

# Enterprise Cloud Strategy



Barry Briggs and Eduardo Kassner

PUBLISHED BY

Microsoft Press

A division of Microsoft Corporation

One Microsoft Way

Redmond, Washington 98052-6399

Copyright © 2016 by Microsoft Corporation

All rights reserved. No part of the contents of this book may be reproduced or transmitted in any form or by any means without the written permission of the publisher.

ISBN: 978-1-5093-0196-6

First Printing

Microsoft Press books are available through booksellers and distributors worldwide. If you need support related to this book, email Microsoft Press Support at [mspinput@microsoft.com](mailto:mspinput@microsoft.com). Please tell us what you think of this book at <http://aka.ms/tellpress>.

This book is provided “as-is” and expresses the author’s views and opinions. The views, opinions and information expressed in this book, including URL and other Internet website references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

Microsoft and the trademarks listed at <http://www.microsoft.com/about/legal/en/us/IntellectualProperty/Trademarks/EN-US.aspx> are trademarks of the Microsoft group of companies. All other marks are property of their respective owners.

**Acquisitions Editor:** Karen Szall

**Developmental Editor:** Karen Szall

**Editorial Production:** Dianne Russell, Octal Publishing, Inc.

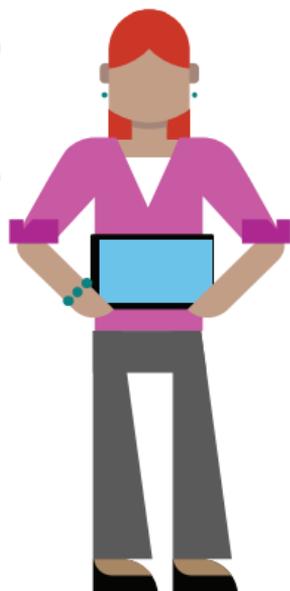
**Copyeditor:** Bob Russell, Octal Publishing, Inc.

**Cover:** Twist Creative • Seattle

Visit us today at

MicrosoftPressStore.com

- **Hundreds of titles available** – Books, eBooks, and online resources from industry experts
- **Free U.S. shipping**
- **eBooks in multiple formats** – Read on your computer, tablet, mobile device, or e-reader
- **Print & eBook Best Value Packs**
- **eBook Deal of the Week** – Save up to 60% on featured titles
- **Newsletter and special offers** – Be the first to hear about new releases, specials, and more
- **Register your book** – Get additional benefits



# Contents

<b>Introduction .....</b>	<b>viii</b>
Acknowledgments.....	ix
Errata, updates, & book support .....	ix
Free ebooks from Microsoft Press .....	x
We want to hear from you .....	xi
<b>Chapter 1: The cloud, efficiency, and innovation .....</b>	<b>1</b>
Economics of the cloud.....	2
Daily efficiencies.....	7
Innovation .....	7
Telenor.....	9
Aviva .....	12
3M Parking Systems.....	15
Heineken.....	18
Learnings.....	21

<b>Chapter 2: Journey to the cloud: the roadmap</b>	<b>23</b>
Don't miss the opportunity to modernize	27
Evolution of the five R's of modernization	30
Cloud migration: three stages	34
<b>Chapter 3: Experimentation</b>	<b>37</b>
Microsoft IT's first cloud application	37
Experimentation and the problem of "shadow" IT	40
<b>Chapter 4: Migrating IT to the cloud</b>	<b>47</b>
Establish strategy and goals	49
Organizational responsibilities in creating the strategy	57
Enterprise architecture	58
Information security and risk management	61
Data classification	65
Enterprise Risk Management	67
Finance	67
Operations	70
Human resources and the evolution of roles	74

Applications teams .....	78
Business units .....	81
Building the catalog.....	82
Top-down portfolio analysis.....	84
Bottom-up portfolio analysis .....	87
The cloud migration plan .....	92
Microsoft IT's experience.....	97
Cloud governance .....	100
Data governance .....	101
Financial governance .....	103
Security and compliance .....	104
Change management.....	105
Information Technology Infrastructure Library and the cloud.....	107
<b>Chapter 5: Transformation .....</b>	<b>112</b>
Platform as a Service architecture.....	113
Containers and microservices.....	115
Storage.....	117
Relational databases in the cloud.....	119
NoSQL (nonrelational) storage.....	121
Analysis .....	130

Integration.....	135
Using services to create rich end-to-end applications.....	137
Conclusions.....	142

**Appendix A: Cloud architectural blueprints..... 144**

Data analytics .....	144
BI and analytics.....	148
Live media streaming.....	151
Video on demand (VOD) .....	154
Line-of-business applications in infrastructure services .....	158
Hybrid cloud storage.....	161
E-commerce website .....	164
Business-to-business (B2B) e-commerce.....	168
Multichannel marketing .....	170
DevOps .....	174

**Appendix B: Sample technology scenarios ..... 179**

Hybrid cloud scenarios.....	179
Hybrid cloud connectivity.....	180

Using the cloud for data backup and recovery .....	184
Hybrid database scenarios .....	192
Development and test .....	193
Application development .....	194
Microsoft SharePoint .....	196
High availability in the cloud.....	196
Connected devices .....	200
Identity and authentication .....	203
Mobile applications .....	207
Enterprise mobility management.....	209
Websites.....	212
Azure Media Services .....	214
Migration strategies.....	217
<b>Appendix C: Recommended references.....</b>	<b>220</b>
Storage references.....	220
Application development and insights references.....	221
Performance best practices references ....	221
Other cloud migration references .....	223
About the authors .....	224

# Introduction

When briefing CIOs and senior IT executives at Microsoft, we are often told that migrating IT workloads to the cloud ranks among their highest priorities. That statement is almost inevitably followed by “How do I start?”; “How should I build a plan for cloud migration for my entire portfolio?”; and “How will my organization be affected by this change?”

This book, based on real-world cloud experiences by enterprise IT teams, seeks to provide answers to these questions. Here, you’ll see what makes the cloud so compelling to enterprises; with which applications you should start your cloud journey; how your organization will change, and how skill sets will evolve; how to measure progress; how to think about security, compliance, and business buy-in; and how to exploit the ever-growing feature set that the cloud offers to gain strategic and competitive advantage.

# Acknowledgments

The authors wish to express their deep gratitude to the following individuals for their support, guidance, and their willingness to freely share their expertise: Scott Woodgate, Javier Nino, Tom Schinder, Venkat Gattamneni, Martin Vliem, Ulrich Homann, Robert Hanegraaff, John Devadoss, Brenda Carter, Michael Washam, Zoiner Tejada, Nadia Matthews, Rob Beddard, Jeff Fryling, Kevin Gee, Colin Nurse, Raman Johar, Walter Myers, Uwe Hoffman, Ashish Sharma, Ashutosh Maheshware, Rich Nickerson, Michel Declercq, Arlindo Alves, Dennis Mulder, and George Moore.

Rob Boucher and Monica Rush created the graphic representations of the blueprints in the Appendixes.

## Errata, updates, & book support

We've made every effort to ensure the accuracy of this book and its companion content. You can access updates to this book—in the form of a list of submitted errata and their related corrections—at:

<http://aka.ms/ECS/errata>

If you discover an error that is not already listed, please submit it to us at the same page.

If you need additional support, email Microsoft Press Book Support at [mspinput@microsoft.com](mailto:mspinput@microsoft.com).

Please note that product support for Microsoft software and hardware is not offered through the previous addresses. For help with Microsoft software or hardware, go to <http://support.microsoft.com>.

## Free ebooks from Microsoft Press

From technical overviews to in-depth information on special topics, the free ebooks from Microsoft Press cover a wide range of topics. These ebooks are available in PDF, EPUB, and Mobi for Kindle formats, ready for you to download at:

<http://aka.ms/mspressfree>

Check back often to see what is new!

# We want to hear from you

At Microsoft Press, your satisfaction is our top priority, and your feedback our most valuable asset. Please tell us what you think of this book at:

<http://aka.ms/tellpress>

We know you're busy, so we've kept it short with just a few questions. Your answers go directly to the editors at Microsoft Press. (No personal information will be requested.) Thanks in advance for your input!

## Stay in touch

Let's keep the conversation going! We're on twitter: <http://twitter.com/MicrosoftPress>

# The cloud, efficiency, and innovation

Most people now agree that the cloud has become a core element of any enterprise's technology strategy. Indeed, in the past few years we have seen the conversation around cloud adoption move from “if” to “when” and “how.” It is, in short, a fact of life.

