

微软技术丛书

Microsoft

Windows Azure 从入门到精通

Windows Azure Step by Step

(意) Roberto Brunetti 著
龙吟晖 译

Step by Step



清华大学出版社

微软技术丛书

Microsoft

Windows Azure 从入门到精通

Windows Azure Step by Step

(意) Roberto Brunetti 著
龙吟晖 译

Step by Step

清华大学出版社
北 京

内 容 简 介

本书针对微软的云计算平台 Windows Azure, 循序渐进地介绍了如何构建和管理云端的可扩展应用, 一次一个知识点, 同时辅之以适当的练习, 可帮助读者轻松掌握基本的编程技能, 掌握 Windows Azure 云计算平台的核心服务和特性, 是一本理想的入门教程。

本书实用性强, 操作性强, 适合对 Windows Azure 感兴趣的所有读者阅读。

Windows Azure Step by Step by Roberto Brunetti(978-0-7356-4972-9)

Copyright © 2011 by Roberto Brunetti

Original English Language Edition Copyright © 2011 by Roberto Brunetti.

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U.S.A.

本书中文简体版由 Microsoft Press 授权清华大学出版社出版发行, 未经出版者书面许可, 不得以任何方式复制或抄袭本书的任何部分。

北京市版权局著作权合同登记号 图字: 01-2011-5335

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

Windows Azure 从入门到精通/(意)布鲁内蒂(Brunetti, R.)著; 龙吟晖译. --北京: 清华大学出版社, 2012. 1
(微软技术丛书)

书名原文: Windows Azure Step by Step

ISBN 978-7-302-27825-2

I. ①W… II. ①布… ②龙… III. ①计算机网络 IV. ①TP393

中国版本图书馆 CIP 数据核字(2011)第 277283 号

责任编辑: 文开琪

封面设计: 杨玉兰

责任校对: 周剑云

责任印制: 杨 艳

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京密云胶印厂

经 销: 全国新华书店

开 本: 185×260 印 张: 15.75 字 数: 372 千字

版 次: 2012 年 1 月第 1 版 印 次: 2012 年 1 月第 1 次印刷

印 数: 1~4000

定 价: 39.00 元

产品编号: 044042-01



序

Roberto 邀请我们为这本书写序的时候，我们因为几个方面的原因而欣然应允。首先，向读者引荐 DevLeap 的会员作品是我们的荣幸。DevLeap 是一个专家组织，因提供高质量的教学材料和课程而有口皆碑。其次，本书讨论的主题是我们未来专业方向一个最重要的问题：云计算。具体说是 Windows Azure 平台，它是专为云计算而构思和建造的第一个应用程序基础结构。

网络技术的变革和 Internet 的迅速发展，在已经改变了人们的消费内容之后，正处在一个基本概念(比如计算能力和存储能力)也在发生重大改变的阶段。以前思考这些东西时，都是把它们同本地物理机器关联，但随着云计算的问世，访问计算和存储资源的方式也在发生变革。感谢 Windows Azure 平台，巨大的数据中心很容易提供计算能力、存储和服务。开发人员可利用这些资源开发解决方案，并把它们转换成服务供用户按需使用。

在一个按需使用的基础结构中提供服务，有助于节省管理和软硬件成本，还有助于降低许可证的价格。与此同时，可用的计算能力和存储容量也在不断提升。

在本书中，Roberto 将和我们一起畅游 Windows Azure 平台和服务的世界，详尽解释新平台主要组件的目标及功能，描述如何利用它来托管新服务和应用程序。从 Windows Azure 平台的核心到它的部署与监视服务，从 SQL Azure 到和开发工具的完全集成，本书以一个具体的、面向开发的角度指引你了解新技术的各个主要部分，提供了许多实际的例子，其主题思想和 DevLeap 不谋而合。书中介绍了有关服务使用的案例研究和实践技巧，还介绍了最高效的实现策略，这些都是使用 Windows Azure 平台时的最佳方式。掌握这些第一手的资料后，从一开始就可以轻松驾驭新技术，更好地挖掘计算、存储和基本服务的潜力。

