Windows Azure Platform

TEJASWI REDKAR

Apress®

Windows Azure Platform

ISBN-13 (pbk): 978-1-4302-2479-2

ISBN-13 (electronic): 978-1-4302-2480-8

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

President and Publisher: Paul Manning
Lead Editor: Ewan Buckingham
Technical Reviewer: Fabio Claudio Ferracchiati
Editorial Board: Clay Andres, Steve Anglin, Mark Beckner, Ewan Buckingham, Gary Cornell, Jonathan Gennick, Jonathan Hassell, Michelle Lowman, Matthew Moodie, Duncan Parkes, Jeffrey Pepper, Frank Pohlmann, Douglas Pundick, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Project Manager: Anita Castro
Copy Editor: Heather Lang, Tiffany Taylor, and Mary Ann Fugate
Compositor: Kimberly Burton
Indexer: Toma Mulligan
Artist: April Milne
Cover Designer: Anna Ishchenko

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit `http://www.springeronline.com`.

For information on translations, please e-mail `info@apress.com`, or visit `http://www.apress.com`.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at `http://www.apress.com/info/bulksales`.

The source code for this book is available to readers at `http://www.apress.com`. You will need to answer questions pertaining to this book in order to successfully download the code.

This book is dedicated to my grandmother Vimal Sharad Redkar. She has been my inspiration for whatever I do. I thank my wife Arohi and my sons Aaryan and Dhruv for supporting me in writng this book. Arohi, I will never forget your sacrifices for making this book happen. I thank my sister, Aasawari for being with me when I needed her in the time of crisis. Finally, I thank my parents for their teachings that has shaped up my life.

Contents at a Glance

- ABOUT THE AUTHOR xv
- ABOUT THE TECHNICAL REVIEWER xvi
- ACKNOWLEDGMENTS xvii
- CHAPTER 1: INTRODUCING CLOUD SERVICES 1
- CHAPTER 2: WINDOWS AZURE PLATFORM OVERVIEW 53
- CHAPTER 3: WINDOWS AZURE 105
- CHAPTER 4: WINDOWS AZURE STORAGE PART I – BLOBS 205
- CHAPTER 5: WINDOWS AZURE STORAGE PART II – QUEUES 267
- CHAPTER 6: WINDOWS AZURE STORAGE PART III – TABLES 313
- CHAPTER 7: APPFABRIC: ACCESS CONTROL SERVICE 371
- CHAPTER 8: APPFABRIC SERVICE BUS 427
- CHAPTER 9: SQL AZURE 505
- INDEX 585

Contents

About the Author xv

About the Technical Reviewer xvi

Acknowledgments xvii

Chapter 1: Introducing Cloud Services 1

Defining Our Terms 3

Evolution of Cloud Services 4

Planning 7

Developing 7

Stabilizing 7

Deploying 7

Operating 7

Modeling Cloud Service Offerings 7

Consumer Services Layer 8

Enterprise Services Layer 9

Middleware Layer 9

Platform Layer 9

Shifting to the Cloud Paradigm 10

Understanding the Cloud Services Ecosystem 12

Service Providers 12
Software Vendors 12
Independent Software Vendors 13
Enablers 13
Businesses 13
Cloud Services Drivers and Barriers 13
Security 14
Total Cost of Ownership 15
Control 15
Culture 16
Competition 16
Time to Market 16
Reliability 16
Service Level Agreement 17
Understanding Cloud Architecture 17
Getting to Know Some Cloud Services Vendors 19
Amazon Web Services 19
Google 39
SalesForce.com and Force.com 44
GoGrid 45
3Tera 46
Microsoft 46
Summary 50
Bibliography 51

Chapter 2: Windows Azure platform Overview 53

Windows Azure Platform Overview 54

Windows Azure Platform Architecture 54

Basic Azure Scenarios 61

Scenario 1: Azure Software Development 63

Scenario 2: Cross-Enterprise Application Integration 63

Scenario 3: Enterprise Process Offloading 63

Scenario 4: Consumer Services (Live Mesh Synchronization) 64

Windows Azure Platform for Developers 64

Developer Roles 64

Developer Readiness 65

Getting Started 65

Windows Azure Developer Portal 68

AppFabricDeveloper Portal 75

SQL Azure Developer Portal 78

Building the Development Platform 79

Operating System 80

Software 80

Getting Started with Windows Azure Platform Development 81

Setting the Objectives 81

Understanding the Service Architecture 81

Understanding the Developer Workflow 82

Developing the Service 83

Deploying the Service 92

Example Summary 104

Summary 104

Bibliography 104

Chapter 3: Windows Azure 105

Windows Azure Architecture 105

The Compute Service 110

Windows Azure API Structure 112

Developer Environment 113

Windows Azure Tools for Visual Studio 113

Windows Azure SDK Tools 128

Service Models 129

Development Fabric 138

Development Storage 140

Diagnostics 141

Logging 143

Developing Windows Azure Services with Inter-role Communication 145

Objectives 145

Adding Diagnostics and Inter-role Communication 145

Running the HelloAzureCloud Service 152

Developing a Web Role to Worker Role Message Exchange 165

Objectives 165

Service Architecture 165

System Information Message 167

The Components of the Solution 169

Geolocation 189

Enabling Geographic Affinity 190

Content Delivery Network 193

Windows Azure Service Management 194

Service Management API Structure 195
Programming with the Service Management API 195
Windows Azure Service Life Cycle 199
Architectural Advice 202
Summary 203
Bibliography 203
Chapter 4: Windows Azure Storage Part I – Blobs 205
Storage Service Architecture 206
The Blob Service 207
Blob Limitations and Constraints 209
Blob Architecture 209
REST API 213
Account Operations 226
Programming Example 229
Container Operations 232
Blob Operations 249
Blob Summary 265
Bibliography 266
Chapter 5: Windows Azure Storage Part II – Queues 267
Queue Limitations and Constraints 268
Queue Service Architecture 268
Windows Azure Storage Account 269
Queues 269
Messages 270
REST API 271
Request 272

