服务供给和日常运维中会执行大量脚本，而这些脚本就是通过脚本管理来注册和集中管理的。

脚本通过脚本管理注册到云管理平台中，在进行云服务定义时，将脚本与具体的云服务绑定，云服务部署再由云管理平台负责将脚本打包推送到具体的机器上。脚本管理主要包括脚本包信息的新增、修改、删除、查看、批量上传功能和脚本信息的新增、修改、删除、查看、上传功能，脚本管理以树状形式维护脚本包和脚本，与物理存在的文件夹、脚本一一对应。

1）脚本包的内容主要有：

① 包名称：脚本包名称，用于在云服务定义和日常运维中选择脚本。

② 负责人：用于在脚本出现问题时或者需要优化时找到相应维护人员。

③ 包路径：用于标识脚本包在脚本服务器的路径以及下发到被管理服务器的路径；包路径需要全局唯一。

④ 功能模块：用于说明该脚本包的用途，主要分类有健康检查、合规检查、服务启停、软件安装等。

⑤ 所属应用：用于标识该脚本包所属应用，如果有关联应用，则只有该应用的管理员可以查看该脚本包，也只有这个应用下的设备可以执行该脚本包下的脚本；如果没有关联应用，则所有管理员均可查看该脚本包，所有设备都可执行该脚本包下的脚本。

⑥ 阈值文件：健康检查、合规检查类脚本都需要用户设定阈值，这些阈值就保存在阈值文件中；如果在脚本包中设置了阈值文件，则其下级脚本包不允许再设置阈值，其平级可以设置阈值文件。

⑦ 备注说明：用于描述脚本包的具体用途。

2）脚本的对应物理存在的脚本文件，内容主要有：

① 脚本名称：标识脚本用途，用于在云服务定义和日常运维中选择脚本。

② 文件名称：对应的脚本文件名称。

③ 执行用户：运行该脚本的系统用户。

④ 回退脚本：如果该脚本运行失败并需要自动运行回退脚本，则可在此设置需要自动执行的回退脚本。

⑤ 阈值：健康检查、合规检查类脚本在执行时会读取阈值文件中的阈值，此处设置阈值名称（对应脚本中的读取参数）、阈值输入类型（数字型、时间、文本等）。

⑥ 是否自定义阈值：如果选是，则该脚本在用户设置阈值时，用户可自行定义阈值的关键字和值。

10.2.1.3 云服务定义

云服务以服务目录的形式提供给用户，服务目录是将识别的技术服务按照服务类型和特点归类，并提供给使用者，方便使用者学习、查询、使用的工具。建立服务目录的过程就是一个识别服务、限定服务对象和内容、确定服务级别和规范的过程。

云服务定义包含：云服务基础信息、服务套餐信息、服务操作模型和云服务参数4部分。

1）云服务基础信息用于方便用户选择云服务，记录云服务的基础信息，其内容包括：

① 云服务名称：用户所选择的服务目录即云服务名称，简要描述云服务的内容。

② 云服务类型：主要有计算、存储和网络三类，方便用户选择所需云服务。

③ 高可用类型：如果云服务类型为计算，则需要选择高可用类型，其包括HA、RAC、集群、单机等，也用于方便用户选择所需要的云服务。

④ 服务器基数：计算云服务的高可用类型分为HA、RAC、集群和单机四种，HA和RAC需要服务器成双提供，集群是三台，单机是一台；云服务都是成套提供的，每套服务器的数量即为服务器基数，用户在申请时，只需填写云服务套数即可，系统会自动计算出需要提供的服务器数量。

⑤ 负责人：用于在云服务出现问题时或者需要优化时找到相应维护人员。

⑥ 业务功能：用于在云服务目录展现云服务的业务功能。

⑦ 系统镜像：选择云服务所使用的系统镜像。

⑧ 脚本集合：选择云服务在供给中所用到的脚本，这些脚本在服务器供给时会先

下发到服务器上。

2）服务套餐是指云服务在设备上的配置，主要包括 CPU、内存、磁盘大小、网络带宽等。通过服务套餐的设置，可以指定云服务在供给时所需使用的资源大小。

3）操作模型是设置云服务的部署模式，分为供给模式、扩容模式和回收模式。

4）云服务参数是用于设置云服务所独有的一些参数，这些参数会在云服务供给时用到，这些参数可以通过前台页面配置添加，而无需另外开发，比较方便快速添加或优化云服务。

10.2.2 云服务部署

云服务部署是指云管理平台根据用户选择的云服务以及申请信息自动化部署云服务的过程，最终用户在自助服务门户提交服务请求后，云管理平台利用流程引擎结合自动化工具实现云服务自动部署。云服务部署的过程应该是可监控的，也就是说云服务操作模型的各种计划流程执行过程对系统管理员是可监控的、可操作的。云管理平台依靠流程引擎执行操作模型的各种计划流程，将云服务部署的各种步骤串联在一起，实现云服务的自动化部署和供给。流程工具支持图形化的设计和监控界面，同时支持流程的异常处理和断点接续机制。

云服务部署包括云服务供给、云服务扩容和云服务回收。这里将以“PaaS_AIX_Oracle RAC 联机交易数据库云服务”为例详细介绍云服务部署的设计和开发方法。

1. PaaS_AIX_Oracle RAC 联机交易数据库云服务供给

PaaS_AIX_Oracle RAC 联机交易数据库云服务是提供联机交易的高可用数据库服务，由 AIX 操作系统平台的两个虚拟机作为 Oracle RAC 的两个节点，加上共享磁盘的方式组成。每台虚拟机均安装 AIX 7/Oracle 11g 软件。该服务一般工作在数据处理层，对关系型数据提供创建、查询、更新、删除等处理能力，通过双节点集群方式保障高可用性。其领域操作及操作编排如图 10-6 所示。

2. PaaS_AIX_Oracle RAC 联机交易数据库云服务扩容

用户通过云服务申请设备后，如需对设备进行扩容或者缩容，可以通过选择“虚拟机扩容/缩容”服务请求来进行调整，云管理平台会自动根据设备当初对应的云服务选择相应的操作模型。

PaaS_AIX_Oracle RAC 联机交易数据库云服务扩容操作模型如图 10-7 所示。

10.2.3 资源池管理

资源池是云平台的重要组成部分，资源池管理主要用于对计算资源池、存储资源池、网络资源池等的定义和管理，以及资源的自动化安装与注册。

10.2.3.1 计算资源池

计算资源池主要用于定义计算资源，其对应的是物理主机及虚拟机。计算资源池分

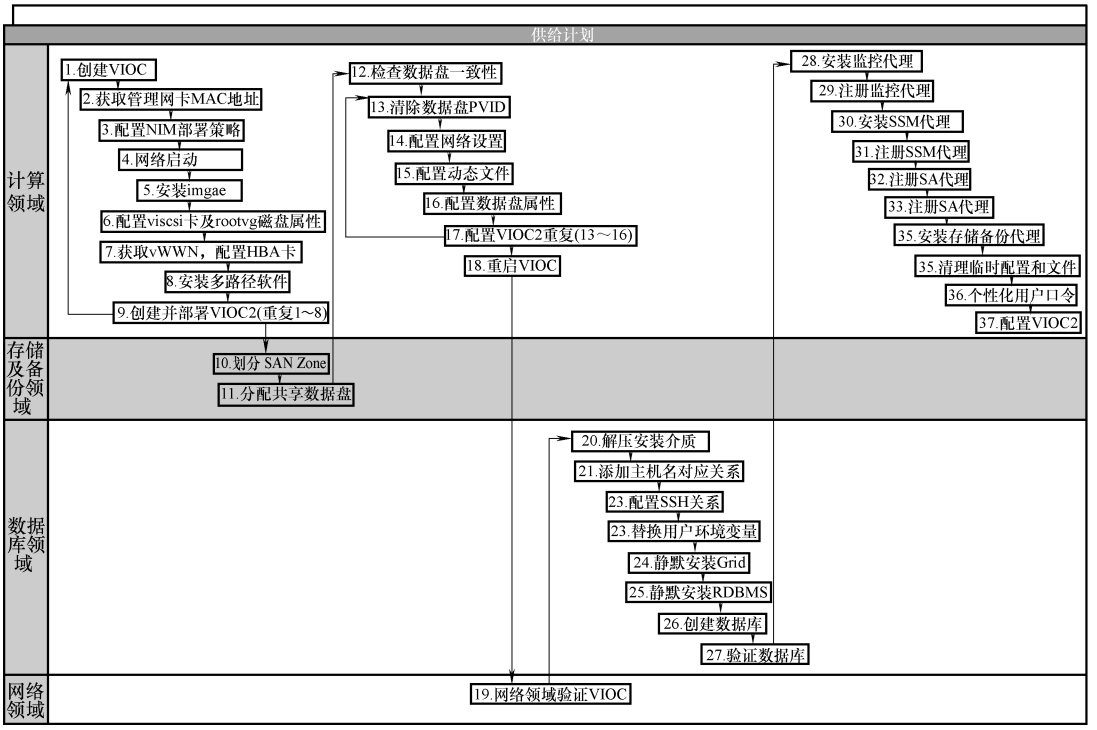


图 10-6　PaaS_AIX_Oracle RAC 联机交易数据库云服务的领域操作及操作编排

注：MAC：物理硬件地址；NIM：网络安装管理，对 AIX 操作系统进行安装和软件维护；；image：镜像；vWWN：虚拟全球唯一名字；HBA：主机总线适配器；VIOC：IBM 研发的虚拟 IO 服务器；SAN ZONE：存储区域网络分构区；PVID：物理卷 ID；SSH：安全外壳协议；Grid：网格；RDBMS：关系型数据库管理系统；SSM：服务调度管理系统；SA：服务器自动化系统。

层结构如图 10-8 所示。

其中，数据中心为所有资源池所共享，即在任意一个资源池维护页面中定义了数据中心后，其他资源池即可看到该数据中心的信息。通过该维护页面，用户可以定义数据中心、计算资源池、部署单元、集群，然后将物理主机与集群进行关联。数据中心下层即是资源池，资源池的信息如下：

1）名称：资源池名称，用于资源池管理。

2）英文名称：资源池英文名称。

3）服务类型：资源池可分为接入资源池、应用资源池、数据资源池和管理资源池；接入资源池用于分配 Web 应用所使用的虚拟机，应用资源池用于分配 AP 应用所使用的虚拟机，数据资源池用于分配数据库所使用的虚拟机，管理资源池用于 IT 运维管理类软件所需要使用的虚拟机。

4）平台类型：是指资源池所管理的硬件平台，包括 X86、AIX 和 HP，而这个类型的设置也表明每一个资源池只能管理一种类型的硬件平台。

5）安全区域：指资源池所管理的服务器及其上所部署虚拟机所在的网络区域，包

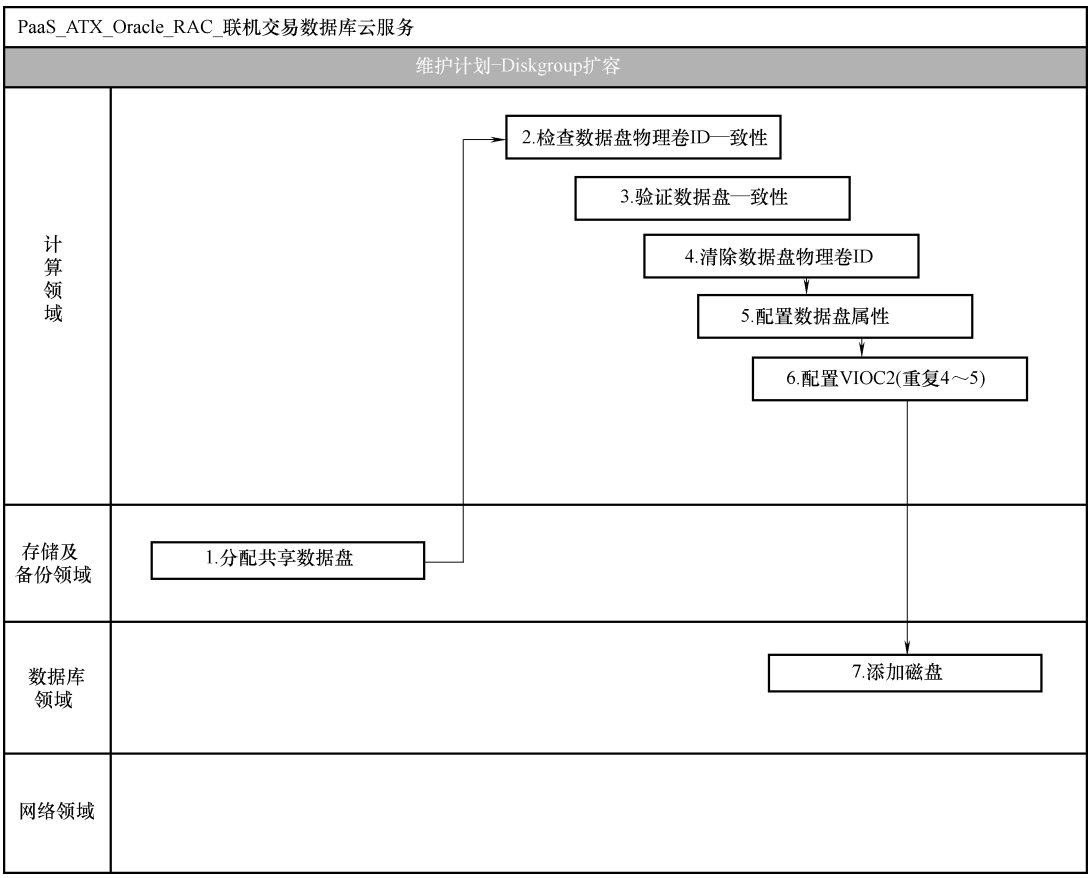


图 10-7　PaaS_AIX_Oracle RAC 联机交易数据库云服务扩容操作模型

括互联网 DMZ 区、外联网 DMZ 区、开放服务区、运维管理区、灾备区等。

6）主机操作系统类型：是指资源池管理的服务器上安装的 Hypervisor 软件类型。通过 VMware 虚拟化产品实现的虚拟化资源池安装的是 Exsi；通过 AIX 虚拟化产品实现的虚拟化资源池安装的是 VIOS；通过 KVM 虚拟化产品实现的虚拟化资源池安装的是 Linux；HP 资源池使用的是物理机模式供给，安装的是 HPUX。

7）虚拟比：一台物理机上可以分配虚拟机的最大个数；如果 CDP、集群设置了虚拟比，则分配虚拟机时参照最下层的设置，如果没有设置则使用资源池的设置。

8）CPU 分配比：一台物理机最大可以超额分配 CPU 比例，比如一个 X86 物理机的 CPU 为 48C，CPU 分配比设置为 150%，则该物理机最大可分配 CPU 为 72C，与虚拟比一样，资源池、CDP、集群均可设置该数值。

9）内存分配比：一台物理机最大可以超分配内存比例，算法同 CPU 分配比。

10.2.3.2　存储资源池

存储资源池主要用于定义存储资源，其对应的是存储设备。存储资源池分层结构如图 10-9 所示。

通过存储资源管理维护页面，用户可以定义数据中心、存储资源池、构建单元、部

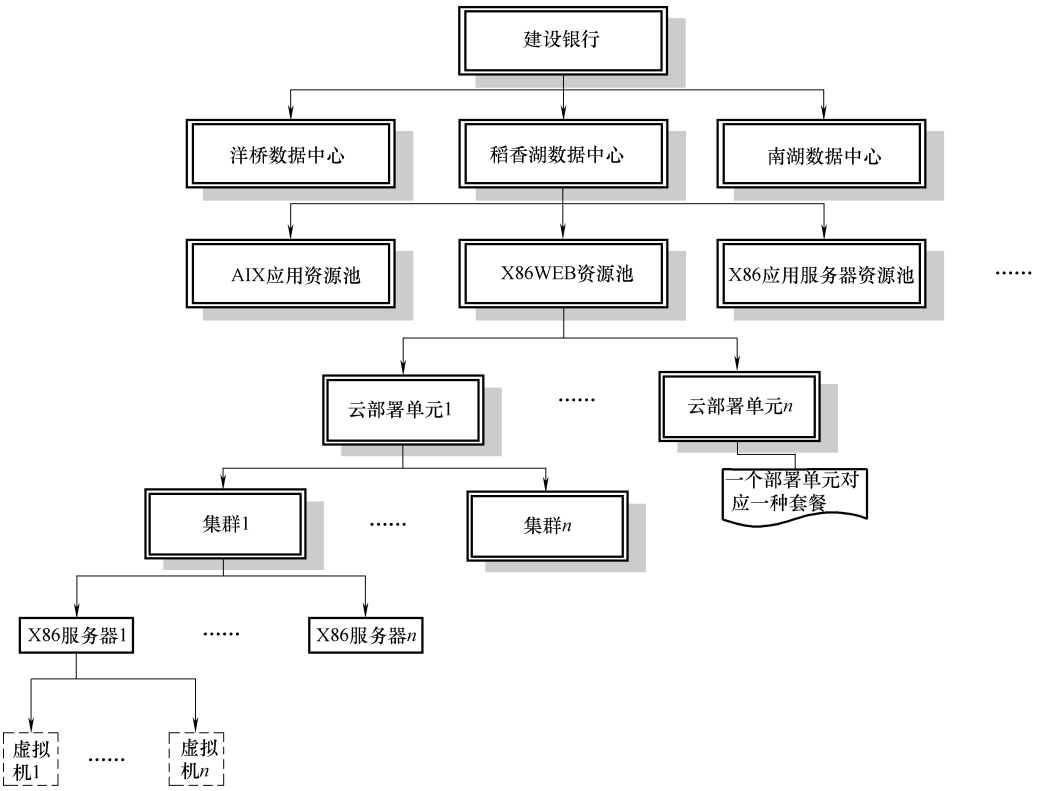


图 10-8　计算资源池的分层结构

署单元，然后将存储设备与构建单元关联。存储资源池用于区分存储使用的技术和服务级别，具体信息如下：

1）名称：资源池名称，用于资源池管理。

2）说明：资源池的用途说明。

3）资源池类型：存储资源池主要有两类，NAS 和 SAN。

4）服务级别：SAN 资源池分为白金级、金级和银级；NAS 资源池分为白金级、银级和铜级。

10.2.4　资源动态分配

云管理平台可以持续不断地监控主机集群中资源池的利用率，并能够根据业务需要在虚拟机中智能地分配其所需的资源。通过将资源整合、池化，并利用高度自动化的管理工具实现资源的动态分配和共享，在规模化的基础之上实现了对底层 IT 资源的充分利用，降低了单位 IT 资源的投入成本。

针对应用需求，资源分配时应遵循如下分配原则：

1）根据应用对计算、存储和网络资源的需求选择相应资源池。

2）根据当前资源使用情况选择资源池中云部署单元。

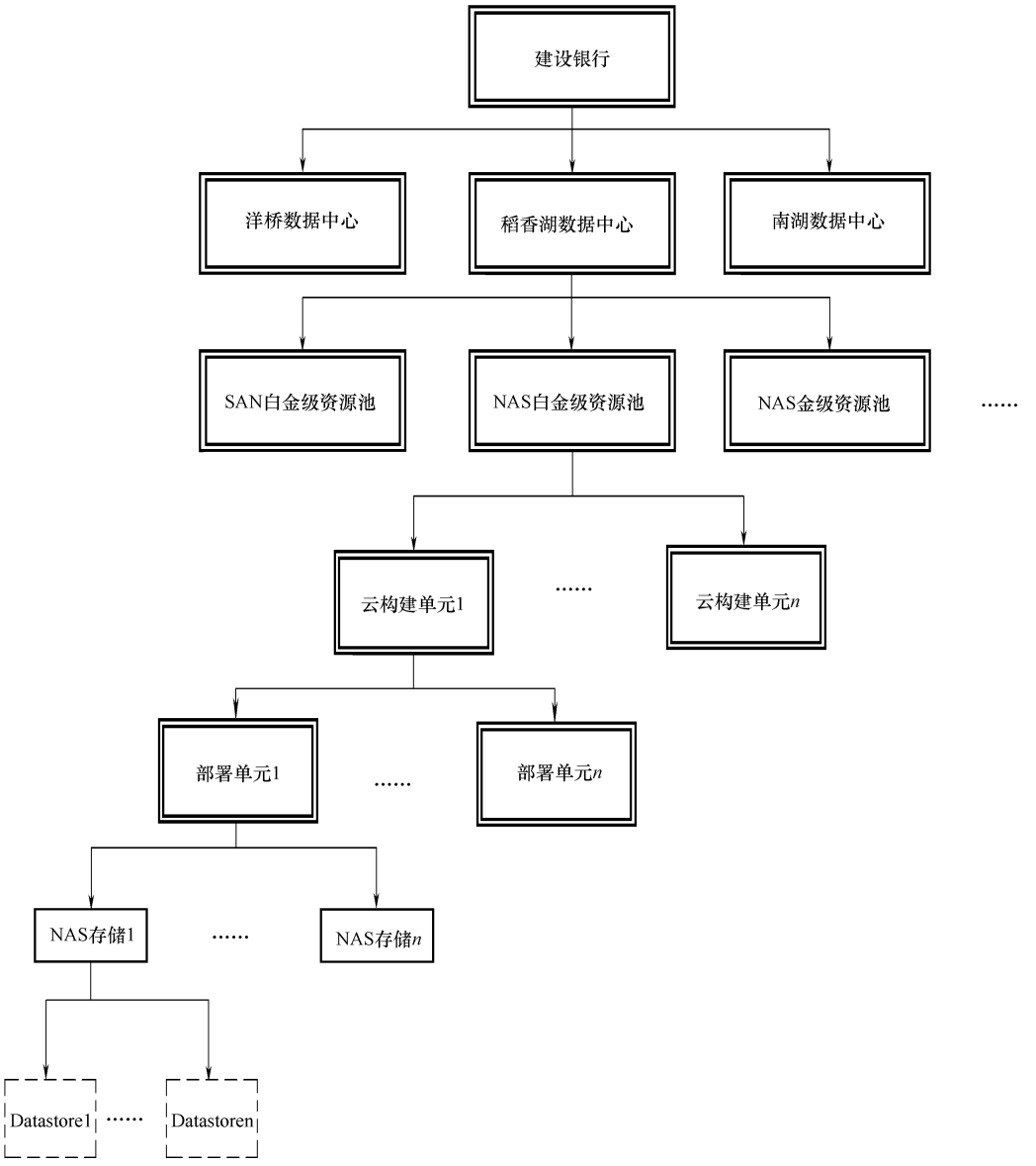


图 10-9　存储资源池的分层结构

3）每个应用平均分配在同一云部署单元的不同集群内。

4）对一个群集内所分配的虚机，应部署在不同的物理机设备上。

5）虚拟机的分配优先选择空闲资源最多的物理机设备上。

例如：某项目需要 33 台 Web 服务器资源，根据资源分配原则，在 Web 服务器资源池内每个服务器集群分配 11 台 Web 服务器虚拟机，这 11 台虚拟机所属于不同的物理服务器；后期另一个项目上线，需求为 62 台 Web 服务器资源，根据资源分配原则按照 3 的倍数部署 63 台 Web 服务器，从资源使用率最低的物理服务器上分配虚拟服务器资源。资源动态分配示意图如图 10-10 所示。

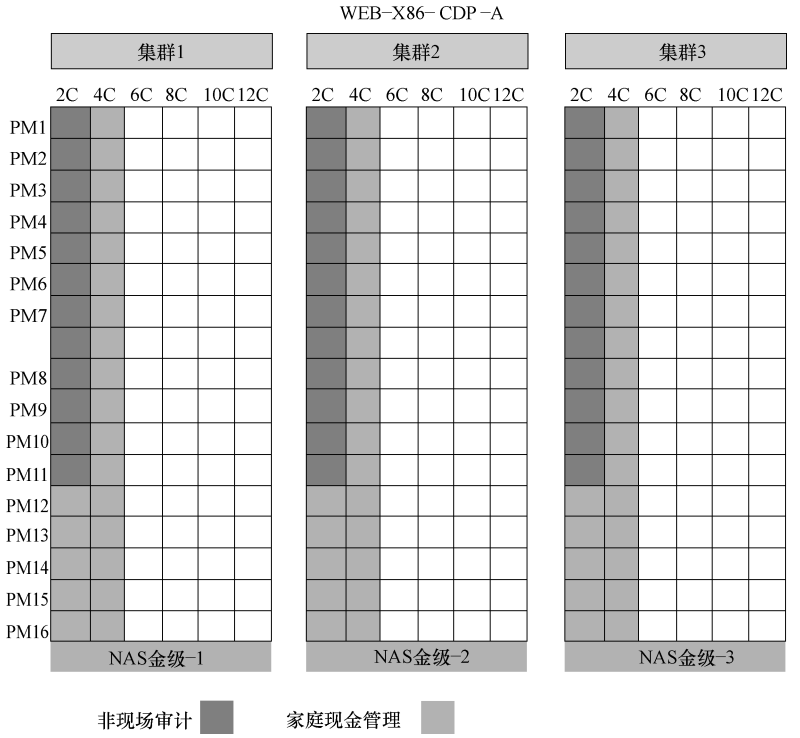


图 10-10　资源动态分配示意图

10.2.5　资源自动化管理

10.2.5.1　物理服务器自动化安装

建行云管理平台支持 X86、AIX 小型机、HP 小型机的物理机操作系统自动化安装，其架构如图 10-11 所示。

对于 AIX 和 HP-UX 来说，由于机器数量较少，所以使用了原厂提供的 NIM 和 IG-NET 方式，同时对这两种方式进行了包装，提供了统一的安装界面。

对于 X86 服务器，由于数量和需求众多，建行云管理平台采取了创新模式，在传统服务器安装的基础上，开发了一个小型操作系统，命名为 BootOS。BootOS 的设计思路如下：

1）一个基于 Linux 的精简操作系统，比较小，只有 100～200MB，能够快速地从网络下载。

2）从网络启动并完全运行于内存中。

3）能通过 SSH 远程登录，并具有常用的 Linux 管理命令。

4）能与云平台通信，接收并执行云平台命令，返回结果。

5）能按需求扩展。

利用 BootOS 系统可以在物理机上自动化批量统一安装 ESXi、RHEL、CentOS 等操

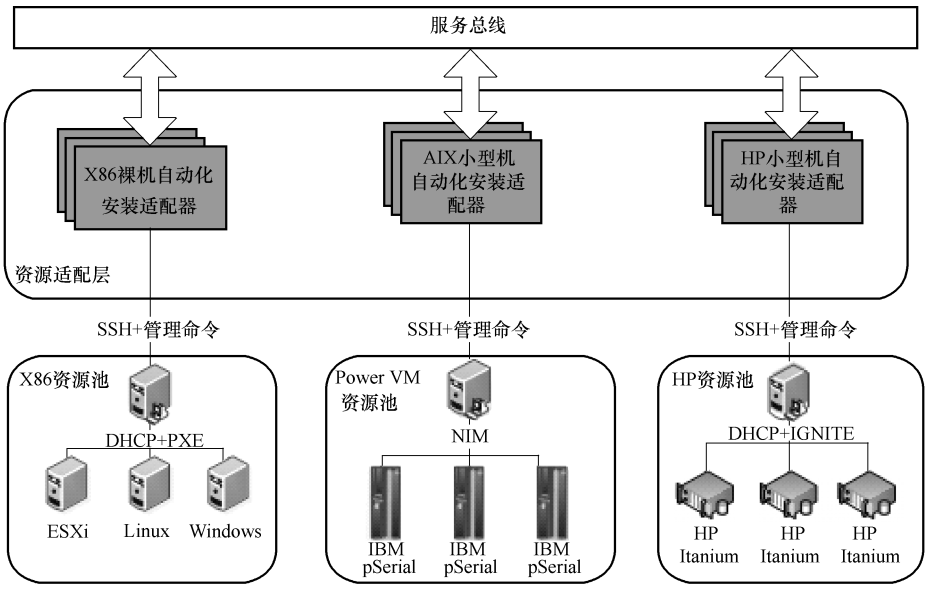


图 10-11　物理机操作系统安装架构

注：SSH：安全外壳协议；DHCP：动态主机配置协议；PXE：预启动执行环境；NIM：网络安装管理，对 AIX 操作系统进行安装和软件维护；PowerVM：IBM Power 系列虚拟机；IBM pSerial：IBM P 系列服务器；HP Itanium：HP 安腾服务器；IGNITE：HP 操作系统网络安装服务器。

作系统。安装过程如下：X86 裸机通过网卡启动后，首先发送 DHCP 请求获得 IP 地址，然后裸机下载 BootOS 的镜像文件，下载完毕后 BootOS 完全在内存中独立运行，同时 BootOS 提供接口可以接收来自云平台的消息并执行任何云平台发送的指令。

除了统一安装操作系统外，BootOS 还具有以下特有的功能：

1）自动化采集服务器硬件配置信息，并能做到实时准确，将人员从传统的手工输入中解放出来，并且还可以避免因手工输入造成的信息不准确。

2）在服务器安装系统前对其进行硬件配置，如划分 RAID，使得相关人员不必进入机房即可完成指定的任务，极大地减轻工作压力。

3）服务器进入机房加电后，所有的系统安装、配置操作完全自动化。

4）BootOS 的设计分为六个模块，分别是：

① 网络模块：BootOS 启动之后，通过 DHCP 为对应的网卡获取地址，保证网络畅通。

② 心跳模块：BootOS 启动之后，会在服务器上起来一个 Client 进程，该进程会周期性地向云平台 BootOS 后台服务发送心跳信息。BootOS 发送心跳信息的目的是为了让云平台感知自己的存在。

