HZ BOOKS
华章科技

云计算技术系列丛书

云计算架构

解决方案设计手册

Cloud Computing Architected

Solution Design Handbook

（美）John Rhoton
Risto Haukioja 著
赵龙刚 金振林 等译

宏观上透彻地讲解云计算系统架构与设计的策略、流程和方法

微观上深入阐述云计算的模型、主流平台、安全性、集成技术、数据存储、可伸缩性和可靠性、部署、运行维护，以及营收和推广

机械工业出版社
China Machine Press

云计算技术系列丛书

云计算架构

解决方案设计手册

Cloud Computing Architected Solution Design Handbook

（美）John Rhoton
Risto Haukioja 著
赵龙刚 金振林 等译

机械工业出版社
China Machine Press

前　言

本书描述了云计算应用的基本组件，介绍了一些可用来创建跨管理域的大规模、分布式应用的架构级选项。

云计算的需求对于软件工程有着深远的影响。在构建云应用时，我们必须保证该应用能够提供灵活且有弹性的服务，而且要针对能够远程使用由各种可靠和不可靠来源所提供的各项功能而进行设计。

在面对这些挑战时，架构师需要在如下几个领域中考虑可伸缩性和多租户的影响：

- 新工具可以使开发人员更有效地开发、测试和运行云服务。
- 基于互联网的交付会促使人们使用浏览器及其他一些精简客户端来作为展示工具。
- 多租户和身份联邦迫使我们要使用一些新的身份认证模型。
- 这是一个由众多迥然不同的服务及服务提供商所组成的碎片化的市场，除非将这些服务有效地集成在一起，否则它们无法一起协调地完成工作。
- 与云规模相关的信息量的爆炸式增长要求我们使用一些新的存储机制和数据模型。
- 可用性和伸缩弹性越来越依靠冗余，必须以一种既高效又可靠的方式来对冗余进行编排。
- 面向效用的服务需要使用新的商业模式和盈利策略才能获得收益。
- 平台服务能够给整个软件开发生命周期带来效率，使我们可以更快地完成部署和采纳变更。
- 自动化程度的提高使我们可以采用一种更高效的运作模式，这种运作模式本身需要从以基础设施为导向转变为以服务为导向。

本书所关注的就是上述这些（和其他一些）可能会受到云计算影响的应用软件架构领域。

假设

本书作者的理念基于这样一个大胆的假设：未来的大多数（即使不是全部）应用都应设计为一种云就绪（cloud-ready）的应用。当然，并非所有软件都会从一开始就将自己的目标定位于那些必须具有互联网级的规模和多租户支持的机会。然而，设计一个云就绪解决方案的价值在于：它可以通过不断成长来满足日益增加的需求，并且还能够挖掘到最初可能没有规划的一些潜在收入流。

促使我们设计一个云就绪应用的因素（即使这种需要可能没有立即显现出来）来自以下三方面的考虑：

- 坚实可靠的设计有内在的价值。
- 伸缩弹性是在当今快速变化的商业环境中必不可少的一个条件。
- 灵活的知识产权才最有价值。

坚实可靠的设计有内在的价值。面向服务的架构（SOA）使得服务在其整个生命周期都具有更大的灵活性，从而使得服务可以应对一组处在不断变化中的需求。如果能够从一开始就在软件中置入可靠性、安全性和可伸缩性，那么该软件就会健壮很多，并且在处理突发事件和新的要求时也更加容易。

伸缩弹性是在当今快速变化的商业环境中必不可少的一个条件。客户或销售的增长不是你可能需要扩大服务的唯一原因。一个可伸缩的架构可以通过合并和购买的方式来应付需求的瞬时高峰。一个灵活的设计可以使我们能够将应用复用于其他的商业目的和用户群（如合作伙伴和供应商）。

灵活的知识产权才最有价值。有这样一个令人信服的说法：即便在构建一个仅供内部使用的应用时，我们也应该在设计架构时考虑好可伸缩性和多租户的问题。如果一项服务只是重复实现了一些公共的功能，那么你就应该质疑自己：为什么在第一个地方需要这一服务，为什么不能从另一个提供商那里获得相同的服务。反之，如果某项服务具有独特性并且具有自主知识产权，那么你随时都可以选择将服务以授权许可或打包成一