Response....274
Storage Client APIs....275
Account Operations....281
Queue Operations....287
Create Queue....289
Set Queue Metadata....292
Message Operations....295
Put Message....297
Get Messages....300
Queue Scenarios....306
Scenario 1: Windows Azure Web and Worker Role Communications....306
Scenario 2: Worker Role Load Distribution....307
Scenario 3: Interoperable Messaging....308
Queue Summary....311
Bibliography....311
Chapter 6: Windows Azure Storage Part III – Tables....313
Table Service Architecture....314
REST API....320
Example Table Model....333
Account Operations....340
Table Operations....340
Entity Operations....350
Table Service Summary....368
Windows Azure Storage Summary....369
Bibliography....369

Chapter 7: AppFabric: Access Control Service 371

Concepts and Terminology 373

Identity Provider 373

Relying Party 373

Security Token (SAML token) 373

Secure Token Service (STS) 373

Request for Security Token (RST) 374

Request Security Token Response (RSTR) 374

Claim 374

Identity Federation 374

Windows Identity Foundation (Geneva Framework) 375

Active Directory Federation Server (ADFS 2.0) (Geneva Server) 375

Web Resource Authorization Protocol (WRAP) and Simple Web Token (SWT) 375

Claims-Based Identity Model 375

Access Control Service Usage Scenarios 378

Scenario 1: Enterprise Cloud Application 378

Scenario 2: Cross-Enterprise Application 380

Scenario 3: ISV Cloud Service 382

Retrieving Tokens from ACS 384

Access Control Service Management Portal 385

Managing Access Control Service Resources 388

Service Namespace 388

Token Policy 389

Scope 390

Issuer 391

Ruleset 392

Rule 392
Programming Access Control Service 393
Creating an ACS solution Using SWT 394
Integrating ACS with a SAML Token Provider 410
Deploying the Web Service in Windows Azure 421
Summary 426
Bibliography 426
Chapter 8: AppFabric Service Bus 427
Enterprise Service Bus (ESB) 428
Security and Access Control 429
Connectivity Infrastructure 429
Enterprise Naming Scheme 429
Interface Contracts 429
Evolution of the Internet Service Bus (ISB) 429
Relay Service 431
Introduction to the AppFabric Service Bus 432
Security 433
Naming Service 438
Service Registry 441
Messaging Fabric 442
Programming with the AppFabric Service Bus 446
ProAzure Energy Service Example 447
NetOnewayRelayBinding 450
netEventRelayBinding 467
NetTcpRelayBinding 475
HTTP Relay Bindings 486

Message Buffer....497
Programming Message Buffer Applications....499
Summary....503
Bibliography....504
Chapter 9: SQL Azure....505
SQL Azure Architecture....506
Infrastructure Layer....507
Platform Layer....507
Services Layer....508
Client Layer....508
SQL Azure Limitations and Supported Features....508
Database Features....509
Application Features....509
Administration Features....510
SQL Azure Data Access....510
Code-Near Connectivity....510
Code-Far Connectivity....512
Getting Started with SQL Azure....513
Creating a SQL Azure Database....515
Connecting to a SQL Azure Database....517
Developing Windows Azure Services that Use SQL Azure....532
Service Description....532
Processes for Curtailment....532
Technical Architecture....534
Database-Migration Strategies....577
Data Definition Migration....578

Data Migration....580
Business Logic Migration....581
Application Migration....581
Database Growth-Management Strategies....582
Summary....583
Bibliography....584
Index....585

About the Author

Tejaswi Redkar is a software architect with a passion for writing. He has been working with Windows Azure since its first announcement during PDC 2008.He been working with Windows Azure early adopter customers and the product team for the past one year. He believes that the best way to master a new technology is to either teach it or write a book on it. Tejaswi has designed large-scale cloud as well as on-premise applications in diverse industries ranging from financial, manufacturing, oil & gas, pharmaceutical, retail and technology. In the past, Tejaswi has not only written on conceptual topics like C# and VB.Net Threading, but also on broader topics like MSMQ and Offhore project governance Tejaswi has a Master's Degree in Computer Engineering from San Jose State University and an MBA from University of Wisconsin, Whitewater.

Tejaswi Redkar resides with his wife, Arohi, and two sons Aaryan and Dhruv, in the beautiful San Francisco Bay Area.. When Tejaswi is not working, he is either engrossed in music or finding out reasons to avoid this week's laundry.

About the Technical Reviewer

Fabio Claudio Ferracchiati is a prolific writer on cutting-edge technologies. Fabio has contributed to more than a dozen books on .NET, C#, Visual Basic, and ASP.NET. He is a .NET Microsoft Certified Solution Developer (MCSD) and lives in Rome, Italy. You can read his blog at `http://www.ferracchiati.com`.