③ 监听模块：BootOS 不仅需要向云平台主动推送消息（心跳、注册），也需要接收来自云平台的命令，因此 BootOS 上必须有能够监听来自云平台消息的服务。我们采用 Apache 提供的 Http 服务。

④ 命令模块：BootOS 收到监听模块的命令之后会将命令转交给命令模块，命令模

块负责解析和执行命令。

⑤ 硬件模块：硬件模块负责按照云平台命令进行硬件配置，这部分实现了与硬件交互的作用，主要包含驱动安装和脚本编写。在这个模块中，主要应针对 RAID 配置和 ILO（Integrated Lights-Out，服务器远程管理系统）配置进行设计。

⑥ 日志模块：BootOS 在整个生命过程中都会有详细的日志记录，包括命令执行结果、心跳发送状态、注册状态、接收云平台命令等。所有的日志都有标准的格式，使得日志易于解析，并进行了分级。

BootOS 自动化安装 ESXi 流程较复杂，因为它不仅要求服务器按标准统一配置，同时还要求各个服务器有不同的配置，如不同的 IP、不同的密码等。自动化安装基本步骤如下：

1）云平台触发批量安装，输入待安装服务器的设备 ID。

2）云平台为待安装主机分配 IP、VLAN 等配置信息。

3）安装服务生成各个服务器特定的安装配置脚本。

4）安装服务向 BootOS 发送命令。

5）BootOS 按命令进行配置，根据自身状态立即执行或者等待执行命令重启。

6）服务器进入安装状态。

7）服务器向安装服务报告安装完成。

8）安装服务调用 vSphere API 纳管 ESXi。

9）记录安装日志。

10.2.5.2 虚拟化服务器自动化管理

建行云管理平台通过虚拟化适配层，实现对于多种平台虚拟化软件的支持。当前已经实现了对 X86 VMware ESXi 虚拟化软件、AIX 小型机 PowerVM 虚拟化软件支持，能实现完整的虚拟化服务器安装、配置和管理，并且能支持按需扩展。

对于 X86 服务器虚拟化的管理，为了更好地自主掌握核心技术、摆脱对厂商软件的依赖，建行云管理平台本着减少与厂商管理软件的交互而更多调用底层 API 的原则，在设计上更多地调用各个物理机上 Hypervisor 的 API，将大量管理层的功能封装在云管理平台，而不通过厂商管理软件直接实现，使得云管理平台在同样的管理模式和管理策略下，不仅可以管理 VMware ESXi 虚拟化软件，还可以管理华为虚拟化软件以及 KVM 虚拟机。

对于 AIX 小型机 PowerVM 虚拟化软件，云管理平台则通过封装 HMC 命令实现 PowerVM 的创建、管理、资源分配和虚拟机启停。

10.2.5.3 网络自动化管理

建行云管理平台一期项目实现了 IP 地址的自动化配置和管理、F5 负载均衡的配置管理，相关设计要点如下。

1. IP 地址分配

云管理平台根据判断网络资源请求的来源确定 IP 地址类型，再根据用途确定 IP 地

址（C 类地址段）。IP 地址分配涉及的属性包括安全区域、设备类型和 IP 地址类型，另外如果 IP 地址为生产地址，则还包括安全分层属性。根据计算资源池的申请请求，确定所申请 IP 地址的基本属性。

IP 地址分配分为物理机地址分配和虚拟机地址分配两部分。

（1）物理机地址采用预分配方式　物理机地址分配根据计算资源池的规划进行预分配，即为计算资源池中当前已部署或以后将要部署的物理机预先分配 IP 地址。

为保证资源池中各部署单元、集群、服务器 IP 地址的连续性，物理机地址在新建资源池时进行预分配，且按照单个资源池的最大需求分配。

X86 机器地址分配策略包括资源池地址预分配、部署单元地址预分配、集群单元地址预分配、物理机地址分配和 NAS 存储地址分配；AIX 机器地址分配策略包括资源池地址预分配和部署单元物理机地址分配（仅有一个部署单元一个集群）。

（2）虚拟机地址采用现用现分配方式　虚拟机地址分配在用户申请虚拟机时一并分配，即根据用户的实时需求实现 IP 地址的自动化弹性分配。虚拟机地址分配策略包括 VMware 虚拟机地址分配和 PowerVM 虚拟机地址分配。

2. 负载均衡自动化

负载均衡用来给后台的 Web 服务器提供负载均衡服务。服务器采用了虚拟化技术，实现了按需自动部署，云管理平台将通过使用负载均衡的接口实现负载均衡服务自动化供给，以满足应用的快速部署需求。云管理平台通过调用 F5 设备的 iControl 接口实现了负载均衡自动化配置。

在云管理平台中，应用管理员将以应用部署单元（部署单元是按照模块或应用功能划分子模块）为单位进行资源申请，包含计算资源、网络资源和存储资源。在应用组件部署单元的资源申请到位后，通过云管理平台完成应用的负载均衡服务交付。

负载均衡自动化的实现根据应用的生命周期，实现了包括应用上线、应用扩容、应用下线和应用维护四个环节的负载均衡的自动化配置。

（1）应用上线　负载均衡服务请求将在应用申请完计算资源后创建，用来满足系统的负载均衡需求。应用管理员填写基本需求参数后提交服务请求，由网络管理员负责初审同时完善参数信息后进入 ITIL 审批流程，待审批通过后，云管理平台将自动生成关联变更单，网络管理员将在变更单关联设备，填写执行时间等信息，确认后进入自动化变更审核阶段。待审核通过后，云平台按照应用管理员和网络管理员填写的配置参数，通过调用负载均衡设备的接口进行自动化配置，实施负载均衡服务请求。

（2）应用扩容　应用管理员在云管理平台新建扩容负载均衡服务请求，审批通过后，将自动生成变更单，网络管理员将待审核通过后，手动或者自动触发自动化操作。

应用扩容申请资源流程中，直接发起对于负载均衡服务的申请，所有负载均衡策略仅可沿用已有策略。

（3）应用下线　应用下线时，在计算资源已下线的前提下，需要将应用关联的负载均衡资源进行释放。具体操作需与网络管理员沟通。

（4）应用维护　应用运行期间，实现对 POOL MBMBER 一键式启停功能和连接限

制设置功能，使得应用系统能根据需求实现负载均衡弹性配置。

10.2.5.4 存储自动化管理

存储资源是重要的运维资源，是整个自动化运维的关键一环。存储自动化之前，存储的分配是通过手工来完成的，从 ZONE 的划分到 VIEW 的创建，存储的选择、LUN 的选取、端口组的选择，一系列过程都需要人工来操作。存储自动化实现了这一过程，通过程序来模仿人工操作，来达到自动分配存储的目的。云平台采用 SMI-S 进行对 EMC、Cisco 的设备进行统一管理。

SMI-S（Storage Management Interface Specification 存储管理接口标准）是 SNIA 开发的一种标准管理接口，旨在减轻多厂商 SAN 环境的管理负担。SMI-S 发布至今已经取得多家 SNIA 成员企业的认可与支持。SMI-S 的目标是，在存储网络中的存储设备和管理软件之间提供标准化的通信方式，从而使存储管理实现厂商无关性，提高管理效率、降低管理成本，促进存储网络的发展。采用 SMI-S 能够在 SAN 中轻松集成和管理来自多厂商的组件，从而提升了灵活性、可管理性和可靠性；同时，用户的资源利用率也将获得极大的提高。

云平台的存储自动化在整个自动化运维平台中属于自动化组件，与服务器自动化、网络自动化和虚拟化管理共同组成自动化组件，给上层提供服务。从层次上看，将系统划分为两层：业务层：主要负责将用户的需求转换成服务层可以处理的消息，并调用服务层相应的方法；服务层：主要负责通过 API 与设备进行交互，并在设备上进行不同的操作。

存储自动化系统服务于工作流，通过工作流引擎来调用存储自动化系统。根据存储自动化的实际需求，结合工作流的调用方式，将系统设计分为 8 个步骤，在工作流中体现为八个节点，每个节点代表一个功能，对应着系统的一个功能模块。这个八个步骤基本上对应系统的六个功能模块。

（1）选择存储功能模块　云平台首先根据应用的分配需求，在存储资源池中选择合适的存储设备，同时筛选出合适的 LUN。然后按照给定的规则，选择合适的存储资源池，并发送给服务层。服务层选择存储接口会相应地被调用，接着在每个构建单元里选择一台存储设备进行容量计算。

（2）选择端口模块　主要在存储的端口组中选择合适的端口。存储设备往往有多个前端端口，为了便于维护和使用，需要对这些端口进行分组。所谓选择端口组就是在这些端口组中，按照一定的规则筛选出其中的一个端口组。

云平台首先获取不在任何 PG（Port Group，端口组）中的前端端口，然后云平台通过前端端口名获取到对应的 wwn，这样就可以获得对应的端口模块。

（3）获取 ZONE 信息模块　主要目的就是获取并创建 ZONE 所需要的一些信息。

云平台业务层会根据所给的 Fabric 名称依次获取该 Fabric 名称下所有的交换机，然后获取交换机的端口，接着依次循环交换机的端口，查看主机 HBA 卡是否连接到该端口。如果是，则获取该 HBA 卡所属的交换机名称；如果不是，再查看存储前端端口是否连接到该端口。依此循环，直到所有的主机和存储的相关信息查询完毕。

（4）创建 ZONE 信息模块　主要目的是根据给定的 ZONE 信息，在交换机上创建相应的 ZONE。

云平台首先对需要创建的 ZONE 以 VSAN ID 进行分类，将相同 VSAN ID 的 ZONE 存放在一起。然后依次循环 VSAN ID，进行 ZONE 的创建。创建 ZONE 完毕后添加 ZONE MEMBER，接着将 ZONE 添加到该 VSAN ID 中处于激活状态的 ZONESET 里，最后对 ZONESET 进行激活。

（5）创建 VIEW 模块　该模块主要完成在存储设备上创建相应的 IG（Initiate Group）、PG、SG（Storage Group）和 VIEW。云平台按照业务层传递的信息，按照以上过程最终创建对应的 VIEW。

（6）结果验证模块　该模块主要在存储分盘完成后，通知主机扫盘，然后将主机的扫盘结果和存储分盘结果进行比对，验证两者是否一致。

云平台首先获取主机的扫盘结果，将相同的 LUN ID 进行合并，并统计个数。然后获取程序分配给该主机的结果信息，按同样的规则进行合并，然后比较这两者的信息是否一致。由于扫盘获得的结果中有可能包含主机系统分配存储之前就存在的盘，所以需要用程序分配的结果信息去匹配扫盘结果信息。

10.2.5.5　工作流功能架构

工作流（Workflow）就是工作流程的计算模型，即将工作流程中的工作如何前后组织在一起的逻辑和规则在计算机中以恰当的模型进行表示并对其实施计算。工作流管理系统（Workflow Management System，WFMS）的主要功能是通过计算机技术的支持去定义、执行和管理工作流，协调工作流执行过程中工作之间以及群体成员之间的信息交互。工作流需要依靠工作流管理系统来实现。工作流管理系统为建造工作流模型、运行控制工作流、用户和工作流之间交互三个方面提供了功能支持。

要实现云管理平台运维系统的自动化，工作流技术是驱动业务流程、协调运维工作、约束任务执行的灵魂。工作流技术替代了分散、复杂的线下人工操作，避免了手工操作复杂、繁冗易出错，线下流程不易管理控制的问题。引入工作流可以有效减少运维工作量，将日常 IT 运维中大量的重复性工作由过去的手工执行转为自动化操作。同时，工作流的引入能够实现流程化运维管理，通过流程驱动、组件封装的方式实现专业化、标准化和流程化的运维自动化管理。此外，还能够通过 Web 化的流程作业，实现透明化、可视化的运维管理。

根据云管理平台的功能需求，工作流模块主要需要完成以下功能：工作流引擎、流程设计器、组件管理、服务管理、流程模板管理以及流程实例管理。

1）工作流引擎用于解释流程定义，支持自动任务和手工任务，支持多任务并发/顺序执行。在流程控制方面，在流程活动之间进行导航，包括顺序或并发的操作、最后期限调度，支持条件路由、顺序流、分支流、并发流、嵌套子流、关联子流等流转功能。

2）流程设计器包括开始、结束、分支、聚合、子流程、容器等默认组件，并支持自定义流程组件，通过拖拽的方式绘制流程模板，设计工作流程。

3）组件是工作流流程的重要组成部分，组件通过封装业务表单，实现工作流与业务流程的结合。组件管理模块通过实现流程组件的新增、删除、修改、查询来管理自定义流程组件。

4）服务包括脚本、命令、API 等业务操作，是工作流引擎所加载的具体业务工作内容，通过将服务注册绑定组件来实现业务接口的封装，服务管理模块通过实现流程服务的新增、删除、修改、查询来管理服务。

5）流程模板管理可以实现所有已定义的流程模板的查询功能以及对具体模板的查看、修改、删除等管理功能。

6）流程实例需要有执行、暂停、强制结束、激活、单步执行、监控以及查看运行结果等管理操作，此类操作通过流程实例管理模块实现，并包括对所有流程实例的查询和删除、修改等管理操作。

图 10-12 展示了云平台管理员中云平台用户和各个组件之间的关系。

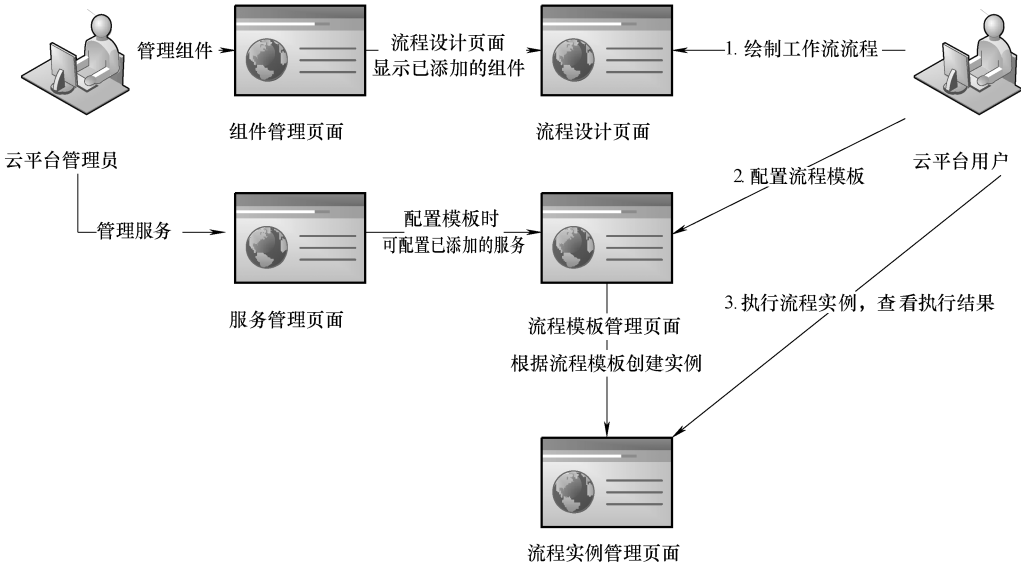


图 10-12　云平台用户和各组件之间的关系

针对工作流模块的需求分析，将工作流组件的组成设计如图 10-13 所示，主要分为工作流工具和工作流引擎两部分。工作流工具包括流程设计器、组件管理、服务管理、流程模板管理及流程实例管理，在图中对应“流程运行监控管理”部分。此部分功能展现并集成到云管理平台门户中，供平台管理员完成服务流程的设计、流程运行管理以及流程状态监控。工作流引擎完成实际的流程运行流转，云管理平台根据实际需要提供定制的客户化流程组件（如：各种与资源层交互的具体资源操作插件）、人工交互任务处理等。

目前云管理平台工作流的功能如下：

（1）流程模型定义　以 JBPM3. 2 为基础构建流程引擎；同时对 JBPM3. 2 流程模型中的流程节点、流转控制、流程活动、上下文传递等进行封装和拓展。

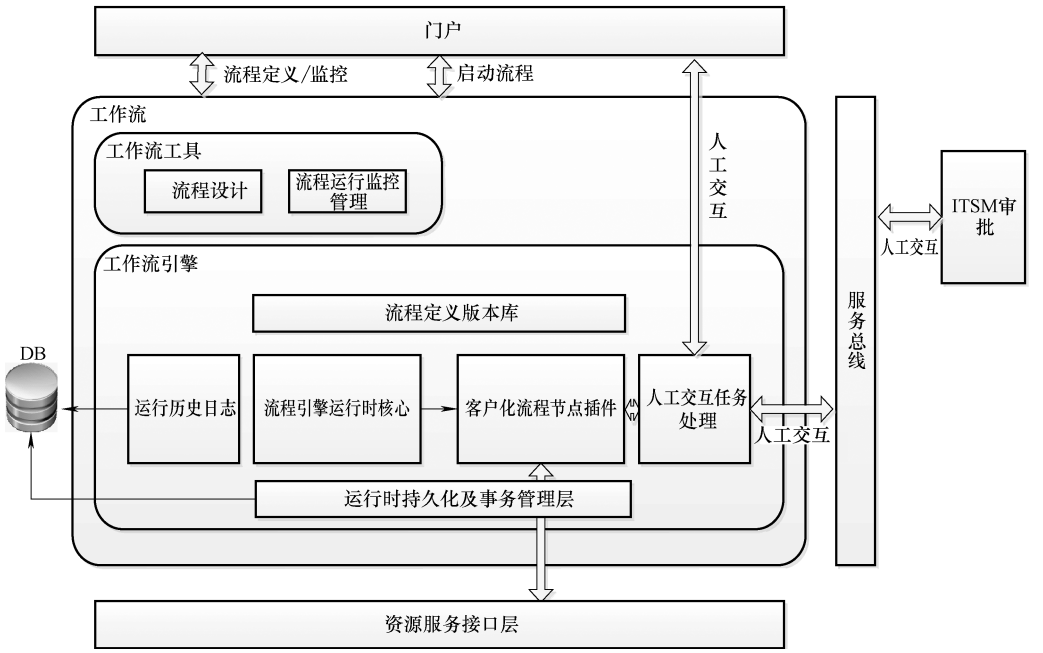


图 10-13　工作流逻辑架构

根据业务场景的抽象定义，设定 3 个基础节点（开始、结束、容器）、3 个路由节点（决策、分支、聚合）和 4 个业务节点（人工、自动、等待、子流）。所有的流程元素都可以绑定事件和动作，来完成用户定义的相关操作，可以根据业务需求定制化地进行封装和拓展。

1）流程节点：

① 开始节点。作为流程的起始端，其业务属性为：完成客户端传递的或预定义的业务参数和流程参数的初始化，以及初始动作的定义。

② 结束节点。作为流程实例执行的终节点，其业务属性为：流程实例运行完成后，触发的其他动作或改变状态等。

③ 容器节点。作为递归嵌套的节点，其业务属性为：在流程定义中产生一些层次，可以任意分组任何节点到一个容器中。同时可以把容器节点的特定事件和业务的某些定义、边界处理、分组定义等进行关联来表达业务。

④ 决策节点。作为单向的路由节点，其业务属性为：决定流程实例的唯一分支的执行路径。选择流程执行路径的方式有两种：一种是由预定义的判定条件来决定流转路径；另一种是根据接口编写特定的程序判定并指定流转路径。该类型节点适用单路路由的判定逻辑。

⑤ 分支节点。作为多路分支的路由节点，其业务属性为：把流程实例的执行路径分割成多条并发执行的子路径，各条子路径可以独立运行、互不影响、不分先后。在分割路径的同时，分支节点为每条子路径创建一个子令牌，这些子令牌和进入该分支节点的令牌属于父子关系。该类型节点适用于多路条件（默认）并发的业务逻辑。

⑥ 聚合节点。作为多路聚合的路由节点，其业务属性为：合并多条并发执行的子路径，并回收所有子路径的令牌。在默认情况下，聚合节点认为到达该节点的令牌同属一个父令牌，当由某个分支节点生成的子令牌全部进入聚合节点时，聚合节点销毁这些子令牌，并通过唯一的输出变迁传递令牌，继续执行流程实例。同时，在聚合时可以做相应的业务定义及实现。聚合节点与分支节点必须成对出现。

⑦ 等待节点。作为一个等待属性的业务节点，其业务属性为：等待外部应用程序或外部系统的调用来完成流转，等点节点负责记录外部驱动的执行、接收外部传递的信息并进行相应的业务处理。

⑧ 自动节点。作为一个自动执行属性的业务节点，其业务属性为：自动执行预定义的业务执行，并当业务执行完成后完成流程的流转。所有的自动执行的业务活动都在此创建，调用执行阶段则交给任务调度端处理。

⑨ 子流节点。作为调用子流的接口节点，其业务属性为：完成子流程的业务调用，从子流定义上包括预定义的子流和根据业务参数动态加载的子流；从执行类型上包括流程引擎内部驱动和等待外部驱动；从执行方式上包括同步执行和异步执行。

⑩ 人工节点。作为资源（参与者）分派的业务节点，其业务属性为：通过和组织权限的集成，来定义该人工节点的组和成员，并控制组和成员的功能权限和数据权限。同时，依据人工节点的属性并集成业务实际，动态的创建任务，完成基础的任务分派，并实现会签、传阅、加签、转交、委托代办等功能。

2）流转和状态控制。

① 流转控制是流程实例在运行中基于变迁的路由指定以及记录流程的流转信息。首先在流程节点元素中已包括了单路条件（默认）执行、多路条件（默认）并发执行和多路聚合的实现，则在流转控制阶段只要完成路由指向的相关功能，即退回提交、回退、特定路由（如：同意/驳回）、任意路由指定等流转控制场景。

② 流程的状态控制包括启动、暂停、激活、结束、强制结束、重启等，在 jBPM3.2 流程引擎本身的实现上已经包含，只需要把通过任务调度的状态控制部分进行业务上的暂停和激活的重新定义。

3）流程活动。流程活动包括业务节点动作、基于特定事件的动作和基于脚本的通过流程参数的路由判定。其具体的调用通过实现 jBPM 的标准接口，并通过 Web Service 的调用来完成业务端的具体业务执行的调用。标准的流程活动实现包括：

① 基于自动任务节点的自动任务的调用，包括 API 的调用、脚本执行、命令执行等。

② 基于人工节点的任务分派及消息通知。

③ 基于子流节点的子流调用。

④ 基于等待节点的外部接口任务的调用。

⑤ 基于 Beanshell 的条件表达式的封装与解析。

4）上下文传递。流程的上下文是指流程实例运行过程当中产生的流程数据和业务数据。其中流程的上下文传递通过标准的 JBPM3.2 实现，业务的上下文传递通过在流

程实例开始阶段的定义和业务执行当中的上下文的交互定义来实现。

（2）流程建模工具　通过图形化的方式描述业务过程，将业务过程的执行逻辑、业务环节的执行规则和内容进行定义，实现业务过程的流程化定义。

流程设计器分四块区域，即操作菜单、流程定义列表、流程组件（基础组件和业务组件）和流程画布。

1）操作菜单提供新建、导入、保存和发布流程定义四项操作。

2）流程定义列表，可以通过右键菜单进行编辑、删除、复制、导出、发布等功能操作。

3）流程组件中的基础组件为默认加载，业务组件是通过流程组件动态加载，容器组件的业务化实现在流程画布的分组中。

4）流程画布作为绘制流程的操作台，通过拖拽的方式实现流程定义。画布上方的菜单栏提供方便快捷键辅助画图等。

另外，基础流程组件根据业务功能的定位，都绑定了一个默认的配置表单；业务流程组件和服务模板的集成也在流程设计器上进行数据配置。

（3）模板及组件管理

1）流程模板。流程模板作为一个完整的业务过程的流程化映射，业务过程的实例化都源于流程模板。在流程模板中通过版本来管理业务过程的变更，同时也可以对业务环节的数据进行修订。

2）服务模板。对业务场景的抽象化、模型化的定义。

① 基本信息：名称、返回码（业务执行结果的判定标识）、业务类型（基于业务的分类）、执行类型（流程组件的关联）和执行路径（流程引擎对应的处理业务执行的接口）。

② 属性信息：以 key-value 的格式进行定义，包括 IO 类型及其对应的操作。其中 IO 的操作包括对 IO 数据进行传递和解析并可以完成三种操作：变为全局的业务变量、变为全局的流程变量、变为全局的业务和流程变量。

3）流程组件。建立服务和流程引擎的接口的关联。流程组件包括名称、类型（工作流节点类型的业务选择）、页面路径（流程组件和业务表单的关联）和接口路径（流程组件调用业务执行的接口）。

（4）流程实例　流程实例是基于流程图显示业务执行过程，对流程进行人工控制，对自动化环节进行人为干预的模块。

1）自动化节点的通用执行类型的定义：

① 自动执行：自动执行预定义的自动化操作。

② 手动执行：通过人工发起业务操作的执行。

③ 跳过执行：跳过某节点的执行，直接流转到下一节点。

2）自动化节点的通用操作类型的定义：

① 单步执行并流转到下一环节。

② 仅单步执行。

③ 不执行业务直接跳转到下一节点。

④ 查看运行日志。

3）自动化流程的流转及状态控制。

① 流转控制包括任意节点的指向、重新启动。

② 状态控制包括暂停、激活与强制结束。

4）权限控制。默认情况下在流程实例的权限控制是开放的，其泛化的约束为：

① 未执行的自动环节无执行的操作权限。

② 已结束的流程无任何的操作权限。

③ 执行完成的环节但流程未结束可以反复执行。

④ 进行分组的前提下，除了默认约束外，只有指定组内的成员可以操作分组下的自动环节。

5）业务执行的异常控制定义：

① 异常挂起等待人工处理。

② 忽略异常继续流转。

③ 强制结束流程。

（5）任务管理　人工任务在人工任务节点的定义及流程实例运行时的任务分配、待办事宜的创建、消息的通知提醒、用户自定义业务页面嵌入流程的流转控制以及预定义数据权限和功能权限的页面引用的接口提供。

（6）流程监控　监控管理分为两部分：对流程实例的监控和对运行时异常的管理。其中流程实例的监控通过业务环节执行状态和流程流转的图形化呈现进行表达；运行时异常是对由于内部代码缺陷或者外部环境问题造成的流转中断进行恢复。

10.3　发布及变更

10.3.1　变更管理设计及实现

变更管理、发布管理和配置管理是 IT 服务管理中最核心的环节，但流程和工具的衔接断层是 IT 服务管理实施的一大难点，导致了 ITIL 最佳实践的“最后一公里”无法顺利落地。针对这一业界难题，建行的云管理平台不仅实现了基础设施的自动化，还将配置管理、变更管理、发布管理的自动化作为重要目标。

在业界“配置管理关联变更”的最佳实践基础上，建行的云管理平台的一大创新是进一步实现了“配置管理驱动变更”，实现了配置管理、变更管理、发布管理的一体化和自动化。将 ITIL 最佳实践固化到系统中，利用云平台集中统一的自动化管理，强制用户在运维管理过程中执行最佳实践，实现 ITIL 关键流程从理论到落地延伸的巨大转变。建行的云管理平台和服务管理流程平台进行了深度的集成，实现了服务申请、服

务审批、服务实施、服务交付等环节的关联与互通，达到了运维流程的自动化。具体来说，就是由流程平台负责人工审批环节，由云管理平台负责自动化环节，实现变更过程的授权审批与变更执行的一体化、标准化与自动化。由于人工审批环节的设计与实现与企业的组织架构和管理紧密相关，本节主要着重描述建行变更自动化的设计与实践经验。

建行云平台变更自动化管理系统的总体功能在结构上分为配置管理层和自动化执行层。配置管理层主要提供配置参数信息的维护和管理功能，自动化执行层则负责变更的自动化执行。其中配置管理分为应用配置管理和平台配置管理两部分。应用配置包括应用逻辑部署结构、应用操作用户、应用变更操作定义、应用变更相关脚本等。平台配置包括操作系统参数、用户、用户组、NAS、ulimits、JDK 等基础软件。在“配置管理驱动变更”的原则指导下，建行云平台实现了日常变更大部分操作的标准化和预定义。自动化执行层基于开源软件 Puppet 和 MCollective 实现，根据建行的实际应用需求，进行了大量的二次开发，实现了开源软件与云平台的深度整合，并做了大量的功能扩展。通过与虚拟机供给流程的整合实现了 Puppet 的自动化安装与纳管，提供了开箱即用的自动化能力。

在应用变更方面建行云平台主要做了以下方面的实践：

1. 应用系统逻辑结构标准化

建行的应用按照子系统、部署单元两个层次进行管理。子系统对应逻辑上的应用系统，部署单元指的是同一种功能类型（如 AP、WEB、DB），并且提供相同业务功能的一组应用服务器集群。建行云管理平台提供了部署单元的录入、查看功能，并以部署单元作为管理的标准单位。应用系统管理员以部署单元为单位，录入所需要的应用系统配置和操作信息，云平台负责实施后续操作。