个服务的方式提供给别人，从而创造出一种额外的收入来源。

范围

本书的写作目的既不是对云计算进行高层次的概述，也不是对云计算的商业利益进行详细说明。在市场上有许多讲述这类主题的书籍，比如《Cloud Computing Explained》[⊖]和“参考文献”中所列的其他一些书籍。

作者也没有深入论述任何一种具体平台或服务。因此，寄希望于仅参考本书就能设计出一个基于云计算的解决方案是一个不现实的想法。不过，各服务提供商一般都会为自己的产品提供一份内容丰富的文档，而且本书也提供了另外一些优秀资源的链接，它们可以在技术和技巧两方面提供很多具体的建议。

相反，本书弥补了企业管理层与程序员之间在视角上的差异。本书讲述了各种可能与云计算相关的架构级选项，但本书论述的深度仅足够评估设计是否适当，并不以详细说明实现细节为目标。

读者

正如前面所暗示的那样，本书的主要读者是那些咨询师、架构师、技术专家和策略师，他们可能在大型企业参与 IT 实现的规划工作或者在参与规模须可扩展到数百万用户的、面向消费者的服务的设计工作。

本书的第二类读者是那些仅在外围参与上述设计工作的人员。无论是系统管理员、程序员还是高层管理人员，几乎每个 IT 人员最终都与云计算有着某种联系。每一个希望对云软件架构有透彻理解的人都能从本书关于这些选项和设计权衡的概念性描述中受益。

⊖ 本书中文版《云计算：企业实施手册》已由机械工业出版社引进出版，ISBN：978-7-111-35177-1。

反馈

你可能会从本书中发现一些错误和遗漏之处，也可能会发现一些真的非常有用和令你感兴趣的内容。无论你要反馈什么样的信息，只要您有话要说，我们就很乐意倾听您的意见。请随时与我们联系，邮件请发至：

john.rhoton@gmail.com 或 risto.haukioja@gmail.com

我们不能保证会回复每一封邮件，但我们会竭尽所能，以感谢您的付出！

致谢

本书（英文版）由 Lightning Source（Ingram Content Group）出版和发行，该公司的生产过程非常高效和敏捷，令我们非常赞赏。此外，我们还得到了 Gill Shaw 的大力帮助，他在校对和编辑方面为我们提供了卓越的帮助。我们还要感谢 Elisabeth Rinaldin，她为本书的封面和版式设计做出了杰出贡献。

我们非常感谢评审者为本书所做出的技术上的宝贵付出，他们是：John Bair、Jamieson Becker、Patrick Joubert、Kevin Laahs、Franz Novak 和 Vikram S。我们要向本书最后所列出的诸多资源喝彩，当我们要深入了解本书所介绍的那些话题的细节时，它们会给我们带来极大的帮助。最后但并非最不重要的是，我们想指出：如果没有那些在 Amazon、Google、Microsoft、Salesforce.com 以及其他云服务提供商处工作的富有远见和创造力的工程师，也不会有本书所要介绍的这些内容。

目　录

第六部分 信息

第七部分 弹性

第一部分
模　　型

第一部分将讲述我们对云计算的看法以及它对软件开发人员和架构师的含义。首先，我们将从对一些与基于云的交付和云服务架构相关的主要理论概念进行定义和概述开始。

其次，我们会定义并阐明一个高层次的参考架构，并以此作为本书其余部分的结构基础。

第 1 章　云计算定义

最近几年，关于云计算的各种声音有很多。时至今日，你可能已经很清楚“云计算”一词的含义了，甚至对于它的工作原理你也很清楚了。但是，如果你要负责设计系统或应用，那么仅仅对这一 IT 新趋势进行理论上分析还远远不够。

相反，你需要了解该如何利用这些新的选项来构建一个更大、更好且更便宜的解决方案。本书将探讨的内容包括如何利用这些新工具、如何实现伸缩弹性、如何集成不同的云服务、如何为海量数据建模、如何使运营更有效率，以及如何充分发挥新商业模式的作用。