下面就让我们开始一次全新的、有趣的新技术之旅。

Giuseppe Guerrasio

架构顾问

开发与平台技术部

微软(意大利)公司

Gabriele Castellani

开发与 IT 架构部门经理

微软(意大利)公司



致 谢

一本书的封面一般只写作者的名字。

首先感谢 Vanni Bencinelli 从技术和语言角度审阅全书。这是我第一个用英语写的书。没有他，编辑恐怕会拒看所有章节。

还要感谢 Russell Jones 迅速和准确地响应我的所有初级问题，而且从一开始就支持我。

非常感谢 Brian Keller，他不仅审阅了全书，而且每行代码都检查了两遍，因为在本书写作期间，微软发布了 SDK 1.3。在配置和 Visual Studio IDE 加载项方面，这个版本和上一版本有很大区别。Brian 还重新检查了所有门户截图，因为 2011 年初，Windows Azure 用户界面用 Silverlight 进行了全面重新设计。

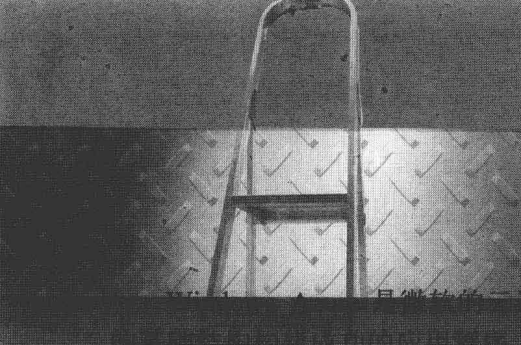
我想感谢的最重要的人是 Victoria Thulman，是他指导我编辑所有章节。对于我的许多错误和不正确的假设，他都非常耐心地予以指正。

Gabriele Castellani 和 Giuseppe Guerrasio 为本书写序。从 2008 年微软在 PDC 08 首次宣布 Windows Azure 起，他们就一直在支持我。

Fabio Santini 和 Mario Fontana 使我有机会和微软(意大利)公司的许多合作伙伴工作，并提供了大量应用场景，本书许多地方都反映了我的这些经历。

最后感谢 Paolo Pialorsi，他在一年前把我引荐给 Russell Jones。

现在，本书终于完成了，再次感谢你们所有人！



前言

Windows Azure 是微软的云计算平台，允许开发人员利用其强大的、可伸缩的和容错的架构构造成功的应用程序。

本书采取了良好的组织，指引你认识 Windows Azure 平台及其相关技术。这是一本入门性质的参考书。它通过许多简单的、你可以亲自实践的操作过程讨论已发布的每个组件及其相关理论。还提供了可下载的示例代码，可通过它们熟悉 Azure 平台，并以此为基础开始自行探索。

本书覆盖了写作时为止微软发布的每个 Windows Azure 平台组件，还讨论了一些相关的技术，比如 WCF Data Services、OData 和 ADO.NET Entity Framework。

本书面向的读者

本书目标是希望使用 Windows Azure 平台组件的 .NET 开发人员提供帮助。这些组件从操作系统到 SQL Azure，再到 Windows Azure AppFabric。如果很熟悉 .NET Framework，那么书中的示例代码很容易理解，使用 Visual Studio 完成本书的实战演练也很容易。软件架构师如果想对云解决方案总体架构中包含的组件有一个基本了解，也值得读一读本书。

本书不适合的读者

如果你已经在使用 Windows Azure 平台，本书也许不适合你。本书是介绍如何利用该平台开发应用程序的入门性读物。

本书前提条件

本书要求你至少对 .NET 开发和面向对象编程概念有一个基本了解。虽然 Windows Azure 能运行所有 .NET 语言平台和许多第三方(以及开源)“运行时”，但书中只提供了 C# 的例子。如果还不熟悉 C#，推荐阅读 John Sharp 的《Visual C# 2010 从入门到精通》(John Sharp 著，周靖译，清华大学出版社，2010)。