Nevertheless, it remains one of the most disruptive changes in computing in years, and it is worth reviewing what makes the cloud so compelling to enterprise IT. Its value proposition

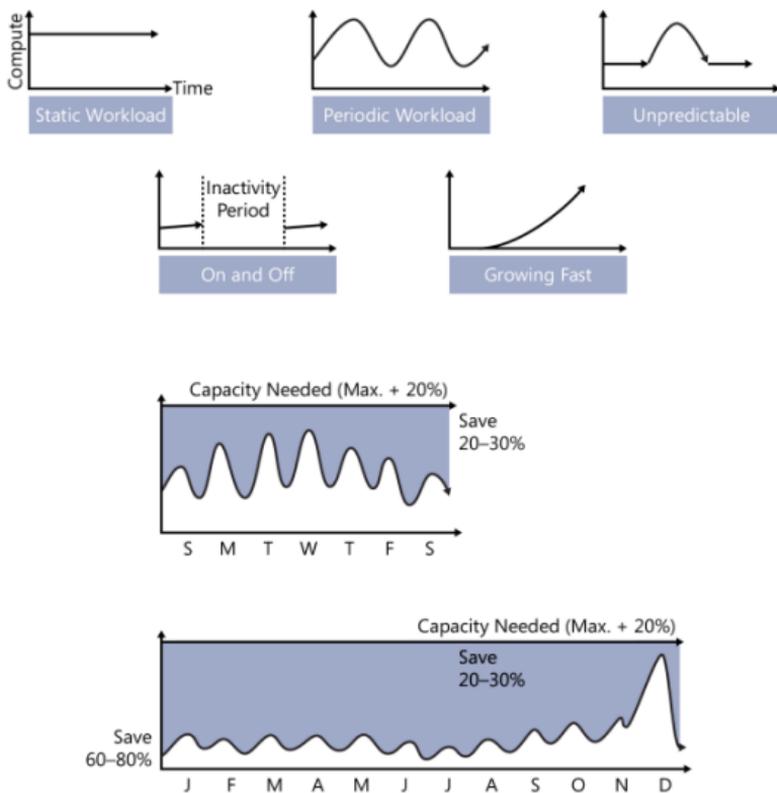
is many-faceted, ranging from significant cost savings over a traditional datacenter approach to the ability to quickly build robust, resilient applications that can scale up as traffic spikes, and scale down as it recedes.

## Economics of the cloud

In cloud computing, enterprises pay for what they use, much as they would a telecom provider. If demand decreases and you no longer need capacity, you can turn off systems and you are not charged. This simple model stands in stark contrast to the *traditional* model of enterprise computing, which is a capital-intensive function, requiring expensive datacenters, electricity, air conditioning, servers, networks, storage, and 24x7 operations staff. For most companies, maintaining a large IT presence in this model implies large capital expenditures and a nontrivial amount of accounting and record-keeping to track depreciation, tax considerations, and so forth. Moreover, when you purchase the hardware and the software, they become *yours* in every sense of the word. Operations staffs are responsible for hardware swaps, networks, backups, updates for operating systems, and upgrades to the system software

and applications. The traditional model is a “capital expense” model.

The cloud, being subscription-based, is an *operating expense* model. In the cloud, computing becomes a service for which customers are billed a monthly charge. Like other such services, it is metered by usage. The more compute, network, and storage resources that you use, the higher will be your bill. Of course the reverse is also true: the less you use, the less you are charged. Indeed, most IT organizations find wide variations in system utilization: some applications (for example, retail shopping) are seasonal; other applications (for example, training applications) run for a short period of time before being shut down; others are simply unpredictable. The cloud addresses this variability (shown in Figure 1-1) perfectly via its “pay for what you use” model.



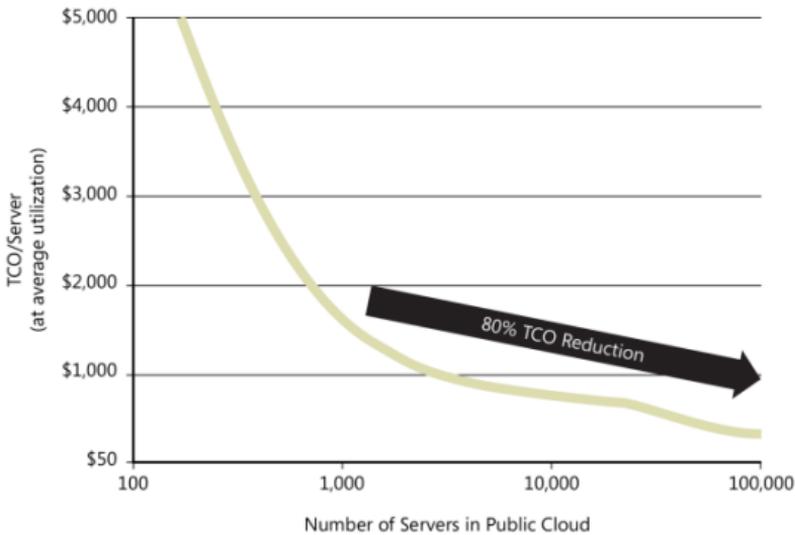
**Figure 1-1:** Common application utilization models

(It is worth mentioning that in the on-premises datacenter, the *maximum* utilization must be planned for and provisioned, which is financially far more inefficient than in the cloud.)

But, there is more to it. Operating in the cloud frees enterprises of the mundane tasks of system backup, network maintenance, patches, and software upgrades, because the cloud provider can handle these in their entirety. The cloud provider in turn is heavily incented to utilize and

in many cases pioneer best practices for system maintenance; the benefits are then passed to the customer.

Moreover, cloud providers such as Microsoft can achieve economies of scale by buying hardware in massive bulk, tens of thousands of servers at a time, for example. Very large datacenters hosting public clouds can also achieve economies in purchasing other resources; cloud datacenters pay only a quarter of the average cost of electricity in the United States. Figure 1-2 shows how overall total cost of ownership (TCO) per server declines dramatically at scale.



**Figure 1-2:** Economies of scale in the cloud

These savings can, and are, passed on to customers of the cloud service.<sup>1</sup> Indeed, although an IT department can certainly create a *private* (internal) cloud of a thousand or so servers, using the *public* cloud can result in savings up to ten times!<sup>2</sup>

Later, we will discuss how IT departments can quantify the savings they can expect to achieve by adopting cloud computing.

Perhaps most important, the cloud is not an “either/or” proposition. It is certainly possible, and indeed in many cases desirable, to leave some applications running in a local, traditional datacenter while others are migrated to the cloud. Providers such as Microsoft have made huge investments in this *hybrid cloud* model that securely connects applications in the cloud to those remaining in a customer’s datacenter.

---

<sup>1</sup> See “Economics of the Cloud” at <http://news.microsoft.com/download/archived/presskits/cloud/docs/TheEconomics-of-the-Cloud.pdf>

<sup>2</sup> John Rex, CFO Microsoft North America, analyst presentation

# Daily efficiencies

After there is an on-demand computing service available, all sorts of other efficiencies become possible. For example, systems devoted to development and application testing often constitute a large cost area for IT departments, yet in the end do not actually provide any direct value to end users. With the cloud, developers and testers can quickly allocate cloud-based resources, use them for their work, and then free them up when done. Similarly, with the vast, capacious amounts of cheap storage available in the cloud, data backup to the cloud, and across multiple geographies if desired, becomes a straightforward and inexpensive function. We will cover more of these in the course of the book.

## Innovation

Of course, at the end of the day, the goal of any enterprise strategy is to create competitive differentiation and advantage, and little doubt remains that IT has become a key element in modern strategy. IT now drives transformative innovation, enabling enterprises to compete more effectively by instantiating processes that deliver ongoing competitive advantage.

As we will see, the emergence of a global computing cloud heralds the arrival of entirely new classes of innovation across applications and markets. Indeed, such new forms of innovation can actually *transform* an organization, and a business.

Transformational innovation drives a different culture and mindset than most organizations currently have. Affecting both IT and the leadership of the enterprise as a whole, this culture requires a close alignment between IT and business leadership.

In the next few pages, we will examine a number of case studies from various global companies, all of which have reaped rewards by their use of the cloud. The first, Telenor, shows how even a simple migration of on-premises applications can make it possible for it to be far more cost-effective in its operations. The second, Aviva, demonstrates how an insurance company used mobile phones and the cloud to create an innovative approach to dynamic insurance pricing. In a third case study, 3M Parking Systems opted for a cloud-based solution when it needed a way to track its thousands of devices in the field. Lastly, the beverage giant Heineken chose the cloud as a way to deliver a global media campaign tied in with the release of a

major motion picture, and the following year, with major sporting events.

## Telenor

The Telenor Group is a Norwegian telecommunications company with worldwide operations serving almost 150 million mobile subscribers. To modernize its intranet and collaboration sites and deliver better search within and across business units, Telenor will soon migrate from the Microsoft SharePoint 2007 web application platform to SharePoint 2013. With 13 different business units spread across 12 countries, Telenor's prime business objectives were to improve collaboration and best-practice sharing, bolster process efficiency, and facilitate a more agile and responsive organization. Through the enhanced capabilities of SharePoint 2013, Telenor could also reduce significantly the complexity and maintenance cost of the 150-plus custom features installed on its IT network. That network, with approximately 40,000 users, utilizes two SharePoint farms to support more than 20,000 site collections, 70 web applications, and 100 content databases.