2. 应用变更操作标准化

通过对日常应用变更步骤的汇总分析，建行云平台将应用变更操作标准分为 8 个步骤，包括启动或停止应用服务、标准备份、定制备份、标准应用分发、定制应用分发、系统文件分发、标准回退和定制回退。在标准操作之外的操作由各系统自己定义，云平台提供相应的统一的自定义操作入口。标准操作的执行方式、脚本编写、文件存放都按照统一规范进行约束。操作标准化一方面固化了应用变更的操作步骤，这样就有利于经验的积累和继承；另一方面也减少了数据中心运维人员与开发中心开发人员之间沟通的成本。

3. 应用变更与工作流的紧密集成

建行云平台的应用变更底层依赖工作流驱动，通过工作流进行变更步骤的编排。变更执行人员可以挑选变更操作灵活组装变更流程，或者通过线下编写并通过审批的变更控制表导入生成变更流程。在流程执行过程中，用户还可以选择变更操作的机器，对于大批量的变更，可以先选择部分机器进行验证然后批量执行，降低变更操作风险。在流程执行过程中，用户也可以实时地查看操作日志，了解变更操作的结果。通过工作流的引入，实现了变更操作的灵活编排和变更的可视化，既提高了变更的效率也能有效地降

低变更风险。

4. 应用变更与平台变更紧密集成

目前在各商业银行数据中心日常运维中，应用变更操作与平台变更操作往往由不同的部门负责，一旦遇上大的变更则需要两个部门的大量沟通与配合工作，比如，先由应用运维人员停止应用服务，接着平台运维人员重启操作系统，然后应用运维人员启动应用服务，最后由平台运维人员启动监控等运维工具等。可以看到一个变更需要多个不同部门的人员协同配合，在一切顺利的情况下还好说，一旦某个环节遇到问题就乱成一团了。既耗费了大量的沟通时间，还容易因为沟通原因引起操作失误。建行云平台通过工作流引擎将应用变更操作和平台变更操作编排在一起，运维人员在一个操作界面上就可以完成整个变更操作，不仅变更过程一目了然，而且能够有效地提高变更效率，减少沟通成本，降低变更风险。

5. 配置参数统一管理

建行云平台配置参数管理的设计目标是集中、统一管理平台系统配置参数，利用自动化工具实现配置批量更新，减少手工变更风险，提高工作效率。平台配置参数管理的逻辑架构如图 10-14 所示。

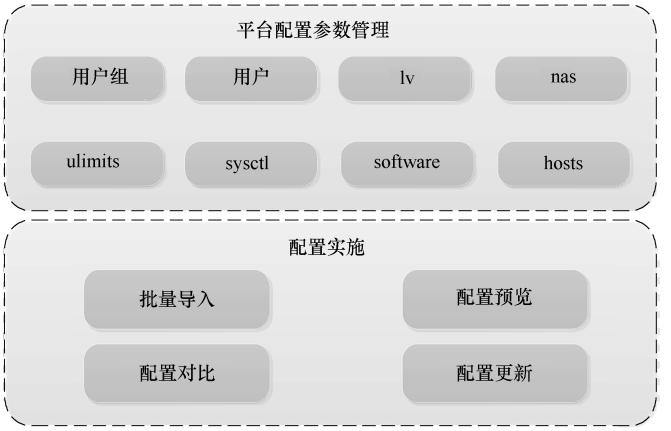


图 10-14　平台配置参数管理的逻辑架构

注：lv：逻辑卷；nas：此处指 LINUX 操作系统上的 NAS（网络附属存储）挂载点配置；ulimits：用于设置 shell 启动进程所占用的资源，是 LINUX 操作系统上的配置文件；sysctl：用于查看、设置/proc/sys 目录中内核设置；software：用于管理操作系统所需安装的软件；hosts：一个没有扩展名的系统文件，其作用就是将一些常用的网址域名与其对应的 IP 地址建立一个关联“数据库”。

建行云平台以部署单元为单位管理常用的平台配置参数，配置项可以根据需求定义扩展。

在配置实施层面，建行云平台支持以下功能：

1）支持配置数据批量导入，用户可以通过 Excel 编辑需要变更的配置参数，批量导入云平台配置库。云平台负责对导入数据的合法性、逻辑性进行校验。对于大批量新上线的机器而言，配置参数的批量导入是效率最高的操作方式。

2）支持配置文件版本比对。通过配置版本比对，能够及时发现配置差异，有效避免变更风险，能够了解配置的变更历史，起到配置审计的作用。

3）配置预览。在配置实施前，可以先进行配置更新预览，用云平台最新的配置参数与实际机器上的配置参数进行比对，清晰地了解本次配置变动的内容，能够有效避免误操作。

4）配置项、目标机器灵活选择。在实施配置更新时，云平台支持选择需要更新的配置项和更新的目标机器，用户可以选择一个或者多个配置项，大大地提高了变更的可操作性和灵活性。

在平台变更管理方面，建行云平台主要做了以下方面的实践：

1）对开源软件进行深度整合和二次开发。建行云平台配置管理底层基于 Puppet 开源软件实现，首先对 Puppet 的使用方式进行接口标准化，整个云平台使用统一接口与 Puppet 进行交互，其次对 Puppet 的功能进行扩展，针对建行的需求开发大量新的模块和 Facter，扩展 Puppet 的功能。另外，为了降低用户使用的复杂性和减少操作的风险，云平台基于 Puppet 和 MCollective 专门开发和封装了一些专用的命令和脚本，大大方便了日常的运维操作。

2）两地三中心的支持。建行云平台建设之初就考虑了多数据中心支持的问题。具体的平台配置层面，基于 Puppet 的架构特点，建行云平台设计了一套管理系统管理多套 Puppet 系统的架构，在每个数据中心都部署了一套独立的 Puppet 配置管理系统，所有的 Puppet 都由建行云平台进行统一调度管理，实现了用户操作入口和操作模式的统一，也减少了云平台自身的运维压力。另外，各数据中心 Puppet 系统互相隔离，能够有效隔离操作的风险，避免互相影响。

3）多种使用模式支持。相对于应用变更基本上都是以系统为单位，平台变更的应用场景更具多样性。既有以操作系统为维度的变更，比如按 Linux、AIX、Windows 划分，也有按照物理机、虚拟机为维度的变更，也有按照资源池、集群维度的变更运维。建行云平台支持多种维度的变更范围选择，既可以按照数据中心—资源池—集群维度，也可以按照操作系统属性、机器类型、中间件类型等属性选择变更的设备，使得云平台可以支持不同需求的变更操作。

4）分级权限控制。建行云平台对不同风险级别的操作采取不同的权限控制策略，对于高风险的操作都进行二次授权控制，并且对所有的变更操作都记录操作日志，方便进行操作审计。

10.3.2 应用发布管理设计及实现

应用自动化发布是实践 DevOps 的关键环节之一，应用发布管理为应用程序、应用配置和应用数据提供了自动化操作接口。建行云平台通过工作流引擎模块，实现部署单元各原子操作的组合，并完成应用版本发布、应用日常运维的相关操作。各应用系统以部署单元为单位，定义和执行各系统的自动化运维操作。

建行云平台将应用发布和日常运维涉及的操作步骤分解为多个独立的原子操作，通过原子操作的编排和组合，实现灵活的自动化操作。通过对变更步骤的总结与分析，建行云平台把应用发布操作分为两类：一类为标准操作，另一类为用户个性化的自定义操作。

（1）标准原子操作　云平台提供了 8 种原子操作。

1）启动或停止应用服务：完成应用的启动或停止操作。由云平台统一规定启停脚本路径和名称，各个应用系统负责按照统一的规范编写脚本内容。

2）标准备份：针对变更前的应用版本文件进行本地备份。由云平台提供备份脚本，应用系统提供备份的源路径、目标路径、排除文件。

3）定制备份：如标准备份无法满足备份需要，应由应用系统脚本按照统一的规范自行编写。

4）标准应用分发：由开发人员按照云平台的打包规范，将测试完成的程序代码构建成版本包。云平台提供标准脚本，用户提供分发的应用版本名称、文件路径等，云平台自动从版本服务器获取应用版本，实现版本包的文件下发、解压、权限设置、更新和校验。

5）定制应用分发：如果标准应用分发无法满足需要，应用系统可以编写分发脚本实现个性化的应用分发需求，云平台负责分发版本包和执行用户定制的脚本。

6）系统文件分发：实现操作系统级文件的分发，考虑操作风险，目前仅支持部分文件的分发。

7）标准回退：针对标准备份文件的回退，采用标准回退时由云平台提供回退脚本，按照备份时指定的源和目标进行回退。

8）定制回退：针对定制备份文件的回退，回退时云平台执行由应用管理员开发的回退脚本。

（2）自定义操作　当标准原子操作无法满足需要时，应用系统管理员或者开发人员可以根据云平台统一的规范自定义操作，实现个性化的需求。

自动化的前提是标准化和规范化，为了保证自动化运维的顺利推进，建行云平台制定了一系列的规范，对自动化涉及的脚本编写和版本文件打包等操作进行了标准化和规范化。

10.4　运维大数据分析

云计算和大数据是一个硬币的两面，云计算是大数据成长的驱动力，而另一方面，由于数据越来越多、越来越实时，这就需要云计算进行处理，两者相辅相成。同时，云计算环境的复杂度越来越高，新技术应用越来越多，这就为数据中心运维带来了更大的挑战，那么利用大数据技术分析运维环境的各项指标，就可以提供准确、实时、有效的运行分析结果，为不同维度的管理、运维人员提供决策依据，提高运维效率和服务

质量。

10.4.1 大数据技术

10.4.1.1 数据采集技术

1. Flume

Flume 是一个高可用的、高可靠的、分布式的海量日志采集、聚合和传输的工具，最新版本为 1.5.2，Flume 支持在日志系统中定制各类数据发送方，用于收集数据；同时，Flume 提供对数据进行简单处理，并写到各种数据接受方的能力，Flume 可以从 console、Thrift、text、tail、syslog、exec 等数据源上收集数据，并可以将数据写入本地文件系统、HDFS、ElasticSearch 等目标。Flume 架构示意图如图 10-15 所示。

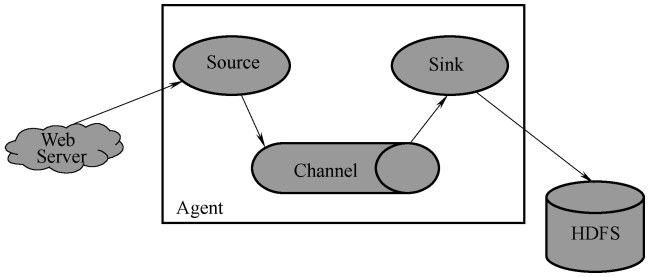


图 10-15 Flume 架构示意图

注：Web Server：Web 服务器；Agent：一个独立的 Flume 进程，包含组件 Source、Channel 和 Sink；Source：数据；Channel：中转数据的一个临时存储，保存有 Source 组件传递过来的数据；Sink：从 Channel 中读取并移除数据，并将数据传递到管道中的下一个处理节点；HDFS：分布式文件系统，用于存储数据。

2. Logstash

Logstash 是一个事件与日志管理工具。用户可以用它来收集日志并加以解析，还可以把它们存储起来以备后续使用，比如检索。Logstash 通常与 ElasticSearch、Kibana（ELK）一起使用。

10.4.1.2 全文检索技术

1. Solr

Solr 是一个基于 Lucene 的全文检索平台。它提供了基于 Http 的 Rest-like 操作接口，具有高可扩展的开放架构，提供了强大的 WEB 管理界面。有多种客户端，即 Ruby、PHP、Java、Python、.NET、Perl 和 JavaScript。

Solr 在版本 4.0 以后发布了 SolrCloud。SolrCloud 是基于 Solr 和 ZooKeeper 的一套分布式搜索方案，具备了集中式的配置信息、自动容错、近实时搜索、查询时自动负载均衡、自动分发的索引和索引分片、事务日志等特性。

2. ElasticSearch

ElasticSearch（以下简称 ES）也是一个基于 Lucene 的实时检索与分析引擎，虽然诞生较晚，但是具有天生的分布式架构以及良好的可扩展性，提供了全文搜索能力、多语言支持、专门的查询语言、支持地理位置服务、基于上下文的搜索建议、自动完成以及搜索片段的能力。近年来，ES 社区发展迅猛，已经发布了重大更新版本 2.0。其生态系统也日臻完善，不但可以和其子项目 Logstash、Kibana 无缝集成，也支持诸多的客户端及接口。

ES 发布的可以通过其插件 Elasticsearch for Apache Hadoop 与 Hadoop 深度集成。Hadoop 在设计上就是一个分布式的，面向批处理的平台，用以处理大数据集。虽然它是一个非常强大的工具，但它的批处理本质意味着在处理结果时需要花费一定时间。此外，用户必须重新为各种操作编写代码，用户门槛较高。Elasticsearch for Apache Hadoop 为 Hadoop 提供了直接的整合功能，因此用户使用起来没有任何障碍，支持 Map/Reduce、Cascading、Pig、Spark 和 HIVE。这样可以以 HDFS 一样的方式访问 Elasticsearch 的数据了。

3. Splunk

Splunk 是一个 IT 数据、日志分析软件，支持的平台包含 Windows、Linux、Solaris、FreeBSD、AIX、MacOS 和 HP-UX。Splunk 可以支持任何 IT 设备（服务器、网络设备、应用程序、数据库等）所产生的日志，其对日志进行处理的方式是进行高效索引之后让管理员可以对日志中出现的各种情况进行搜索，并且通过非常好的图形化方式展现出来。

Splunk 是一款对 IT 管理员非常有用、非常专业的工具。也正因为如此，与之前介绍的产品不同，Splunk 是一个商业软件，但是其提供了可以自由使用的 Splunk 测试版且可以免费下载，免费版每天最大新增数据量为 500MB，在使用免费版时，如果在 30 天之内，有 7 天的索引数据量超过 500MB，那么就不可以再搜索了，而如果需要海量授权及更多的功能，比如分散式搜寻（Distributed Search）、排程告警（Schedule Alert）、权限（Access Control）等功能，则需要购买企业版。

Splunk 的四大应用是：运维 IT 数据整合、IT 安全信息及数据的整合、应用程序 IT 数据整合和 IT 数据的法规遵从。Splunk 的六大功能是：Index（索引）、Search（搜索）、Alert（警报通知）、Report（报告）、Share（资源共享）和 Secure（安全功能）。

Splunk 可以实时对任何 App、服务器或网络设备的数据进行索引并提供搜索，这些数据可以是日志、配置文件、消息和告警等。利用 Splunk 可监控分布部署在多个数据中心的成千上万台服务器，可以管理 PaaS 云的基础设施，可监控云交付 SaaS 解决方案的性能，并可监控 SaaS 与托管混合型的数据中心。

10.4.1.3 数据分析技术

数据分析领域，既有开源的 R 语言与 Mahout，也有商用产品 SAS 和 SPSS。

1. R 语言

R 语言是一套完整的数据处理、计算和制图软件系统。其功能包括数据存储和处理系统；数组运算工具（其向量、矩阵运算方面功能尤其强大）；完整连贯的统计分析工

具；优秀的统计制图功能；简便而强大的编程语言：可操纵数据的输入和输出，可实现分支与循环，以及用户可自定义功能。

与其说 R 语言是一种统计软件，还不如说 R 语言是一种数学计算的环境，因为 R 语言并不是仅仅提供若干统计程序，使用者只需指定数据库和若干参数便可进行一个统计分析。R 语言的思想是：它可以提供一些集成的统计工具，但更多的是它提供了各种数学计算、统计计算的函数，从而使使用者能够灵活机动地进行数据分析，甚至创造出符合需要的新的统计计算方法。

2. Mahout

Mahout 是基于 Hadoop 的数据挖掘和机器学习的算法框架，Mahout 的重点同样是解决大数据计算的问题。Mahout 目前已支持的算法包括，协同过滤、推荐算法、聚类算法、分类算法、三层贝叶斯概率算法、朴素贝叶斯算法和随机森林算法。这些算法中大部分都是距离的算法，可以通过矩阵分解后，充分利用 MapReduce 的并行计算框架，高效地完成计算任务。

但是，Mahout 的很多数据挖掘算法，很难实现 MapReduce 并行化。Mahout 的现有模型都是通用模型，可直接应用到的项目中，计算结果只会比随机结果好一点点。Mahout 二次开发，要求有深厚的 Java 和 Hadoop 的技术基础，最好兼有“线性代数”、“概率统计”、“算法导论”等基础知识。因此，想玩转 Mahout 真的不是一件容易的事情。

R 语言同样提供了 Mahout 支持的大多数算法（除专有算法外），并且还支持大量的 Mahout 不支持的算法，其算法的增长速度比 Mahout 快得多，并且开发简单，参数配置灵活，对小型数据集运算速度非常快。

然而，Mahout 同样可以进行数据挖掘和机器学习，但是和 R 语言的擅长领域并不重合。因此，在适合的领域选择合适的技术，才能真正“保质保量”地开发软件。

3. SAS

SAS（Statistical Analysis System）是一个模块化、集成化的大型分析软件系统。它由数十个专用模块构成，功能包括数据访问、数据储存及管理、应用开发、图形处理、数据分析、报告编制、运筹学方法、计量经济学与预测等。

SAS 系统基本上可以分为四大部分：SAS 数据库部分，SAS 分析核心，SAS 开发呈现工具，以及 SAS 对分布处理模式的支持及其数据仓库设计。

4. R 与 SAS 之争

R 语言开源、免费，扩展包丰富，但是语言本身特性比较匮乏，可读性不高，处理数据量不大的研究时比较方便，而处理数据量很大时就得借助数据库。R 语言相对 SAS 的一个最大优势就是扩展包相当多，而且相当全面，对于大部分研究问题而言都不需要自己去实现底层的计算方法，但是扩展包比较零散，很多时候需要自己去淘金。

SAS 很庞大，但其主要优势在于入门简单，语言可读性强，语言思路比较符合统计分析的思路：大体上是数据步整理筛选数据，过程步做数据分析等。编写 SAS 代码的过程就是告诉 SAS 系统做什么，而不是怎么做，减少了很多编程细节上的麻烦。

10.4.2 大数据应用

10.4.2.1 运维中的大数据

在数据中心，大数据的整个生命周期包括采集、存储、分析、展示等环节，数据源主要是基础设施产生的机器数据。通常会在设备上安装采集代理，收集各类性能数据和日志，或者通过网络协议采集网络设备的数据，这些数据存储到大数据平台后，可以为日志分析、交易监控、性能管理等诸多领域应用。数据中心大数据平台架构如图 10-16 所示。

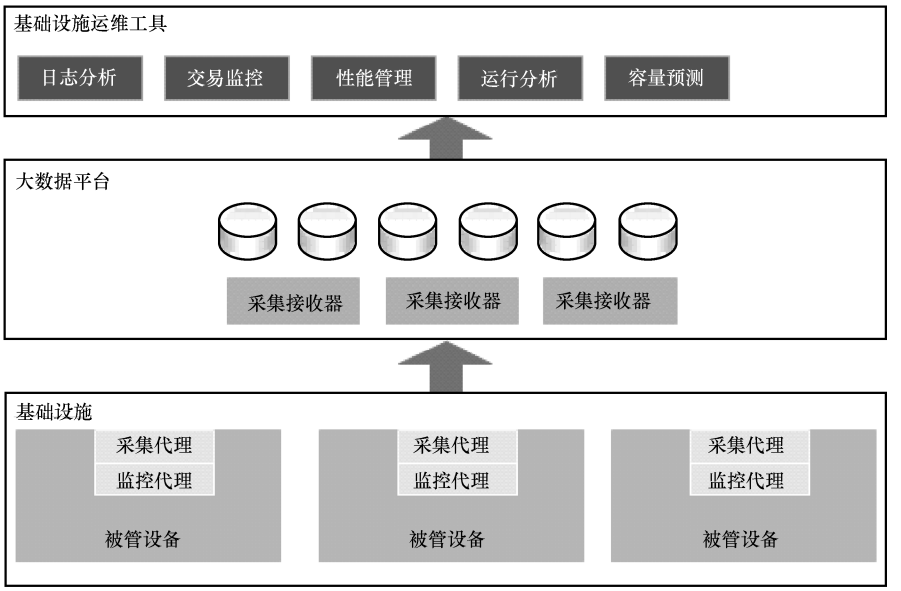


图 10-16 数据中心大数据平台架构

数据中心常用的大数据技术包括以下几个方面：

首先是大数据采集技术，采集代理必须能够高效地从庞大的基础设施中采集各种信息，包括日志、性能数据等，采集代理应采用分布式架构，在采集接收端出现容量或性能出现瓶颈时可以平滑地横向扩展，并能够高效、低延时地完成数据的初步处理和入库。

其次，面对海量的结构化数据和非结构化数据，传统的关系型数据库无法满足数据存储的需求，必须使用分布式的文件系统，提供高容量、高负载、高可用性的存储能力，HDFS 就是一个不错的选择，也可以使用一些应用级别的存储技术，比如分布式的全文索引，全文索引在面对基于时间的性能数据和日志这类非结构化数据时，提供了更好的存储和检索能力。

大量的数据采集完成后，要通过一定的数据分析工具，才能产生价值，由于较高的时效性要求，可以使用流式计算技术快速处理海量的性能数据和日志，生成告警事件并

通知到相应的人员；也可以使用复杂事件处理（CEP）技术，在各个事件之间进行关联分析，减少重复报警、过滤表象事件和定位根源事件，并确定故障的业务影响，为后续的故障处置提供决策依据。我们也可以使用一些机器学习算法，比如，使用回归分析来进行容量预测，使用聚类分析寻找故障发生的模式进行故障预测。

最后，我们要打通数据和人之间的壁垒，通过可视化技术在数据和人之间建立起桥梁，通过各类分析图表，尤其是非传统类型的图表，分析数据之间的关系，为性能管理、容量管理等日常工作提供决策支持。

10.4.2.2 日志采集与分析

运维人员分析系统问题的一个主要途径就是分析日志，原来一套系统最多也就十几台设备，一台一台地去登录，逐个日志的手工查看还可以忍受，但是在虚拟化、云计算的环境下，运维人员要面对成千上万台设备，原来手工的分析方法显然已经无法满足运维工作的需要，这就需要对基础设施的各类日志进行统一采集和集中存储，提供统一的查询控制台进行日志的检索，并结合文本分析工具进行各种统计分析，以快速定位问题原因。日志集中采集及分析平台逻辑架构如图 10-17 所示。

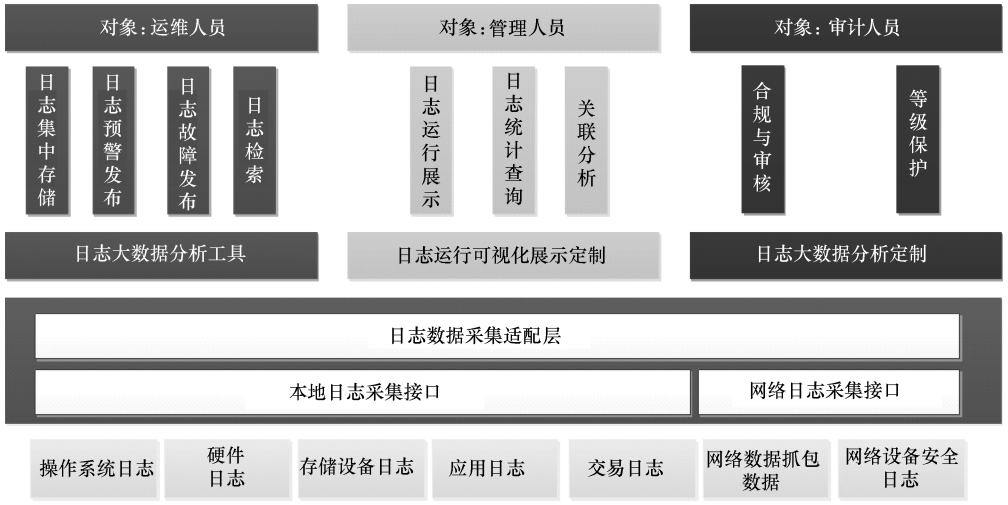


图 10-17 日志集中采集及分析平台逻辑架构

同时，采集到的日志可以使用规则引擎和流计算技术对日志进行监控，监测异常的关键字，对多个日志进行关联分析，通过模式分析发现潜在的系统异常，实现故障预测。比如，当某个极少出现的词在短时间内频繁出现时，可能意味着系统出现了不同寻常的行为，需要运维人员及时关注，这种行为可能是正常的，也可能是异常的，通过不断对分析模型进行调整，就能不断提高故障预测的准确性。

此外，日志分析还可以在入侵监测、司法协查、安全审计等领域发挥作用。

10.4.2.3 应用质量分析

传统的监控大多数集中在基础设施领域，比如操作系统、中间件、数据库等，对应用自身的监控大多仅限于进程和日志，缺少一种直接掌控系统业务运行情况的监控手

段。借助大数据分析，我们可以通过对交易日志的分析，提供面向交付的端到端的交易监控能力，可以监控包括流量、性能和可用性三方面的指标，比如交易量、交易成功率、交易响应时间等。

通常一笔交易要经过多个系统的处理才能完成，以往交易处理出现问题，整个交易线上的所有系统都跟着一起排查，费时费力。通过对多个交易日志的关联分析，就可以对一笔交易在各个系统的处理情况进行深入剖析，快速定位问题在哪个系统上。

通过对一笔交易的分析可以定位问题，对海量的交易进行分析就可以勾勒出整个企业交易线的拓扑结构，从而分析应用架构是否合理、交易线是否过长等问题，进而对交易线进行全局的管理和优化。

对交易监控进行延伸，就可以进行交易质量分析。通过对交易监控数据的分析，找出交易失败的分布特征，发现客户行为、员工操作、业务参数配置、营业时间限制等问题，引导客户、员工、项目组合理使用 IT 系统资源，降低无效交易比例，降低数据中心运营成本，提高客户体验。

通过交易质量分析，可以识别出长期不使用的交易，及时将这些无用交易进行注销，找到响应时间较长的交易，进行针对性的优化，识别成功率较低的交易，看看到底是客户的原因造成的交易失败还是由于 IT 系统造成的失败。如果是客户原因造成的，比如客户对系统不了解，对系统错误地使用，那就要让客服人员及时与客户联系，引导客户正确使用系统；如果是 IT 系统造成的失败，比如在日常批处理时段发生一些只能在联机时段处理的交易，那就要对系统进行优化，不要发送这些无效交易，避免对 IT 系统造成无谓的消耗。

交易质量管理工作需要企业多个部门协作，在流程上形成闭环。首先由数据中心通过应用监控平台发布生成交易质量报告，由架构管理部门对报告进行分析，提出优化建议并对开发部门下达整改任务，并对这些任务进行跟踪，开发部门完成整改后，需要由应用监控平台再次发布交易质量报告并验证优化的效果，由此循环往复，形成一个闭环，不断地提高系统的服务质量，从而提高客户体验。

10.4.2.4 性能管理

在私有云中环境下，资源众多，环境复杂，包括计算资源、存储资源、网络资源、物理层资源和虚拟层资源。因此，私有云环境下的性能管理要比传统的性能管理复杂得多。大数据技术可以应用于性能管理领域，以确保云环境资源的服务质量。数据中心通常会在各个专业领域实施一些性能管理工具，这些工具由不同的厂商提供，架构各异，提供的能力不一致，无法互联互通，并且会形成对厂商的依赖，替代成本较高，对于科技力量较强的数据中心，可以只使用这些产品基础的采集功能，降低对产品的依赖，通过自主研发的采集代理，将各种性能工具采集到大数据平台，进行集中的存储与分析，这些性能数据还可以为容量管理提供数据支持。

性能数据的采集应以实时采集为佳，一种为有代理方式，另一种为无代理方式。有代理方式一般在虚拟机或物理机中安装一个采集代理，通过接口或者文件采集的方式，在一定周期内轮流采集被管机的性能数据；无代理方式一般采用 Syslog、Snmp 协议或

者适配设备厂商的 API，以集中的采集点采集被管机的性能数据。对于那些无法实时采集的性能数据，可以采用批量采集方法，但这种方法由于时效性低（通常是 T + 1），因此丧失了对云环境实时性能分析的能力，不利于问题分析及定位。

性能分析通常使用图表方式进行，可交互式的图表以及可灵活配置的仪表盘可以大大提高性能分析的效率。另外，可以通过报表工具定期生成性能分析报告，进行周期性的分析。

10.4.2.5 容量评估及预测

容量评估主要是基于历史性能及容量数据，评估基础设施是否出现了容量瓶颈或容量热点，以便及时对应用进行调整或者采购扩容。在容量评估过程中，一般不使用平均值或最高值指标，因为平均值会被部分业务低谷时段的样本值拉低，而最高值会被业务高峰时段的少数高点拉高，都不能准确描述基础设施的资源使用情况。可是使用百分位数或者等价的指标来描述容量使用情况。百分位数是指如果将一组数据从小到大排序，并计算相应的累计百分位，则某一百分位所对应数据的值就称为这一百分位的百分位数。可表示为：一组 n 个观测值按数值大小排列，如处于 p% 位置的值称为第 p 百分位数。一般可以取 95% 百分位数或者 99% 百分位数作为容量评估的指标。

容量预测是结合各类性能管理的历史数据，建立业务量与基础设施资源使用情况的分析模型，通过回归分析、神经网络、卡方、决策树等模型进行基础设施资源使用率预测与业务量预测，评估系统容量，分析资源可能出现的瓶颈，为基础设施扩容与采购提供决策支持。