Web Role 示例假定读者事先已基本了解了 ASP.NET Web Forms 技术，尽管该示例代码未用到任何高级的 ASP.NET 功能。

本书的组织

本书共 11 章，每章都着眼于 Windows Azure 平台的一个不同的方面或者技术。

- 第 1 章概述云计算技术以及 Windows Azure。
- 第 2 章分析 Windows Azure 平台的各个组件，指导读者在 Windows Azure 管理门



户中使用这些组件。

- 第3章介绍托管服务、实例、虚拟机和角色。
- 第4章创建简单的应用程序，利用存储帐户来存储和获取二进制大文件(blob)。
- 第5章的重点还是存储帐户，解释表格和队列的主要概念，介绍工作者角色。
- 第6章深入讨论一些读者关心的问题，比如记账、安全、管理证书和地缘组。
- 第7章讲解 Windows Azure AppFabric，这是在 Windows Azure 操作系统顶部构建的组件。
- 第8章讨论 WCF 数据服务，指导读者创建实体数据模型，并在 Astoria 项目中通过 REST 和 OData 协议公开这个模型。
- 第9章是 SQL Azure 专题，它是 SQL Server 的云版本。
- 第10章讲解如何从本机应用程序和不同的编程环境中使用 Windows Azure 服务。
- 第11章讲解应用程序架构。

本书约定

本书采用以下约定使文字更容易阅读和理解。

- 书中大多数实战演练都是用 Visual C#写的。示例代码和操作步骤故意设计得很简单，即使你是 Visual Basic 程序员，也能通过学习 C#示例轻松理解一个新概念。
- 每个实战演练都由一组任务构成，采用编号步骤(1, 2 等)来展示。必须按步骤操作来完成整个演练。
- “注意”等补充内容提供了成功完成一个步骤的替代方案或者额外信息。
- 要求你输入的文本(代码除外)用粗体显示。
- 两个按键名称之间的加号(+)表示两个键要同时按下。例如，在说到按 Alt+Tab 时，是指按住 Alt 不放，再按 Tab 键。
- 两个菜单项之间的竖线，比如“文件”|“关闭”，是指先选择前一项，再选择后一项，依此类推。
- 由于译者硬件和服务环境有限，部分截图只能沿用英文截图，还请读者见谅。

系统要求

需要准备好以下硬件和软件环境来完成本书的实战练习。

- Windows 7, Windows Server 2008 Service Pack 2 或者 Windows Server 2008 R2。
- Visual Studio 2010 的任何版本(免费的 Express Edition 产品可能要求下载多个组件)。
- SQL Server 2005 Express Edition 或之后的版本(2008 或 R2 版)，SQL Server Management Studio 2005 Express 或之后的版本(Visual Studio 自带; Express Editions 要求单独下载)。SQL Azure 要求 SQL Server Management Studio 2008 R2。
- 一台能运行 Visual Studio 2010 的计算机。
- Windows Azure 要求 Internet 连接。所有示例都可使用本地模拟器来试验。

取决于具体的 Windows 配置，可能需要本地管理员权限来安装或配置 Visual Studio

2010 以及 SQL Server 2008 产品。

示例代码

本书大多数实战演练都允许交互式尝试在正文中学到的新内容。所有示例项目(演练前后有两个版本)都可从以下网址下载:

<http://go.microsoft.com/fwlink/?Linkid=217915>

请点击“Download the companion content”链接。

安装和使用示例代码

将下载回来的压缩文件解压到一个指定的目录。示例代码存储在和章对应的目录中。每一章的目录都包含 Visual Studio 解决方案,可以直接使用。第 7 章包含四个目录,每个都有一个解决方案。

每章都包含了逐步骤的操作过程来重建示例,所以为了完成本书的实战演练,并不需要任何现成的示例代码。要完成一个实战演练,按顺序完成步骤即可。每章的项目都是独立的,完成一章的实战演练不需要先完成上一章的练习。