Telenor's IT structure is highly distributed across its business units, with ownership of corporate-wide initiatives held by a central team at the

Telenor Group level. The various business units are empowered to make the best decisions for their particular business. Although this structure fosters initiative, speed, and agility, it also results in decreased company-wide integration between business units (the classic IT tradeoff between individualized flexibility and central control). Telenor sought to balance and mitigate this tradeoff by modernizing its shared infrastructure and processes so that its business units could still function independently but stay within an efficiently managed, cohesive, company-wide infrastructure.

To accomplish this goal, a Telenor project team estimated that more than 80 servers, plus additional servers for load and scale testing of the architecture, would need to be brought online as part of developing, testing, and running demos of SharePoint 2013. The time and cost of getting this huge infrastructure up, tested, and operational was estimated to exceed any reasonable timeframe and budget using Telenor's standard IT approach. Furthermore, a SharePoint project of this size required a significant amount of server infrastructure across all environments, as well as inclusion of Microsoft Active Directory and Exchange.

The project team quickly realized that it needed to take a different approach.

Microsoft Azure Virtual Machines, which are built on Azure's Infrastructure as a Service (IaaS), made it possible to quickly create the development and testing environment essential to Telenor's successful deployment of SharePoint 2013. The expected three-month window to set up the environment for a system of this size was reduced to two weeks, a huge savings not only in time, but in costs.

SharePoint 2013 is a sophisticated product that integrates with other Microsoft products such as Windows/IIS, SQL Server, and Active Directory. With SharePoint 2013, both a scale-up and a scale-out strategy can be adopted, meaning that the different parts of SharePoint can have multiple instances across different servers, either virtual or physical. For example, if a network needs more search servers, it's simply a matter of adding more servers running these components. For Telenor, development and testing SharePoint 2013 in Virtual Machines meant that IT could scale resources up or down, quickly and easily, with no up-front capital expense.

"Because of Microsoft Azure Virtual Machines, Telenor saved 70 percent on test, development,

and demo that could be turned off when finished to minimize its capital outlays,” says Marius Pedersen, associate systems architect, Microsoft, Norway. “They loved how quickly they were able to implement, and the scalability of the solutions, all without the need for a huge capital investment. There was simply nothing else that could solve their overall big picture for this deployment like Virtual Machines.”

“Testing a big new deployment like this is essential to success, but development and testing can take up a lot time and it normally requires that we buy many extra servers that, once testing is concluded, we don’t really need anymore. And that costs us considerable money and other resources,” declared Andreas Høgberg, director, Telenor.

## Aviva

A leading provider of insurance, savings, and investment products, London-based Aviva serves 43 million customers worldwide. The company wanted to design an innovative pricing model that would reduce premiums for the appropriate customers, but first it needed a better understanding of driving habits. Traditionally, car insurance premiums were determined not just by

the driver's history, but also by statistical probabilities, including age and gender.

Aviva sought a better approach. "We wanted to give people an individual price," says Jason Vettraino, application architect at Aviva. "We didn't want to say 'You're in your forties, so you must drive like my dad.'"

Until very recently, this kind of approach would have required purchasing and installing individual black boxes in vehicles to collect the data and transmit it back to the company's datacenter, which would need to be scaled-out to handle the increased storage and computing capacity needed to process all the data. The expense of this approach would have been prohibitive.

Advances in consumer mobile devices and cloud computing opened up new opportunities, and Aviva realized it had alternatives to building out its datacenter and installing black boxes in its customers' vehicles. "Suddenly in 2012, all of the constraints we faced before had eased," says Vettraino.

Aviva looked for a hybrid cloud-based solution that would take advantage of its customers' own mobile phones. The company needed a flexible,

highly scalable infrastructure that would integrate with its existing on-premises quote system as well as external web-based services and secure applications running on those mobile phones.

Aviva began refining its rating algorithm and strategy for integrating social networks such as Facebook and Twitter. Next, the company worked with Microsoft Visual Studio 2010 and the Microsoft .NET Framework 4 to build its Aviva Drive app for mobile phones. The development platform included Azure SQL Database and the Azure Table storage service. Developers also used tools in the Azure SDK—including Azure Compute Emulator and Azure Storage Emulator—to test the solution.

In February 2012, Aviva began a 90-day trial project with a fleet of commercial vehicles. The vehicle operators used the app to compete against one another and evaluate performance, while Aviva tested scalability and data accuracy. Satisfied with the results, the company then worked on integrating the app with its on-premises quote system.

In July 2012, the company released Aviva Drive in a consumer pilot project that initially captured driving data from a phone, stored the

information in Azure, and connected the mobile app to the company's website for insurance quotes. Three months later, the app had real-time connectivity with Azure so that it could collect telematics information for the quote process. Aviva officially launched Aviva Drive in November 2012.

"By using a Microsoft Azure-based solution to learn more about our customers' driving, we can help them save money," says Vettrano. "After drivers use our application and receive a score, we can give them up to a 20 percent discount on their premium."

Being able to deliver a unique, personalized pricing model based on data collected by the customers' own mobile devices (protected with Microsoft Azure security and privacy controls) provided a major competitive advantage. The fact that this entire system was accomplished in such a short amount of time is indicative of the benefits of the close collaboration between the business process drivers and IT.

## 3M Parking Systems

Minneapolis-based 3M Parking Systems had recently purchased parking, tolling, and automatic license plate reader businesses and

required better insight into these acquisitions. Chad Reed, global business manager for 3M, says: “With thousands of installations across the world, we couldn’t keep track of our software and hardware deployments, which made it difficult to understand our market penetration.”

3M wanted a tracking application that sales staff could use to get real-time information about the type and location of 3M products in parking lots and garages. The solution had to provide access to data anytime, anywhere, and from an array of mobile devices so that it could be used on site with potential customers.

The company chose Azure Mobile Services for a secure, scalable platform that would easily integrate and store data from 3M equipment and other sources. It created native apps that run on multiple mobile operating systems to display real-time information about 3M installations around the United States. Whenever a salesperson enters new data, the information is immediately available to others in the field through Azure Notification Hubs, a push notification engine in Mobile Services. The solution also takes advantage of the mapping and GPS technology built in to each mobile device to automatically provide highly visual, location-specific information.

In just two days, 3M created a tracking solution that connects multiple types of mobile devices, thousands of machines and data sources, and a cloud platform. The 3M team credits its success to a streamlined development environment.

"Integration with Xamarin Studio and Visual Studio, along with built-in functionality, made Azure Mobile Services the best choice for a mobile-services back end," says Jason Fox, mobile application architect at 3M. "Having the right tools and capabilities to put a stable, robust, and functional solution together in two days is a great story."

"The platform provides us with an opportunity to quickly scale a full solution and provide updates within a very short response time," says Jason Rivera, manager of product development at 3M. "The benefits of the Azure Mobile Services platform place the power in the hands of our development team."

With real-time access to data on mobile devices, 3M sales teams can work more efficiently. "The number-one benefit to the sales teams is ease of use," says Fox. "With apps powered by Azure Mobile Services, they can immediately see where we have equipment installed without having to call a home office."

## Heineken

Heineken, which sells its flagship beer in 178 countries, has long run innovative marketing campaigns around the world. Traditionally, its marketing operation had been decentralized. The campaigns might have been global, but their implementations were not. Those decisions had been left to the company's national and regional marketing divisions. Rollout dates, for example, were left to the divisions and, consequently, global campaigns were launched gradually over a period of months.

For a worldwide promotion based on the release of the James Bond movie *Skyfall*, Heineken wanted to launch the campaign at the same time everywhere on the planet. That created unprecedented challenges, particularly given that the primary digital content for the campaign was a 100-megabyte movie that had to play flawlessly for millions of viewers around the globe.

Previously, Heineken had supported digital media at its outsourced datacenter. But, that datacenter lacked the computing resources needed for such a global event, and building them—especially to support peak traffic that would total millions of simultaneous hits—would

have been both time-consuming and expensive. Nor would it have provided the geographic reach that Heineken needed to minimize latency worldwide.

To help deliver that successful campaign, the company used the Azure Content Delivery Network (CDN) to make the digital content available quickly, reliably, and globally to 10.5 million consumers.

The next year, Heineken faced another digital marketing challenge. This time, Heineken based a global campaign on UEFA Champions League (UCL) soccer games. The campaign would launch simultaneously in more than 70 markets and 30 languages. It would require not only that the company host a giant website to serve content, as the *Skyfall* campaign had, but the UCL campaign also needed real-time computing on a global scale.

That's because the centerpiece of the UCL campaign was a pinball game for consumers to play live against other players anywhere in the world. The solution had to support multiple leaderboards for each player, based on the number of friends and family that an individual played with, and the leaderboards required real-time updating, Heineken wanted the technology

to support one million simultaneous users. And, in the words of Lennart Boorsma, digital marketing manager at Heineken, “It couldn’t fail.”