Acknowledgments

I would like to thank the following individuals for their contributions to my professional life:

- Smt. Laxmi Natarajan, the only school teacher who once told me I had what it takes to be an author
- Prof. M.B Unde from NCL, India for teaching me the importance of teaching and writing in learning new engineering concepts
- Jamal Haider from Microsoft for believing in me and encouraging the author in me
- Ewan Buckingham from Apress for believing in my idea for writing an advanced book on Windows Azure Platform
- Penny Tong, in teaching me that software is not only about development but also about delivery and support.
- My seniors and friends in University of Mississippi (Olemiss) who convinced me to enroll for a Computer Science Master's degree instead of continuing PhD. in Chemical Engineering.
- Justin Smith from Microsoft for providing me the right answers at the right time on some key releases.
- Mark Kottke, Sanjeev Karande, Eric Golpe, Patrick Butler Monterde and all of the Windows Azure OneTAP team for giving me access to Microsoft internal cluster and early adopter customers
- The Microsoft Consulting Services leadership team for fostering an atmosphere promoting the creation of intellectual property.
- Kui Jia for being a mentor and the right person at the right time for encouraging me in joining Microsoft.
- Ed Koch, Dan Hennage and the Coactive Networks leadership team for inspiring the architect in me and teaching the whole telemetry and energy management business.

My professional life is incomplete without a personal network of amazing friends, coworkers, educators and students who have played an important role in shaping my professional as well as personal life. Finally, special thanks to my wife, Arohi. Without Arohi's selfless help and support, this book wouldn't have been possible at all.

CHAPTER 1

Introducing Cloud Services

As an introduction to our discussion of cloud services, let's consider a situation that's typical in today's medium to large enterprises. Assume a business unit has an immediate need to deploy and maintain an analytics application that it has built internally. The application will provide the business users with valuable business insight that will make the company much more competitive in the marketplace. The business unit has the budget but not the time to implement the solution, and this deployment needs to happen in the next three months.

The IT hosting team members understand the requirement, but to deploy an application with IT resources requires coordination among hardware, software, operations, and support teams. Perhaps ordering hardware and preparing the enterprise operating system build itself takes two months. After that, IT has to go through its standard testing process and operations procedures to make sure all the support needs are identified. So, the earliest application delivery date would be in six months.

The business owner escalates the urgency of the issue but cannot get past the process boundaries of the enterprise. Ultimately, the business owner establishes an independent IT department funded by the business and delivers the application in three months. Even though the application is delivered, it doesn't have the enterprise support and operations quality.

Now, the CEO and the CTO evaluate the situation at the enterprise level and come to the conclusion that there are too many application instances running redundantly across the enterprise and costing the enterprise millions of dollars in resource and maintenance costs. Therefore, they decide to create a mandate that all the applications need to be migrated to the IT application-hosting infrastructure. Eventually, the business unit ends up creating an exception for its situation and continues running its own IT department, thus costing the overall enterprise on redundant resources.

I see these scenarios on a daily basis, and I don't see a clear solution to the problem unless the entire process and structure in which these organizations operate is revamped, or technology like cloud computing takes off and enterprises embrace it wholeheartedly.

How will cloud computing help in this area? To understand, let's go back to the original business requirement: the business owner has an immediate need to deploy and maintain an analytics application, and the time frame is within three months. The biggest hurdles IT has in deploying this application are not in the application itself but in the dependencies and the process involved in provisioning the infrastructure required to deploy and maintain it. If the cloud computing dream is realized, it will eliminate the need for the application hosting team to be dependent on most of the hardware team requirements, because abstraction of hardware is one of the main tenets of cloud computing, and this abstraction is provided by cloud service providers' data centers. If the servers', load balancers', routers', and switches' dependencies are eliminated, the application hosting team could focus solely on deploying the application in the cloud service of its choice, with business approval. In this case, the overall IT agility will improve and better align with the business goals of the enterprise.

Of course, considering the current state of cloud service providers, I am discounting several facts like security, performance, reliability, political atmosphere, on-going maintenance costs, and overall company culture. But all technologies start slow and with skepticism from large enterprises. Skepticism fades away as early adopters of the technology embrace it and provide valuable feedback, which goes back into the product helping it mature over time. As the technology matures, larger enterprises start embracing it. Some larger enterprises do play parts as early adopters, but very rarely because companies typically become risk averse as they grow in size and their processes become streamlined and mature.

As discussed in the scenario earlier, cloud services platforms relieves you of investing in expensive hardware and IT resources for maintaining a highly available and scalable infrastructure. Cloud platforms are designed to be used on demand. The cost of the platform is directly proportional to its usage. The more you use the platform, the more you pay, and vice a versa. These dynamic capabilities allow you to proportionately balance the service operating costs to its usage and thus make your business more elastic and responsive to change. For example, if you have an e-commerce site that peaks during the Thanksgiving and Christmas seasons and attracts fewer but consistent numbers of users for the rest of the year, then you can dynamically increase the capacity of the underlying platform during the holiday season and reduce it for the rest of the year. This dynamic capability offered by service providers is called *utility computing* and is analogous to your utility service providers' model: they charge you by the amount of energy you use. You can scale back your energy bill by reducing the energy usage, or your energy bill will go up during peak load. If you invest in a highly available and scalable infrastructure on premise, scaling in either direction based on demand is difficult. So, in the long term, cloud platforms are designed to reduce your average operating costs by giving you the flexibility to commission and decommission infrastructure depending on the business needs. Currently, the level of performance, flexibility, and reliability offered by heavily invested on-premise infrastructures may not be matched by the currently available cloud service offerings. But, as the offerings mature, they are expected to provide you with lower total cost of ownership without compromising the service reliability.