目 录

第 1 章 云计算基础	1
1.1 云计算的方式	1
1.1.1 基础结构即服务	1
1.1.2 软件即服务	2
1.1.3 平台即服务	2
1.1.4 定义云服务	3
1.2 长期展望	4
1.3 Windows Azure 作为 PaaS 方案	5
1.3.1 小公司的大好机遇	6
1.3.2 大公司的大好机遇	8
1.4 Windows Azure 和云计算	9
1.5 小结	11
第 2 章 Windows Azure 平台基础	13
2.1 操作系统	14
2.2 服务创建	16
2.3 Windows Azure 存储	22
2.4 工作者角色	25
2.5 虚拟机角色	26
2.6 Windows Azure AppFabric	26
2.6.1 服务总线	27
2.6.2 访问控制	28
2.6.3 缓存服务	29
2.6.4 集成服务	30
2.6.5 复合应用程序	31
2.7 SQL Azure	32
2.8 小结	32
第 3 章 创建 Web 角色项目	33
3.1 SDK	33
3.2 Windows Azure Tools for Visual Studio	34
3.3 Web Role 项目模板	35
3.4 云项目	40
3.5 部署到 Windows Azure	44

3.6	配置和升级	47
3.7	服务定义文件	54
3.8	角色属性	55
3.9	小结	59
3.10	快速参考	59
第 4 章	Windows Azure 存储	61
4.1	本地存储	61
4.2	Windows Azure 存储帐户	66
4.3	Windows Azure Management Tool	69
4.4	Blob API	75
4.5	小结	83
4.6	快速参考	83
第 5 章	表、队列和工作角色	85
5.1	表服务	85
5.2	队列服务	98
5.3	小结	105
5.4	快速参考	105
第 6 章	Windows Azure 操作系统	107
6.1	Live ID、订阅和记帐	107
6.2	地缘组	111
6.3	内容交付网络(CDN)	112
6.4	证书	114
6.5	诊断	120
6.6	小结	123
6.7	快速参考	123
第 7 章	构建 AppFabric 解决方案	125
7.1	Windows Azure AppFabric 组件	125
7.2	服务总线	127
7.3	直接连接	140
7.4	绑定	143
7.5	HTTP for Management URI	146
7.6	小结	147
7.7	快速参考	148
第 8 章	WCF 数据服务和 OData	149
8.1	Astoria 项目	149

8.2	WCF 数据服务.....	150
8.3	基本构建单元.....	151
8.4	WCF 数据服务.....	159
8.5	查询和关系.....	162
8.6	筛选、排序和分页.....	167
8.7	WCF 数据服务客户端.....	168
8.8	.NET Framework 客户端.....	172
8.9	安全性简介.....	175
8.10	小结.....	178
8.11	快速参考.....	179
第 9 章	使用 SQL Azure.....	181
9.1	SQL Azure 功能.....	181
9.2	SQL Azure 数据库访问.....	183
9.3	在云端创建数据库服务器.....	185
9.4	SQL Azure 访问.....	191
9.5	SQL Azure 关系引擎功能.....	196
9.6	现有数据库的迁移.....	197
9.7	SQL Azure 迁移向导.....	198
9.8	小结.....	201
9.9	快速参考.....	201
第 10 章	从任何地方访问 Azure 服务.....	203
10.1	创建存储帐户项目.....	203
10.2	从 PHP 访问存储帐户.....	213
10.3	使用 HTTP 和 REST.....	215
10.4	小结.....	220
10.5	快速参考.....	220
第 11 章	应用程序构架.....	221
11.1	多层解决方案的特征.....	221
11.2	数据访问层.....	223
11.3	服务代理.....	235
11.4	小结.....	238

第 1 章 云计算基础

本章旨在教会你：

- IaaS、SaaS 和 PaaS 的差异
- 理解 Microsoft 为其云策略选择的路线
- 理解 Windows Azure 平台的基础

本书是我学习、讲授以及开发云解决方案时的经验之谈。我的云解决方案利用了 Windows Azure 平台的多种组件，包括 Windows Azure Compute，Windows Azure Storage，Windows Azure AppFabric 以及 Microsoft SQL Azure。本章介绍云计算的理念，它是所有云项目的基础。