To meet these requirements, Heineken expanded its use of Azure from one datacenter to four—one each in Europe and Asia, and two in the United States—gaining geo-redundancy and low latency. Data was stored in Azure Table storage for asynchronous updates. The storage was structured with 10,000 partitions—up from 10 initially—for the requisite scalability. Heineken developed the solution using Microsoft Visual Studio 2013. The architecture was tested by using a Visual Studio load-testing cluster. Microsoft Services consultants helped develop and load-test the solution and resolve performance issues.

Heineken used Azure to achieve 100 percent reliability on a massive scale. The platform exceeded its service-level agreement with perfect performance in the UCL campaign, supporting 2 million gameplays per hour with capacity for more than 40 million players in all.

“Azure didn’t let us down,” says Boorsma. “More than that, it gave us a way to assure senior management that we could support this massive,

global campaign. It put our stakeholders at ease, knowing that we had them covered. When you bring out a global campaign with such bravura, you really do need to make sure that all your homework is done. With Azure, it was.”

## Learnings

What have we learned? These examples demonstrate the potential offered by the cloud; customers could:

- Build and rapidly deploy applications with reach and scale that would have been impossible from their own datacenters;
- Take advantage of Internet-connected devices all over the world;
- Tap into Big Data and analytics services for personalization, better products, and more efficient processes;
- Enjoy unprecedented development, test experimentation, and innovation cycles.

Every IT department is charged with safeguarding its company’s information assets; and this function is, and always will be, a critical component. Yet IT must also enable and foster innovation, both to make existing processes

faster and cheaper as well as to support new and emerging business models.

With the cloud, the balance between maintenance and innovation shifts. As we shall see, operating in the cloud provides many cost advantages, allowing IT departments to focus more on innovation. Running in the cloud can reduce the need for rote operations such as system software upgrades and patching, thus permitting IT to redirect staff at revenue-centric activities. And, new capabilities in the cloud make new kinds of powerful applications possible. As we have seen in the preceding examples, more and more companies now see the cloud as a way to accelerate business innovation and competitive differentiation.

But, as with any great technological change, this kind of transformation cycle involves much more than pure technology. It also requires a shift in corporate culture, enterprise and IT processes, individual roles, governance, and (for that matter) engineering. How an enterprise achieves this transformation is the subject of the remainder of this book.

# Journey to the cloud: the roadmap

What if you were able to achieve both efficiency and innovation in all the business domains and applications across your entire portfolio? What if you could take advantage of the cloud and all of its resources and features to get a “the whole is greater than the sum of its parts” effect? With a good roadmap to lead the way, you can. This chapter covers what it means to move your enterprise to the cloud. We’ll provide examples and learning experiences from

## Microsoft's own journey, as well as from those of our customers.

In any transformative change, it's important to understand what the destination is and what the waypoints along the journey will be. There are multiple potential destinations for any application, and IT cloud deployments will be a mixture of them:

- **Private cloud** In a private cloud, cloud technologies are hosted in an on-premises datacenter. Private clouds can be useful because they can implement a technology stack that is consistent with the public cloud. This might be necessary in scenarios for which certain applications or data cannot be moved off premises. However, private clouds do not provide the cost savings and efficiencies that the public cloud can, because private clouds require a significant capital expense budget and a (potentially large) operations staff.
- **Infrastructure as a Service (IaaS)** In IaaS, the application virtual machines (VMs) are simply moved from on-premises to the

cloud. This is the easiest migration strategy and has many benefits, including cost savings. But, it still means that your operations staff will need to perform such tasks as patch management, updates, and upgrades. Nevertheless, IaaS is one of the most common cloud deployment patterns to date because it reduces the time between purchasing and deployment to almost nothing. Additionally, because it is the most similar to how IT operates today, it provides an easy onboarding ramp for the IT culture and processes of today.

- **Platform as a Service (PaaS)** In PaaS, the cloud provider maintains all system software, removing the burden of upgrades and patches from the IT department. PaaS is similar to the traditional three-tier model of enterprise software, having a presentation layer (called “Web Role”), a business logic layer (called “Worker Role”), and persistent storage (Microsoft Azure SQL Database or other database). In a PaaS deployment model, all that the enterprise needs to focus on is in deploying its code on the PaaS machines; the cloud provider ensures that operating systems, database software, integration software, and other features are

maintained, kept up to date, and achieve a high service level agreement (SLA).

- **Software as a Service (SaaS)** In SaaS, you simply rent an application from a vendor, such as Microsoft Office 365 for email and productivity. This is by far the most cost-effective of all the options because typically the only work involved for the IT department is provisioning users and data and, perhaps, integrating the application with single sign-on (SSO). Typically, SaaS applications are used for functions that are not considered business-differentiating, for which custom or customized applications encode the competitively differentiating business models and rules.
- **The hybrid cloud** Many enterprises might choose to keep some applications on-premises—perhaps they are based on nonstandard systems or out-of-date software, or perhaps they will remain on-premises while waiting for their turn to be migrated to the cloud. In this model, some applications run in the cloud, whereas others remain on-premises, requiring a secure, high-speed communications path between the two environments. In a way, then, the

cloud becomes an extension of the existing datacenter, and vice versa.

## Don't miss the opportunity to modernize

Before we go on, it's worth noting that the cloud provides an opportunity to consider the IT ecosystem as a whole and how it can be *modernized*. As you shall see, cloud migration at scale involves looking at each application and determining how it should be thought of in this new environment called the cloud. Is further investment in certain applications justified? Should they be retired?

Many enterprises have held their applications for far too long without assigning to them a maintenance or retirement schedule. Therefore, for fear of complexity, lack of documentation, resources, source code, or other reasons, applications remain untouched.

Even for applications that remain on-premises, modernization can save time and money. An internal Microsoft IT study in 2010 demonstrated that the number of problem reports ("tickets") and the time to resolve them increased with the

age of the application and system software. (This analysis led to a focused effort to ensure that all applications were on the latest version of the operating system and other systems software such as database.)

Moreover, and more important, migration to the cloud provides an opportunity to evaluate and modernize applications and, in particular, their business logic. This activity can provide great returns on investment and impact to the top-line revenue.

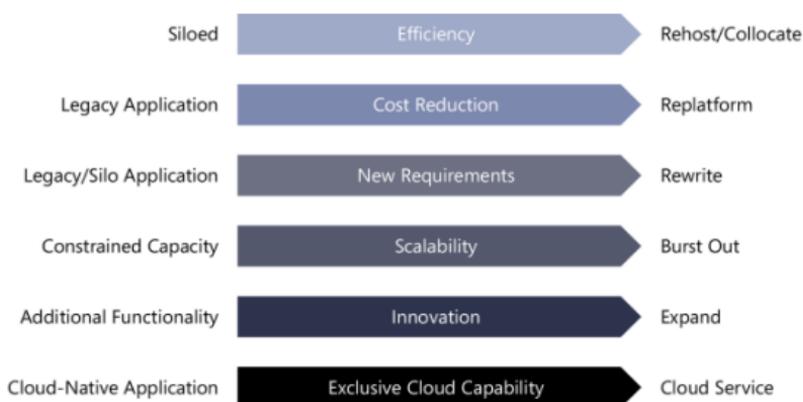
There are many motions that one can take to modernize application and services portfolios (see Figure 2-1), such as the following:

- **Rehost** Move a VM or an operating environment from the on-premises datacenter to a hoster or a cloud. This model is also known as *co-location*.
- **Replatform** A legacy environment becomes unsustainable based on cost or operational requirements; a solution is to “retain and wrap” the application without making changes to the code, possibly compromising the integrity and security of the operation.

- **Retire and Rewrite (or Reenvision)** When there are sufficient new requirements that cannot be met by the older environment, the best way to proceed is to rewrite the application in a newer, better-suited environment. Often this occurs when examining the portfolio of applications and consolidating several that have similar function.
- **Burst out** With all of the new compute, data, and service models that are being provided in cloud environments, each providing capabilities and capacities that were never before accessible to an IT environment, many applications are bursting out to the cloud. These applications are doing innovative types of analytics, reporting, high-performance computing, visualization, and so on. Keeping frequently used (“hot”) data locally while aging-out infrequently accessed (“cold”) data to far cheaper cloud storage is another common pattern.
- **Expand** Enterprises are now exploring how to expand their older applications and how to add functionality to provide to mobile devices and web front ends the same capabilities that previously were limited to a

computer screen. They are even moving to enhance the applications with search or video services, as an example.

- **Cloud-Native Applications** As companies begin their investigation of the cloud, they frequently realize that there are new forms of applications like Big Data, new types of analytics, entirely new capabilities such as machine learning, and applications for the Internet of Things (IoT) that are uniquely fitted to live in the cloud.



**Figure 2-1:** Types of modernization initiatives

## Evolution of the five R's of modernization

To focus our efforts on guidance for existing applications, let's proceed with the most convenient way to think about modernization,

which is commonly called “the five R’s”:<sup>3</sup> retire, replace, retain and wrap, rehost, and reenvision. It’s likely that no single approach will be appropriate for all of an enterprise’s legacy applications, and a mix of differing approaches might be warranted, based on the value that an application delivers versus the cost of any given approach. Because these approaches depend highly on the situation, application, and types of cost involved, there is no one-size-fits-all solution.