一般情况下的容量预测可以分为以下几个步骤：

1）找到与业务量相关性较高的指标，再对数据进行平滑处理，剔除离散度较高的异常数据。

2）建立容量分析模型，设置置信区间，按照 3 份数据预测 1 份数据的规律，计算预测值。

3）对预测结果进行解读，通常容量预测不仅要看预测值，还要考虑到业务推广、系统升级等模型之外的因素，单纯依赖预测结果，可能会产生较大的偏差。

10.4.3 数据可视化

我们常说，大数据是一座金矿，数据是 21 世纪的一种新型自然资源，这座金矿必须通过挖掘和消费才能产生价值，否则，数据放在库里，随着时间的流逝，价值就会急速降低。

可视化技术是使用大数据的有效手段，借助可视化技术，可以在人与数据之间搭建沟通的桥梁，从现阶段来讲，依靠人的视觉进行观察和分析，是任何机器学习算法无法替代的。

可视化不但要提供丰富的展现形式，也要提供参数化、模块化的定制能力，能够让

用户快速地定制所需要的仪表盘。最终用户不需要编码，甚至不需要精通大数据的技术细节，就可以享受大数据带来的便利，在降低了运维人员使用大数据的门槛后，每一个人都可以是大数据的最终用户，所有的运维人员都可以成为一名数据分析师。

10.5 监控智能处置体系

IT 系统的可用性、可维护性以及灾难可恢复性一般被视为最重要的 IT 建设指标。监控系统对于促进这三个指标水平的提高都可以发挥重要的作用。在传统监控体系中监测系统的可用性一直作为监控的主攻方向，其也是衡量监控系统是否合格的主要指标。随着云计算、大数据、可视化等技术的不断发展，自动化手段得到了前所未有的丰富，监控系统与自动化系统的界限已变得非常模糊，由于监控系统其本身在问题发现上所具备的先天优势，监控系统对于可用性处置、系统可维护性促进、灾难可恢复性评测的贡献也在显著地增强。由于所涉及的技术与管理环节众多，所以本节仅讨论监控智能处置的一些设想与运用。

（1）监控智能处置体系　先从监控体系说起，一般来说，一个基本的监控体系模型是由相关专业岗位人员、被监控对象、衡量对象各组件工作状态的监控指标、各领域监控所需数据采集工具、数据处理与分析手段、相关维护与处理工作流程等要素构成的。在此基础上针对智能处置，监控智能处置体系还扩展了时间维度（主要包括识别与处置的时间）、处置对象、案例、场景、处置方式、交互方式等要素。这里定义案例为基本的处置单位，主要包含处置的方法步骤，需要强调的是案例中并不包含处置对象、处置时间等信息，以保证案例的通用性。案例的形式并无一定之规，可以是一段文字，也可以是自动处置脚本甚至表现为一个手工触发的按钮等。除了案例，场景也是模型中一个重要的要素，场景是案例、报警对象、报警指标、报警级别、处理对象、报警时间窗口、处置时间窗口的集合，场景被定义为报警处置的最小单位，可简单地理解为案例的实例，场景中主要包含识别信息与处置信息两部分内容。

（2）监控智能处置　监控智能处置就是将监控智能处置体系中的要素串联起来，以场景识别为流程入口，借助规则引擎、工作流引擎、大数据、可视化等技术定位故障与影响，进而对事件进行自动、半自动或者人工处理，并对整个过程进行可视化展示的活动。

监控智能处置的基础是监控，监控需要涵盖 IT 的各个领域，监控的数据量越大、覆盖面越广场景识别的能力就越强，分析定位的准确性就越高，处置的效果才会更好。监控对象应包括应用系统整体构成的所有组成环节，而不仅仅是应用程序，还应包括其运行所依赖的存储环境、网络环境、服务器资源、各类系统软件、外部链路资源、维护操作任务、内部配置数据、系统间会话、业务数据交换任务等。

一套完善的监控智能处置系统，不仅包括对监控对象的监测、处理，同时也包括系统自身的自我完善机制。其能够将预警前移到故障发生之前，预测故障和容量危机，为

运维管理提供风险提示；可以发现系统中各个组件间的依赖关系，进而推导故障影响；能够总结场景与故障根源的对应关系，形成故障树，分析故障原因，并为业务部门提供决策支持；可以对软件版本、硬件型号进行质量分析，从而影响采购决策，促进运维质量，降低运维成本；能够统计分析系统中的监控指标与阈值，形成基准指标集，推进监控管理质量。

如何构建并运用一套有效的监控智能处置体系，为银行 IT 系统安全运行提供基础技术保障，是我们一直在探索的课题之一。总结之前研究与实现的一些实践经验，接下来将按照规划、体系、实现和应用四个不同的方面进行简要介绍。

10.5.1 规划设计

在此之前各个系统已具备了很强的监控能力，具备各种检测手段，可以检测开放系统、网络、大型机、环境、硬件、交易等领域，但是如何将这些监控所获得的信息纳入到监控智能处置体系中来，统一运用，并使智能处置系统可以不断地学习、完善，是在本节需要探讨的内容。接下来通过认知、识别、处理和进化四个维度进行阐述。

1. 认知

认知是指对监控智能处置体系所涉及的要素梳理、提炼、保存的过程，监控智能处置体系中最主要的要素是监控对象、监控指标、处置对象、时间、案例与方案。

1）监控对象的认知：监控智能处置体系中的监控对象是指能够对外提供服务的个体以及个体间的关系，对于监控对象来说，理论上其颗粒度应该越细越好，其颗粒度的粗细直接影响场景识别的敏感度。但另一方面颗粒度的每次扩展都会带来监控对象维护成本的显著增加，尤其是对象关系，其增加量非常巨大。只存储必要的数据属性，剔除无效数据，就变得非常重要。一般来说，对象属性分为两类：自然属性和管理属性。自然属性是指对象自身所具有的公理类型属性，此类属性不会因为管理体系的更替而产生变化，往往是可以通过技术手段获取的；管理属性主要是指由于职能分工为对象带来的附加属性，这类属性的变化频率更高、获取的难度更大，往往需要人工维护，一般来说，属性的维护成本较高，使用场景更多与安全、客户交互体验有关，如权限、视图分类、分组查询、报警通知等。

2）监控指标的认知：监控指标是描述监控对象“态”的要素，一般分为状态指标、性能指标和容量指标。状态指标是指可以精确定位监控对象当前“态”的指标，比如进程“启”与“停”（注：0% 和 100% 一般也认为是状态指标）；性能指标是衡量提供对外服务质量的指标，其指标与各服务的承诺有关，存在更大的容忍区间。性能指标是影响分析的主要考量指标。容量指标是指可以精确度量但无法独立定位监控对象当前“态”的指标，相对于其他两类指标，容量指标的判别标准与对象个体差异的关联更为紧密，容量指标是故障预警的主要考量指标。与传统监控指标不同的是，监控智能处置体系中的监控指标也具有和监控对象类似的关系要素，指标关系包括必然影响关系

和可能影响关系，一般来说状态指标和性能指标多产生必然影响关系，容量指标产生可能影响关系。

3）处置对象的认知：顾名思义，处置对象就是需要对“谁”进行智能处置。处置对象的管理与监控对象类似，甚至可以重复利用。但需要注意的是，在同一事件当中报警对象和处置对象有时并不相同。随着处置逻辑复杂度的提高，处置对象与监控对象的关系将从自包含关系向多对多关系转换。

4）时间的认知：时间是监控智能处置体系中重要的维度指标，时间主要包括两部分，即报警时间和处置时间。某些情况下监控对象尽管自身的状态没有改变，但由于时间的改变，监控目标和处置方式都会发生改变。比如性能指标为了适应这种变化有时会采用基线阈值告警的方式。时间还有分段的属性，单位时间内的操作往往被要求不能重复。另外，时间还有顺序的属性，操作是有顺序的，判定什么时候操作也是有顺序的，但是由于监控方式的不同原本先发生的事件可能由于是轮询采样比后发生的事件到达监控系统更晚，这些都是需要在监控智能处置中考虑的。

5）案例与方案的认知：如前所述案例就是处置执行的步骤，方案是案例的实例，所以在认知时这两个要素往往需要一起考虑。它们的认知需要通过相关专家梳理获得。梳理获得有很多种方式，这里介绍两种，即指标梳理法和事件回顾法。这两种方式在梳理时都需要考虑触发条件和处理方式两个部分的内容，指标梳理法是以当前监控系统中存在的监控指标作为基础，梳理每一个指标在触发告警后的处置方法，这种梳理方法适合处理方法明确的、触发条件清晰的监控指标。状态指标的梳理往往采用这种方式，其处理方法一般都是以对“态”的恢复为目的，同时以“态”的变化作为触发条件。性能指标、容量指标产生报警的成因往往比较复杂，相同报警在不同时间、环境之下其根本原因可能也会不同，我们可以通过故障树的方式来对报警成因以及对应的处理方法进行管理。故障树的形成方式有很多种，这里介绍的是通过对事件进行回顾的方式。所谓事件回顾就是对历史上发生过的事件进行汇总分析，主要分析报警对象、报警指标、处理方法和根本原因。以报警指标作为故障树的根结点，先梳理报警原因，再梳理确认方法，之后是不同原因的处理步骤（注：报警对象如果会决定报警原因则包含在报警原因中一同梳理）。

2. 识别

识别是对触发条件的确认，包括事件的获知与确认两个主要阶段。

事件的获知一般有多种渠道，比如电话报障、定期巡检、监控报警等。由于电话报障在时效上不可控，所以只能作为事件获知的辅助手段。定期巡检是监控的有效补充，其特点是监控覆盖面广、时效性低，可以作为事件获知的主动手段。监控报警是由监控系统自身产生的，具有敏感度高、内容标准化程度高的特点，更有利于根源定位与后续处置，所以作为监控智能处置体系中首推的发现手段。如何逐步降低报障在事件获知中的比例是本阶段的重要工作。

事件确认主要有三部分工作：确认事件是否真实发生，确认事件是否已经恢复，以及确认事件所匹配到的方案。

1）由于事件获知的渠道不同，每个渠道由于主体的不同，其判断方法、认知范围都不尽相同，所以当事件获知后首先要做的是确认事件是否真实发生。确认的方法可以分为主动方式和被动方式两种，主动方式一般采用巡检、健康检查、日志查询、性能容量报表分析等方法；被动方式是在监控系统中查询报警事件，以是否发生过报警事件作为事件是否真实发生的判别标准。

2）确认事件是否已经恢复一般涉及两个阶段，即事件发生阶段和事件处置完成阶段。由于时序问题，性能、容量事件有时会在事件发生阶段自我消除，所以事件处置前的确认操作是非常必要的；另一方面，处置完成之后通过对事件是否恢复的确认来衡量处置的效果也是监控智能处置流程中的重要步骤。

3）当确认事件确实真的发生了，同时尚未得到任何缓解后，使用何种处置方案就变得非常重要。方案的确认主要是通过对监控对象、监控指标、发生时间的综合判断来实现的。

3. 处理

处理是分析与处置的广义描述，包含事件方案匹配后到事件恢复前的所有操作。分析主要包括根源分析与业务影响分析两部分，根源分析可以通过监控对象、监控指标的关联关系以及故障树作为分析依据，业务影响分析可以根据交易监控的性能数据辅以监控对象、监控指标的关联关系获取。对于传统处置理论来讲，处置对象的颗粒度越细越好，但是颗粒度的细化必然带来维护成本的巨大压力，同时由于数据量的提高必然会对需要在短时间内完成事件处置的内在要求造成挑战。随着云技术、硬件技术的不断发展，对于处置对象颗粒度的要求也有所转变，监控智能处置体系对于处置颗粒度的定义是在业务允许的前提下，处置的颗粒度越粗越好。其实很好理解，当计算机硬件价格堪比黄金的年代，硬件故障经常通过更换电容来解决，之后发展到整个主板的更换，现在可以进行整机更换，未来可能会发展到整个集装箱式机房的替换。同样，虚拟机的替换也在技术方面为应急提供了更加便捷的手段。基于以上原因监控智能处置体系要求监控对象的颗粒度越细越好，而处置对象的颗粒度越粗越好。除了监控对象，故障树的复杂度对于处置管理也有很大的影响。这里提出一个故障树线性化的设想，也就是说当故障树成为一条具有多个节点的直线时，故障树的管理是最简单的。那么如何将故障树线性化呢？主要有两种手段，即扩大处置范围、拆分细化监控对象与监控指标。

处置的方式一般包括自动、半自动和人工方式。半自动方式又包括人输入机器判断、机器输出人触发等方式。另外，需要特别提出的是，事件不一定会命中系统已有的知识，在这种情况下启动系统之外的应急流程也属于监控智能处置体系需要考虑的范畴。

4. 进化

事件处置完毕后，系统应进入新知识的转化阶段。转化阶段分为两类：定期转换与实时转换。定期转换包括新对象的梳理、新指标的梳理、通用与专项方案的梳理等；实时转换包括对象关系、指标关系、基线阈值、高阶阈值等数据的实时更新。

10.5.2 体系构建

传统监控体系的构成，是以有效的人员岗位分工与合作为主体，辅助以适用的工具、有效的知识及规范，形成发现问题、改进问题的不断自我完善机制。主要的岗位分工及工作过程要点如下：

1）一线值守人员按操作规程及时处理各类监测数据，并详细跟踪、记录处理过程。

2）监控管理人员跟踪并分析各类报警数据的有效性，发现问题并设计优化策略。

3）监控技术人员按需求开发各类监测工具及手段，维护监控资产信息，并确保监控工具及功能能够被有效地交付到一线值守人员、二线技术人员等岗位。

4）二线技术人员及时响应并处置报警，并根据实际使用效果，及时提出监控优化需求，同时针对各类运行问题与隐患，不断完善相关技术规范，避免同类问题的反复发生。

5）运维质量管理人员，应采集各类监测与运行数据，形成常规分析机制，及时通报当前关键问题、工作优化建议，推动整个体系的不断优化，消除各类技术与工作隐患。

在此基础上监控智能处置体系中增加监控智能处置管理岗负责评估系统中缺失的告警对象与指标，并针对方案进行定期分析，查找冲突方案与缺失方案；归纳二线技术人员处置报警的方式方法，将应急处置的短暂时间分散到日常梳理工作当中；注重各岗位的信息沟通，强调将各岗位所关心问题的标准化，并利用工具采用实时推送方式，将信息呈现给相关岗位，最大限度减少操作无序带来的时间损耗；规范操作流程，尤其是分析与处置流程，避免事件处理停滞或者影响范围意外扩大。

10.5.3 实现技术

有效的监控智能处置体系，始终离不开相关技术平台的支撑，而要搭建有效的技术平台，需要解决如下几个技术层面的关键问题。

监控智能处置整体技术架构应包括数据采集层、专业领域监控工具处理层、统一事件管理层、数据分析层、事件处置层、展现与运用层。支持监控智能处置的完成处理流程，满足控制整体生产环境监控部署，满足从监测数据中发现问题、处置问题的技术需求。监控智能处置流程如图 10-18 所示。

1）数据采集：对于生产环境中各类运行数据的采集，首先要尽量规避对生产系统的直接影响（一般规定监控系统占用被监控系统的资源使用率小于 3%），其次是保证数据的实时性与准确性，最后是对于原始监测消息的准确加工与判断，最终形成有效的监控采集数据。

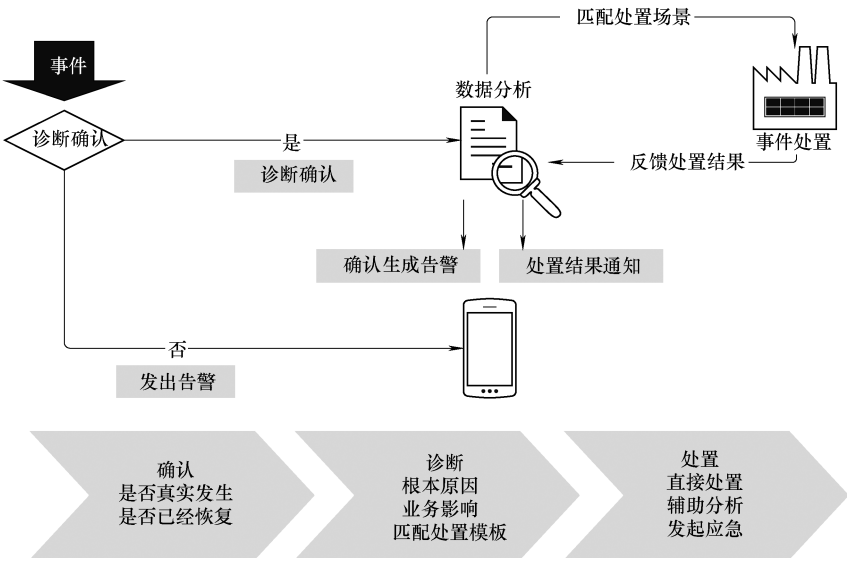


图 10-18　监控智能处置流程

2）专业领域监控工具：由于各领域的特点，采集方式和计算方式差异很大，定义领域监控工具负责各自领域采集数据的加工，其负责事件生成前和生成过程中的规则（包括计算公式）管理工作。

3）统一事件管理：负责对各领域事件生成后的规则处理，包括事件标准化、过滤、压缩、压制、通知、工单、关联处理等。一般采用规则引擎的方式处理事件。

4）数据分析：当获取到监控报警后，首先要解决的是确认其有效性，即是否真的发生了生产故障，其次是针对已发生的生产故障，按照对象和指标关系进行定位分析，并联合故障树进行故障诊断，并匹配故障处置场景。此外，数据分析还负责对历史事件的聚类分析，更新对象、指标关系，以及硬件、软件版本的质量、计费分析等。数据分析主要解决的是保证监控报警的有效与准确，为后续处置提供精确导航以及自学习的问题。

5）事件处置：当故障处置场景匹配后，按照处置模板调取各类服务，按照预设条件完成模板中的处置逻辑。

6）有效展现与运用：对于有效的监控报警数据、处置结果与过程，应按需展现与运用，分别针对操作人员、技术人员、管理人员的不同工作目标，提供不同的视图与界面。

10.5.4　应用场景

有效应用监控智能处置体系，可建设并不断维护其良好运转。在数据中心运维管理工作之中，监控智能处置体系可以在以下方面发挥作用：

1）运行事件管理前移：通过不断完善智能处置体系，在事件发生的征兆阶段发现

相关线索，并建立长效工作机制，跟踪报警的处理结果，不断优化与改进预测策略，通过对策略的不断演练，提升监控预警能力，将运行事件的管理与控制前移到预警阶段，以减小故障对于生产的影响。

2）智能辅助分析：由于生产环境的技术复杂性，一定会发生成因复杂的故障，并很难进行根源分析，在这种情况下就需要人工的介入。通过研究发现，90% 以上的专家分析都离不开长期历史图表、故障时刻数值、错误日志以及常规命令，系统可以通过规则的不断完善，在故障现象发生时，按照技术逻辑与经验，抓取推送相关数据，为技术人员故障分析提供详细资料，节约宝贵的应急时间。

3）监控基准指标管理：对于管理了大量设备与系统的数据中心来说，将监控指标标准化是非常重要的课题。但现实往往是理论与实际脱节，通用的指标和阈值自生效之日起就开始不断地被改来改去。可以通过对系统中指标和阈值的归类分析，动态确认和调整基准指标，监控纳管更加科学有效。

4）无效报警事件分析：对历史报警事件进行分析，查找监控对象、监控指标相同的重复发生事件，分析其处置模式、处置效果、故障持续时间与正常时间的比率关系、报警通知成本等，查找疑似无效报警事件，交由相关管理员分析解决，降低监控系统虚警率。

5）产品质量分析：根据产品厂商、版本等指标分类分析历史事件，绘制产品的质量基线。同时更重要的是汇总其关联事件，进一步确定产品质量对运维和业务的影响，为采购、追责等工作提供有效依据。

6）系统运行容量规划：目前生产系统的构成环节多、关联性广，因此对于生产系统运行容量的预估与管理，需要实时采集各技术层面数据，并能够进行有效整合，发现各个环节运行数据之间的变化联动性、放大系数等规律，然后根据业务请求的变化情况，去评估系统整体容量变化规律、系统容量瓶颈等关注点。

7）客户交易动态数据分析：在运行监测体系工作过程中，能够通过海量非结构化日志采集、网络报文镜像分析等手段，获取客户交易行为的动态数据，一方面能够及时发现客户交易过程中的各类故障及异常现象，另一方面也能够分析客户的交易趋势及喜好等，为业务流程优化、应用系统功能优化等提供分析数据，为数据中心从运维中心向运营中心发展提供技术基础。

10.6 私有云实施收益

建行私有云在经过新一代 1 期、新一代 2.1 期以及南湖搬迁几次大规模建设后，已初具规模，形成北京、武汉两地部署，未来在稻香湖建成后，将真正形成两地三中心部署模式。

目前，北京、武汉两个数据中心已建立 30 个资源池，近 400 个应用系统，新、老一代物理服务器数量超过 4000 台，虚拟机数量超过 5000 个，管理的操作系统总数也超

过 9000 个，已形成覆盖 Web、AP、DB 三层功能结构，Linux、AIX、HP Unix、Windows 等多种平台，X86、小型机多种物理服务器类型的基础设施资源池，提供了 38 种云服务、119 个套餐、自动化应用发布、平台参数管理等多种运维自动化手段，为老一代、新一代提供了高效、快捷的运维模式。

10.7 私有云实施难点和建议

通常商业银行采用成熟、可靠的传统 IT 技术路线，在产品选择上往往采用通用的信息技术和商业软件，在信息技术实施、支持和保障上很大程度上依赖信息技术供应商，存在技术标准不统一、新技术应用和技术创新缓慢、投入产出低等问题。而云计算则代表了一种采取互联网思维的全新技术路线，其核心思想是在低成本、标准化的硬件和开放的开源软件的基础上，通过分布式系统实现系统处理能力的无限扩展，并借助合适的应用架构弥补基础软硬件的不足，满足其可靠性的要求。

对于银行而言，云计算的本质是实现“去 IT 化”，即通过科技创新，实现 IT 和业务融合，完美快速交付 IT 服务，使 IT 部门从成本中心、服务提供者的角色，转变为企业内各部门业务发展的战略伙伴，实现科技引领业务。因此，银行数据中心的“云化”，重点不是关注技术概念，也不是单纯思考如何利用技术重建或翻新数据中心，而是根据自身业务发展现状和未来规划，逐步实现技术、人员、资源和过程四个核心要素的有机融合和持续改进。换句话说，数据中心的“云化”就是从技术转向服务的过程。因此，云计算将对数据中心运营服务管理能力产生巨大影响，也对传统 IT 运营团队提出了更高的要求，要求具备更新、更全面、更高层次的技术、运营、管理和服务等综合能力。

同时，云计算对银行的数据中心运维模式也将产生深远的影响。以前烟囱式的系统部署和运维分工模式必然被打破，系统整合与一体化运维将是未来的趋势。同时，标准化的系统建设、自动化工具将在银行业迅速普及，对银行的运维工作和安全生产将产生深远的影响和革命性的变革。

1）复杂的系统越来越依赖工具的运维。

2）运维的组织架构将产生变化，一体化的运维需要一体化的组织架构。

3）当前的银行流程和制度是根据人工运维的模式制定的，随着工具的普及，流程将会进行革新和重新制定，才不会制约企业的发展。

4）传统 IT 的运维人员将需要更新。实施云计算后，运维的核心是自动化工具系统，以及支撑运维工具的专家和系统开发人员，DevOps 的理论会进一步普及，传统运维人员需要进一步走向业务。

安全是云计算实施遇到的另外一个难题。云计算包含大量新的开源软件，商业银行的 IT 架构还缺乏对此进行维护的经验，将会引入新的安全问题。同时，随着云计算和 SDDC（软件定义数据中心）的发展，硬件、软件的边界越来越模糊，很多传统的安全

分层和安全设计已经不再适用，比如很多公有云就取消了传统的防火墙，改之以软件的方式实现，防止防火墙成为瓶颈。另外，向云计算转型期的系统安全需要特别注意。云计算描绘的前景很好。但传统商业银行的 IT 架构向云计算的转型也不能一蹴而就，要根据自身的技术实力统筹规划，做好转型期的风险防范工作，保障金融系统的稳定运行。

基于以上考虑，对商业银行的数据中心云化，有如下建议：

1）基于安全性考虑，私有云是云计算较好的实施方向和切入点。建行基础设施私有云研究与应用取得的突破，积累了大型商业银行在生产系统实施云计算的经验，并通过私有云技术有效解决了传统数据中心面临的资源管理复杂、运维操作风险高、服务响应慢等难题，节约了成本，增强了业务敏捷性。随着深入应用云计算技术，未来银行业可能形成内部私有云和外部行业公有云相结合的混合云模式。

2）私有云实施的前提是做好标准化和规范化工作。不同企业之间存在差异，云计算并不能形成一套可以直接使用外购的通用解决方案，因此企业自主可控能力建设非常重要。建行在私有云实施过程中，自主完成了云计算底层最核心的资源池和云服务的业务逻辑设计，将底层独立的服务器资源池、存储资源和网络资源池封装成一体化的资源向外提供，并根据业务特性细分成不同类型的服务池，在成本和可靠性上取得平衡，形成了基础设施统一的标准规范。

3）实践证明，大型商业银行根据自身需求，通过自主研发、量身定制云计算标准、云计算规范和云管理平台，实现传统数据中心“云化”管理切实可行。并且，综合了金融行业稳定性和互联网行业敏捷性特点，自主研发方案在技术先进性、业务适用性、自主可控性上比直接外购商业软件具有更明显的优势，与互联网行业的“云”相比也有独到之处，会对银行业信息技术国产化进程发挥积极促进作用。保持一支长期稳定的、由行内人员作为核心成员的实施团队，是自主研发成败的关键所在。

4）云计算实施成败和整体效果由整个企业 IT 战略决定，不仅仅取决于技术因素和实施团队。需要从战略、规划、架构、开发、运维多个领域进行统筹规划，提升部门间的沟通和协作水平，促进应用架构与基础设施架构融合，设计出更合理的应用架构，提升应用研发速度和用户体验。使业务人员和最终用户切身感知到效果才是真正的云计算。

5）相对于互联网行业，银行业云计算数据中心建设还处于起步阶段，经验和方法还没有形成完整的最佳实践；同时，云计算技术的核心是理念、流程、技术的深度融合，技术本身没有较高的壁垒。当前云计算还在快速发展中，这给银行业实现后发先至、弯道超车、打造先进云数据中心的跨越式发展提供了良好机遇。

第 11 章
中国邮政储蓄银行开发测试云建设

11.1 开发测试云建设背景

2014 年，工业和信息化部针对云计算的“十三五”规划已经启动，推动云计算产业链的快速发展。与此同时，国家发展改革委、财政部、工信部等部委进一步组织实施 2014 年云计算工程，将专项重点支持云计算服务平台建设、基于云计算平台的大数据服务、云计算和大数据解决方案及推广项目 3 个领域。云计算“十三五”规划的进一步落地，未来信息设备国产化的步伐有望进一步加快，云计算行业有望迎来爆发式增长。

为了赶上业务高速发展的要求，在商业竞争中获得业务优势，中国邮政储蓄银行（以下简称邮储银行）应用系统的开发量激增，开发任务也非常繁重，并行建设的应用系统年平均在 100 个以上。为了解决开发测试的效率问题，提高开发应用系统的可靠性和稳定性，构建良好的开发测试环境已成为提高应用系统开发效率及质量、加快系统上线速度、提升业务竞争力的一个重要因素。因此，建设符合银行业需要的开发测试云环境，替代传统应用系统的开发、测试模式就成为一种业务发展的必然要求。

开发测试云，是基础架构云在开发测试环境下的一种典型应用。通过建立开发测试云，解决开发测试环境下，环境部署和配置效率低下、管理难度较大的问题；实现开发测试环境下，按需驱动的资源调度；实现自动化的部署和配置硬件、操作系统、中间件和应用；实现实时地追踪、监控系统资源使用状况；实现虚拟化管理容量和镜像管理。从根本上降低人工运维成本，降低潜在的配置管理错误，提高开发测试环境下的生产效率。

11.2 银行开发测试环境现状分析

11.2.1 硬件及机房现状

目前邮储银行系统开发测试环境设备比较多，类型多样，由多台 PC 服务器、多台小型机、数据存储和网络设备等共同组成。设备利用率不高，测试环境设备管理依旧采用纸面单据定时巡检的方式，依靠大量的 Excel 表格来更新维护资产信息，管理人员劳动强度非常大，但仍无法做到实时、准确、主动式管理，对 IT 基础设施的配置和资源使用情况缺乏了解。造成这一现状的原因是设备资产的多样性、分布范围的广泛性和关联关系的复杂性。邮储银行开发测试环境所使用服务器的平均 CPU 使用率往往不到 10%，宽带利用率不到 15%。资源的浪费比较严重，导致了投资回报率的降低。我行的开发测试环境分布在不同的中心，不同中心之间网络带宽有限，环境中存在着诸多运行在不同品牌和不同操作系统的应用系统，这些应用系统种类繁杂、部署分散。尤其对于其中 X86 架构的服务器，大多数采用一对多的应用部署模式，即为一个应用部署一台 X86 服务器，同时准备一台备用机以保证突发事件发生时，能够比较快速地恢复应用，由此导致银行内部有着数量众多的 X86 服务器。同时，测试环境中其他厂商的小型机、服务器也是类型众多，比如 Power 系列服务器、各种刀片式服务器、小型机等。这种情况会造成以下问题：