Internet itself is a vast phenomenon to be branded as a cloud. "Cloud" is a subset of the Internet. The term specifically means applications, platform, infrastructure and consumer services offered by service providers to build applications for the Internet. Acronyms like software as a service (SaaS), platform as a service (PaaS), software plus service (S + S), and database as a service (DaaS) all represent a piece of cloud services in one way or the other. In this chapter, I will go over the evolution of Internet services into cloud services and look at the offerings from major industry players like Microsoft, Google, Amazon, SalesForce.com, GoGrid, and 3Tera.

There has been a sudden burst of interest in cloud computing not only because of the cost savings it offers but also the quality of infrastructure the cloud service providers promise. The credit should go not only to SalesForce.com for revolutionizing their SaaS platform in the small business market, but also to VMWare who created a new market for operating system and hardware virtualization. I credit VMWare for market awareness they brought around virtualization and its enormous possibilities. Microsoft and Amazon followed the trend by investing heavily in virtualizing not only operating systems and software but also data center assets. Virtualization is a key piece in building low-maintenance platforms for cloud services, because a virtualized platform can be moved and scaled without any dependency on the underlying hardware. It abstracts the operating system from the hardware and applications from operating systems. Virtualization makes the concept of utility computing a reality in terms of cost and manageability. As you read this book, you will learn how virtualization plays an important role in Microsoft's Windows Azure platform infrastructure.

The cloud services platforms are in their infancy and have not yet matured either in business models or technology. But, the addition of software vendors like Amazon, Microsoft, Google, and IBM to the list of cloud service providers adds credibility to its future success. These software vendors are going to drive the cloud services industry by offering their enterprise-grade software products to businesses of all scales. So far, businesses clearly value the cost savings but are still concerned about the security and reliability of their data in the cloud. Cost, control, reliability and security are the four main quality

attributes enterprises will evaluate before deciding to adopt a cloud services platform. Enterprises can also adopt hybrid models, where some services are hosted on-premises and others off. For example, the Microsoft Dynamics CRM Online service offers on-premise option that can be switched to off-premise anytime by the enterprise. These kinds of models help enterprises slowly transition a fully on-premise application to an off-premise or a hybrid solution. This helps critical technical resources in the enterprise focus on important strategic initiatives instead of worrying about day-to-day operational issues. After reading this chapter, you will have good understanding about the cloud services industry and some major players in it.

■ **Note** Throughout this book, depending on the context of the conversation, I have used the terms "cloud services" and "cloud applications" interchangeably to generally represent cloud services. A cloud service may be thought of as a collection of cloud applications in some instances, but in the context of this book, both mean the same thing.

Defining Our Terms

Before diving deep into cloud services, I would like to introduce you to the terminology used in this book. "Cloud" is an overloaded word because the platform is not a standardized yet. There are different flavors of interpretations and perspectives about it in the technology industry. To be consistent in this book, I have developed this section for introducing and defining some important terms used herein. Table 1-1 lists the common industry terms and their definitions as they relate to this book.

Table 1-1. *Terminology in This Book*

Term	Definition
Azure	Microsoft's Windows Azure Platform
Azure Services	The components of Windows Azure Platform (e.g., Windows Azure, SQL Azure, AppFabric, and Live Services)
Cloud	The cloud services platform (e.g., the Windows Azure platform)
Cloud application	An application deployed to a cloud services platform and typically part of a larger cloud service
Cloud platform	A service offering by a cloud service provider for deploying cloud services (e.g. Windows Azure platform offered by Microsoft and EC2 offered by Amazon)

Continued

Table 1-1. *Continued*

Term	Definition
Cloud service	An end-to-end service deployed to the cloud platform that may contain one or more cloud applications
Cloud services platform	The same as a cloud platform, which is defined earlier in this table.
On-premise	Refers to applications or services deployed and managed by an enterprise on its own and at its location
Off-premise	Refers to applications or services in the cloud
Service	When used on its own in this book, refers to the cloud service
Solution	When used on its own, refers to a collection of multiple applications and/or cloud services designed for a specific business purpose (e.g., a payroll solution consisting of three cloud services and four on-premise applications)

Evolution of Cloud Services

The Internet service platform has evolved from a simple dial-up access provider to an enterprise-grade software applications platform. The evolution of its maturity is depicted in Figure 1-1.