- **Retire** Of course, if a legacy application is providing little value compared to its costs, the enterprise should consider it a candidate for retirement. When few people are using an application relative to its cost impact, the enterprise needs to run a cost-benefit analysis to determine if it is worth the expense. Additionally, some functionality provided by legacy systems may be rolled into a consolidated modern application running in the cloud, allowing some applications to be retired while others are replaced and modernized.

---

<sup>3</sup> Based on “Gartner Identifies Five Ways to Migrate Applications to the Cloud,” Gartner Inc, 2011.

<http://www.gartner.com/newsroom/id/1684114>

- **Replace** Often, a legacy application is providing some value, but an off-the-shelf replacement with a lower total cost of ownership (TCO) is available. Many legacy applications were originally built because there was no alternative at that time. A modern, readily available application that is better suited to running in the cloud—most cost-effectively of all, a SaaS application—may now exist that can be used to replace the older one. Also, when a legacy application is replaced with a more comprehensive modern solution, there might be a chance to consolidate functionality from several older applications, thereby replacing multiple applications with a single system.
- **Retain and wrap** If a legacy application is providing good value and not incurring a high TCO, the best approach might be to retain it but put a modern “wrapper” around it in order to gain additional value and benefits. Examples of the “retain and wrap” approach include the following:
  - Wrap a legacy application within C# in Microsoft Visual Studio, add web services to the application there, and

then add a layer of orchestration around those web services.

- Extend a legacy application with third-party tools; for example, using a C# wrapper around an older technology such as COBOL. Apply the benefits of the wrapper on top of the core technology in new, more modern ways, such as facilitating the development of mobile tools.
- **Rehost** If a legacy application is providing good value but is expensive to run, it might be a candidate for rehosting. Rehosting involves keeping the same basic functionality, but moving it to the cloud where it is easier to manage and less expensive to run. This is also called “lift and shift.” In a rehosting situation, the legacy application might be currently located either on a local VM or on local hardware. Some VMs might be eligible to move with a simple migration. Those on local hardware might be able to be converted with a physical-to-virtual migration and then hosting the VM on the cloud. Some VMs, especially older ones, might not migrate easily to the cloud without some significant work. In those cases, you might want to consider

reenvisioning and building the application in the cloud.

- **Reenvision** If a legacy application is providing good value but cannot be easily migrated, the best solution might be to reenvision it and build it again on the cloud. Reenvisioning is a process of rebuilding the application in the cloud using modern technology, a new architecture, and best practices; it normally also involves adding more business value to core functionality, such as improving market differentiation. Reenvisioning an application might require rewriting the main logic by using a modern development language and tools and making it service oriented. Reenvisioning an application can be facilitated by starting with VMs in the cloud, which can be instantiated in a matter of minutes.

## Cloud migration: three stages

When planning migration to the cloud, there are many ways to think about a roadmap. From our experience, however, we've seen three basic stages: *experimentation*, *migration*, and *transformation*.

In the essential experimentation phase, two processes take place. In the first, the engineers and others create the IT department's first cloud applications, with the objective of learning what the cloud is all about: how to develop for it, how to test, how to deploy, and how to monitor and maintain a cloud application. Concurrently, businesses and IT departments envision the art of the possible; design new solutions to demonstrate how to advance the status quo; and envision a newer, expanded, more agile and better application or service.

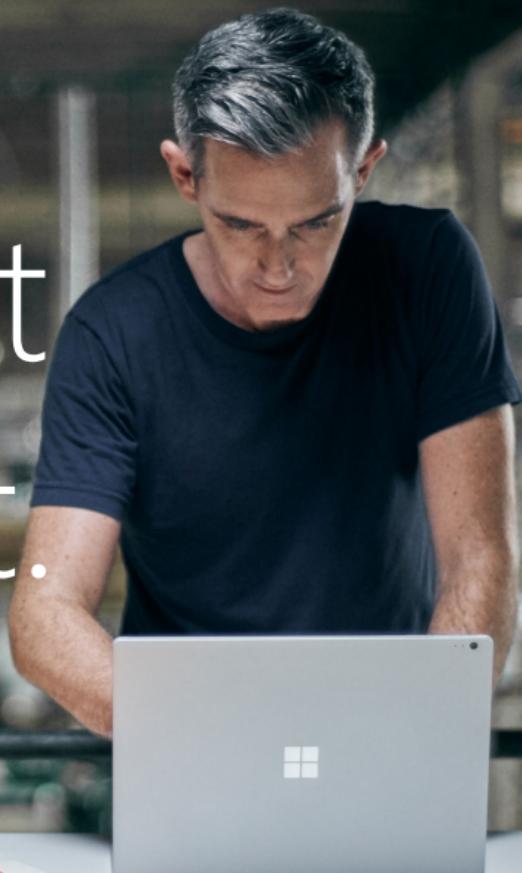
In the migration phase—which in many ways is the most demanding of the phases—the bulk of the IT portfolio is moved to the cloud in one form or another. This requires cooperation and collaboration across a number of different enterprise functions, including the technical staff, the operations staff, as well as the executive team, business sponsors, security professionals, regulatory compliance staff, legal, and HR.

In the transformation phase (which will often be coincident with the migration phase), selected applications are redesigned to take maximum advantage of the cloud—using the PaaS model—affording greater scale, greater integration with other cloud services, and numerous other advantages.

Moving forward, the now–cloud-native applications can take advantage of cloud services such as machine learning, big data, streaming analytics, and many others—making them much, much richer in function and feature than before.

The following chapters cover each phase in detail.

# Hear about it first.



Get the latest news from Microsoft Press sent to your inbox.

- New and upcoming books
- Special offers
- Free eBooks
- How-to articles

Sign up today at  
[MicrosoftPressStore.com/Newsletters](https://MicrosoftPressStore.com/Newsletters)



# Experimentation

There is always a first cloud application. In every IT organization, some brave soul will either move an existing application to the cloud or create a new one there. In so doing, this person will gain an understanding—beyond all the hype—of what developing, testing, deploying, and maintaining a cloud application is all about.

## Microsoft IT's first cloud application

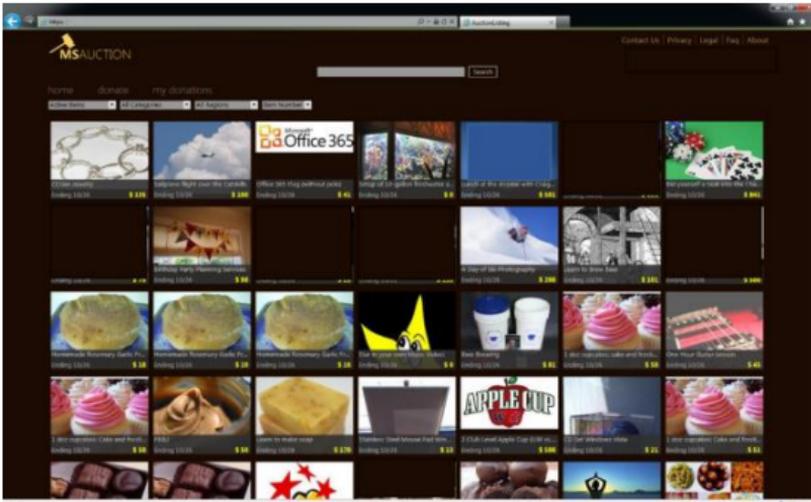
Microsoft IT developed its first cloud application in 2010. It was an employee auction application, used once a year as part of the Microsoft

charitable giving campaign. With it, employees donate items (ranging from mentoring sessions, to cooking classes, to software, and even the use of an executive's car for a day!) and others buy them, with all the proceeds going to charity. The auction, typically held in October, runs for a month.

Why did we pick this as our first cloud application? A number of factors led us to this decision: first, it was *not* a business-critical application. Therefore, news of any application problems would not cause damage to the company's finances or reputation or appear on the front page of any newspaper.

Second, we could see the scalability features of Microsoft Azure in action. As the end of October approached, traffic on the application continually rose, reaching a peak in the last few days of the auction.

Finally, it was a relatively simple application whose deployment in the cloud did not require updating other applications in concert.



**Figure 3-1:** Microsoft internal auction application, circa 2010<sup>4</sup>

In the end, the application was very successful and the auction met its goals (incidentally, over the years, Microsoft's employees have raised more than one billion dollars for charity). Microsoft IT learned many lessons on cloud development and deployment, which we used in subsequent stages of our own journey. We saw the application easily scale to meet the increased demand during the course of the month. At the end of the auction, we could shut it down and no

---

<sup>4</sup> Mentoring, tutoring, and personal one-on-ones with executives are always among the items offered for auction. In the interest of privacy, we have removed the faces of the individuals offering these sessions from the screen shot; hence, some squares are blank.

longer pay for resources required to run it (as we would have—for servers, operations staff, and so on—had we run the application in our own datacenter). By every measure, then, this first experiment was a success.