本书有意识地没有花费过多的篇幅来对云计算进行基础性的概述。如果你对如何从面向业务的视角来看待在企业中实施云计算的优势和挑战感兴趣，那么你可以从阅读《Cloud Computing Explained》一书或本书“参考文献”所列的其他一些介绍性的书籍开始。本书假定你已很好地掌握了那些基础知识，并已准备好进入一个更高阶的水平了。

尽管如此，我们还是希望能够确保大家在同一个概念基础上探讨云计算。你对云计算的认识可能完全正确，但同时又可能与我们的认识有着显著的差异。为了尽量避免混淆、确保使用同一套框架和术语，我们仍需简单地刻画一下云计算对于我们来说意味着什么，并且描述构成云计算基础设施及解决方案的那些主要服务和交付模型。

1.1　云属性

简单地说，云代表一种网络，更具体地说，就是全球的互联网络。云计算，言下之意就是：使用远程托管并通过互联网进行交付的计算资源。这就是该术语所蕴含的基本理念。对于你的那些不搞技术的朋友和同事来说，这样理解可能就足够了，但对于本书的读者来说，这还远远不够。

如果你曾试图通过寻找一个权威定义来分离出“云计算”的核心意义，那么你很快就会发现这一术语蕴含着许多不同的概念。对于云计算这一技术上的根本性转变有哪些构成要素，专家们也有不同的意见。有些专家能够将自己的观点表达得比别人的观点更优雅，但这并不意味着他们的观点得到了大家普遍接受。

目前人们普遍接受的定义是由 NIST（美国国家标准技术研究所）所阐述的（2009）：

> 云计算是一种模型，人们可以使用它来方便地按需通过网络访问一个可配置的计算资源（如网络、服务器、存储、应用和服务等）的共享池，只需最小化的管理工作量或服务提供商干预就可以快速地开通和释放资源。

然而，无论是 NIST 的这一公式化定义，还是上述关于云是何意的解释，都没有得到所有人的认可。要理解这一术语的各种常见解释，最有实效的方法是研究一下各种典型云解决方案的属性分类。这并不意味着云计算的每一项云属性都是必不可少的，也不意味着只要将这些属性任意组合在一起，就能形成一个符合云计算思想的方案。对于云计算的概念来说，这些属性本身既不是必要条件，也不是充分条件。然而，这些属性在某个实现上应用得越多，其他人就越有可能将该实现认可为一个云解决方案。

我们非正式地调查了关于此话题的博客、微博以及公开发表的一些文献，从中发现了如下一些关键部件：

外部部署（off-premise）：服务被托管在一个所有权归服务提供商所有的地点，并从那里提供给用户使用。这通常有两层含义：一是该服务要通过公共互联网来提供；二是其处理过程要在公司防火墙之外进行。换句话说，该服务必须同时跨越物理和安全这两道边界。

伸缩弹性（elasticity）：云计算的主要好处之一是服务提供商所固有的、可以提供给最终用户的可伸缩能力。云计算模型在提供弹性开通机制方面进行得越深入，在需要资源时，就可以越快地扩展或者收缩资源。因为资源通常都是按照效用计费的，所以这种伸缩弹性可以等同于节约了直接成本。

灵活的计费方式：将对资源使用情况进行细粒度的计量与按需开通服务相结合，可以很容易地得出多种客户计费的可选方案。可以订阅为基础收取费用，也可根据实际使用（或预留）资源的数量来收取费用。除了制定详细的合同和采用集中计费之外，还可以通过放置广告的方式或依赖简单的信用卡收费来获利。

虚拟化：云服务通常要通过一个抽象化的基础设施来提供。它们会充分利用各种虚拟化机制，通过多租户来实现成本的优化。

服务交付：云功能往往要以某种服务的形式来供用户使用。虽然这些服务

在本质上有很大的差异，但它们通常除了具有用户界面之外还提供一些可编程访问的接口。

通用访问：资源民主化（resource democratization）意味着将资源放入一个资源池中来供那些经过授权的用户访问。与此同时，地点无关性和高水平的弹性可以使用户获得一种始终连接的用户体验。