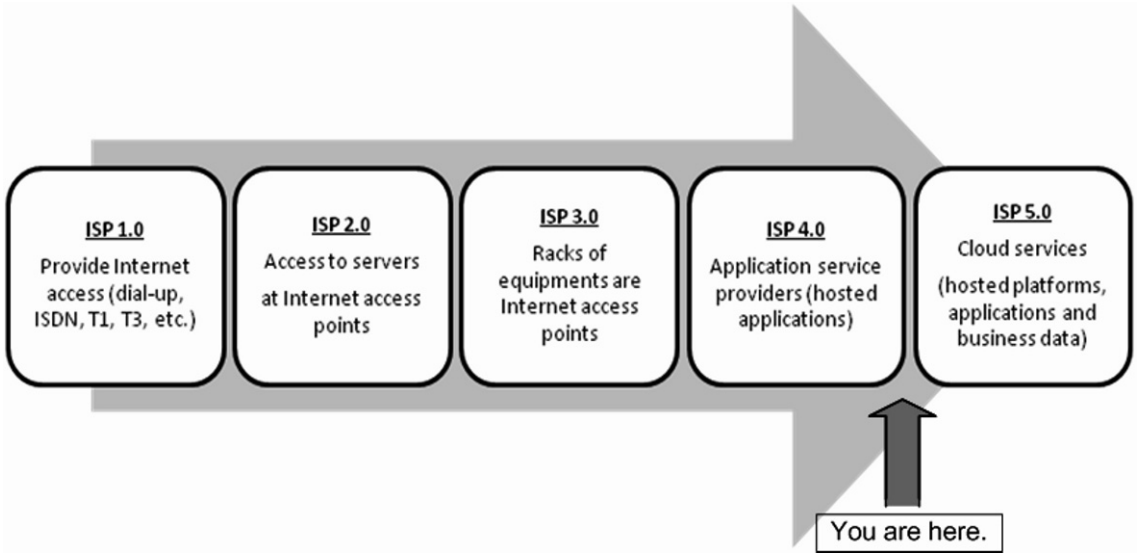

Figure 1-1. *Evolution of ISP into cloud services (Source Data: Forrester Research Inc.)*

The ISP 1.0 era was in the early to mid-1990s, and the focus was on building Internet access networks for consumers and businesses. This era was dominated by companies like AOL, NetZero, Comcast, and Time Warner. Businesses were also heavily involved in building their own internal network infrastructure. In the ISP 2.0 era, the focus shifted to providing access to the servers in the ISP infrastructure. Businesses and consumers could host their web sites on ISP servers with limited capabilities. The ISP 3.0 era brought the colocation concept into the ISP equation. Businesses could host their servers with the ISP, thus leveraging the ISP's massively scalable, efficient, and redundant infrastructure. Companies like Rackspace.com and AT&T were the leaders in this space. Even though ISP 4.0 could achieve economies of scale in the network and power infrastructures, it had to keep up with the technology and business demands to achieve economies of scale at the application and platform levels. This gave rise to the ISP 4.0 era, where the application service providers (ASP) built scalable business software services and abstracted the intricacies of the data centers from the enterprises. Enterprises just had to subscribe to the software services like the CRM services offered by SalesForce.com and Microsoft Dynamics CRM Online without worrying about the underlying data center infrastructure. In this era, the software vendors took the initiative to offer their software services to businesses over the Internet. We have not fully graduated out of the ISP 4.0 era; I would say that we are on the border of ISP 4.0 and ISP 5.0. ISP 4.0 still faces the connectivity, security, and integration challenges between on-premise and cloud services. SalesForce.com, Microsoft Dynamics CRM Online, SharePoint Online, and Exchange Online are viable services that businesses are subscribing to. In the ISP 5.0 era, the ISP infrastructure will mature into a scalable on-demand platform, called the cloud, ripe to be leveraged for building and hosting business applications.

Later in this book, you will see how Microsoft has built an operating system in the cloud comprised of virtually enabled nodes of Windows operating system for building Internet-scale applications. In the ISP 5.0 era, there is complete transparency in application hosting. Enterprises will be able to deploy custom applications into the cloud without worrying about the hardware and platform requirements for the application. This will create transparency between on-premise and cloud applications for

businesses, as they will interoperate seamlessly. You will see in future chapters how Windows Azure achieves some level of this transparency.

A critical success factor for ISP 5.0 is the quality of service (QoS) offered by the cloud service providers. Cloud service providers like Amazon, Microsoft, Google, and IBM are in the process of creating massively scalable data center infrastructure, but there is little focus on the QoS for businesses as of yet. Cost, control, reliability, and security will be the determining factors cloud service providers will have to focus on to convince businesses to use their services. The biggest difference between ISP 4.0 and ISP 5.0 is the entire application life cycle hosting support offered by ISP 5.0. This means applications can be planned, developed, stabilized, deployed, and operated around cloud services with little dependence on on-premise infrastructure. Figure 1-2 shows the ISP 5.0 as an application development and deployment platform.

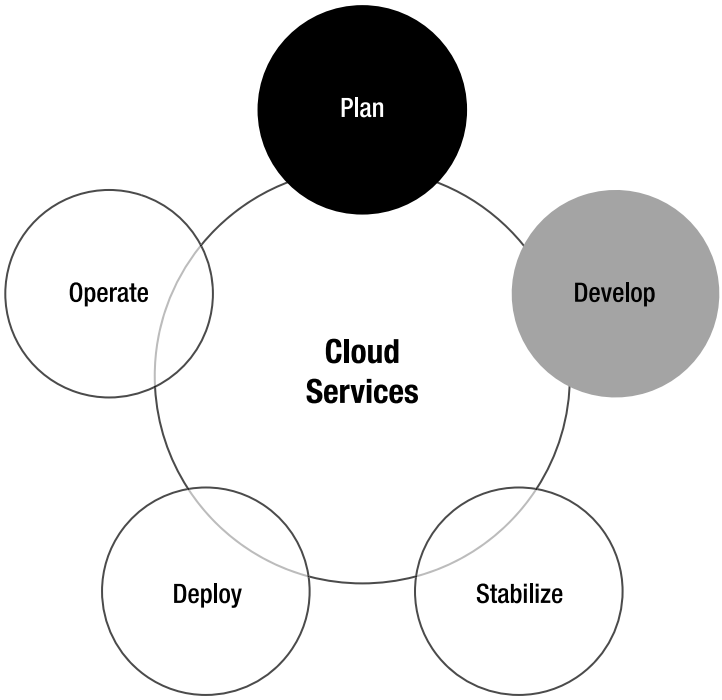

Figure 1-2. *Cloud services application platform (ISP 5.0)*

In Figure 1-2, the planning phase is conducted completely on site, similar to an on-premise application. The deviation from the on-premise application life cycle happens in the development phase, where developers have to work directly with the cloud for unit and functional testing, even though the actual software development may take place on-premise. From the development phase onward, the control of cloud over the service increases and in the deployment and operation phases the cloud is completely in control of the service. The cloud manages the deployment, availability, scalability, and connectivity of the service.