Windows Azure 官方主页(<http://www.azure.com>)如下声明：

Microsoft Windows Azure 提供可扩展和容错的环境，允许开发人员创建强大的应用程序，同时不需要购买和配置硬件及操作系统。相反，只需遵循 PaaS (Platform as a Service)模型租用自己需要的东西。

注意：由于 URL、UI 名称以及操作步骤会随着时间的变化，书中提到的这些内容在你读到本书时可能已经过时。遇到这种情况，建议访问 Windows Azure 主页 (<http://www.azure.com>)了解 Windows Azure 产品的最新信息。书中提供的信息在写作时是准确的。

1.1 云计算的方式

任何云计算方案的基本思路都是只为所用的东西付费，能根据业务需求扩展或收缩。支持云计算的厂商会对这个基本思路进行不同的诠释，提供不同等级的服务来达到最终的结果。云计算的三种方式是“基础架构即服务”(Infrastructure as a Service, IaaS)、“软件即服务”(Software as a Service, SaaS)以及“平台即服务”(Platform as a Service, PaaS)。

1.1.1 基础结构即服务

有的厂商提供了构建解决方案的基础结构，你租用服务器、负载均衡器、防火墙和线缆，远程配置并在上面安装自己的解决方案。需求增加了，可请求更多的服务器并重新配



置负载均衡器，不需要购买更多的硬件。需求减少了，任何时候都可以重新配置从云服务提供商租用的基础结构。这种云计算方式称为“基础结构即服务”(IaaS)，因为客户可以租用基础结构，无需提前预测并满足自己最高的需求。采用这种方式，要由你正确配置租用的基础结构。IaaS 的关键概念包括：

- 硬件层面的东西由厂商管理；
- 很少有厂商实际提供操作系统。仍然要由你管理从操作系统到应用程序的一切；
- IaaS 最明显的优点是不需要准备大量物理机或虚拟机。

1.1.2 软件即服务

另一种方式是租用厂商提供的服务，通过厂商提供的界面管理服务，而不需要知道厂商用什么基础结构提供服务。这种方式称为“软件即服务”(SaaS)，因为要付费使用定义好的服务。例如，Microsoft Exchange Online 就是按邮箱收费的。为了配置它，要使用由厂商提供的 Web 应用程序来请求邮箱，进行命名和指定容量。完成交易后，会收到邮箱密码，其他一切都不需要，用户能立即访问邮箱。

这种方式和许多人熟悉的、安装到企业生产场所(on-premises)的机器上的 Microsoft Exchange 有很大区别。在 SaaS 模型中，既无法控制、也不由你负责安装服务的硬件。类似地，你控制不了运行服务的操作系统，也控制不了除了在 Web 用户界面上所公开的之外的其他软件设置。换言之，厂商提供了运行应用程序所需的一切，隔离了所有底层组件。

1.1.3 平台即服务

第三种方式是“平台即服务”(PaaS)。采用这种方式，要租用平台来部署应用程序，但不需要配置基础结构，同时也没有 SaaS 方式的限制。维基百科全书对 PaaS 的定义如下：^①

……将计算机平台和解决方案作为服务提供。PaaS 简化了应用程序的开发，避免了购买和管理底层软硬件和提供主机托管所需的费用和复杂性，为了支持通过 Internet 来使用的 Web 应用程序和服务的完整生命周期，一切必要的功能都由 PaaS 提供。

PaaS 提供了应用程序设计、开发、测试、部署和托管功能。还提供了应用程序服务，比如团队协作、Web 服务集成和调度、数据库集成、安全性、伸缩性、存储、持久化存储、状态管理、应用程序版本控制、应用程序 instrumentation^②和开发人员社区功能。这些服务可以通过 Web 作为集成解决方案来提供。