（1）硬件采购成本高居不下　采购设备数量很多，导致采购成本较高，无法有效降下来。

（2）运营和维护成本高　面对数量庞大、复杂的服务器资源，运维人员要逐一管理各类服务器，运营、维护的工作量、强度非常大，造成人力成本的上升。

（3）电力及空间资源浪费　由于存在数量众多的各类服务器，数据中心内部机柜空间、电量消耗、温度控制等资源都存在相当程度的浪费。

（4）服务器部署效率低下　由于业务服务器数量众多，难以管理，大大降低了新业务应用测试加载时间，无形中导致了不同程度的损失。

（5）维护和变更管理周期长　如果有些硬件发生故障，会导致业务测试从几小时到几天的中断，同时在维护和变更管理时也可能需要维持数小时的时间，同样影响了业务的运行。

（6）系统兼容性差　由于众多不同品牌、不同型号的服务器共存，在系统和应用迁移时需要部署兼容系统的工作。这样的迁移通常非常困难，无疑也导致了一些业务无法正常使用。

（7）资源利用率低　X86 平台的应用，平均硬件资源利用率较低，大量的硬件资源被浪费，长期不能发挥应用的性能。

11.2.2 操作系统及应用系统现状

测试环境构成比较复杂，操作系统有 Unix、Windows、Linux、AIX 等，中间件及数据库种类较多，基础环境软件版本组合情况较多。

邮储银行每年有近百项工程并行开发、测试，参与开发测试的人员规模上千，开发测试环境资源往往是比较紧张的。每个应用系统所需要的环境至少要包括：开发测试环境、验证测试环境和联调测试环境。每套环境都需要相应的基础应用系统，一些开发项目没有独立的开发测试环境，只能与其他开发项目共用环境，各种软件、不同版本交织在一起，导致应用频繁在不同机器上进行迁移，很容易出现各种错误，延误项目进度。相对于生产环境，开发测试环境具有系统架构复杂，应用负载不均、资源变更频繁等特点。

11.2.3 人员及管理现状

开发测试环境管理人员承担着整个银行项目开发测试环境的系统运维工作，但用于维护的人员数量不足。开发测试环境日常管理工作一般包括：搭建或回收环境、释放或修改物理资源、监控应用系统的故障、统计系统资源利用率等。这些传统的管理方式，基本上都需要靠手工方式来完成，管理效率低下，不堪重负，也容易出错，影响软件交付时间。开发测试资源申请周期一般都比较长。一个半年左右的开发测试项目需要大约 1 个月的时间花费在申请资源、审批资源、归还资源等事务性、重复性的工作上。这些工作无疑拖延了开发测试的进度。

另一方面，近几年 IT 资源云化趋势明显加快，云环境和传统物理机环境的资产管理最大的区别是，云环境资源供给更加弹性化、动态化，绝大多数业务系统以虚拟机的形式运行，而虚拟机的资源规格可以动态调整以及在不同物理机之间动态迁移，如何实时准确地体现虚拟机和物理机的映射关系以及虚拟机规格是传统的资产管理所无法做到的，在这种背景下如何有效管理和维护成倍增加的虚拟设备资产成为一个重大的挑战。

11.3 基于云技术的开发测试环境探索

11.3.1 新一代开发测试环境探索

随着 IT 建设规模不断扩张，基于对虚拟化技术的理解，邮储银行计划建立了新一代云应用，首先从软件开发测试环境入手，通过整合不同架构的计算资源，提高 IT 管理全过程的风险管控能力，实现人员、设备、服务的精细化管理。新一代开发测试环境

重点考虑了以下问题：一是控制 IT 投资成本，包括各类软硬件、机房设备的投资，实现更加精细化的 IT 成本控制；二是加速项目开发部署和业务需求响应速度，缩短硬件部署周期，进而缩短项目整体实施周期；三是持续加强开发环境安全管理，限制外部人员对移动设备的使用，防范数据信息泄露的风险。

通过深入细致的调研与评估，摸索出一条符合自身情况的云计算建设之路，即以虚拟化技术为基础，综合运用多种软硬件技术，对传统开发环境进行重构，按分步走的方式打造安全灵活的开发测试环境。

循序渐进地对服务器进行了 P2V（Physical to Virtual，物理机向虚拟机迁移），将部分项目的开发测试工作放在虚拟化环境下进行。采用以虚拟化技术为核心的解决方案，大大缩短了系统环境的部署周期，实现了基础平台对敏捷开发的支持，有效推进了项目进度。同时，各类 IT 投资成本（包括硬件成本、采购周期的时间成本、环境部署的人工成本等）得到有效控制。

将虚拟桌面与终端配合应用，通过采用桌面虚拟化技术，有效解决了外部流动人员自行携带便携式计算机工作模式带来的安全问题；统一监控管理数据、源码、文档等信息，降低了信息安全风险；综合运用磁盘管理、网络管理、终端管理、信息加密等各类软硬件技术，对传统开发环境进行重构，不断完善管理制度和流程，实施开发测试环境多维度管理体系建设。

在开发测试环境中，通过云计算平台统一管理包括虚拟服务器、虚拟桌面在内的各类 IT 资源。云计算平台同时具有应用级别的自助交付功能，可实现自助化申请、审批、回收等流程，并可实现按需求进行自动备份、归档，按资源进行成本分摊等。监控智能处置流程如图 11-1 所示。

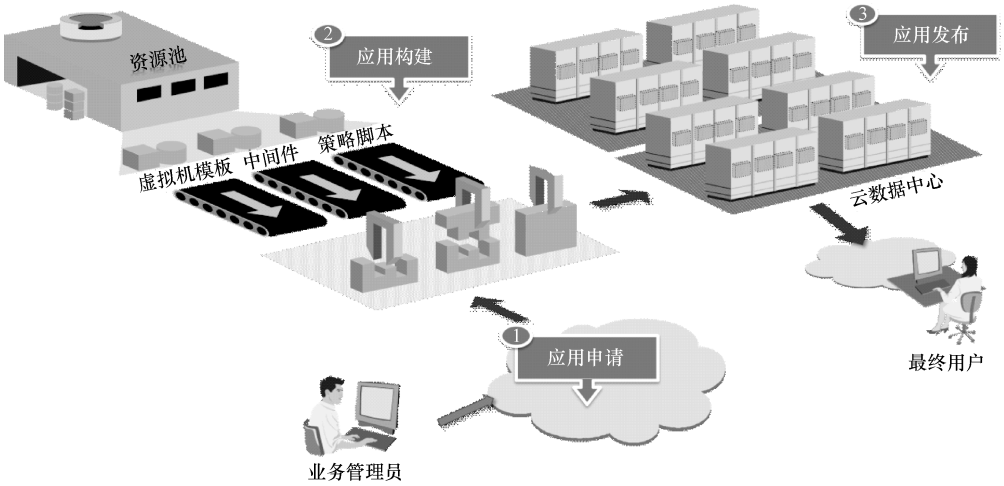


图 11-1　监控智能处置流程

11.3.2　云计算助力企业运维管理

在云计算环境下，运维人员面向的是所有的云资源，完成对不同资源的分配、调度和监控。云环境下的运维管理的目标是适应上述的变化，通过改进运维方式和流程来实

现云资源的运行维护管理。利用云计算提升企业运维管理水平主要体现在以下几个方面：

1）资源调度。提供基于策略的资源调度，系统自动判断当前资源状态，并且执行自动的工作流来分配及部署资源，按照既定的适用规则实现实时响应服务请求，根据用户需求实现资源的自动化生成、分配、回收和迁移，用以支持用户对资源的弹性需求。

2）统一监控。提供操作系统层、虚拟化层、物理硬件层、应用层等多维度的统一监控平台，实时监控、捕获资源的部署状态、使用和运行指标、各类告警信息，生产丰富的报表，供平台运维人员决策和处理。

3）统一管理平台。提供监控面向管理维护人员的统一管理平台，屏蔽异构的基础硬件以及虚拟化平台环境，将服务、资源的各项管理功能构成一个统一的工作台，来实现管理维护人员的配置、监控、统计等功能需要。

4）自动化运维。提供自动化运维手段，简化运维人员的资源部署、调整、查看等复杂度。

5）自服务门户。自服务门户将支撑基础设施资源、平台资源和应用资源以服务的方式交互给用户使用，提供基础设施资源、平台资源和应用资源服务的检索、资源使用情况统计等自服务功能，需要根据不同的用户提供不同的展示功能，并有效隔离多用户数据。

6）身份与访问管理。身份与访问管理提供身份的访问与管理，只有授权的用户才能访问相应的功能和数据，对资源服务提出使用申请。

7）服务质量管理。服务质量管理遵循服务水平协议要求，按照资源的实际使用情况进行服务质量审核与管理，如果服务质量没有达到预先约定的服务水平协议要求，将自动进行动态资源调配，或者给出资源调配建议由管理者进行资料的调配，以满足服务水平协议的要求。

8）服务交付管理。服务交付管理包括交付请求管理、服务模板管理、交付实施管理，实现服务交付请求的全流程管理，以及自动化实施的整体交付过程。

11.3.3 探索的创新与意义

通过引入虚拟化等一系列成熟技术，可以有效降低系统开发工作量和 IT 基础架构设施的耦合度，提高现有 IT 资源的利用率，实现开发环境中终端、网络、服务、存储等资源的集中管理和动态扩展。在开发环境管理、数据安全、基础设施及资源整合等方面，具有以下创新特点。

1. 有效控制 IT 成本

通过实施云计算平台，银行对开发测试环境的 IT 资源可以进行重新规划和利用，实现 IT 资源按需分配与及时回收，在提升 IT 资源使用效率的同时，实现对软件开发测试环境中的硬件成本、机房运营成本和外包人员管理成本等的有效控制。不仅如此，利用云计算平台对 IT 资源统一监控、管理的功能，银行能够实现根据当前资源使用情况

和增长预估，制订合理的资源扩容计划。

2. 应用开发环境灵活快捷

通过建立云计算平台，可以实现开发环境管理流程的自动化。开发环境的准入、资源申请、设备申请等流程的自动配置和流转，大大缩短了应用开发项目中环境准备、人员入场、应用软件安装等环节所需要的时间。借助云计算平台，邮储银行得以构建灵活快捷的应用研发环境。

3. 实现虚拟环境软硬件“热插拔”

借助云计算平台，银行能够实现虚拟环境软硬件“热插拔”。在云计算平台上，IT基础设施实现松耦合，软硬件基础设施可轻松分离，按需部署在不同的地理位置并完成各自的任务，当任务完成后又可迅速整合。IT 基础设施灵活、可靠、稳定的特性，充分满足了银行 IT 项目封闭开发、多地开发等应用环境要求。

4. 打造开发测试环境信息安全体系

将云计算技术与环境准入、桌面管理、数据加密等技术相结合，银行能够构建开发测试环境的信息安全体系。在虚拟桌面环境中，对 IT 资源进行统一配置和管理，确保数据运行和存储在平台内部，数据、信息无需通过网络传输，彻底隔绝台式计算机、便携式计算机等设备接入网络带来的风险隐患。

结合自身软件开发测试的现状，利用虚拟化技术搭建云计算平台，达到提高工作效率、简化管理流程、降低操作风险等目的。云计算平台具有全面性、系统性、兼容性、前瞻性、先进性等特点。

11.4 开发测试云建设过程

11.4.1 开发测试云的建设需求

基于邮储银行开发测试环境现状分析，采用以云计算技术为载体、以虚拟化架构为基础、以自动化部署为核心的银行开发测试云，并以此实现资源的最大化利用、软件的自动化部署、系统的实时监控，提升银行开发测试环境管理水平，满足应用系统大规模建设的要求。

11.4.1.1 业务支撑层需求

需要向开发测试云平台的终端用户提供统一服务的门户。门户提供以下的基础服务：

1. 用户管理

平台中有两大类用户，分别是开发测试云平台管理员和开发测试云平台使用者。管理员负责审批对云平台上所有资源分配请求，增加或减少资源数量，修改项目时间，终止或删除项目。平台使用者能够向平台申请资源，包括增加或减少，变更使用时间以及

终止使用资源。

2. 软件管理

平台管理员可以对软件进行管理，包括维护可部署的操作系统镜像及软件包、定制或客户化某些应用软件或者商业软件的部署。

3. 使用计量

提供资源使用情况统计报表。平台管理员能给出每个使用者使用的物理资源数量、时间和利用率等统计信息。平台使用者能够实时获取报告，了解自己申请的资源使用情况。

4. 部署管理

能够实时了解平台的基本性能状况，提供对当前申请使用的虚拟服务器的状态进行实时监控。

5. 安全管理

未来保证每个项目的安全性和隔离线，平台应具备完善的访问策略。用户可以通过两种方式访问系统：访问 Web 界面和访问项目虚拟机。位于 Web 界面的访问需要使用用户 ID 及密码。对于虚拟机的访问可以通过 VPN 设备对用户进行认证。

11.4.1.2 运营支撑层需求

开发测试云平台应该能够集中管理不同虚拟化平台，具有跨越异构平台的功能，有效整合异构资源，支持 kvm、VMware、Xen、Hyper-V 等多种虚拟化软件。能够实现按需分配的弹性资源使用模式，提高资源利用率实现自动化虚拟机管理和软件的安装，减少人工操作，提高系统交付时间。平台的扩展性要强，也允许客户进行二次开发，以及部署自定义软件。

开发测试云建设主要解决提高开发测试环境资源利用率、提升开发测试环境资源部署效率和提升服务水平。为了解决这三个问题，可采取多种技术手段加以解决。开发测试云建设的主要内容是通过实现虚拟化、监控，使用计划、自动化供应，流程管理，平台和工具建设，搭建一个完整的开发测试平台系统，以实现开发测试云平台资源的有效利用。其涉及自动化平台的建设、虚拟化架构的搭建和管理流程的梳理和开发、多种自动化软件的部署，以及其他自动化部署、监控、存储管理的软件，流程自动化管理软件，虚拟化平台软件等。

11.4.2 开发测试云建设过程

11.4.2.1 建设目标与原则

1. 建设目标：

1）引入云计算管理平台，优化 IT 组织架构、制度和流程，实现资源标准化、自动化、流程化，提高业务响应效率，同时采用统一管理及监控，实时导出业务、资源等维度的视图，从而更好地进行业务决策。

2）引入虚拟化技术，构建云计算资源池，实现 IT 系统资源共享与按需分配，是云计算的基础技术实施方向。

3）IT 系统实施云计算资源池，需要在 IT 系统建设和运维管理方面进行变革。云计算资源分配如图 11-2 所示。

① 集中的、跨应用系统，打破烟囱式限制，IT 系统资源不再局限在某个应用甚至某个部门。

② 资源池的建设、配置、采购不应与具体项目捆绑。

③ 与具体应用无关的横向维护，在应用和 IT 资源之间的界面清晰，运维管理各司其职，责任明确。

④ 资源管理依赖于自有专业团队而不是原有各系统集成商，能够最大化缩短需求响应时间，满足应用按需申请资源、快速上线的需求。

⑤ 建立合理的 IT 资源服务和资源使用质量考核机制，真正促进资源的合理应用，降低成本并提高效率。

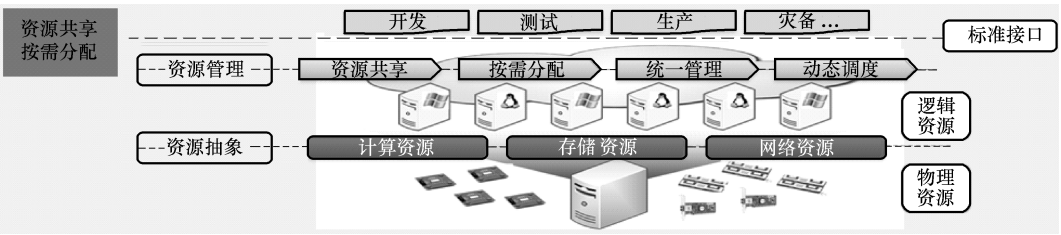


图 11-2　云计算资源分配

2. 建设原则

1）高可靠性原则：平台不间断、持续可用。

2）可扩展性原则：可以动态伸缩，满足应用和用户规模增长的需要。

3）资源灵活分配原则：根据业务需求可进行灵活的资源动态分配。

4）信息安全原则：明确的数据安全访问、存储、备份机制。

5）开放性原则：支持 Unix 和 X86 平台统一管理，支持多种虚拟化技术的统一管理；支持异构存储的统一管理。

11.4.2.2　云平台建设要点

每个云平台根据用户群的不同，业务模式的不同，在建设中，需要重点考虑的问题也不一样。对于云平台的建设，建议根据分步走的规划，特别是在初次搭建云平台的客户，建议遵循以下建设要点：

1. 资源池

云计算采用池化资源管理。所谓“池”就是公共资源，资源并不属于某一个应用或业务，而是根据其要求，从公共资源池中划分资源。

2. 自动化

云平台采用服务管理流程化、自动化的方式集中管理，减少人为参与，为平台的规模化扩展提供条件。

3. 易用性

对于业务系统，作为云计算平台的用户，不需要关心资源的来源及原理，只需要登录系统，使用资源。

4. 快速响应

当业务需求变化的时候，云平台可以通过弹性伸缩机制和自动化来快速响应，以适应业务的变化。

5. 可度量性

云平台的各种资源服务，如存储、CPU、内存、网络带宽和软件许可证等，是可以监控、控制和计量的。平台可以更好地统计 IT 资源使用率，为服务水平管理提供依据。

6. 高可扩展性

平台建设规模会随着业务类型增加和业务量的增加而迅速扩展，因此，高可扩展性是平台的重要特征。

7. 高可靠性

云平台通过多副本容错和计算资源同构可互换来提高服务的可靠性。对可靠性比一般的云平台更高，因此在资源选择上，就要采用可靠性高的服务器和存储。

11.4.2.3 计算资源池构建

服务器是云计算平台的核心，其承担着云计算平台的“计算”功能。对于云计算平台上的服务器，通常都是将相同或者相似类型的服务器组合在一起，作为资源分配的母体，即所谓的服务器资源池。在这个服务器资源池上，再通过安装虚拟化软件，使得其计算资源能以一种云主机的方式被不同的应用和不同用户使用。在 X86 系列服务器上，其主要是以云主机的形式存在。一般虚拟化软件由以下两部分构成：

（1）虚拟化内核平台　运行在基础设施层和上层操作系统之间的“元”操作系统，用于协调上层操作系统对底层硬件资源的访问，减轻软件对硬件设备以及驱动的依赖性，同时对虚拟化运行环境中的硬件兼容性、高可靠性、高可用性、可扩展性、性能优化等问题进行加固处理。

（2）虚拟化管理系统　主要实现对数据中心内的计算、网络和存储等硬件资源的软件虚拟化，形成虚拟资源池，对上层应用提供自动化服务。其业务范围包括虚拟计算、虚拟网络、虚拟存储、高可靠性（HA）、动态资源调度、云主机容灾与备份、云主机模板管理、集群文件系统、虚拟交换机策略等。

采用虚拟化平台对多台服务器虚拟化后，连接到共享存储，构建成计算资源池，通过网络按需为用户提供计算资源服务。同一个资源池内的云主机可在资源池内的物理服务器上动态漂移，实现资源的动态调配。

计算资源池的构建可以采用以下六个步骤完成，即计算资源池分类设计、主机池设计、集群设计、云主机设计、云主机模板设计和高可用性设计。

1. 计算资源池分类设计

在搭建服务器资源池之前，首先应该确定资源池的数量和种类，并对服务器进行归类。归类的标准通常是根据服务器的 CPU 类型、型号、配置、物理位置和用途来决定

的。对云计算平台而言，属于同一个资源池的服务器，通常会将其视为一组可互相替代的资源。因此，一般都是将相同处理器、相近型号系列并且配置与物理位置接近的服务器，比如相近型号、物理距离不远的机架式服务器。在做资源池规划的时候，也需要考虑其规模和功用。如果单个资源池的规模越大，可以给云计算平台提供更大的灵活性和容错性：更多的应用可以部署在上面，并且单个物理服务器的宕机对整个资源池的影响会更小些。但是同时，太大的规模也会给出口网络吞吐带来更大的压力，各个不同应用之间的干扰也会更大。

初期的资源池规划应该涵盖所有可能被纳管到云计算平台的所有服务器资源，包括为搭建云计算平台新购置的服务器、用户内部目前闲置着的服务器以及现有的并正在运行着业务应用的服务器。在云计算平台搭建的初期，目前正在为业务系统服务的服务器并不会直接被纳入云计算平台的管辖。但是随着云计算平台的上线和业务系统的逐渐迁移，这些服务器也将逐渐地被并入云计算平台的资源池中。

针对开发测试的需要和邮储银行实际情况，我们按照用途将云计算资源池划分为云主机和云存储区资源池、云管理和服务区资源池，以便云计算平台项目实施过程以及平台上线以后运维过程中使用。

在云计算平台搭建完毕以后，服务器资源池可以如图 11-3 所示。

虚拟化管理平台体系将云计算资源池的物理服务器资源以树形结构进行组织管理，云资源中的被管理对象之间的关系如图 11-4 所示。

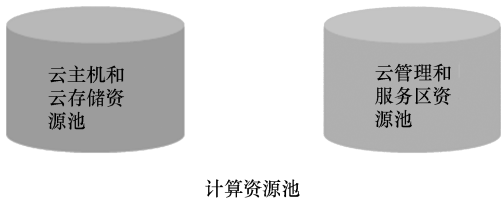


图 11-3　云计算资源池

2. 主机池设计

完成在云计算软件体系架构中，主机池是一系列主机和集群的集合体，主机可纳入群集中，也可单独存在。没有加入集群的主机全部在主机池中进行管理。

3. 集群设计

集群目的是使用户可以像管理单个实体一样轻松地管理多个主机和云主机，从而降低管理的复杂程度，同时，通过定时对集群内的主机和云主机状态进行监测，如果一台服务器主机出现故障，运行于这台主机上的所有云主机都可以在集群中的其他主机上重新启动，保证了业务的连续性。

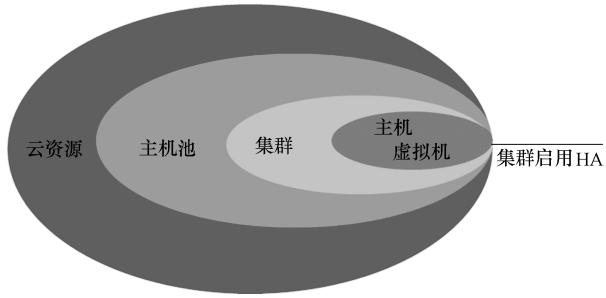


图 11-4　云资源对象关系

4. 云主机设计

每台云主机都是一个完整的系统，它具有 CPU、内存、网络设备、存储设备和 BIOS，因此操作系统和应用程序在云主机中的运行方式与它们在物理服务器上的运行方

式没有任何区别。与物理服务器相比，云主机具有如下优势：

1）在标准的 X86 物理服务器上运行。

2）可访问物理服务器的所有资源（如 CPU、内存、磁盘、网络设备和外围设备），任何应用程序都可以在云主机中运行。

3）在默认情况下，云主机之间完全隔离，可以实现安全的数据处理、网络连接和数据存储。

4）可与其他云主机共存于同一台物理服务器，从而达到充分利用硬件资源的目的。

5）云主机镜像文件与应用程序都可以封装于文件之中，通过简单的文件复制便可实现云主机的部署、备份以及还原。

6）具有可移动的灵巧特点，可以便捷地将整个云主机系统（包括虚拟硬件、操作系统和配置好的应用程序）在不同的物理服务器之间进行迁移，甚至还可以在云主机正在运行的情况下进行迁移。

7）可将分布式资源管理与高可用性结合到一起，从而为应用程序提供比静态物理基础架构更高的服务优先级别。

8）可作为即插即用的虚拟工具（包含整套虚拟硬件、操作系统和配置好的应用程序）进行构建和分发，从而实现快速部署。

在计算资源池中，一般物理服务器与云主机的整合比平均不超过 1∶8、单台物理服务器上所有云主机处理器内核数之和不超过物理机总内核的 1.5 倍、单台物理服务器上所有云主机内存之和不超过物理内存的 120%。

5. 云主机模板设计

云主机模板包括云主机的处理器、内存等参数，主机应根据主要应用系统负载量的不同提供不同的规格。

在采用云计算来向开发测试人员交付服务时，用户通过云门户自助申请的 IT 服务资源就是业务应用模板，因此需要提前设计好相应的 IT 服务模板向云门户发布，当用户申请该服务时，云平台根据模板进行资源编排，快速生成云主机相关资源并交付给开发测试人员使用。

6. 高可用性设计

1）云主机之间的隔离：每个云主机之间可以做到隔离保护，其中一个云主机发生故障不会影响同一个物理机上的其他云主机。

2）物理机发生故障不会影响应用：故障物理机上运行的云主机可被自动迁移接管，即云主机可以在同一集群内的多台服务器之间进行迁移，从而实现多台物理服务器之间的相互热备，实现当其中一个物理服务器发生故障时，自动将其上面的云主机切换到其他的服务器，应用在物理机宕机情况下保证零停机。

11.4.2.4 存储资源池构建

在开发测试云计算管理平台的建设中，存储资源池设计主要采用以虚拟化分布式存储为主要存储资源，以传统 SAN 存储为补充的混合资源池。

1. 虚拟化分布式存储

虚拟化分布式存储技术，融合了计算虚拟化和存储虚拟化，将计算和存储聚合到一个硬件平台，形成可横向扩展（Scale-out）的云计算基础架构。在开发测试业务应用区部署虚拟化存储方案，运行在这种架构上的云主机不仅能够像传统层次架构那样支持vMotion、动态资源调度、快照等，而且数据不再经过一个复杂的网络传递，其性能就会得到显著提高。由于不再需要集中共享存储设备，整个云平台基础架构得以扁平化，大大简化了IT运维和管理。利用虚拟化分布存储技术方案构建云平台存储资源池，有效利用服务器资源，降低能源消耗。

虚拟化分布式存储解决方案的逻辑架构如图11-5所示。这种架构的基本单元是部署了虚拟化系统的X86标准服务器。在提供虚拟计算资源的同时，服务器上的空闲磁盘空间被组织起来形成一个统一的虚拟共享存储：虚拟存储系统。虚拟化存储在功能上与独立共享存储完全一致；同时由于存储与计算完全融合在一个硬件平台上，无需像以往那样购买连接计算服务器和存储设备的专用SAN网络设备（FC SAN或者iSCSI SAN）。

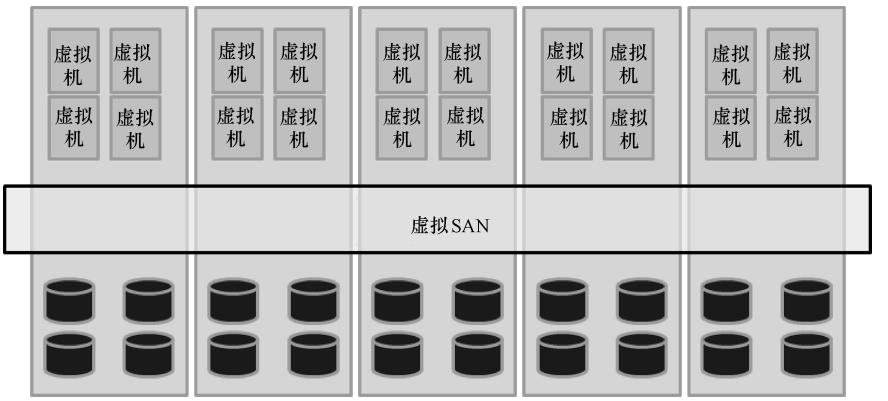


图11-5　虚拟化分布式存储解决方案的逻辑架构

虚拟化分布存储方案技术实现方案如下：

（1）分布式LUN（Logic Unit Number，逻辑单元号）设计　在虚拟化存储架构中，每台服务器同时也是一个存储节点。除了安装平台软件的系统盘外，每个节点上的其他所有磁盘空间（包括系统盘的剩余分区）都能被用于虚拟化存储。虚拟化存储会使整个LUN尽量均匀分布在所管理的全部节点和物理磁盘上，这样的设计使得对LUN的I/O操作能利用整个系统中全部节点和磁盘的性能。当管理员创建一个LUN时，虚拟化存储并不会马上为该LUN分配实际的物理存储空间，而是采用精简模式，在有数据写入的时候才分配存储空间。精简模式使用户在存储空间有限时，能按未来的业务发展需要提前规划LUN的容量。

（2）高可用性设计　用户可以根据业务需要为数据设置副本数量。虚拟化存储支持用户为每个LUN设置2~5个副本，并且使得不同的副本分布在不同的服务器和物理磁盘上，从而提供最大的容错性。当一个服务器出现故障，甚至多台服务器出现故障时，虚拟化存储仍能正常工作，而且数据不丢失。