There were many other early experiments in this period, trying out new approaches, testing new features, and so on; we learned that developing a “culture of experimentation” was useful in that we could be continuously trying new things and innovating.

## Experimentation and the problem of “shadow” IT

IT departments often live in a world of contradiction. On the one hand, they must “keep the lights on,” by keeping servers and networks up, by delivering reports on time, and by ensuring that systems and data meet regulatory obligations such as Sarbanes-Oxley and other forms of compliance. These requirements are nothing if not rigorous—and essential.

On the other hand, they and their business partners desire innovation: new programs and new applications to support both new and

evolving business opportunities, to better serve their customers, and so forth. Yet the costs of IT operations—sometimes 70–80 percent of the overall budget—reduces the ability of IT to spend on new programs and innovation.

In many cases (in fact, in every enterprise we know), there are occasionally applications created and deployed outside of the IT department in response to critical business needs. These unofficial applications are often referred to as “shadow” IT. Instead of going through the usual budget, requirements analysis, design and deployment phases typical in the creation of a new IT application, a marketing department publicizing a new campaign might simply create a new website on their own.

Because it eliminates the capital-expense investment component (i.e., servers, storage, and network) of application development, the cloud makes this sort of rapid innovation much, much easier. In effect, all that is needed are a few coders to write the application—and a credit card.<sup>5</sup>

---

<sup>5</sup> The elimination of such capital expenses has greatly accelerated the pace of startups, as well.

IT executives should realize that this sort of innovation and experimentation is inevitable, and in many cases actually desirable. As the business climate rapidly evolves, it is critical for both businesses and IT organizations to foster rapid experimentation and innovation. It will be important to educate businesses on the importance and consequences of regulatory issues and noncompliance, of course. IT departments can actually help them by providing controlled, managed access to critical data, such as customer information, rather than letting them gather and manipulate the data on their own.

As soon as a company starts this process of envisioning and creates the culture of experimentation, it learns a disruptive truth: in the cloud era, you must experiment, fail fast, and learn fast. It is as important to experiment in order to learn quickly both from successes and from failures. Learning from how you succeed and what makes you fail provides the basis for delivering the disruptive innovation and value from the cloud.

As you can by now expect, these phases shape the cloud migration principles used for the rest of the process, these principles are *go fast, push the boundaries, make data-driven decisions,*

*simplify*, and, finally, *communicate* to succeed. Table 3-1 provides an overview of them, followed by detailed descriptions of each.

**Table 3-1:** Cloud migration principles

<b>Go Fast</b>
<ul style="list-style-type: none"><li>• Fail fast, learn fast</li><li>• Try many, use best</li></ul>
<b>Push the boundaries</b>
<ul style="list-style-type: none"><li>• Design new applications and capabilities for PaaS/SaaS</li><li>• Refactor legacy apps for PaaS/SaaS</li><li>• Build your plan-of-record to take advantage of cloud capabilities</li><li>• Think “Experience”</li></ul>
<b>Make data-driven decisions</b>
<ul style="list-style-type: none"><li>• Manage your costs</li><li>• Use telemetry to gain insight into operational efficiency</li><li>• Understand your blockers</li><li>• Manage your plan-of-record</li></ul>

## Simplify

- Retire, retire, retire legacy applications wherever possible
- Aggressively right-size
- Review frozen and cold servers weekly
- Clean up Configuration Management Database (CMDB) data

## Communicate to succeed

- Communicate customer and stakeholder impacts—transparency is key
- Share learnings and best practices

- *Go fast* exemplifies the spirit of the experimentation phase. For some, it might represent a new way of thinking for IT because, with the cloud, you can “spin up” new projects quickly with a few clicks rather than having to plan, allotting datacenter space, procuring equipment, and so on. We call this the *try many, use best* approach because the cloud uniquely facilitates the ability of IT departments to choose the best of many solutions.

- *Push the boundaries* suggests that wherever possible, IT should not simply adapt to the new paradigm of the cloud, but embrace it and adopt new architectures and processes quickly to best exploit the new opportunities.
- *Make data-driven decisions* proposes that you carefully track and measure the numbers, including the cost effectiveness of the cloud for financial reasons, system telemetry for technical efficiency reasons, and so on. Following the data carefully will make it possible for you to make informed decisions about which applications are generating the most return, about which you should prioritize, about which are performing well in the cloud, and where potential problem areas exist.
- *Simplify* focuses on retiring, right-sizing, and consolidating as many services and applications as possible. Applications that are infrequently or rarely used often generate significant costs for an IT organization, with little return. Retiring them and consolidating them with applications that perform similar functions can, conversely, generate savings in a number of areas such as hardware, system software

licenses, and maintenance. Consider generating metrics around “hot” and “cold” applications based on CPU, network, and database utilization; for example, an application that averages two percent of CPU and has few authenticated users might be just such a “cold” application.

- *Communicate to succeed* is the single most important mechanism that guarantees continued success and not just the migration of a single application or a service. Establish a clear and continuous communication channel for stakeholders to visualize success and impact as well as to understand the failure and the lessons learned from them. Key stakeholders remain engaged and continue to invest when they feel their participation in the joint effort required to make this a continuous journey and not just a single trip. These lessons set us up for the migration phase, which we cover in Chapter 4.

# Migrating IT to the cloud

Sooner or later, it becomes obvious that running a large number of the IT portfolio—perhaps even the majority of it—in the cloud makes sense from a variety of perspectives. In most cases, running in the cloud provides substantial cost savings; reduces or eliminates the need for an enterprise to maintain its own datacenters; reduces or eliminates the need to manage hardware and software updates; and enables the sort of innovation we discussed in

Chapter 1. The cloud is very compelling; yet the migration phase typically involves many more applications and many more people, and potentially impacts more of IT's customers than any other—by far.

It can be daunting when a large enterprise IT department manages hundreds or thousands of applications running on perhaps tens of thousands of virtual machines (VMs). Which ones to move first? How to prioritize? How does operating in the cloud affect regulatory compliance, data security, and enterprise processes? What does it mean for organizational roles, training, and change management? And, last but certainly not least, how to do all this while continuing to serve the business?

Where to begin?

In the next few sections, we describe how to establish strategy and goals for a cloud

migration activity; what roles the various organizations in the enterprise play; how to prioritize application migration; and how to extend IT governance to cover the cloud.

## Establish strategy and goals

Every journey must have a sense of its destination, its route, and when it will arrive; the migration journey to the cloud is no different. It is time well spent to engage senior members of IT to understand all aspects of the cloud and which of the many options and approaches to take.

With Microsoft IT, as in many enterprises, the journey began with the creation of a Cloud Strategy Team, driven (in our case) by the CTO and consisting of members of the enterprise architecture team, IT finance, the most senior technologists from the various IT applications groups (HR, finance, and so on), and similar leaders from the infrastructure, security, and

networking teams. Figure 4-1 shows the structure of the Cloud Strategy Team.<sup>6</sup>



**Figure 4-1:** The Cloud Strategy Team at Microsoft IT

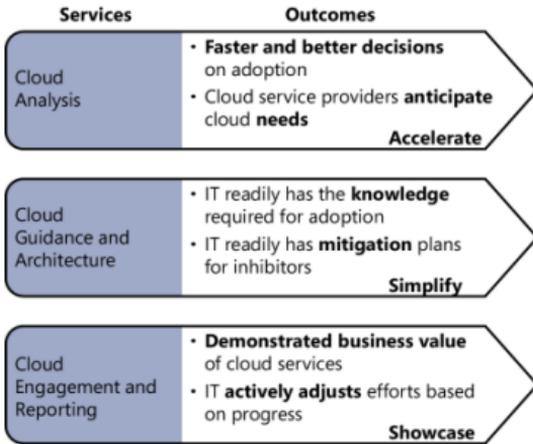
The Cloud Strategy Team was chartered to lead the cloud analysis and experimentation phase previously described. In addition, it built (or facilitated building) the architectures, patterns, and guidance for deployment of the re-envisioned applications or services to finally manage the communications to key stakeholders and promote the success and learnings from the program. The creation of this team is one of the key forcing functions to promote

<sup>6</sup> Microsoft IT’s “First and Best” team ensures that Microsoft IT is Microsoft’s “First and Best” customer by testing all of the company’s products in IT prior to their general release to the public, a practice often referred to as “dogfooding.”

long-term commitment to the journey. It also establishes a practice for continuously evaluating and experimenting to help determine what is migrated to the most appropriate platform, such as the following:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Services (SaaS)

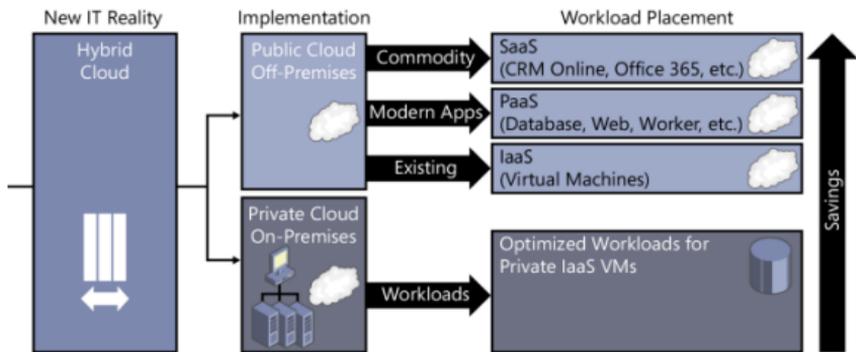
Among its first tasks, the team spent its initial time educating itself and ensuring that all participants were on a “level playing field.” For better or worse, cloud technology comes with its own set of acronyms (IaaS, PaaS, SaaS) and new terms (private cloud, public cloud, hybrid cloud, containers); learning to speak a common language early accelerated future conversations. The team also spent time familiarizing itself with the offerings from platform, tools, and cloud applications providers.