① http://en.wikipedia.org/wiki/Platform_as_a_service

② instrument 是仪表，instrumentation 是仪表化的意思，这些“仪表”有什么用呢？用处就是监视和跟踪。简单地说，instrumentation 是监视或测量产品性能的一种能力，目的是诊断错误、衡量性能以及输出跟踪信息。程序员使用代码指令来实现 instrumentation。这些指令监视系统中的特定组件(例如，指令可在屏幕上输出日志信息)。包含 instrumentation 代码的应用程序必然能用某个管理工具进行管理。——译注

Windows Azure 平台在很大程度上归于 PaaS 一类，因为它不提供对底层虚拟化环境的访问，也不允许访问操作系统细节，比如网络接口、IP 配置和磁盘管理等。

PaaS 的关键概念包括：

- 平台厂商提供和管理一切，从网络连接到运行时环境；
- PaaS 支持平台运行时和相关应用程序服务，为开发人员“减负”；
- 开发人员可立即开始为应用程序创建业务逻辑；
- 和传统托管方案相比，PaaS 优势在于显著提升了生产力，因为云提供商负责管理云平台的所有硬件和运作问题。

1.1.4 定义云服务

图 1-1 总结了你和厂商各自承担的责任。^①

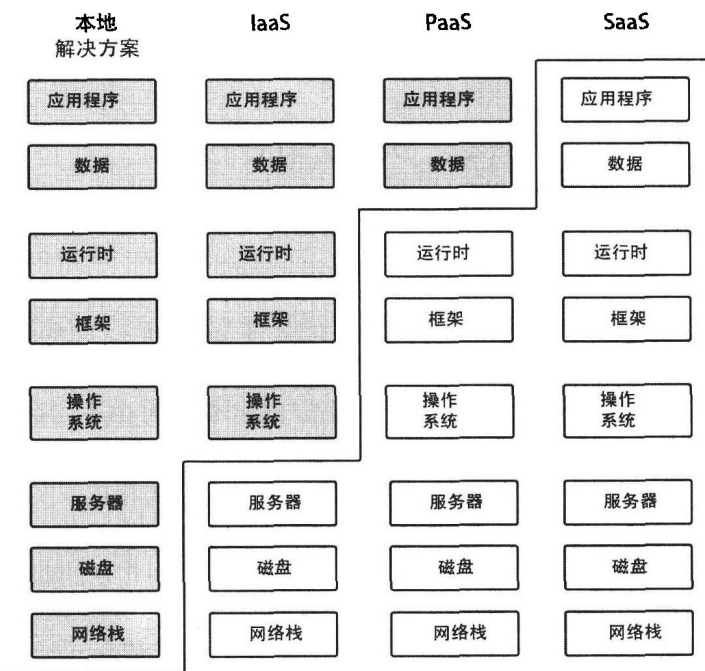


图 1-1 深色块的内容由你负责，白色块的内容由厂商负责

如图所示，虽然云计算工业的各种方案存在显著区别，但共同点是所有厂商都提供了可供租用的服务，避免你花费时间和精力管理它们(线下方的白色块)。

维基百科对云计算的定义如下：^②

^① 注意图中的“本地解决方案”，原文是“On-premises solution”。premises 是生产场所、经营场所的意思。所以“On-premises solution”的意思是将整套解决方案都安装到本地(你的经营场所)，而不是安装到“云端”，正好和“云解决方案”相反。

——译注

^② <http://zh.wikipedia.org/wiki/云计算>

云计算是一种基于互联网的计算方式,通过这种方式,共享的软硬件资源和信息可以按需提供给计算机和其他设备。整个运行方式很像电网。

云计算是继 20 世纪 80 年代大型计算机到客户端-服务器的大转变之后的又一种巨变。用户不再需要了解“云”中基础设施的细节,不必具有相应的专业知识,也无需直接进行控制。云计算描述了一种基于互联网的新的 IT 服务增加、使用和交付模式,通常涉及通过互联网来提供动态易扩展而且经常是虚拟化的资源。云其实是网络、互联网的一种比喻说法。因为过去在图中往往用云来表示电信网,后来也用来表示互联网和底层基础设施的抽象。典型的云计算提供商往往提供通用的网络业务应用,可以通过浏览器等软件或者其他 Web 服务来访问,而软件和数据都存储在服务器上。云计算关键的要素,还包括个性化的用户体验。