（3）高性能设计　虚拟化存储采用分布式系统设计，其存储容量和性能随着服务器节点的增加而线性增加。由于每个 LUN 都横跨全部节点服务器和物理磁盘，所以每个 LUN 都可以利用全部服务器和物理磁盘的性能，从而提供比传统存储更高的性能。如图 11-6 所示，显示虚拟化存储的 IOPS，和吞吐能力随节点服务器数量的增加而呈现线性增加的状态。

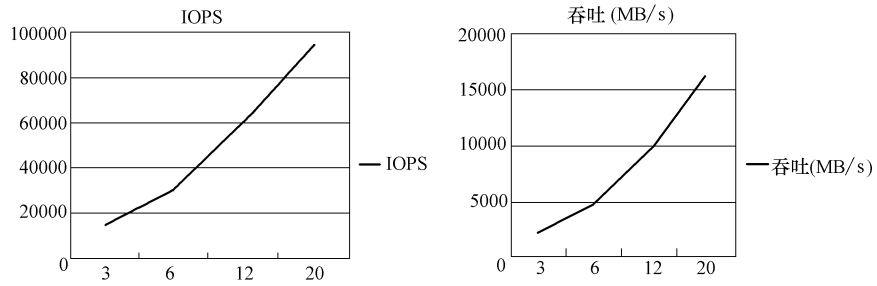


图 11-6　IOPS 随着节点的增加线性增加

此外，不同于传统 RAID（Redundart Arrays of Independent Disks，独立冗余磁盘阵列）以专用空闲磁盘作为热备，虚拟化存储由系统自动提供热备空间，并且将热备空间均匀分布在全部物理磁盘上。当数据重构启动后，全部节点服务器及物理磁盘都可以参与重构，从而实现最佳的重构效率。

可以在每个服务器节点上配置 RAID 卡缓存来增强 I/O 性能，根据存储容量的不同可以配置容量不等的 RAID 卡缓存。在追求更高性能时，还可以在每台服务器上配置 SSD（Solid State Disk，固态硬盘）作为热点数据高速缓存。

（4）自动化管理设计　虚拟化存储采用无中心架构，每个节点服务器的角色完全一样，这样用户无需像传统分布式存储系统那样管理元数据服务器。而且整个虚拟存储系统的元数据采用分布式设计，由系统自动管理，无需人工干预。

当系统扩容时（比如增加物理磁盘或者增加服务器节点），用户只需几条简单命令（通过命令行或者图形化管理界面）将物理部件加入集群，系统上原有的数据将自动重新均衡，原有 LUN 将自动扩展到新的物理设备上。

虚拟化存储具备强大的自愈能力，一般硬件故障无需人工干预。物理磁盘或者服务器节点故障后，系统可在数秒内自愈，自动恢复业务。

（5）按需扩展　虚拟化存储是一种可横向扩展（Scale-out）的分布式架构，当云平台需要更多计算和存储资源时，只需以服务器为单位进行扩容，即能实现计算与存储资源的同步扩展，而无需像传统存储阵列那样，哪怕是扩容一个硬盘，也需要再购买整套存储设备。采用每次增加一台节点服务器的扩容方案，虚拟化存储存储容量可从几太字节逐步扩展到几千太字节。新加入的节点服务器将在业务不中断的情况下，自动纳入统一资源池，极大缩短了扩容部署时间。

（6）管理简便　虚拟化存储系统采用无中心分布式架构，无需人为设置管理节点，从而避免了传统集群存储系统复杂的元数据服务器管理。系统具备强大的自愈能力，一

般硬件故障无需人工干预。一个节点故障后，系统能自动自愈，用户只需选择合适的时间窗口更换故障件即可使整个系统恢复到原样。当增加一个服务器节点时，无需中断业务，该节点的存储空间将自动纳入整个系统，系统中已有的数据将自动重新均衡分布。这一切都无需人工干预。

由于整个系统具备强大的自我管理能力，用户在使用虚拟化存储提供的存储资源时，只需简单地创建、删除存储资源，查看存储资源使用情况，以及通过日志了解系统运行状况，管理十分方便。

2. 传统 SAN 存储

在数据库等要求高性能或者稳定性需求比较高的业务系统，暂时仍旧采用传统 SAN 存储设计，以保证业务的稳定性和数据的可靠性，同时要求传统 SAN 存储硬件具备虚拟化或者支持第三方云平台管理的特性，方便通过云管理软件进行统一管理，以便形成虚拟化分布式存储资源池与 SAN 存储池整体统一管理；并且通过与虚拟化分布式存储进行比较，通过 IOPS 管理和部署难易程度，稳定和可靠程度，在开发测试云环境中进行充分测试，为未来在生产环境的应用提供实践和参考。

11.4.2.5 网络资源池构建

服务器虚拟化技术的出现使得计算服务提供不再以主机为基础，而是以云主机为单位来提供，同时为了满足同一物理服务器内云主机之间的数据交换需求，服务器内部引入了网络功能部件虚拟交换机 vSwitch（Virtual Switch），如图 11-7 所示，虚拟交换机提供了云主机之间、云主机与外部网络之间的通信能力。IEEE 的 802.1 标准中，“虚拟交换机”即为“Virtual Ethernet Bridge，VEB”简称 VEB，或“vSwitch”。

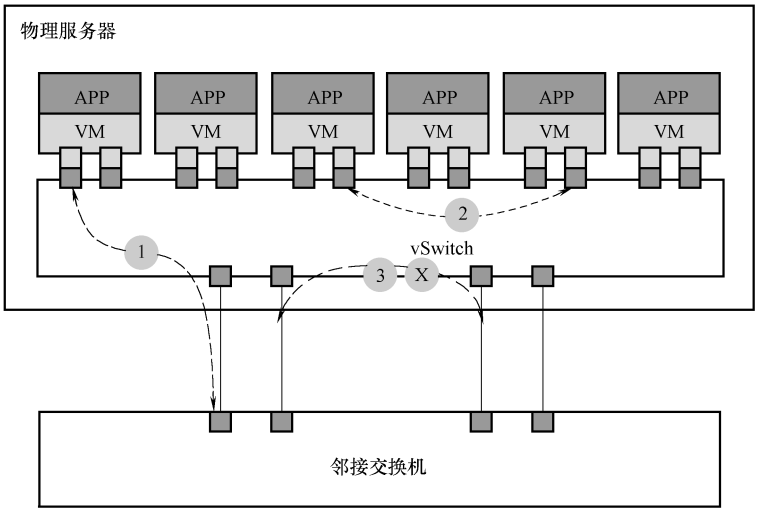


图 11-7　云主机交换网络架构

vSwitch 的引入，给云平台基础网络的运行带来了以下两大问题：

1. 云主机的不可感知性

物理服务器与网络的连接是通过链路状态来体现的，但是当服务器被虚拟化后，一

个主机内同时运行大量的云主机，而此前的网络面对这些云主机的创建与迁移、故障与恢复等运行状态完全不感知，同时对云主机也无法进行实时网络定位，当云主机迁移时网络配置也无法进行实时的跟随，虽然有些数据镜像、分析和侦测技术可以局部感知云主机的变化，但总体而言目前的云主机交换网络架构无法满足虚拟化技术对网络服务提出的要求。

为了解决上述问题，邮储银行的解决思路是将云主机的所有流量都引至外部列中交换机，目前将所有的流量均经过物理交换机，因此与云主机相关的流量监控、访问控制策略和网络配置迁移问题均能在物理交换机上得到很好的解决。该方案的核心思想是：将云主机产生的网络流量全部交给与服务器相连的物理交换机进行处理，即使同一台物理服务器云主机间的流量也将发往外部物理交换机进行查表处理，之后再 180°调头返回到物理服务器，形成了所谓的“发卡弯”转发模式，如图 11-8 所示。

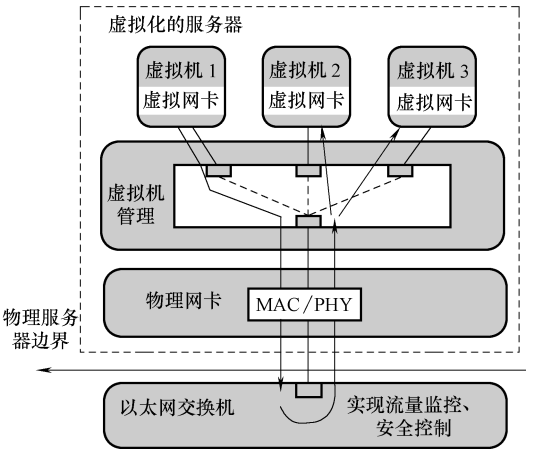


图 11-8　网络传输基本架构

云主机与网络之间的关联标准协议，使得云主机在变更与迁移时通告网络及网管系统，从而可以借助此标准实现数据中心全网范围的网络配置变更自动化工作，使得大规模的云主机云计算服务运营部署自动化能够实现。

将支持相关标准协议的 vSwitch 嵌入虚拟化软件，是为云平台基础架构提供最优化的虚拟化管理技术方案，该产品通过将云平台 IT 资源的整合，不仅能够达到提高服务器利用率和降低整体拥有成本的目的，而且能简化劳动密集型和资源密集型 IT 操作，显著提高系统管理员的工作效率。

2. 多租户的隔离

由于云主机及服务器数量的增多，网络技术方案需要保证多个租户使用的资源能够有效地隔离，通过 Overlay（一种封装在 IP 报文上的新的数据格式）虚拟化方式用来支撑云与虚拟化的建设要求，并实现更大规模的多租户能力。

Overlay 的本质是 L2 Over IP 的隧道技术，相应的技术方案称为（Virsual extensible LAN，虚拟可扩展局域网），目前在服务器的 vSwitch、物理网络上技术框架已经就绪。VXLAN 网络架构有多种实现，就纯大二层的实现可分为主机实现方式和网络实现方式；而在最终实现 VXLAN 与网络外部数据连通的连接方式上，则有多种实现模式，并且对于关键网络部件将有不同的技术要求，包括基于主机的 VXLAN 虚拟化网络和基于物理网络的 VXLAN 虚拟化两种大的实现方式。在已经进行虚拟化的环境下，这里将采用基于主机的 VXLAN 虚拟化网络方案，如图 11-9 所示。

VXLAN 运行在 UDP 上，物理网络只要支持 IP 转发，则所有 IP 可达的主机即可构

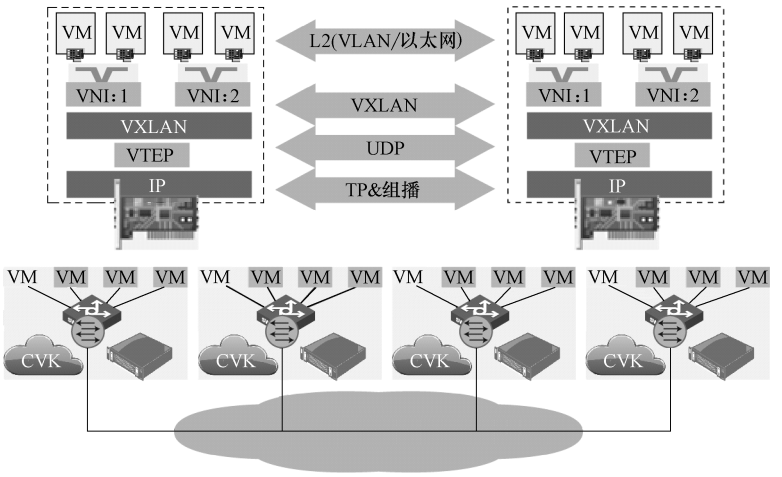


图 11-9　基于主机的 Overlay 虚拟化网络

建一个大范围二层网络。这种 vSwitch 的实现，屏蔽了物理网络的模型与拓扑差异，将物理网络的技术实现与计算虚拟化的关键要求分离，几乎可以支持以太网在任意网络上的透传，使得云的计算资源调度范围空前扩大。

另外由于 VXLAN 把 12 bit 的 VLAN 标签扩展成了 24 bit，这样能实现远远高于传统 VLAN 4096 的数量，解决了 VLAN 数量不足的问题，满足更多的租户隔离需要。

为了使得 VXLAN Overlay 网络更加简化运行管理，便于云的服务提供，通过使用集中控制的模型，将分散在多个物理服务器上的 vSwitch 构成一个大型的、虚拟化的分布式 Overlay vSwitch（如图 11-10 所示），只要在分布式 vSwitch 范围内，虚拟机在不同物理服务器上的迁移，便被视为在一个虚拟的设备上迁移，如此大大降低了云中资源的调度难度和复杂度。

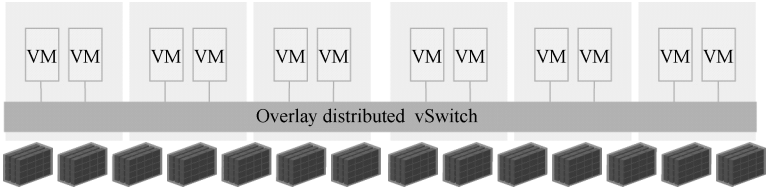


图 11-10　分布式 Overlay vSwitch

对于计算资源丰富的数据中心，Overlay 网络使得虚拟机不再为物理网络所限制。

11.4.2.6　虚拟化管理平台

1. 虚拟化管理系统

虚拟化管理系统是虚拟化平台的核心组件之一，主要实现对数据中心内的计算、网络和存储等资源池的管理和控制，对上层应用提供自动化服务。

管理平台可以集中管理数千台物理服务器和数万台云主机，通过一个统一的管理平台来对所有相关任务进行集中管理，管理员仅需要键盘和鼠标便可实现云主机的部署、配置和远程访问。

虚拟化管理系统可以实现以下功能：基于集群的集中管理、共享存储管理、虚拟交

换机管理、资源使用状况监控。

（1）基于集群的集中管理　虚拟化管理系统将服务器主机和云主机都组织到集群中，单个集群支持 5000 台以上的物理机，1PB 以上的分布式共享文件系统存储，另外单个集群支持的并行任务调度数量不低于十万级。集中管理提供了清晰的分层结构视图，直观地展示了数据中心、主机池、集群、主机和云主机之间的关系，大大简化了资源管理的工作量。

基于集群进行集中管理的好处在于：利用集中化管理功能，管理员能够通过统一的界面对整个 IT 环境进行组织、部署、监控和配置，从而降低管理成本。

由多台独立服务器主机聚合形成的一个具有共享资源池的集群不仅降低了管理的复杂度，而且具有内在的高可用性，通过监控集群下所有主机，一旦某台主机发生故障，虚拟化管理系统就会立即响应并在集群内另一台主机上重启受影响的云主机，另外支持集群的在线扩容，从而为用户提供一个经济有效的高可用性解决方案。

（2）共享存储管理　虚拟化管理系统中的虚拟机文件系统是一种优化后的高性能集群文件系统，允许多个计算节点同时访问同一虚拟机存储。由于虚拟架构系统中的虚拟机实际上是被封装成了一个档案文件和若干相关环境配置文件，通过将这些文件放在 SAN 存储阵列上的文件系统中，可以让不同服务器上的虚拟机都可以访问到该文件，从而消除了单点故障。

（3）虚拟交换机管理　虚拟交换机是用软件实现的 IP 报文转发与控制模块。在物理环境中，物理服务器通过物理交换机连接到网络，在云平台中，云主机通过虚拟交换机连接到网络。为了让维护人员直观易懂，在虚拟化管理系统里会对虚拟交换机有一个直观易懂的管理界面。

虚拟交换机在设计时将整个虚拟交换机以物理交换机的面板呈现，并通过绿色的、闪烁的端口表示云主机连接虚拟交换机的网卡，每一个闪烁的绿色端口都表示一个活动的虚拟端口，可以显示端口名称、连接端口的云主机名称、云主机网卡对应的 MAC 地址等。

（4）资源使用状况监控

1）物理服务器性能状态监测，如图 11-11 所示。提供物理服务器 CPU 和内存等计算资源的图形化报表及运行于其上的云主机利用率前 5 名的报表，为管理员实施合理的资源规划提供详尽的数据资料。

2）云主机性能状态监测，如图 11-12 所示。提供云主机 CPU、内存、磁盘 I/O、网络 I/O 等关键资源进行全面的性能监测。

3）虚拟交换机状态监测。提供虚拟交换机上各个虚拟端口的流量统计与模拟面板图形化显示。

2. 自动化、多异构资源、多服务模板管理

云平台采用服务管理流程化、自动化的方式集中管理，减少人为参与，为平台的规模化扩展提供条件。当业务需求发生变化时，云平台可以通过弹性伸缩机制和自动化来快速响应，以适应业务的变化。

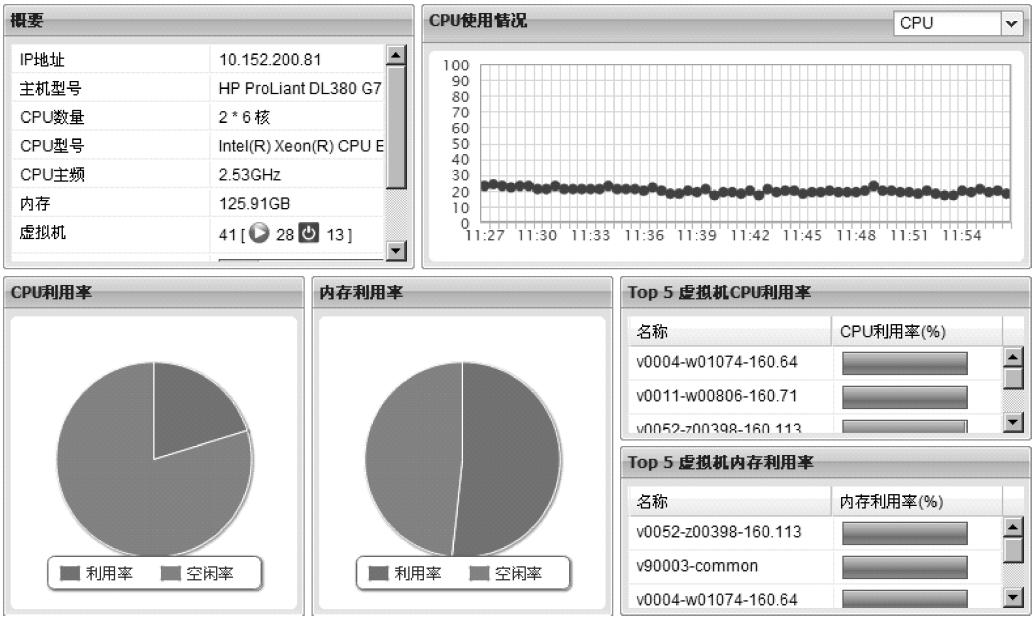


图 11-11　物理服务器性能状态监测

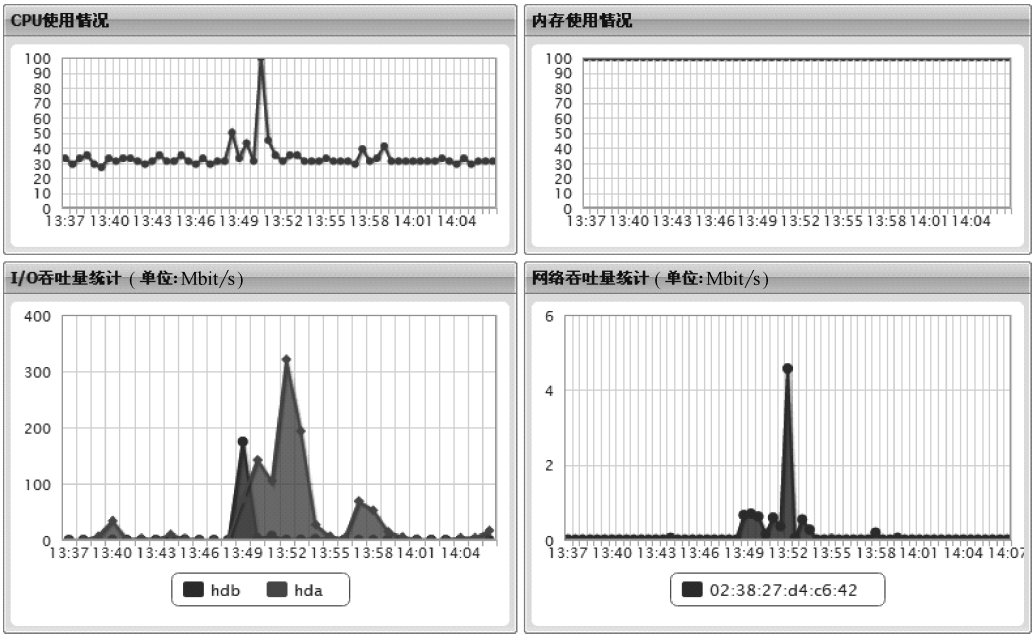


图 11-12　云主机性能状态监测

在长期的信息化建设中，用户积累了很多不同技术路线、不同架构、不同品牌的硬件产品——服务器、存储、网络设备，会部署各种各样的应用，可能已部署 VMwarevSphere、CitrixXenServer 等多种虚拟化环境。数据中心内积累了大量的软硬件资源，如何将这些资源迁移到云环境中，如何继续利用旧有设备，如何统一管理数据中心内种

类繁多的硬件资源、异构的虚拟化系统和不同的业务应用，是每个客户在实施云计算过程中难以回避的难题。只有开放融合的云计算方案，才能更好地整合和管理数据中心各类异构资源。

云计算数据中心提供的资源使用方式通过服务进行表达。服务模板是一些资源和资源集合的组合，用户可以根据自己的需求对这些服务模板进行配置订购。这些服务在经过用户订购以后将会生成和真实资源或资源集合一一对应的服务实例。所有的云计算中的资源包括网络、存储、计算能力以及应用服务，都需要以服务的形式向用户提供。任何云计算资源都可以按单个或多个有机整合的方式发布为云服务。系统提供服务目录管理功能。对服务的创建和发布、变更、激活、挂起、撤销进行全生命周期管理。一旦用户申请了服务，系统提供对服务实例的全生命周期的进行管理，如服务申请、操作、变更、终止等功能。

3. 应用部署

在完成云计算平台的建设后，应考虑建立应用模板，将业务应用逐步部署到云计算平台上。云计算现状如图 11-13 所示。

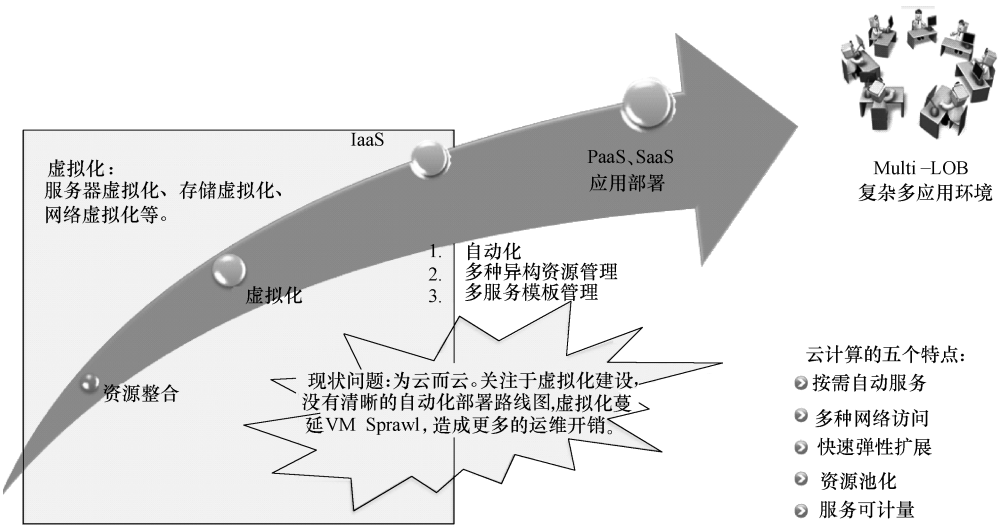


图 11-13　云计算现状

11.5　开发测试云部署方案

11.5.1　概述

建立开发测试云，部署有两个主要组成部分。

1）用于性能测试的硬件架构。

2）用于开发和非性能测试的虚拟机环境。

常备开发测试环境是建立在虚拟机环境下，充分利用主机虚拟化的能力，满足开发测试的需求。当有需要时，临时部署一些虚拟机测试环境以满足随机的需求，这样的工作通过临时测试环境来随需准备和管理。对于系统集成测试、压力测试等对性能要求较高，并且要求尽量符合生产环境的测试，通过性能测试环境的方式，实现先在虚拟环境下完成整个测试环境的准备工作，包括系统、应用、网络、负载发生器的联调。然后部署到实际的性能测试环境中，并通过合理的安排达到充分利用性能测试的硬件环境，建设测试中的意外情况发生，并协调自动化工具实现自动部署的实现。

11.5.2 云平台总体部署架构

通常情况下，基于银行多数据中心机器的分布情况，考虑到节省宽带和方便管理，应该在某个数据中心部署统一管理整个开发测试系统的云管理平台，开发测试云平台管理员可以通过统一的门户管理银行整个开发测试环境，包括项目开发测试环境的自动部署、系统资源监控以及系统资源的动态调度。按照开发测试系统的目前物理位置，构建几个资源池，不同的数据中心分别部署各自的镜像库和中间件安装介质库。

从逻辑角度来看，系统架构描述的是开发测试系统整个的设计思路，主要组件及其相互间的关系，以及组件、用户和外部系统之间的关系。逻辑架构从三个方面描述部署：首先是从总体角度阐述架构的宏观视图，描述业务目标和为支持开发测试云平台应具有的能力，并定义了主要的系统特征和系统需求；然后从功能模式、部署模式进一步描述整体架构，从系统功能组件和运行环境上给出宏观设计。

开发测试云平台未来将成为邮储银行开发测试环境的统一运行平台，实现基础设施资源共享，该平台的主要特点如下：

1）应用最新虚拟化技术形成高度虚拟化资源池，优化物理资源和简化资源管理，实现资源自动化快速部署。

2）快速自动安装、配置银行开发、测试环境需要的中间件，节省人力，减少人为失误。

3）IT 资产信息管理为客户提供 IT 资产的可视化管理。随时掌握 IT 资产和运行情况。

4）建立面向服务的 IT 服务提供模式，为 IT 基础设施的用户提供对 IT 基础设施环境能力的快速理解和资源使用需求的有效管理。

邮储银行开发测试环境的服务器架构通过水平方式对服务器系统进行整合，即通过资源分布进行整合，通过多个物理服务器或服务器分区共享应用负载，从而合并多个节点的计算资源，这样可以使系统的可靠性、可扩展性、多功能性及性能得以提高。服务器虚拟化主要通过服务器集群计算技术、服务器分区技术、虚拟 IOServer 等技术实现。

11.5.3 资源统一管理

在云管理平台中，资源管理模块的总体功能架构如图 11-14 所示。

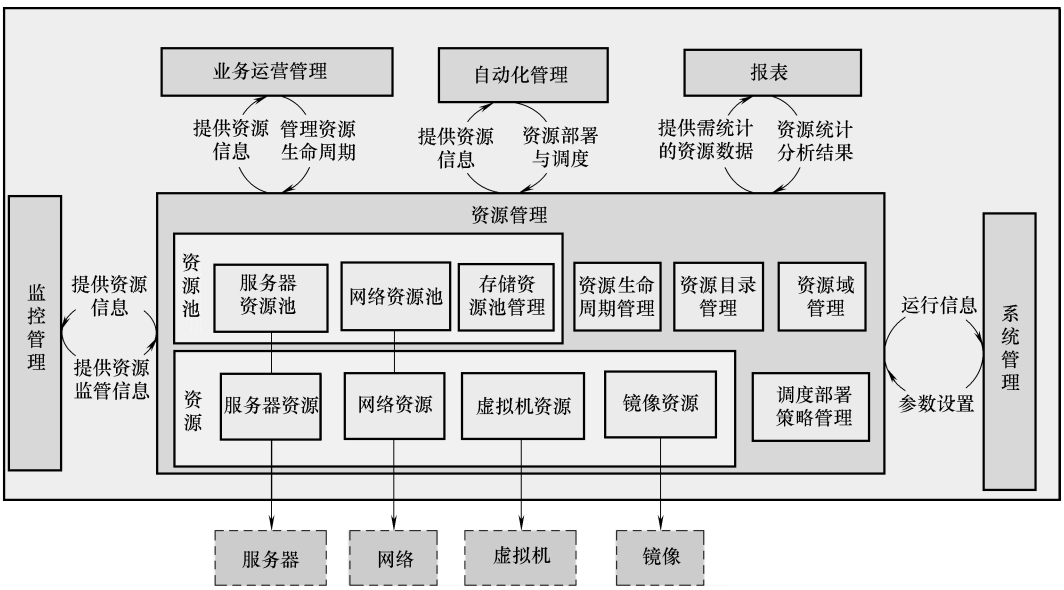


图 11-14 资源管理模块的总体功能架构

服务器资源管理提供了对服务器、小型机等物理设备的基本信息记录和系统管理接口的封装。通过对物理服务器的控制接口，可以对物理服务器进行开关机、重启、加电、断电等操作。云管理平台提供物理主机的容量信息、服务器主机实时性能信息和服务器主机历史性能信息查询。

1. 虚拟机资源管理

在云管理平台，服务实例体现为虚拟机，支持多种虚拟化类型，同时，支持对虚拟机实例的自动发现和配置管理，并和监控模块配合实现对实例运行的即时状态的监控和历史监控状态的回顾。