**Figure 4-2:** Cloud Strategy Team charters

When the team began to draft out the strategy, members understood that not all services or applications would end up in the public cloud, for various reasons. The strategy for Microsoft IT, therefore, was based on the notion of a *hybrid* cloud. This meant that at least for some period of time, certain applications would remain on-premises.

On the other hand, we clearly realized that the optimum strategy from an efficiency and cost point of view was to move as many applications to a SaaS model (as shown in Figure 4-3); whereas the least efficient approach (involving the highest cost and most resources) would be to keep them on-premises.



**Figure 4-3:** Hybrid cloud strategy

Each application was analyzed to determine the best fit for its hosting environment. If the workload was to be retired or no longer receive further investment, we evaluated whether we could host it on a public cloud; if not, it stayed in an on-premises private cloud. If the workload could be placed on an IaaS cloud environment, we proceeded to migrate it there to obtain benefits of cost reduction. Later, we will show the mechanics of this analysis.

If the assessed workload could be run as a SaaS application, we would make the appropriate migration path to contract the SaaS service and proceed to migrate, create the enterprise change plan, process, and data migration plan, as well as a comprehensive security and compliancy plan to meet with all appropriate requirements. If the workload was not an appropriate candidate or was simply not offered as a SaaS service, we

created a plan and architecture to redesign the application by using a PaaS platform.

In the following tables and Figure 4-4, we stress the importance of a comprehensive enterprise cloud strategy that takes into account SaaS, PaaS, IaaS, and, finally, a private cloud environment as a whole, and in that specific order because this is the sequence by which efficiency and agility benefits are best realized. (In fact, many companies adopt this as an architectural principle: “SaaS before PaaS before IaaS before Private.”)

**Public SaaS evaluation**

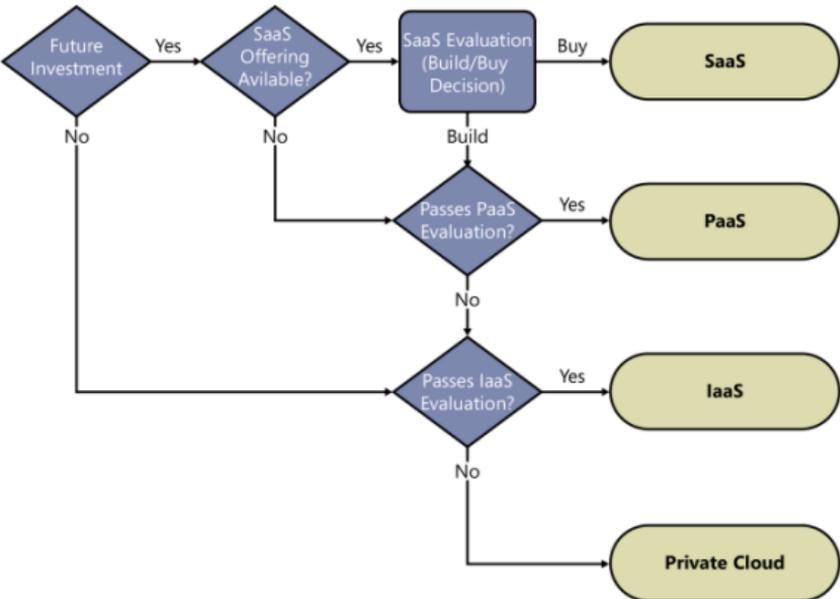
<b>Business factors</b>	<b>Technical factors</b>
Business case (build/buy)	Integration
Competitive technology assessment	Performance and scale
Privacy and compliance	Management
Security	

## Public PaaS evaluation

Business factors	Technical factors
Privacy and compliance	Integration
Security	Resiliency

## Public IaaS evaluation

Business factors	Technical factors
Privacy and compliance	Integration
Security	Connectivity



**Figure 4-4:** High-level workload placement decision tree

The Cloud Strategy Team's deliverable was a document describing the goals of the migration, proposed timeframes, recommended technical strategy (that is, technical platform and tools) and expected results and benefits. For example, the recommendations included statements like the following:

- Majority of existing applications will be moved to IaaS virtual machines.
- To take advantage of scalability and other features, new applications and major releases will be (re-) architected as PaaS applications.
- During the transition, on-premises applications will communicate with cloud applications via a dedicated connection (typically MPLS or WAN) line (such as Microsoft's ExpressRoute offering).
- Applications that provide little competitive differentiation (applications that can be commoditized) will be transitioned to external SaaS providers (for example, Microsoft's Office365 for mail and productivity applications).
- Expected cost savings will be  $x\%$  after the first year and  $y\%$  after the second.

- Certain applications will remain on-premises.
- Security will be through combinations of encryption, cloud identity federated with on-premises identity providers (such as Active Directory), and other controls.
- Operations teams will be trained in cloud deployment and systems management in the cloud, and will evolve to a DevOps model (discussed later).
- The document may include different models and options to facilitate discussion and informed choice.

## Organizational responsibilities in creating the strategy

What sorts of models should be built, and how? This section describes how each of the following organizations contributes to the cloud strategy:

- Enterprise architecture
- Information security and risk management
- Data classification

- Enterprise risk management
- Finance
- Operations
- Human resources
- Applications teams
- Business units

## Enterprise architecture

The enterprise architecture (EA) plays a key leadership role in cloud migration. The goal of any EA team is to ensure that the highest business value is received for most efficient use of technology resources; as such, EA provides the essential bridge between business and IT.

Typically, EA maintains the list of IT capabilities and processes, facilitates the creation and implementation of IT strategies, works with businesses and executives to understand the long-term goals of the company in order to plan for the future, and drives various enterprise-wide governance activities such as architecture review. For such reasons, the EA team is an ideal choice to lead the Cloud Strategy Team.

The EA team overseeing the IT ecosystem as a whole is in a position to provide the appropriate analyses of system capabilities and application impacts of any large-scale changes to the ecosystem. Often, it is EA that creates and maintains the portfolio management system (the catalog of applications) from which the prioritization of applications to be moved to the cloud can be drawn (we will have much more to say about this process later). Enterprise architects should examine what is known about the portfolio and where additional information is needed—for example, whether an application is virtualized. The EA team should add this and other attributes to the knowledge base and engage with other parts of IT to collect the data. Other examples of such metadata will be described shortly.

Cloud migration offers the enterprise architect many opportunities. By using modeling techniques such as business capability analysis<sup>7</sup>

---

<sup>7</sup> A modeling technique that analyzes an enterprise in terms of its business capabilities, independent of organization or technology, pioneered by Gartner. See <https://www.gartner.com/doc/1415831/use-business-capability-modeling-explore>. Capability models are just one possible enterprise architecture modeling

and capability maturity models, it might be possible, as the prioritization process for applications takes place, to *simplify* IT by consolidating applications of similar function. Consolidation will have clear financial benefits both by reducing the compute, data, and network requirements, as well as by simplifying the operations and maintenance functions.

The enterprise architect, and in particular the enterprise information architect, can also use the opportunity afforded by cloud migration to analyze the data models used by applications and update them to enterprise-wide canonical models. Such an effort will streamline application integration and reduce semantic mismatches between disparate data models, which often require manual adjustment in a complex on-premises environment.

In addition, it is the EA team's core responsibility to create and maintain as-is and to-be roadmaps of the overall IT ecosystem. The EA team should be able to easily communicate the various stages

---

methodology; you can use others such as the famous Zachman Framework pioneered by John Zachman or Business Process Model and Notation (BMPN), either with or instead of capability modeling.

of the migration, summarizing the current thinking of the Cloud Strategy Team.

Finally, the EA team should direct the investigation into the use of new cloud technologies to either augment existing capabilities and/or provide entirely new functionality to IT applications, and as these are validated, to add these to the existing roadmaps. Enterprise architects need to experiment with new technologies as well as understand and communicate their business value to IT management and business stakeholders. Successful investigations should lead to the development and publishing of reference architectures that applications teams can reuse.