这个定义突出了两点:使用分布式资源(IaaS, SaaS 和 PaaS),以及对底层技术的抽象。第一点你现在已经了解。第二点也很重要,它使你能轻松管理抽象的资源(比如分布式存储),而不必知道如何配置、加密和分布的技术细节。

1.2 长期展望

我希望将来用户可以完全忽略数据和程序的所有物理细节。但为了达到这个目标,还有很长一段路要走。

现在,人们经常谈到 SOA 这个缩写词,它的全称是“Service Oriented Architecture”,即“面向服务的架构”。这个术语描述了由相互连接的服务构成的一种生态系统。这些服务使用共同的模式和标准来交换数据和共享进程。SOA 服务可由不同系统上部署的应用程序“消费”,这些系统可以使用不同的操作系统,可以有不同的编程环境。SOA 定义了不同系统和平台上的互操作概念。每个服务都可使用不同的方式和技术实现。SOA 只是定义了这些服务相互之间以及与客户端应用程序的通信方式,使服务开发人员能自由地实现他们希望的内部逻辑。例如,在 Microsoft .NET Framework 中实现的服务会用到其他 .NET Framework 组件和 Windows API,而使用 Java 或 Ruby 写的类似服务则完全不同。虽然每个服务在内部使用不同通信模式,但它们必须遵守共同的通信条例以便与其他 SOA 服务或客户端通信。

语言、操作系统和框架的进步已经为本地平台提供了一个抽象层。例如,使用大多数现代编程语言时,开发人员都不必直接管理 RAM。相反,在今天的垃圾回收环境中,只需正确释放对象实例,框架会自动与操作系统协调内存的分配和释放。正是由于这个抽象层的存在,所以无论是安装了 8 GB RAM 的笔记本电脑上的 .NET Framework 环境,还是只有 256 MB RAM 的 Windows CE 设备上的 .NET Compact Framework 环境,都可以运行相同的代码,即使两个环境中的垃圾回收器以不同方式工作。我不是说同样的代码在哪里都能工作,而是说虽然 Windows 和 Windows CE 存在很大不同,但许多不同都因为 .NET Framework 的存在而在开发者面前变得“透明”了。

今天的编译器能够很好地抽象机器码。操作系统抽象了内存、磁盘和图形设备的细节，而像 CLR 和 JVM 这样的“运行时”为你处理物理细节。这是一个很好的起点，下一步是消除资源物理位置和使用资源的代码之间的依赖性，从而创建分布式系统，以便部署应用程序和服务，并提供管理资源的一种抽象方式。

从消费者的角度，重要的不是位置，而是他们的请求能快速和不费力地得到响应。从开发人员的角度，重要的是关注应用程序逻辑，避免纠缠于底层环境的细节以至于分心。

1.3 Windows Azure 作为 PaaS 方案

使用 PaaS 不需要知道每个组件的技术细节。你不需要知道 RAID 0 和 RAID 1 硬盘的区别，不需要关心硬件和速度和容量，也不需要知道盘符是 C 还是 D。只需向平台要求一个位置来存储信息，所有技术细节都由平台照管。

Windows Azure 平台彻底隐藏了这些技术细节，提供了以符合逻辑的方式管理资源的 API。你只需创建存储，选择名称，然后用系统提供的终端点^①管理资源即可。

Windows Azure 的基本思路就是提供分布式操作系统，不需要通过经典的 Windows 界面就可以部署和运行应用程序。例如，不需要将文件复制到 Windows Azure 文件系统，也不必使用 IIS 管理控制台来配置网站、虚拟目录或者应用程序池。事实上，甚至不需要知道幕后是否存在 IIS。