虚拟机资源管理具备如下功能：

1）支持虚拟机资源的创建、修改、查询和删除。

2）支持对虚拟机的生命周期管理（虚拟机的创建、上电下电、开关机、资源变更、删除等）。

3）支持虚拟机实例的迁移操作（手工迁移）。

4）根据虚拟机资源使用情况，虚拟机的自动扩容功能。

5）支持管理员查看物理机-虚拟机视图，可快速浏览虚拟机与物理机的承载关系。

6）支持对虚拟机运行实例的状态监控（CPU 利用率、内存使用率、磁盘空间、网络流量）和信息查询，并可查询虚拟机历史状态。

2. 镜像资源管理

镜像资源实质上应该是一种软件资源，由于镜像资源生成和管理的特殊性，把其单独作为一种资源，方便管理、规划与统一部署调度。

1）支持镜像资源的创建、修改、查询和删除。

2）支持镜像资源的拍照功能。

3）支持对镜像资源的抓取功能。

3. 服务器资源池管理

服务器资源池统一管理云平台的所有服务器资源，通过服务器资源池实现服务器资源的统一规划、统一定位、统一分配、提供服务器资源的统一对外接口。

平台提供对服务器资源池的监控和信息收集。

4. 存储资源池管理

将多个存储设备的资源整合在一起并抽象化，对外提供整体的出口和存储空间管理，让它看上去如同一个资源。通过存储资源池实现存储资源的统一规划、统一定位、统一分配，提供存储机资源的统一对外接口。

5. 网络资源池管理

虚拟机的 IP 通过网络资源池管理的 IP 自动分配得来。系统提供 IP 所属业务域的逻辑管理，不同业务域的虚拟机 IP 地址会在业务域中的 IP 池中分配。运维人员可在网络资源池中添加、删除、修改 IP 地址。系统提供 VLAN 信息的管理，包括 VLAN 号、VLAN IP 的管理。

11.5.4 全面监控

云管理平台中，监控管理功能域主要包含监控管理和告警管理两部分。

对云管理平台涉及的各种资源，系统的监控数据采集层模块通过 SNMP（Simple Network Management Protocol，简单网络管理协议）协议、代理等采集方式，对资源的性能指标进行主动轮询收集，并在此基础上提供对各类资源的故障管理、负载管理等功能，达到实时监控资源健康状态、主动发现故障和及时运维的目的。

云管理平台中监控管理部分需要与运行在云中的业务系统进行集成，最大化利用现有系统实现对核心业务域系统的监控，资源使用审批等功能，并提供统一的管理界面对外提供服务。

监控管理模块如图 11-15 所示。

全面监控是整合的端到端的监控：

1）涉及物理设备、虚拟化层、操作系统、应用程序、业务流程数据采集五个程度。

2）分为容量监控、性能监控、可用性监控。

3）由分层级事件和告警构成指标体系。

4）实时信息和历史信息结合的仪表盘展现。

5）智能数据关联，容量和性能趋势分析，统计分析报表。

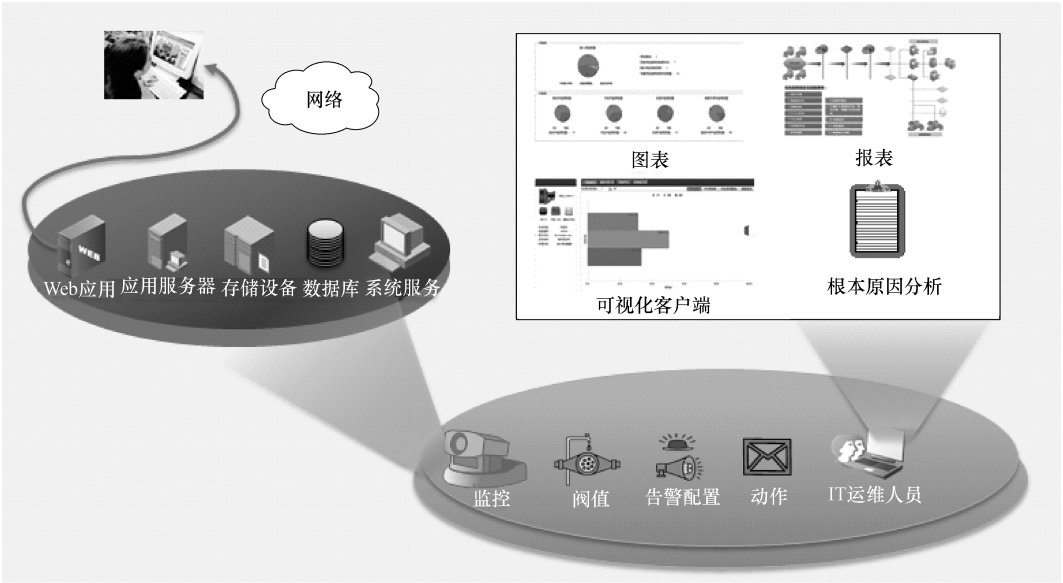


图 11-15　监控管理模块

6）系统消息、邮件和短信通知体系。

7）自动化处理流程，涵盖故障处理与资源调度。

11.5.5　应用软件部署

扩展平台的服务能力，除了计算、存储和网络资源之外，还可以在服务目录提供 HTTP 服务器、中间件环境和数据库环境的集群服务以及应用拓扑部署服务。支持的中间件服务器包括 Tomcat 及 PHP 等；支持的数据库服务包括 MySQL、Oracle、WAS + 等。

通过云计算平台管理软件，实现所有服务器整合为一个统一的云计算服务器平台，抽象出统一的硬件资源，包括 CPU 资源池、Memory 资源池、Network 资源池、Storage 资源池，任意云都可以按需在统一资源池中获得硬件资源并运行。由此实现了统一硬件资源整合，在统一的硬件平台上来实现云的分配、运行和维护，为云计算平台实现高扩展性、高伸缩性提供支撑。

11.5.6　资产台账和统一报表

1. 面向资源管理维度

1）云平台资源统计报表：系统的展示整个云平台资源情况，在现有基础上进行完善，以业务系统的维度进行展示，细化存储资源，以存储池、存储类型进行区分展示。

2）云平台资源使用量统计报表：展示云平台资源使用情况，完善现有使用统计报表，按业务维度区分，以存储类型及存储池进行细化。

3）云平台性能统计报表：完善平台性能统计报表，增加从物理机层面的展示以及

物理机的性能统计。

4）云平台运行状况分析报表：展示平台运行状况数据，包括物理机、虚拟机的CPU，内存、存储、网络资源的运行状况。

5）云平台运行趋势分析报表：根据云平台运行数据展示趋势分析报告。

6）存储池使用报表：已存储池的角度去展示每个存储池的使用情况，包括虚拟磁盘的使用情况。

2. 面向用户或业务系统维度

1）提供用户、业务系统的资源使用量统计报表：按不同的业务系统，根据不同的用户，不同的数据权限展示资源的使用量。

2）提供用户、业务系统的性能统计报表：从不同权限的用户，不同业务系统的角度去展示业务性能报表。

3）提供用户、业务系统的运行状况分析报表：对所关心的业务系统，主机的运行状况进行展示。

4）提供用户、业务系统的运行趋势分析报表：从业务系统及用户的角度去展示运行数据，得出趋势分析报表，以图形化的方式进行展示。

5）监控考核报表：展示考核报表，对业务人员的告警处理情况进行展示。

11.6 开发测试云给银行带来的价值

邮储银行采用开发测试云平台后，将所有IT资源进行整合，然后虚拟化为一个资源池。如果开发测试人员需要使用某种资源，那么只要提交资源使用申请表即可，开发测试云平台会自动安装部署所有需要的IT资源。资源使用完毕后，开发测试云平台也会自动回收资源，重新放回资源池，这样资源就可以得到循环使用，提高了资源的利用率。因此，使用开发测试云平台为软件开发测试通过了IT支持，实现了以下目标：

1. 提高资源利用率

使用开发测试云平台，把所有的IT资源集中投入到一个数据资源池，开发工程师或测试工程师可以根据工作需要来申请IT资源，环境管理工程师可以监控IT资源的使用情况，包括CPU、内存和存储利用率。使用开发测试云平台的IT资源调度管理和资源回收功能可以最大限度地高效利用资源和避免资源浪费。据统计，IT资源的每台PC的利用率不到20%，而使用的开发和测试云平台，IT资源利用率可提高到70%。

2. 降低软件开发生命周期

使用开发测试云平台来部署开发或测试环境，通过标准化的申请管理和自动化软件部署，大大降低了IT资源的部署时间。通常，部署一套环境所需的时间不会超过90min，而且可同时并行处理多项部署请求，大大减少了搭建环境所需的人力和物力，从而间接地缩短了软件开发生命周期，从某种意义上讲也节省了人力成本。

3. 降低IT总用于成本

使用开发测试云平台后，服务器的平均使用寿命是 5 ~ 8 年，而且更加稳定，处理能力更强。考虑到 PC 的使用效率较低，以及服务器 CPU 的处理能力，如果每年更换 4 台，只需花费 60 万元左右，直接节省 50% 的 IT 成本。

4. 优化业务流程

随着开发测试云的投产实施，部门内部管理和变更流程也进行了相应的调整和改变。例如，由于资源池的划分和新的环境搭建模式的建立，环境管理员和网络管理员平时只需维护资源池即可，根据资源池的变化情况增加或减少相关资源，这样对于突发的资源申请，环境管理员就不需要临时协调资源。环境管理员和网络管理员对资源池的管理工作绝大部分只涉及逻辑层面的变更，很少进行物理资源的变更，从而显著地提高环境管理的工作效率。

5. 提升管理水平

随着自动部署、监控、报表等操作的规范化和简易化，进一步明确了操作人员、运维人员、管理人员的职责，达到了简化流程、精细化管理的目的。

邮储银行结合自身软件开发测试的现状，利用虚拟化技术，搭建云计算平台，达到了提高工作效率、简化管理流程，构建新一代开发测试环境方面进行了有益的探索和实践。同时利用互联网金融发展趋势，把握金融信息化发展契机，向基于云计算、大数据的生产应用迈进，形成邮储银行新的竞争优势。

第 12 章
浙江省农村信用社联合社私有云的应用

12.1 在互联网金融上的创新

12.1.1 建设背景

从 2013 年起，浙江省农村信用社联合社（以下简称浙江农信）开始并计划用三年的时间，围绕创业普惠、便捷普惠和阳光普惠三大目标，积极开展网络覆盖、基础强化、扶贫帮困、感恩回馈和创新升级五大行动，让全省广大人民群众普遍享有基础金融服务的权利，享有参与经济社会发展的机会。

普惠金融工作主要包括以下三方面。

1）创业普惠让百姓深受惠。以“丰收小额贷款卡”和“丰收创业卡”为载体，推行农户贷款“一站式”、小微企业贷款“工厂式”、产业贷款“链条式”的“三式”服务模式，支持城乡农民创业创新，扶持小微企业成长。

2）便捷普惠让农信服务广受惠。以“丰收村村通”工程为基础，构建多层次、多渠道、广覆盖的服务网络，积极全面开展乡镇财政国库集中支付、社保卡、新农保、种粮直补等财政业务代理和政府惠民资金的发放，让广大城乡居民不出村、不出社区就能得到金融服务。

3）阳光普惠让城乡居民长受惠。以推进丰收信用工程建设为抓手，与农办联合评定信用村、信用乡；与省农业厅、省工商局联合推进农民专业合作社信用等级评定，完善省、市、县三级信用体系，努力实现“银村、银农、银商、银企”共建。

随着浙江农信业务需求的大幅增加及业务量的爆发性增长，原核心业务系统存在功能范围定位过大，以及承载太多管理类功能等不足。根据省农信联社科技规划 AS400 核心瘦身和去中心化的建设思路，2013 年实施并完成了信用卡、大总账系统剥离，以

确保作为“心脏”的核心业务系统低负荷和高效稳定运转。但剥离部分系统之后的核心业务系统仍然偏重交易控制和以账户为中心的系统设计模式，难以支撑爆发式的互联网金融业务的发展需要。同时，核心业务系统未进行服务标准化封装，子系统模块之间仍是紧耦合的关系。子系统间相互依赖性强、增加新的业务时开发复杂程度高、工作量大，难以支撑快速迭代的敏捷开发模式，更无法适应千变万化、开放式的互联网金融时代“以客户为中心、以产品为基础”的 IT 平台建设需要。

为了发展互联网金融业务，浙江农信势必需要借助云计算、大数据和开源技术组件，以开放式的全新设计理念和 SOA 系统架构模式，通过模块化、参数化、服务化和逻辑分层松耦合的方式，全新构建互联网金融平台。

互联网金融平台相对独立于现有核心业务系统、信贷系统、CRM 系统等传统平台，两者之间通过调用接口服务的方式实现数据交互，确保既满足新的互联网业务可以快速开发、快速上线，又能保持原有核心系统平台不受影响。

12.1.2 建设目标

浙江农信互联网金融业务建设需要实现以下目标：

1）建设一个全新的互联网金融服务系统平台。该平台相对独立于现有核心业务系统平台，通过服务接口调用的方式实现与现有核心业务系统平台之间的数据交互和业务处理，以满足新的互联网金融业务快速开发、快速上线的需要，同时又保持原有核心业务系统平台不受影响。

2）建设一套全新的综合电子账户系统。综合电子账户系统将构建成为多层级的电子账户体系，以“客户为中心”实现客户资产负债信息的完整视图展现，满足网上商城、社区 O2O、金融商城在内的快速支付结算需求，同时为今后直销银行的开放式多产品应用做好账户准备。

3）建设基于互联网金融服务平台，开发符合浙江农信特色的“行社特色馆”网上商城业务系统。采取统分结合的方式，由省农信联社负责搭建系统平台，行社基于该平台开展电商业务，网上商城立足本地，以主打农副产品、农家乐、农资“三农”为切入点和突破口，形成特色化、精品化、差异化的垂直型电商。

4）建设基于互联网金融服务平台，开发符合社区银行需要的 O2O 综合信息服务系统。社区 O2O 服务系统将充分发挥浙江农信点多面广的优势，以社区或中心集镇为基点，通过整合社区或中心乡镇周边的政府平台、物业、购物、餐饮、文化娱乐、健身保健等综合资源，为社区客户提供便捷的生活、消费等综合信息服务，从而支持社区金融业务的快速发展。

5）在网上商城系统和社区 O2O 综合服务系统中嵌入浙江农信的金融服务功能。将现有的支付、缴费、代购、理财等金融服务嵌入到网上商城和社区 O2O 互联网平台渠道中，后期依托商城系统、O2O 平台及外部大数据，重点开发线上实时小额消费信贷业务。

12.1.3 整体架构

互联网金融交互平台将包括电商平台、O2O 平台、综合电子账户平台、大数据平台等子系统组成，遵循统一模型、统一框架、数据集中、数据共享、后端解耦、前端融合的建设原则，其中综合电子账户平台将作为互联网金融交互平台的核心，是实现以“客户为中心”服务战略的基础。

综合电子账户平台作为购物商城、O2O、金融商城的底层核心平台，承担着提供基础电子账户开户、便捷支付结算和资金归集、以客户为视角的集成账户视图等服务，也是与银行核心系统平台实时交易的关键枢纽，分为基础业务平台、公共服务、产品和渠道四个层面。基础业务平台层作为综合电子账户平台的架构基础及核心，支撑各种业务应用服务。公共服务层为用户提供各种业务应用服务。产品层分为行业应用平台、对公业务平台和个人业务平台。渠道层适用于 PC、PAD、手机、ATM、VTM、数字电视、固定电话等多种业务终端，满足各种业务应用场景。

在业务方面，浙江农信通过建设综合电子账户系统，需能有效支撑互联网金融其他子项目，提供包括便捷电子账户管理、多渠道支付结算与资金归集、整合挖掘分析客户信息等功能。

综合电子账户系统定位于一个安全、高效、稳定、开放、可伸缩的平台型系统。作为众多子系统的信任基础、互联网金融项目与银行内部系统的重要纽带，综合电子账户系统须实现绝对资金安全、极低的响应延时、极高的业务连续性，同时兼容各类银行系统与第三方支付系统并支持廉价线性拓展。

12.2 私有云在互联网金融中的定位

12.2.1 传统模式建立互联网金融所遇到的问题

在向互联网金融转型过程中，依靠传统银行信息系统建设过程中碰到了如下困难。

1）随着手机网银等移动设备的广泛应用，各种促销活动的大量展开，系统需要具备大规模的并发能力和快速弹性能力，原有的烟囱式架构无法满足大量增长的业务需要。

2）数据来源越来越多，类型丰富、容量巨大，传统存储模式成本巨大，管理困难，分析时间较长。

3）项目实施过程中，周期较长，无法满足业务快速变化的需要。

4）随着集群和服务器的增加，需要大量的运维人员。

5）大量使用 X86 服务器和开放式的架构，对于系统安全和业务的可靠性的要求

增加。

12.2.2 以私有云为基础的全新的互联网金融服务系统

浙江农信经过周密的评估，采用华为公司的融合架构一体机（FusionCube）产品作为基础设施，FusionSphere 为私有云平台，承载着全新的互联网金融服务系统，云平台系统的设计主要基于以下性能目标：

1）处理性能：系统数据库服务器、应用服务器、Web/App 服务器的处理能力要根据业务量和用户量的具体情况满足系统处理要求；同时，系统的网络带宽支撑必须具有较强的处理能力，避免可能出现的通信拥塞。

2）可靠性：系统数据库服务器、应用服务器、Web/App 服务器主要通过分布式架构来满足高可靠性要求，同时满足应用级同城（或异地）灾备、数据库异地灾备模式，来提升系统和数据安全等级；电子账户后台模块可考虑独立构建峰值系统（同时适用于电商和 O2O 平台），采取长连接和异步化的设计来减少网络拥堵，来达到系统的高可用性。

3）安全性：由公有云服务商提供网络、防 DDOS 攻击等在内的基础实时监控，由系统管理员负责对应用系统、数据库进行实时监控，同时应用部分提供严密的操作权限控制和安全管理措施。

4）可管理性：系统应用设计采用 SOA、组件化、参数化设计，使系统能够便于随着业务需求的变化做出相应的调整和扩展。

12.3 私有云建设方案

12.3.1 私有云的部署架构

浙江农信私有云系统部署架构如图 12-1 所示。

所有 IT 基础设施以云计算资源池的模式向上层提供服务，私有云部署的服务器集群均是由私有云中的虚拟机或由云管理平台统一管理的物理机组成的。

综合电子账户将交易数据、交易数据等结构化数据存储于 DB2 中，采用 DB2 pureScale 的部署方式，电商、O2O 应用采用分布式部署 MySQL 数据库，利用多个数据库节点同步运行一个 MySQL 实例，提升并发数据处理能力。所有访问综合电子账户的请求均通过由云计算所带的负载均衡功能，均衡地分配到云服务器进行处理，以避免单点过热问题。云解析服务快速定位可提供服务的资源，帮助综合电子账户快速实现服务规模的动态伸缩。定制云监控策略，全面监控综合电子账户的运行情况，在第一时间通过邮件、短信、即时通信软件等渠道发送系统异常信息给系统管理员。

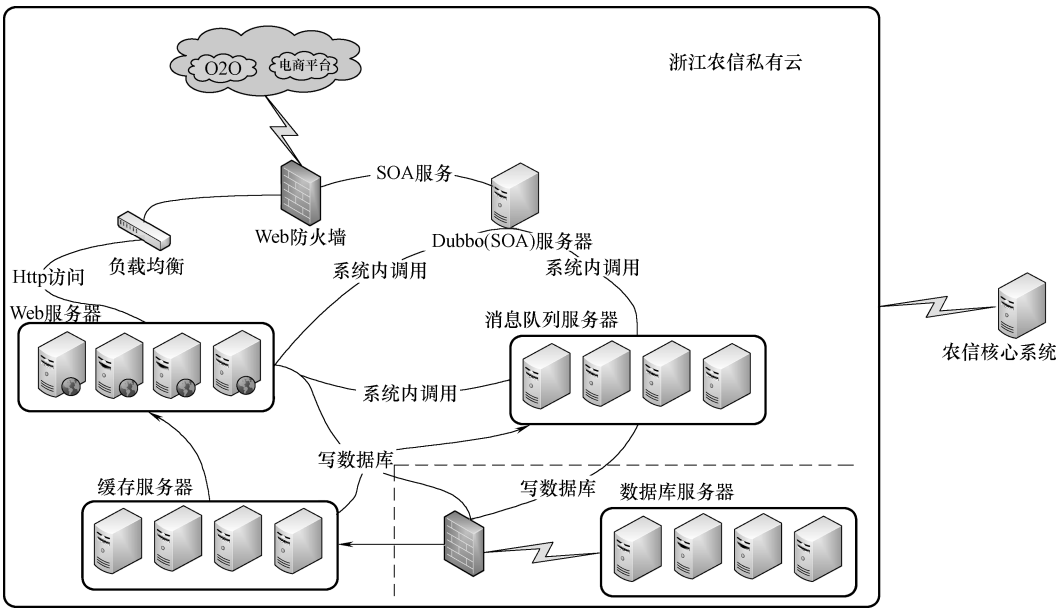


图 12-1　浙江农信私有云系统部署架构

为保证极高的业务连续性、极低的系统延时与极大的系统吞吐量，包括账户中心、交易中心等应用模块采用集群式方式部署。集群底层统一管理调度集群中的各服务器主机。

为避免数据库成为系统服务或计算瓶颈，数据库采用 MySQL 集群式部署，集群中的任意节点均能够进行读写操作。集群能够自动分表并实现冗余备份，任一节点出现故障均不会造成数据丢失。

应用模块采用松耦合架构，视业务需求采用同步或异步的方式交互。同步调用模式中，各模块向服务注册中心注册本身可提供的服务，以供其他模块快速查询并调用。异步模式中，模块集群间通过消息队列传递消息。集群中各服务器主机产生的日志通过海量日志汇集框架或消息队列自动实时归集，理想的日志汇集框架应能够提供自动负载均衡等高级可选配置。

对于高度结构化的数据，系统使用分布式 MySQL 数据库集群存储，保证数据的原子性，一致性、隔离性、持久性。集群式数据库相比单机数据库规避了单点故障的风险，相比热备数据库减少了资源浪费，相比主从式数据库消除了写操作的性能瓶颈。此外，开源 MySQL 集群式部署的方式在业界已经成熟，能够在降低商业数据库版权与设备成本的同时极大地提升数据库性能。

用户行为数据等半结构化数据不必严格保证一致性，同时一旦写入后很少变更，且写操作极为频繁。因此，根据半结构化数据的特性，系统使用 No-SQL 数据库集群存储。

外部接入系统可分为省联社传统银行系统、互联网金融子系统与第三方战略合作伙伴系统。为降低系统的耦合度、提高性能与安全性，综合账户系统对外开放多种接入方

式，保证系统的兼容性与开放性。

综合电子账户系统与省联社系统通过专线方式对接，保障通信的稳定性、安全性与响应时间，并灵活地使用多种接口，通过同步或异步的方式进行对接。此外，系统后端能够提供大批量数据导入导出的专用安全管道。

互联网金融综合电子账户系统架设于私有云内网环境中，并通过子网设置进行业务隔离，因此具备较高的安全性与通信效率。接入方式也可选择多种方式。

与第三方战略合作伙伴系统进行对接时，系统可根据战略合作伙伴系统重要性、紧密性与接入风险，并根据对方系统部署环境、开发语言、接口支持、通信数据不同，具体分析提供的接口方式与接入深度。如通过公共互联网对接，则要求对方系统发送的通信信息经过可认证的数字签名，保证数据的完整性与不可抵赖性，同时可通过加密的方式保护敏感数据。

整体安全性方面，通过配置运行环境（云环境、操作系统与服务器）、完善代码规范、加密敏感数据等多种方式，全方位、纵深式保障系统安全与网络安全。

12.3.2 私有云总体架构

浙江农信私有云平台采用融合架构实现对互联网金融平台的基础支撑。将包括计算、存储、网络资源进行整合，通过虚拟化实现资源的充分利用，并进行所有 IT 基础设施资源的统一管理。私有云总体架构如图 12-2 所示。

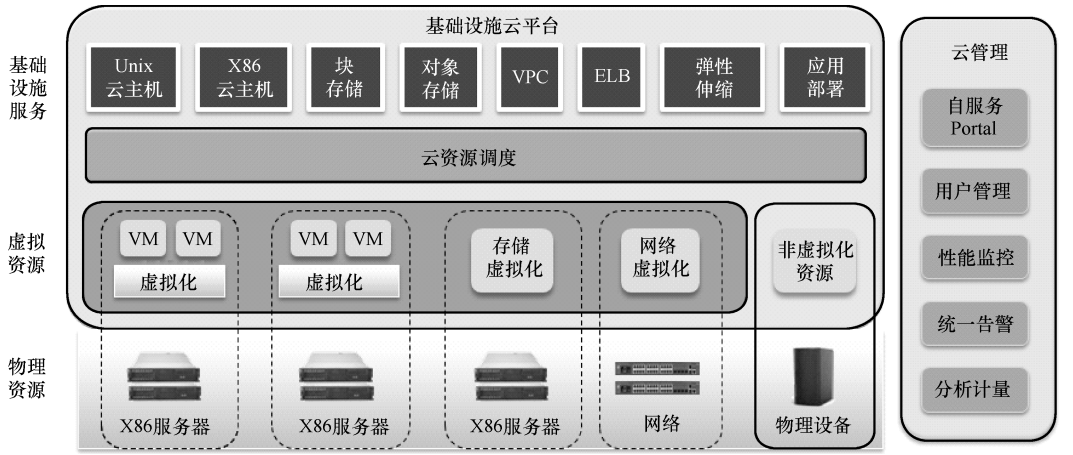


图 12-2 私有云总体架构

基础设施云平台提供基础的计算、存储和网络虚拟化的能力，在资源管理层的管理下，提供统一的计算、存储、网络资源池。基础设施层同时提供本地的基础运维能力，包括对本地基础设施的告警、性能、拓扑的监控等。

云管理提供私有云内镜像管理、业务编排、自动化、弹性及网络服务。资源管理支持对不同虚拟化平台计算资源的统一管理；支持对防火墙、交换机的统一管理；支持对虚拟化平台使用的块存储设备的统一管理。

在物理资源层，服务器、存储、网络等设备组成资源集群。资源集群由一个或多个计算资源、存储资源和网络资源组成。在虚拟资源层，一个虚拟化环境中可以有多个资源集群。资源集群管理包括创建资源集群、删除资源集群、扩容资源集群、减容资源集群、查询资源集群、资源集群性能监控和资源集群的调度策略等。通过资源集群管理，可以提高基础设施资源的利用率和灵活性，统一虚拟化资源管理的能力，对上层应用的发放屏蔽差异，提升了虚拟资源的管理效率，降低了运维成本。模板管理实现包括虚拟机模板的创建、发布、查询、修改等。云管理平台可以实现多数据中心统一管理调度，应用快速部署，弹性伸缩。

12.3.3　云计算融合基础设施

浙江农信采用融合基础设施为私有云提供了承载平台，融合基础设施的单位形态是融合架构一体机，将计算、存储、网络、管理等功能集成在一个机箱中。其组成形态如图 12-3 所示：

1. 硬件组成

融合架构一体机硬件平台中集成了可以混插的计算/存储刀片和 GE/10GE/FC/IB 交换模块。计算和存储刀片可以灵活配置以应对不同的工作负载。单个机框在仅仅 12U 的空间内提供 64 个 CPU 和 12.3TB 内存的计算能力，使融合架构一体机尤其适合需要高计算密度的云计算工作场景，整合的存储和 SSD 缓存对提升应用和数据库性能有很大帮助。

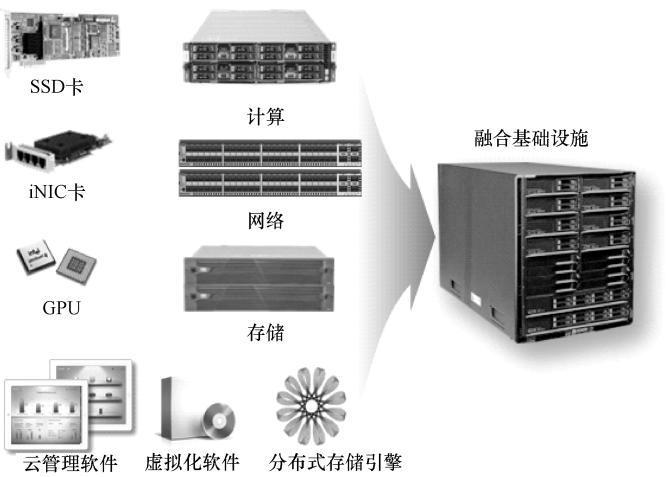


图 12-3　云计算融合基础设施组成形态

2. 虚拟化软件

在融合架构一体机中，私有云的计算虚拟化模块定位于构建针对互联网金融高性能业务需求环境的虚拟化平台，通过对开源 Xen 进行安全加固、功能扩展、性能优化和可靠性保障，打造安全、高效、稳定、开放的虚拟化平台。针对互联网金融大规模特点，虚拟化软件支持大规模运维支撑能力，提供虚拟机完善的生命周期管理能力、虚拟机运行状态查询能力、虚拟机动态调整能力及虚拟机远程安装部署能力，支撑虚拟机大规模运维管理。引入“黑匣子”技术，在系统出现异常或宕机时自动存储 VMM 内核日志、系统快照、内核诊断信息及临终遗言，并保存至非易失性存储设备或自动传送至网络服务器。