管理简化：自动配置可以满足可伸缩性的要求，用户自助服务可以加快业务流程，通过编程方式可访问的资源便于实现与企业管理框架集成。这些都简化了系统的管理工作。

资源价格合理：资源成本大大降低，原因有二。一是不需要固定采购的资本支出。二是服务提供商的规模经济效应使他们能够利用商品化的硬件和大多数公司所难以匹敌的优化运作流程来优化成本结构。

多租户（multi-tenancy）：云要被多家组织（即租户）所使用，并且也包含一系列的机制来保护每个租户并使其与其他租户隔离。让客户通过一个共享池来访问资源是实现可伸缩性和节约成本的重要因素。

服务水平管理：云服务通常会提供对于服务水平的定义，该定义设定了客户对于该服务的健壮度的预期。有些服务可能只带有一些最低限度的承诺（甚至也可能根本没有承诺）。虽然我们仍可以把它们视作云服务，但与其他那些使用了更加精确的承诺进行监管的服务相比，我们通常不会“信任”这样的服务，不会将它们用于那些关键任务的应用。

1.2 云服务

云计算的一个很突出的倾向就是它对于面向服务的强烈关注。我们经常听到“某某（或一切）即服务”（XaaS）这样一类说法。换句话说，云不是某种单一产品，它通常是一些异构服务的一种零碎合并。

你不能仅仅提供一些可整体安装在台式机和服务器上的打包解决方案，或者投资建设一些单一用途的应用，你需要做的是：将用户需要的所有功能分解成为一些可根据需要组装在一起的原语（primitive）。

遗憾的是，除非你对所有可用的服务都了解得非常清楚，否则很难找到一种可以将这些功能聚合在一起的最佳方式。如果你能使用一些结构和一个模型来阐明各种服务之间的相互关系，那这件事就容易做多了。

最常见的分类方法使用的是一种名为 SPI 的模型（NIST，2009），它把服务分成 SaaS、PaaS 和 IaaS 三类，即软件即服务（Software as a Service）、平台即服务（Platform as a Service）和基础设施即服务（Infrastructure as a Service）。Amazon Elastic Compute Cloud（EC2，亚马逊弹性计算云）是 IaaS 的一个经典范例。通常认为 Google App Engine（Google 应用引擎）是一种 PaaS。而 Salesforce.com 则是我们最熟知的 SaaS 的一个例子。

这三种方法向使用者提供了不同程度的共享。基础设施服务共享的是物理硬件。平台服务则还允许租户共享同一个操作系统和应用框架。而软件服务则一般会共享整个软件栈。如图 1-1 所示，这三种方法代表了在对优化和灵活性进行平衡时所做出的不同取舍。在优化方面，需要充分利用多租户和大规模的可伸缩性；而在灵活性方面，则需要能够融合各种不同的约束和自定义的功能。

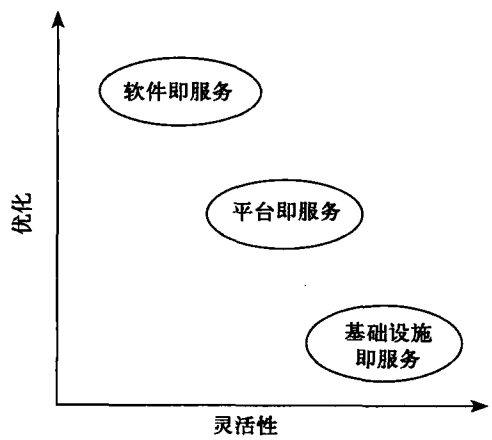

图 1-1　软件、平台和基础设施服务

SPI 模型是一种简单的分类法，它能帮助我们对那些主要的云相关服务建立起一个初步的印象。然而，与其他分类系统经常发生的情况一样，它们实际上的界线并非像图中所示的那样明确。很难将服务准确地归为哪种服务类型。随着时间推移，服务还会在不同服务类型之间漂移。例如 Amazon 公司，该公司一直在增强其 AWS（Amazon