Planning

In the planning phase, you typically envision and plan your cloud service. This may involve designing a new cloud service, migrating an existing on-premise application to the cloud, or creating a hybrid on-premise and cloud service. In this phase, you also decide on the cloud services provider you want to host your service with and open and account with that provider. The most amount of effort in this phase goes into architecting the following attributes of the cloud service:

- Access control
- Network connectivity
- Reliability
- Storage architecture
- Service usage projections

Developing

In this phase, you develop the application in a local development environment provided by the cloud services provider. A *local development environment* is a simulated cloud running on-premise on your development machine used purely for development and functional testing purposes. Development may also involve deploying to the cloud development environment for unit testing.

Stabilizing

In the stabilization phase, the development, testing and the release engineering team iteratively test the application by deploying the service into a testing environment in the cloud. The black box , performance, and business scenarios testing are done in the cloud environment.

Deploying

In the deployment phase, the production version of the application is deployed into the staging environment in the cloud and then further promoted to the production cloud environment for business.

Operating

In the operation phase, the operation logs and usage of the service are evaluated periodically to analyze the usage and health of the service. Usage data is analyzed for billing purposes, and health data is analyzed for improvements to the future versions of the service.

Modeling Cloud Service Offerings

Cloud services offer several different models depending on the industry requirements. For better understanding of the cloud offerings, I have designed a pyramid for categorizing the offerings into four

layers: platform, middleware, enterprise services, and consumer services. Figure 1-3 illustrates the four-layered pyramid model of the cloud service offerings.

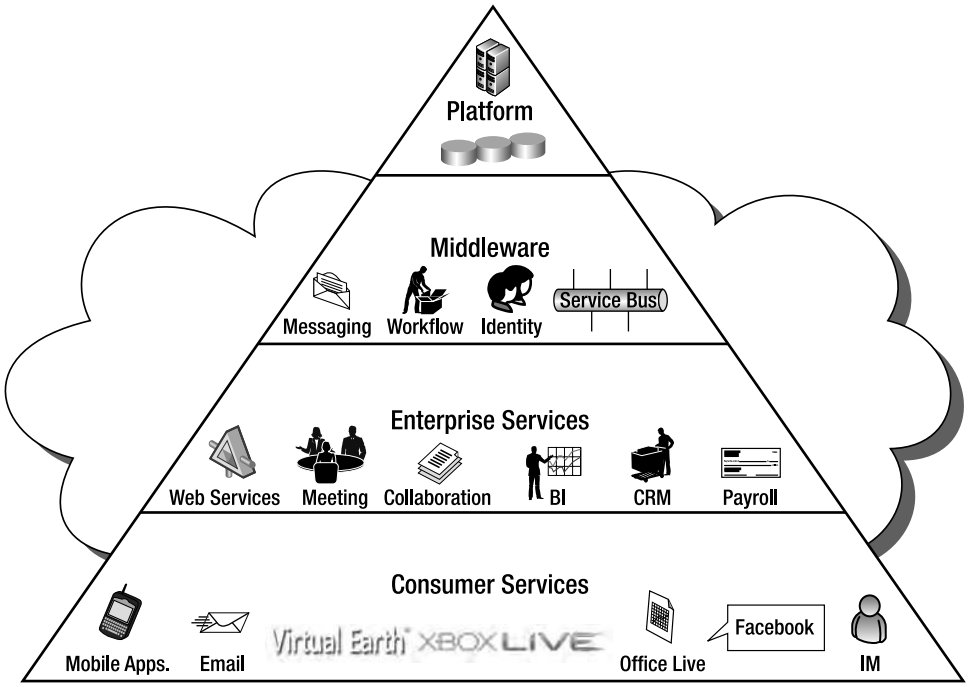

Figure 1-3. *Cloud services offerings pyramid*

Each layer in the pyramid can be considered as a separate offering, but some providers like Microsoft and Google are building complete offering spanning all the layers.

Consumer Services Layer

The consumer services layer represents cloud services that are targeted for the consumers. Some of the services like e-mail, instant messaging, and searching have been available to the consumers from the very beginning of the Internet era, whereas new services like massively multiplayer games, mobile applications, collaboration, social networking, and mapping services have gained significant consumer attention in recent years. The early cloud services like instant messaging and e-mail were developed as dedicated services without any flexibility or abstraction built into their architectures. Every software vendor had its own communication protocol and little effort was made to interoperate across platforms. I call such services as *silos* because each stands on its own, performing a certain function without any cross-platform interoperability. Over the years, these silo architectures have matured with several layers of abstraction and interoperability built in not only the platform but also the infrastructure architecture. These services now support open web services protocols and interoperate across vendor platforms. The consumer layer is built on top the foundation created by the application, infrastructure, and platform

layers. Each of these layers has its own software boundaries, and the consumer market has been and will be the broadest market in terms of end users for cloud services.