3. 云管理软件

融合基础设施采用云管理软件为融合架构一体机提供计算、存储、网络资源的调度管理功能，并对外提供弹性计算、负载均衡、虚拟私有云等 IaaS 服务。统一的管理系统让用户可以通过单一控制台管理所有资源，从交换机到虚拟机，从存储卷到软件应用，从应用部署到自动化和安全。融合基础设施可以从单个机框平滑扩展到整个数据中心，在完成新增硬件的物理连接后，管理系统会自动发现和完成配置，使云计算平台的扩展比以往任何时候都简单。

4. 融合基础设施的优点

采用融合一体机作为云基础设施平台后，系统部署周期从传统模式的 3 个月变为 14 天，现场部署只需 1 天。融合基础设施提供云计算所带来的便利同时，仍然保持了传统数据中心的高性能和高可靠性。

通过更高的资源利用率、更少的连线和网络连接，相比传统的计算、网络、存储分散部署的模式，节省了 50% 以上的机架空间，降低了 IT 机房投资成本。

融合一体机可以线性扩展来适应金融互联网业务，可成为数据中心及跨数据中心规模部署的融合基础设施平台。

融合架构一体机适合互联网金融业务的多种应用场景，能够高效地承载数据库和中间件系统平台。可横向扩展的存储引擎是融合架构一体机的一大亮点，基于 P2P 和 SSD 缓存技术使它不再需要传统的 RAID 控制器却可以提供更高的可用性和扩展性。在条带化后，数据被保存在所有数百到数千个磁盘上，系统中不会有过热或过冷的磁盘，这种做法既增加了磁盘的利用率，又大幅提升了 I/O 性能，相比采用同样数量磁盘的传统存储系统，融合架构一体机可以实现 3 ~ 5 倍的 IOPS 值。

12.3.4 分布式存储

为了应对大容量、高并发的互联网金融业务需求，融合一体机采用分布式存储软件，与融合基础设施硬件平台配合实现高性能、高可靠而且具线性扩展能力。

分布式存储软件作为一种与计算融合的存储软件，通过在通用服务器上部署该软件，可以将所有服务器的本机磁盘组织成一个虚拟存储资源池，在浙江农信的互联网金融业务场景下完全替换外置 SAN。分布式存储软件使计算和存储高度融合，达到高性能、高可靠和高性价比。分布式存储引擎具有以下特点：

1. 突破性的架构和设计

分布式存储软件的创新设计使其能够实现传统架构通常无法达到的性能优化效果。这种优化充分利用系统资源并且在所有的系统硬盘驱动器之间自动分配工作负载。此外，系统还支持一系列高级功能，如精简调配和快照等，而且这些功能不会对性能产生负面影响。

2. 一致的、可预测的性能及可扩展性

分布式存储软件系统在所有磁盘之间优化分配所有工作负载的能力以及强大的分布

式缓存结构允许系统通过添加服务器节点来平滑、线性扩展高性能。这种高性能表现具有不同场景的一致性，无需手动调试，因此用户可在因为业务量和快照使用模式发生变化而造成的高峰期和低谷期享受到相同级别的高性能，即使在组件发生故障时也不例外。

3. 高弹性和自愈能力

分布式存储软件可在硬件故障期间保持高度弹性，继续正常运行，几乎不会对性能产生任何影响。这个解决方案的高级自愈功能使其能够在最初的故障恢复之后抵御更多的硬件故障。

4. 计算存储高度融合

分布式存储软件作为一种把存储与计算融合的存储软件，将所有服务器的本机磁盘组织成一个虚拟存储资源池进行管理，具有管理自动化、运行高性能、安装免工程配置等优点。存储不再作为单独网元进行专门的配置和管理，使计算节点上的资源得到充分利用，企业用户在得到高性能存储的同时，节省了购买外置存储的高昂费用。

5. 可以实现快速自愈

数据分片在资源池内打散，硬盘故障后，可在资源池范围内自动并行重建，1TB 数据重建时间小于 30min（48 个节点），无需热备盘，完全满足互联网金融的业务需求。

6. 支持无限次快照

系统支持无限次快照，不降低系统性能，同时可支持一致性快照组，用于保证对整个应用或虚拟机的一致性。在进行重大升级等可能导致虚拟机故障操作前，可先对虚拟机做快照，以便在升级过程中遇到故障无法成功升级时，可以将虚拟机恢复到快照时刻的正常状态。

12.3.5 私有云运维管理

浙江农信私有云的融合基础设施部署与运维可以实现开箱即用、模板化简易部署和统一化运维。

1. 开箱即用

通过预集成的模式，开箱即可使用的融合基础设施一体机模块，极大地简化了现场的安装、调测时间，从几周甚至数月的调测时间缩减到几个小时之内。预集成包括的内容：

1）硬件预安装：设备上柜、线缆预绑。

2）软件预安装：BIOS 定制、虚拟化软件软件安装和存储管理软件预安装。

3）整机集成调测：齐套性检查和连通性检查。

4）带柜运输：立柜运输。

5）现场开局：上电硬件检测、齐套性检查和连通性检查。

2. 模板化简易部署

私有云系统提供三种模板化部署能力，包括软件包模板、虚拟机模板（能指定虚拟机的规格、虚拟机所安装的 OS、应用软件、是否支持 HA）和服务模板（包含虚拟

机、应用软件、组网信息等)。系统提供模板的导入导出功能，提供模板的发布、创建应用等功能。

私有云平台采用统一 Portal 上提供图形化设计 IT 系统的能力，能通过拖拽的形式，将用户 IT 系统需要多少物理机、虚拟机；虚拟机如何组网；虚拟机安装哪些软件包等信息全部图形化设计完成。

3. 统一运维

私有云可实现统一运维。统一运维是指在融合基础设施平台上，采用统一的用户界面对常见的操作维护功能进行管理，使用户保持一致的体验，大大简化了用户学习与熟悉界面的实践，降低运维人员操作维护出错的概率。统一运维包含的内容包括：

1）统一的用户界面。

2）统一的硬件资源管理：服务器、存储、网络的物理设备资源的统一管理。

3）统一的虚拟基础设施管理：服务器、存储、网络的虚拟化资源的统一管理。

4）统一的云基础服务管理：各种应用软件资源（例如虚拟桌面资源）统一管理。

5）统一的 OM 功能管理：包括 TOPO、告警、监控等。

系统具有拓扑和监控功能，维护管理员通过分权分域功能，可以监控自己权限内的资源信息，掌握资源使用情况和设备健康状况。支持管理员自定义监控统计项。支持按监控对象所属的逻辑节点或按虚拟机的业务类型等多维度分类监控方式，方便用户管理使用。支持监控物理资源、虚拟资源的 CPU、内存、硬盘等使用情况。在云平台方面，可监控 CPU、内存、存储平均占用率，故障服务器数量，虚拟机 CPU、内存、存储资源分配情况等。在集群方面，可监控 CPU、内存平均占用率，故障服务器数量，虚拟机 CPU、虚拟内存分配情况等。对服务器的监控包括服务器 CPU、内存、磁盘占用率、磁盘 I/O，网络流量，虚拟机数量，服务器基本信息等。对虚拟机监控，包括虚拟机 CPU 占用率、内存占用率、运行状态、网络流入和流出流量、磁盘读写速率等。对交换机可监控交换机端口状态、流量。可统计服务器性能趋势、虚拟资源分配统计等信息。

系统支持拓扑自动发现系统资源。支持以拓扑图的形式展示资源、资源关系及状态，其状态包括正常和故障等情况。资源包括集群、服务器、虚拟机和存储。

系统支持告警管理。通过故障管理确保系统正常运行的重要活动，包括系统故障预防设计、故障检测和处理。告警管理是故障管理的重要部分，并提供 Email 和短信通知告警功能。

在日志管理方面，包括日志记录、查看、审计。支持的日志包括操作日志、系统运行日志和黑匣子日志。操作日志包括：管理员访问运维管理平台日志，即管理员的操作日志，包括管理员登录、修改配置、查看告警监控等所有用户操作的日志。黑匣子日志用于业务和系统异常的故障定位。

系统可以生成各种统计报表和运行分析报告。

系统可对用户进行访问控制，支持用户组、分权、密码管理，便于维护团队内分职责共同有序地维护系统，还可设置密码策略，确保密码的保密性。例如：密码长度、密

码是否含特殊字符、密码有效时长等。系统可以初始配置和配置调整，配置的保存和备份，包括网络、告警、对接、业务等。

在软件管理方面，可以实现供应商发货前，将部分云操作系统软件已安装到服务器上。系统同时可以进行自动化批量安装，包括批量安装云操作系统软件、用户虚拟机软件和升级补丁等。云操作系统的安装支持统一安装界面，一次性导入所有服务器的信息，多节点同时加载安装，安装效率较高。用户虚拟机软件的安装可以通过镜像方式创建虚拟机并安装应用软件，且支持批量创建虚拟机，大大减少了用户操作和操作难度。系统可进行升级、打补丁及回退的自动化，包括云操作系统软件升级、打补丁有工具支撑，实现了自动化健康检查、分发软件、升级/打补丁、校验、回退，且支持静默升级，即升级/打补丁不影响业务。

在虚拟机运维管理方面，可以进行虚拟机生命周期管理，包括创建、删除和暂停。对虚拟机操作管理，包括启动、关闭、重启、自动休眠、迁移和查看虚拟资源的使用情况，虚拟资源包括虚拟计算、存储和网络资源。对虚拟机资源调整，包括 vCPU 个数、内存、网卡、磁盘挂载和卸载等。系统可以远程诊断虚拟机。

12.3.6 私有云安全管理

浙江农信私有云从接入安全、数据安全、网络安全、虚拟化安全、管理安全、基础设施安全和物理安全等方面采取严密的安全措施和策略，旨在进行端到端的安全防护。

1. 接入安全

为了防止恶意用户非法接入，私有云系统在用户接入、认证和授权方面进行了周密的防护，以保证只有经过认证和授权的用户才能访问其在云计算系统中的计算资源和存储空间，防止“危险”的终端或者用户对云造成损害。

2. 数据安全

在数据安全保护方面，私有云主要从用户数据隔离、数据访问控制和数据备份三方面着手进行保护。虚拟机所有的 I/O 操作都会截获处理，保证虚拟机只能访问分配给它的物理磁盘空间，从而实现不同虚拟机硬盘空间的安全隔离。

VM 删除或数据卷删除是指系统进行卷（Volume）资源回收时，小数据块链表将被释放，进入资源池。存储资源重新利用时，再重新组织小数据块，这样从新分配的虚拟磁盘恢复原来数据的可能性很小。数据中心的物理硬盘更换后，需要数据中心的系统管理员采用消磁或物理粉碎等措施保证数据彻底清除。

数据存储采用多重备份机制，每一份数据都可以有一个或者多个备份，即使存储载体（如硬盘）出现了故障，也不会引起数据的丢失，同时也不会影响系统的正常使用。系统对存储数据按位或字节的方式进行数据校验，并把数据校验信息均匀的分散到阵列的各个磁盘上；阵列的磁盘上既有数据，也有数据校验信息，但数据块和对应的校验信息存储于不同的磁盘上，当某个数据盘被损坏后，系统可以根据同一带区的其他数据块和对应的校验信息来重构损坏的数据。

3. 网络安全

在网络安全防护方面，私有云主要从传输加密、平面隔离和设置 VLAN 及安全组的方式着手。管理员访问管理系统，均采用 HTTPS 方式，传输通道采用 SSL 加密。网络通信平面划分为业务平面、存储平面和管理平面，且三个平面之间是隔离的。保证最终用户不能破坏基础平台，管理员不能侵入用户虚拟机。通过虚拟网桥实现虚拟交换功能，虚拟网桥支持 VLAN 标签功能，实现 VLAN 隔离，确保虚拟机之间的安全隔离。处于不同物理服务器上的虚拟机通过 VLAN 技术可以划分在同一个局域网内，同一个服务器上的同一个 VLAN 内的虚拟机之间通过虚拟交换机进行通信，而不同服务器上的同一 VLAN 内的虚拟机之间通过交换机进行通信，确保不同局域网的虚拟机之间的网络是隔离的，不能进行数据交换。

4. 虚拟机安全

虚拟机安全组通过 VLAN 和防火墙规则实现是一组虚拟机的集合，达到在一个物理网络中，划分出相互隔离的逻辑虚拟局域网，提高网络安全性。

虚拟化安全措施上，私有云对同一物理机上不同虚拟机之间的资源隔离，避免虚拟机之间的数据窃取或恶意攻击，保证虚拟机的资源使用不受周边虚拟机的影响。终端用户使用虚拟机时，仅能访问属于自己的虚拟机的资源（如硬件、软件和数据），不能访问其他虚拟机的资源，保证虚拟机隔离安全。

5. 管理安全

在管理安全方面，私有云主要采用，对系统日志进行监控分析和审计、管理员分权分域、设定管理员支持设置密码策略等方式，确保密码的保密性。

6. 基础设施安全

在基础设施安全方面，私有云系统分别进行了操作系统裁剪、加固和数据库裁剪、加固的方式增强云基础设施安全。操作系统安装时只安装满足业务需求的“最小化操作系统”，同时对操作系统进行安全配置。

操作系统安全配置包括账号口令安全配置、系统服务安全配置、文件及目录权限的设置等。操作系统和数据库程序数据文件安装在不同 NTFS 分区上，在非系统卷上安装数据库程序和文件。

采用最小化安装原则：只安装业务需要的组件，不安装如升级工具、开发工具、代码示例、联机丛书等不必要组件。限制客户端计算机连接到数据库服务器所能够使用的协议的范围，并确保这些协议的安全性，如限制只使用 TCP/IP 协议。限制客户端计算机连接到数据库服务器所使用的特定端口，不使用默认端口。登录用户设置强账号密码，并定期修改。

采用最小授权原则，以用户组为单位来分配权限。

12.3.7 私有云可靠性设计

浙江农信私有云系统在可靠性设计方面，重点考虑在大规模互联网业务运营环境

下，私有云系统能够实现自动化容错，通过高可靠性私有云系统设计，保证业务的连续性。系统引入虚拟化在可靠性方面的优势，在业务、硬件、软件等层面，提供了强大的可靠性保障，以满足互联网金融业务可靠性需求。按照架构分层，在业务层面实现多数据中心异地业务容灾、快照、备份与恢复等。在平台层面实现资源动态调度、虚拟机高可用（HA）、负载均衡、流控、数据一致性、存储多路径访问等。在硬件层面实现服务器电源、风扇冗余备份、内存软失效保护、网络双平面、硬盘故障预警、热插拔等。在管理层面实现告警管理、日志管理、配置管理、故障管理、升级不中断业务管理以及在线无损扩容等。

1. 计算与存储集群分离

私有云平台采用计算集群和存储集群相分离的架构，极大地提升了系统的可靠性。计算集群完成对虚拟机的按需分配，并支持虚拟机在集群内的热迁移；虚拟块存储集群完成系统卷和用户卷的按需分配，并支持卷数据的跨物理服务器存放。为了满足卷 I/O 操作通信带宽和时延的要求，计算集群一般使用本数据中心的虚拟块存储集群，与此同时，两集群之间通过采用网络存储多路径方式保证卷访问通信的可靠性。

2. 管理节点可靠性增强

私有云平台业务管理节点采用高带宽的心跳线连通。备用节点实时检测主用节点的健康状态，一旦发现主用管理节点故障，备用管理节点将立刻接管主用节点的任务，持续对外提供服务。

针对业务管理节点上的应用进程，通过采用软件狗的方式对运行在管理节点上的进程进行实时监控。如果发现进程吊死或进入死循环，软件狗将会检测到相关进程的异常状态，并触发相关进程的重启恢复。如果发现进程重启后仍不能恢复正常，则进行业务管理节点的主备倒换以保证应用进程的可靠性。

管理节点负责对全系统的业务进行管理，采用主备高可靠性的工作方式，如果主备管理节点同时故障的时候，相关的业务会受影响，例如虚拟机的创建和删除等，但是，对于已经存在并运行中的虚拟机，不会产生任何影响，也就是说即使主备管理节点同时故障，也不会对正在使用的虚拟机产生影响，用户继续使用虚拟机上的应用程序，不会有任何感知。

为了提供稳定的高可用的互联网金融并发业务和避免大流量冲击导致系统崩溃，管理节点针对系统关键流程设计了完善的流量控制机制。首先在网卡接入点采用操作流控措施，从前端抑制系统过载，保证系统的稳定性。其次是针对系统内部的瓶颈环节，增加了鉴权、虚拟机相关业务流控（包括虚拟机迁移、虚拟机 HA、虚拟机的创建、虚拟机的休眠和唤醒、启动和停止），O&M 流控，确保各个环节不因为流量过载导致业务失效。

3. 故障预防机制

私有云系统提供了故障检测和告警的功能，同时它包括了在 Web 浏览器中显示故障信息的工具。一旦集群进入正常状态，系统提供使用数据可视化工具观察集群管理和分配负载的功能，可以帮助用户确定是否有负载均衡问题、失控进程或硬件性能下降的

趋势，将对合理调整、分配系统资源，提高系统整体性能起到重要作用。历史记录允许查看集群每日的、每周的，甚至是每年消耗的硬件资源。通过在每个被监控的节点包括定制化的虚拟机上运行探针程序，OM 系统可以收集被监控节点或者虚拟机的核心指标如 CPU 使用情况、基础网络流量和内存数据等，检测到诸如进程崩溃、管理和存储链路异常，节点宕机、系统资源过载等各种异常，使系统具备完善的故障检测能力。另外，私有云系统中提供了健康检查工具，为技术支持工程师和维护工程师提供的一套日常检查工具，并能输出各部件健康检查报告，方便技术支持工程师和维护工程师快速了解系统的健康状况。通过检查系统当前信息和运行状态，反映系统健康或亚健康状态，在开局、巡检、升级等维护场景中使用。目前健康检查工具可以检测的信息，包括关键配置文件检查（例如软件狗配置文件、双机备份配置文件）、进程检查（例如重要进程内存占用、狗进程状态）、虚拟化信息检查（例如虚拟机 Pvdriver 是否安装检查等）、备份检查（例如检查备份机制是否有效）、软硬件信息检查（例如软件版本号、BIOS 信息、物理节点 CPU 信息等）。

系统提供了数据一致性审计，除了系统本身针对关键资源提供的自审计和恢复能力之外，还支持定时审计虚拟机，卷的相关数据和状态的一致性，发现有异常，会自动记录并针对记录情况提供操作指导，以便维护人员做相应的判断和恢复措施，从而保证系统内部各种相互关联数据的一致性。

4. 灾备

系统提供配置数据和业务数据定期本地和异地备份能力，当管理节点服务异常无法自动修复时，通过本地备份的数据立即恢复；当由于灾难性的故障导致管理节点双点同时故障且不能通过重启等操作进行恢复，可使用异地备份数据立即恢复（1h 之内完成），减少故障恢复时间。

私有云系统内部提供了时钟同步功能，可以保证所有网元（IPSAN、交换机、管理节点、计算节点、服务器硬件管理口、防火墙等）时间一致，还支持外接便准时钟源设备，可以保证全局时间统一且精准，方便系统维护以及各个网元的正常消息交互。

5. 云平台故障处理机制

私有云的虚拟化系统可靠性，除了支持通用的虚拟机热迁移、虚拟机负载均衡、虚拟机 HA 外，还支持虚拟机故障隔离，虚拟机之间彼此相互独立，一个虚拟机故障不会影响其他虚拟机。用户对虚拟机的使用体验和对传统物理机的体验相同，因此在一个虚拟机内的任何操作，不对同一台物理服务器上的其他虚拟机和虚拟化平台自身的可用性产生危害。即使虚拟机的运行出现故障，比如操作系统崩溃、应用程序错误导致死机等情况，同一物理服务器上的虚拟化平台以及其他虚拟机仍然可以正常运行，继续为用户提供服务。

当虚拟机本身发生故障或者虚拟机所在物理服务器发生故障导致虚拟机故障时，系统能够根据用户预先设置的故障处理策略，决定在本地或异地重新启动虚拟机，以尽快恢复业务的运行。用户也可以设置为虚拟机发生故障后不作处理，在这种故障处理策略下，系统即使检测到虚拟机发生故障，也不会做处理。对于虚拟机 OS 内部故障，如

Windows 虚拟机的蓝屏故障，或 Linux 虚拟机的 Panic 状态，这类故障系统都能检测到并处理。通过增强系统的自动化维护手段，减少了维护人力投入，最大限度地减少虚拟机业务中断时间，缩短了平均故障恢复时间，提升系统可靠性。

私有云的虚拟化软件和虚拟化管理软件通过黑匣子功能，在管理节点或者计算节点出现系统崩溃、进程死锁或异常复位故障时，会将“临死信息”备份到本地目录，用于后续故障定位。

黑匣子主要用于管理节点和计算节点上收集并存储操作系统异常退出前的内核日志、诊断工具的诊断信息等数据，以便操作系统出现死机后，系统维护人员能将黑匣子功能保存的数据导出分析。为了让这些系统定位数据不丢失，黑匣子支持把操作系统死机前收集的数据通过 Netpoll 方式实时发送至远端服务器进行备份，如果网络异常则会保存在本地。

6. 网络可靠性增强

私有云网络子系统主要采取以下措施来增强系统的可靠性：网络路径全冗余；网络分平面避免不同消息相互干扰；通过网卡绑定技术提高了服务器端口的可用性；通过交换机堆叠技术将两台交换机虚拟成一台使用在提高链路的利用效率的同时大大提高了列中交换机的可靠性；通过 Trunk 后的 SmartLink 技术接入汇聚交换机；在核心路由器侧，采用 VRRP 技术部署主备两台路由器以便提高网络核心部分的可用性。

网络路径全冗余，按照层次网络分为核心层、汇聚层、接入层和虚拟网络层。当系统规模较小时，核心层和汇聚层可合一。核心层交换设备主要完成数据中心之间的通信互联，同时提供数据中心对外网络出口。核心交换机对外与防火墙/NAT 连接，对内与汇聚交换机或列中交换机连接。通过使用交换机集群，保证了核心层的网络连接冗余。汇聚层交换设备位于各个数据中心机房内部，完成本数据中心内各列中交换机的流量汇聚，对外与核心层交换机通过三层互通，同时对列中交换机提供二层接入功能。通过使用交换机集群，保证了对外与核心层交换设备和数据中心内列中交换机连接的冗余。列中交换机位于机柜内部，负责本机柜内部的服务器接入。通过使用交换机堆叠，保证了对外与汇聚层交换设备和对内虚拟网络层连接的冗余。虚拟网络层位于服务器内部，负责服务器内部的虚拟机之间以及对外通信功能。通过采用多网卡绑定，避免单个网卡故障引发的业务中断。

整个私有云系统网络逻辑上分为三个平面：管理平面、存储平面和业务平面。为了保证各种网络平面数据的可靠性，采用分网络平面的架构方案，不同平面间采用 VLAN 进行隔离，单个平面的故障不影响其余的两个平面继续工作。例如当管理平面暂时故障时，业务平面还能够继续为云终端用户提供服务。

此外，系统还支持基于 VLAN 的优先级设定，使得内部的管理/控制报文具备最高的权限，从而使得在任何时候，管理员和用户均可以管控系统。在服务器内部，通过对多个网卡的合理绑定和分类，允许将管理、业务和存储平面部署在不同物理网卡上，并将其连接到不同的接入层交换设备接口上，从而实现物理层面的网络隔离。

对于物理服务器提供的多块网卡，出于可靠性以及流量负载均衡的考虑，系统采用

绑定模式，多块物理网卡被绑定成逻辑上的“一块网卡”后，同步一起工作，对服务器的访问流量被均衡分担到多块网卡上，这样每块网卡的负载压力被大幅减轻，抗并发访问的能力提高，保证了服务器访问的稳定和畅快。当其中一块网卡发生故障的时候，另外的网卡立刻接管全部负载，过程是无缝的，服务不会中断，避免单个网卡或者链路故障引发的业务中断。服务器绑定多网卡的实际意义在于当系统采用绑定多网卡形成阵列之后，不仅可以扩大服务器网络进出口带宽，而且可以实现有效负载均衡和提高容错能力，避免服务器出现传输瓶颈或者因某块网卡故障而停止服务。

7. 硬件可靠性增强

计算节点支持存储模块的冗余部署，其上虚拟机通过标准协议访问存储系统，并通过多块网卡的负荷分担技术、交换机的堆叠和集群技术提供存储路径的物理冗余。任意一个虚拟机对所挂载的任意一个虚拟卷，都将至少有两个完全冗余的路径来实现卷的多路径访问，并通过多路径软件来实现访问多路径的控制和故障切换，从而避免单点故障带来的系统可靠性问题。

私有云通过采用高可靠性硬件来降低系统整体故障风险，并减轻运维人员的工作负担。服务器在内存软件错误纠正上采用内存 ECC（Error Checking and Correction）技术，采用工业标准的纠错算法，能够检测内存 2bit 错误，并修复内存单比特错误。硬盘采用热插拔硬盘，IPSAN 磁盘框支持系统运行时的硬盘（SAS/SATA）热插拔。对于硬盘 RAID，在 IPSAN 模式下支持 RAID 0、RAID1、RAID5、RAID6、RAID10、RAID50、RAID60，支持 RAID 下另加热备盘的配置，保证了硬盘数据的高可靠性，在 RAID 组的某块硬盘坏掉后，支持数据恢复、RAID 组恢复和在线更换硬盘。IPSAN 控制框可以对 Cache 数据进行保护，既可以提高对硬盘的访问性能，又可以防止意外掉电时数据的丢失。私有云的服务器和存储采用了 SMART（SMART 是 Self-Monitoring Analysis and Reporting Technology，自监测、分析和报告技术）标准来实现对基于 ATA 和 iSCSI 接口的硬盘进行监控和可靠性管理，检查其可靠性并预测磁盘错误。技术原理主要是通过侦测硬盘各属性，如数据吞吐性能、马达起动时间、寻道错误率等属性值和标准值进行比较分析，推断硬盘的故障情况并给出提示信息，帮助用户避免数据损失。私有云服务器配置 2 组电源（PSU），提供电源故障告警，支持电源 1 +1 冗余和热插拔，可以在 1 组电源故障后，系统持续运行而不影响业务；并且可以在线更换故障电源。服务器支持对 CPU、内存等热关键器件的温度实时监控，配合智能的风扇调速和监控，确保系统运行的可靠性。系统可以对风扇、电源、硬盘等关键器件的运行状态监控，设备故障时会产生告警，可以灵活对支持热插拔设备进行在线更换，不支持热插拔设备提前安排好业务后进行下电更换。

12.4 私有云效益

浙江农信建设了以“开放、高效、安全、融合”为特征的私有云，构造了以 Open-

Stack 为架构、性能卓越、便于扩容、端到端的安全防护和自动化容错的基础设施平台。有效支撑了浙江农信的互联网金融业务，包括电商、O2O、综合电子账户、大数据等。系统实现了统一模型、统一框架、数据集中、数据共享、后端解耦、前端融合的建设目标。浙江农信私有云所承载的综合电子账户平台将成为浙江农信互联网金融交互平台的核心，为实现以“客户为中心”服务战略打下了良好基础。

银行业信息科技风险管理高层指导委员会简介

银行业信息科技风险管理高层指导委员会（简称高层指导委员会）是由银监会发起，20家主要银行业金融机构自愿参与建立的行业性、专业性高层组织。高层指导委员会的宗旨是：以全面风险管理为导向，提升银行业信息科技核心竞争力和自主创新能力，提升银行业信息化建设和信息科技风险管理整体水平，推动银行业信息科技持续、健康发展，维护金融稳定和国家安全。高层指导委员会的主要任务是对银行业信息化建设与信息科技风险管理工作进行研究、指导并提供咨询、建议，研究银行业信息化建设重大发展问题，深入传导贯彻信息科技监管政策，开展专业指导和风险分析，开展信息科技课题研究，推动银行业信息科技领域新兴技术研究，促进银行业信息科技领域的交流合作。高层指导委员会自2011年成立以来，建立了风险分析、课题研究、专业指导等常态化工作机制，编制了《金融科技治理与研究》期刊，组织开展了近300项课题研究，组织银行业金融机构协同开展安全可控、自主创新能力建设，为银行业信息科技领域的经验交流、知识分享和资源互补提供了有效的平台。

银行业信息科技风险管理高层指导委员会
银行业信息化丛书

银行数据治理

商业银行信息系统研发风险管控

全球化时代的银行信息系统建设

■ **商业银行私有云设计与实现**

银行信息系统架构

银行信息安全技术与管理体系

商业银行业务连续性管理

金融数据挖掘与分析

银行数据中心基础设施建设与运维管理

银行业信息科技监管

地址：北京市百万庄大街22号
邮政编码：100037
电话服务
服务咨询热线：010-88361066
读者购书热线：010-68326294
010-88379203
网络服务
机工官网：www.cmpbook.com
机工官博：weibo.com/cmp1952
金书网：www.golden-book.com
教育服务网：www.cmpedu.com
封面无防伪标均为盗版

机械工业出版社微信公众号

上架指导 金融

ISBN 978-7-111-52725-1

ISBN 978-7-111-52725-1

定价：79.80元