需要磁盘空间时，只需创建存储帐户^②，用提供的终端点管理资源。使用 PaaS，可以忘记磁盘、存储区域网络和负载均衡配置的存在，直接将数据存储在云端即可。可以使用 REST 和 HTTP 等标准与这种存储交互。文件存储在哪儿？没必要知道！哪个磁盘最快？没必要关心！PaaS 方案的磁盘管理过程(从订购新磁盘到更换损坏磁盘)在用户面前是完全隐藏的。

使用 Windows Azure 平台看不到自己租用的磁盘空间的具体位置，不能选择 UPS，也不能选择磁盘或服务器的硬件制造商。不可以选择自己的 IP，也不用关心计算机名。

在这样的系统中，对资源的访问必须用相关的服务来完成。每个 API 都必须作为一个远程 Web 服务公开。虽然今天的系统没有公开每个 API，但想必你已明白其中的意思。

在实际应用中，PaaS 从各个方面说都是一种 SOA：

- 你要求存储服务保存新文件；
- 你要求存储服务查找文件；
- 你要求平台管理服务根据你的当前需求而增大或缩小规模；
- 你要求存储服务创建新文件夹；
- 服务提供一个响应，你必须分析这个响应。

① endpoint 按照 Windows Azure 官方术语翻译成“终端点”。——译注

② 在本书中，帐户也涉及与金钱相关的账户，两者可等同看待。

以后各章会更详细地讨论一切是如何工作的，但基本思路就是程序的编写、部署和执行都不需要知道二进制指令和数据的物理存储位置。部署阶段完成后，就可以把自己的软件忘掉了(bug 除外)，因为平台会帮助你做下面这些事情，帮你对它进行管理。

- 一有补丁就会立即应用。
- 复制你的数据和“运行时”，提供容错和负载均衡。
- 管理磁盘和其他硬件。例如，如果磁盘失效，系统立即使用备用磁盘，不要求你进行任何介入。你甚至注意不到磁盘曾经失效。
- 数据增长时自动分配更多磁盘，并重新配置负载均衡，不会有任何当机时间；
- 应用程序崩溃，能自动重启系统。
- 任何时候都可请求更多计算能力(还可在 Windows Azure 中自动完成这个任务，本书会具体讲述如何做)。
- 分配给你的应用程序的机器停止响应，就自动将服务移至新机器，不需要你介入。

1.3.1 小公司的大好机遇

在我看来，云计算——尤其是 Windows Azure 平台——对每个公司都是一个巨大机遇。小型软件公司尤其应该抓住这个机遇。没有云计算平台或基础结构，小公司在创建可能有几千个并发用户的应用程序时无法做到与更大的公司竞争。原因很简单，它们买不起昂贵的硬件。

小公司创建新的解决方案(比如电子商务网站、广告应用程序或财务应用程序)时必须考虑诸多因素。在这些因素中，不仅包括初始投入，还包括许多后期投入。其中包括聘请技术人员来配置和维护下面这些东西：

- 生产服务器和过渡服务器^①
- 数据库集群
- 路由器和负载均衡
- 防火墙和安全软件

除此之外，应用程序所需带宽的费用和所有软件的许可证费用是固定的。而且这些投入还不能仅仅满足公司的初始需求就完了。公司必须一开始就购买足够的 Web 服务器来创建容错方案，应用程序从最开始部署就提出了这个要求。有的时候，因为故障(包括服务中断)而蒙受的损失甚至比预防这些故障而进行的初始投入还要大。

除了开发应用程序昂贵的初始投入，小公司还需聘用技术人员配置服务器，使之发挥最高性能。达到这种技术级别的人员必须在公司外部寻找，这意味着系统的初始阶段又得增加一笔支出。等系统投入生产后，一旦出现故障，当初雇佣外人可能带来新问题，因为公司员工对系统知之甚少。除此之外，无论外聘技术人员，或是对内部员工进行培训，都可能耽误发布日期，这可能进一步影响成本。

^① 过渡(staging)服务器作为一个临时性的 stage(舞台)使用，目的是服务器在正式投入运营之前对各项参数(包括性能)进行测试。可理解为对服务器进行“彩排”。——译注