Enterprise Services Layer

The enterprise services layer represents application platforms that can be leveraged by businesses to host their business-specific applications or enhanced by independent software vendors (ISVs) in building additional functionality. Most of today's SaaS applications fall into this category of cloud services, and SalesForce.com and Microsoft's Dynamics CRM Online are good examples of application platforms in the cloud. They also offer web services application programming interfaces (APIs) for custom development and add-ons on top of their basic CRM functionality. From the business perspective, the upfront cost and risk involved in deploying these services is minimal, because they are completely managed by the service provider. Businesses have to adapt to the new interface and make sure the software satisfies their business requirements. In contrast, it will cost much more to build and maintain such software in-house, because the businesses cannot leverage economies of scale like the service provider. Service providers can share the same platform across multiple customers, thus benefitting from economies of scale and passing on these cost savings to the businesses. This layer is the fastest growing in cloud services offerings because of its flexibility, low risk, and low upfront cost to the businesses.

Middleware Layer

The middleware layer is a direct result of the monetization of large-scale middleware software components already built to support massively scalable consumer services. Some examples of these already existing services are Amazon e-commerce systems, Google Search, and Windows Live services. As the consumer services and matured, the middleware layer was abstracted and service providers decided to monetize this intellectual property (IP) by offering their middleware capabilities directly to the businesses. In the middleware layer, businesses can utilize proven scalable software services and infrastructure platform offered by service providers for hosting custom software services. All businesses have custom software services that are developed internally to suite their own business processes. Such services are expensive to scale internally due to licensing, hardware, and labor costs. By deploying these services in the cloud and leveraging the scalable middleware of the service providers, businesses can scale these custom services on demand. Microsoft's AppFabric falls into this category because it offers software platforms like service bus and Access Control Service businesses can leverage for building and scaling custom services.

Platform Layer

The platform layer forms the core foundation for all the other cloud services offerings in the pyramid. The platform layer represents the computational, data storage, and network platforms ISVs, and software vendors can leverage this layer in building middleware, enterprise, and consumer services. In this layer, virtualization is employed at its optimum for providing platform abstraction and dynamic scalability for the rest of the layers. Provisioning, management, and milling of the operating systems and storage are automated to reduce maintenance and deployment costs. Quality attributes like scalability, performance, reliability, and availability are built right into the architecture of the platform layer. The primary audiences of the platform layer are ISV developers and infrastructure architects interested in leveraging this highly resilient platform in building end-to-end cloud services. Amazon's Elastic

Compute Cloud (EC2), Microsoft Windows Azure, and Microsoft SQL Azure fall into the platform payer of the cloud services pyramid.

■ **Note** Throughout this book, I will refer to the cloud services pyramid to explain the differences between the cloud services products that we will be working with.

Shifting to the Cloud Paradigm

The move from a traditional on-premise model to an off-premise cloud model is a fundamental paradigm shift for businesses. Usually businesses are in their comfort zone of managing IT internally. With the cloud services model, even though the cost savings become evident, the challenge for businesses is to get out of their comfort zones and make the paradigm shift of moving to cloud services to stay competitive. The shift does not happen overnight; it takes several months of rigorous analysis, planning, and implementation. Depending on the costs, benefits, risks, and security requirements, a business can either stay on-premise, embrace cloud services fully, or settle on a hybrid model yielding cost benefits while keeping core competencies on-site. Figure 1-4 illustrates the ownership of key enterprise assets in on-premise, cloud, and hybrid scenarios.

The recommended migration process is to move step by step, one application at a time. When the offshore software development model became popular in 2000, businesses faced a similar challenge in getting aboard the outsourcing wagon. Now, many businesses have significant offshore investments and clearly see the payoffs. It took time and learning for businesses to make the paradigm shift in off-shore software development projects. For cloud services to succeed, businesses will be required to make a paradigm shift again.

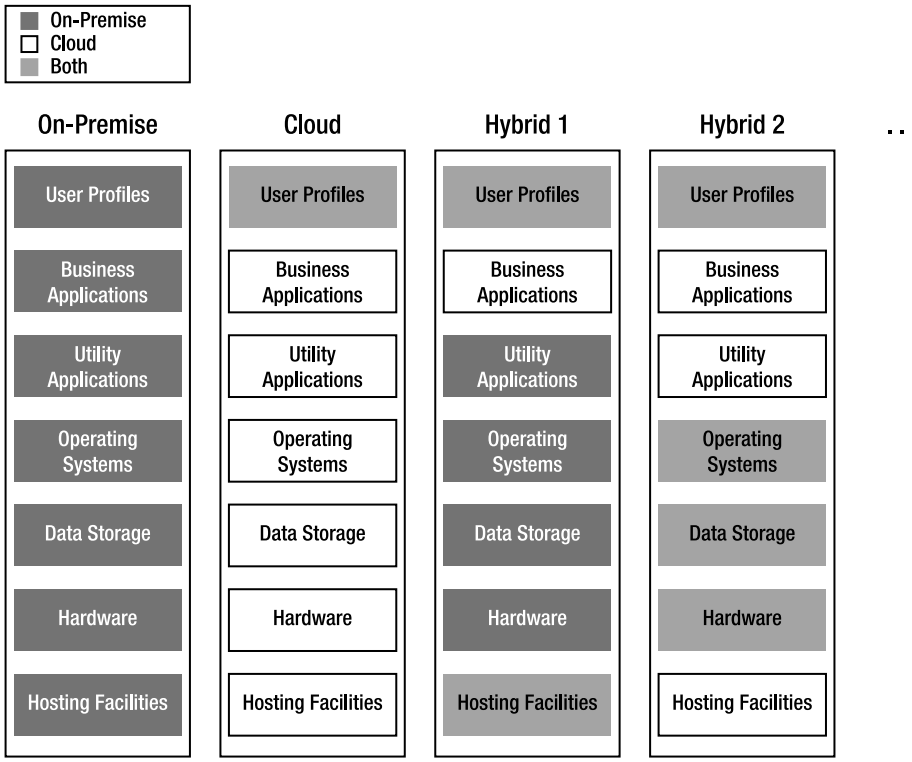

Figure 1-4. *On-premise, cloud, and hybrid scenarios*

In Figure 1-4, the on-premise and cloud scenarios are fairly easy to understand, because either all the assets are on-premise or in the cloud. The user profiles asset is usually required on both the sides because of single sign-on requirements between on-premise and cloud services. In hybrid models, the businesses and the service provider must negotiate and decide which assets and services are better suited for locations on-premise, in cloud, or both. In the Hybrid 1 scenario in Figure 1-4, the user profiles and hosting facilities are present on both the sides; the business applications are in the cloud, whereas the utility applications, operating systems, data storage, and hardware are on-premise. In the Hybrid 2 scenario, the user profiles, operating systems, data storage, and hardware are present on both the sides, whereas the business applications, utility applications, and hosting facilities are in the cloud. Most of the companies typically choose some hybrid model that best suits them.

Understanding the Cloud Services Ecosystem

The cloud services ecosystem consists of five major roles, as shown in Figure 1-5.

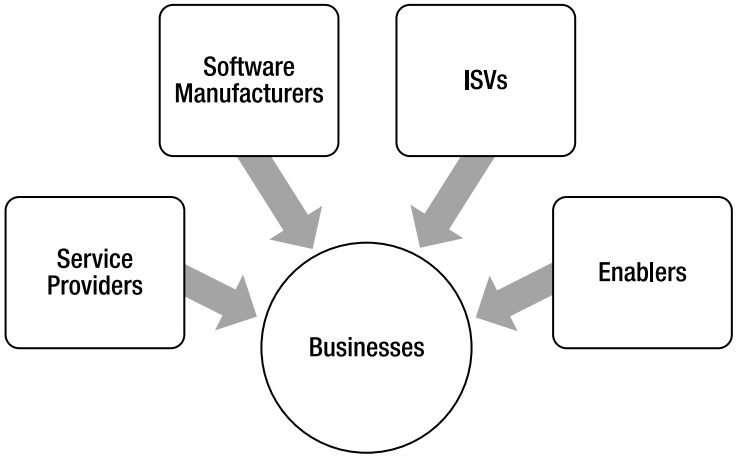

Figure 1-5. *The cloud ecosystem*

Service Providers

The service providers are the companies that provide cloud services to the businesses and to the consumers. These companies run the giant data centers hosting massively virtualized and redundant software and hardware systems. Service providers like Amazon with its EC2 service and Microsoft with its Windows Azure fall into the service providers category. These companies not only have expertise in data center management but also in scalable software management. The service providers may offer services directly to the businesses, consumers, or ISVs.

Software Vendors

Software designed to run on-premise is very different to software designed for cloud services. Even though they both may provide the same business functionality to the end users, architecturally they are not the same. The cloud services must account for multitenancy, scalability, reliability and performance at a much broader scale than on-premise architecture. Cloud services run in data centers offered by cloud service providers. In some cases, there is a significant overlap between the service providers and the software vendors. For example, Microsoft Windows Azure and Google Apps are cloud software running in their own data centers. The software vendors have found it economically feasible to package hardware and software together in the data centers to optimize the service delivery in the cloud.

Independent Software Vendors

Independent software vendors (ISVs) are going to play a key role in the success of cloud services because of their expertise in vertical business applications. ISVs typically build vertical applications on an already existing platform. ISVs identify the business demand for a particular solution in vertical markets and thrive by offering the solution on existing platforms. The cloud offers a great platform for the ISVs to build vertical solutions. For example, an ISV could build a medical billing solution in the cloud and offer the service to multiple doctors and hospitals. The infrastructure required for building multitenant scalable software is already provided by the service providers, so the ISVs have to focus only on building the business solution.

Enablers

Enablers (which are also called *implementers*) are vendors offering services to build end-to-end solutions by integrating software from multiple vendors. Many enterprises purchase software licenses from vendors but never deploy the software because of lack of strategic initiative or availability of product expertise. Enablers fill in the gap by offering consulting services for the purchased software. Companies like Microsoft Consulting Services and IBM Global Services offer customer-specific services regardless of the underlying platform. Enablers play a key role by integrating on-premise and cloud services or building end-to-end cloud services customized for a business. Cloud platform offers enablers an opportunity to expand their service offerings beyond on-premise solutions.

Businesses

Finally, businesses drive the demand for software products and services. If businesses see value or cost savings in a particular solution, they do not hesitate to implement it. To stay competitive in today's market, businesses have to keep their IT and applications portfolios up-to-date and take advantage of economies of scale wherever possible. Cloud service offerings are architected to achieve economies of scale by supporting multiple businesses on a scalable and automated platform. For cloud service offerings to be successful, service providers, software vendors, ISVs and enablers must work together in creating cloud applications and services not only providing cost savings but also a competitive edge to businesses. This search for a competitive edge will drive demand for cloud services.

Cloud Services Drivers and Barriers

Even though cloud computing has gained popularity as the new technology driver, businesses are still evaluating its feasibility for their own business environment. Figure 1-6 shows some of the key business drivers and barriers for cloud services.