## Information security and risk management

Every major change in the way you conduct business entails some amount of risk; few aspects of the cloud have generated more discussion and controversy than those regarding its security and risk. In this time of breaches, nation-state hacking, and growing and profound concern with individual privacy on the Internet, cybersecurity has become a board-level concern, and rightly so.

Begin by understanding the security postures of the cloud platform providers. Issues to examine include the availability of antimalware software for cloud-hosted applications; the presence of intrusion detection software and tools; sophisticated and secure identity management; at-rest and in-motion encryption options; networking options for on-premises and off-premises communications; the ability to do penetration testing; and so on. The requirement to implement “defense in depth” remains; you will need to determine how you can collaborate with your cloud provider to implement and enhance it.

You should also understand the physical security practices of the cloud provider. Are employee background checks required? Does access to the cloud datacenter require biometric authentication?

Next, because the cloud potentially makes it possible to access corporate computing devices from anywhere in the world, the information security team should address what requirements should be levied on these devices to grant them such access. For example, it might require all client devices to have encrypted local storage by using such technologies as Microsoft Bitlocker. Similarly, because typing usernames and

passwords on mobile devices can be tedious, the team should consider the merits of alternate forms of authentication, such as biometrics. Or, it might choose to implement “multifactor authentication,” requiring both a username/password as well as some other form of identity (such as a smart card).

A related capability in the cloud is its ability to accept authentication credentials from a multitude of sources by using the Open Authorization (OAuth) protocol. Information security professionals should decide which, if any, applications may accept (for example) Facebook or Google credentials. E-commerce sites might benefit from usage of these credentials but internal applications likely would not.

Third, verify key regulatory compliance certifications (for example, HIPAA, the Health Insurance Portability and Accountability Act; FISMA, the Federal Information Security Management Act; and the EUDPD, European Union Data Protection Directive). Different industries and different geographies will be governed by different regulations and standards. Learn how to detect a suspected breach and how to report it to the provider, and what the response time SLA is expected to be. The

Microsoft Azure Trust Center provides details on all of these as they relate to its offering. The Cloud Security Alliance is an excellent independent resource bringing together experts from across the industry to develop recommendations for best practices for secure computing in the cloud.<sup>8</sup>

Even though the cloud provides many security advantages, hosting an application in the cloud does not entirely relieve application writers and security professionals of their responsibilities. We strongly recommend developers and testers adhere to the Security Development Lifecycle (<https://www.microsoft.com/en-us/sdl/default.aspx>), which provides a set of steps for anticipating and mitigating threats. Antivirus and antimalware options should be included in your deployments. Penetration testing of deployed applications should be performed.<sup>9</sup>

---

<sup>8</sup> Azure Trust Center: <http://azure.microsoft.com/en-us/support/trust-center/>

Cloud Security Alliance:  
<https://cloudsecurityalliance.org>

<sup>9</sup> However, you should work with your cloud provider to schedule such testing because it will be difficult for the cloud provider to distinguish between a test and a real attack without advance warning.

Security and risk professionals will also be deeply involved in cloud governance, which is discussed in that section.

## Data classification

The next step is to think about the data your applications can store in the cloud and how they might influence security and risk. Many companies classify their data according to its sensitivity: a marketing document has a very different security requirement than, say, a draft of a 10-K filing prior to earnings release.

One possible schema is to divide data into several categories, based upon the impact to the business in the event of an unauthorized release. For example, the first category would be public, which is intended for release and poses no risk to the business. The next category is low business impact (LBI), which might include data or information that does not contain personally identifiable information or cover sensitive topics but would generally not be intended for public release. Medium business impact (MBI) data might include information about the company that might not be sensitive in and of itself, but when combined or analyzed could provide competitive insights, or some personally identifiable information that is not of a sensitive

nature but that should not be released for privacy protection. Finally, high business impact (HBI) data is anything covered by any regulatory constraints, involves reputational matters for the company or individuals, anything that could be used to provide competitive advantage, anything that has financial value that could be stolen, or anything that could violate sensitive privacy concerns.

Next, you should set policy requirements for each category of risk. For example, LBI might require no encryption. MBI might require encryption in motion. HBI, in addition to encryption in motion, would require encryption at rest. You should also consider creating audit requirements, access control, and other security guidelines based on these categories. The cloud strategy team working with the information security group might, in fact, choose to prioritize applications that manage low-security data (LBI) to migrate to the cloud first because it represents the least risk. High-risk data (HBI) such as customer personally identifiable information (PII) might require a security review before being migrated, whereas LBI applications might not.

## Enterprise Risk Management

If you have an Enterprise Risk Management (ERM) team, work closely with it to determine how the cloud affects its risk models. Most ERM teams have a detailed, documented list of enterprise risks along with the likelihood of them happening and the impact if they do. To address these risks ERM teams will implement controls and establish teams to either remediate or monitor the risk, depending upon its severity. The cloud, as with any significant change, will introduce transformations to the existing risk model as well as new risks, and it is important that these be examined and discussed. For example, in the extremely unlikely case of a cloud datacenter failure, IT departments should consider geo-replicating data to mitigate the risk of data loss.

## Finance

It is imperative to involve your CFO and your enterprise's finance department in developing your cloud migration plan. You will need to work with them on developing cost models comparing on-premises IT operations (in the datacenter) against in the cloud. You'll also need to build models showing how purchasing and procurement of new hardware draws down over

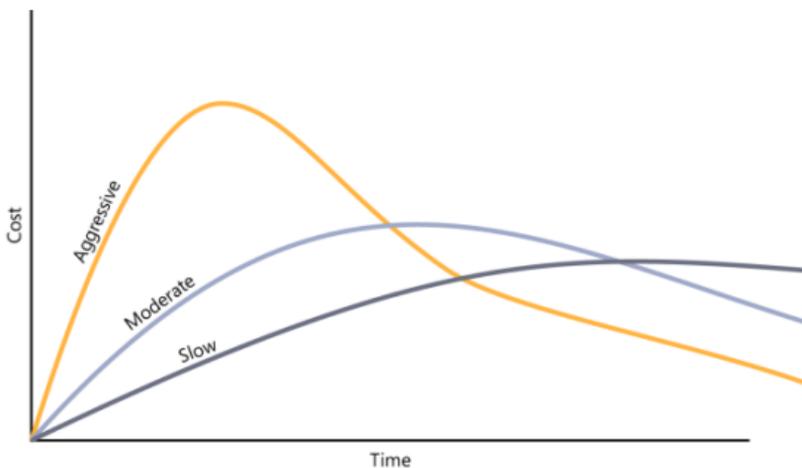
time. You might also even build models showing when and how datacenters can close.

Develop some key measurements to quantify the savings more particularly. For example, one measurement we used in Microsoft is called “cost per operating system instance (Cost/OSI).” (We used this so as to include both applications and operating systems running on bare-metal servers as well as those running in VM’s as a single metric.) Cost/OSI includes hardware, licensing, facilities, network, operations staff, and, in general, all the costs of running an operating system and its applications in an on-premises datacenter. You can segment systems if this is useful: we used “t-shirt sizing” and had a metric for small, medium, large, and extra-large deployments.

With this metric you can now compare the cost of running an on-premises system against one in the cloud. Of course, the parameters for Cost/OSI in the cloud are different and include size of the application, number of cores required, amount of storage, and estimated network traffic. And, unlike the on-premises case, you can spin-down servers in the cloud when they’re not needed or not used, and thus reduce or even eliminate charges.

You should determine your Cost/OSI currently as a baseline. Then, you can forecast costs for various operations in the cloud. Most cloud service providers, including Azure, provide cost estimation tools to help you determine what your Cost/OSI will be under various configurations and requirements.

Work with your finance department to develop several scenarios for your cloud migration, including aggressive, moderate, and slow migration plans, as shown in Figure 4-5. An aggressive plan might involve moving 50 percent of your workloads to the cloud in the first year, whereas a moderate plan might be 30 percent, and a slower plan might be 10 percent. Aggressive plans will potentially save you more, but this must be weighed against greater risk and higher migration costs.



## Figure 4-5: Adoption rates and costs

Of course, finance leaders need to understand that the journey to the cloud is about more than just cost savings. They need to view the enterprise's data as a valuable asset that can be made to have greater value based on what we can do with it. Using new types of data; analyzing the data to discover insights on your products, customers, and processes; frequent experimentation to determine how to maximize the impact from these insights; and scaling these innovations will add significant value to your data. In turn, these actions will provide increase control and reduction of risk to a company's operation; something about which all CFOs care deeply. The more you can quantify the increased value of data as well as cost savings with moving to the cloud, the easier it will be to get more of the highest-level decision makers to support the move.

## Operations

Cloud migration has a very significant impact on daily operations in an IT department. Although *functionally* the requirements of this team remain intact, the *mechanics* of how many of these functions are performed changes in some important ways. Consider some of the following

operations tasks and how they will change in the cloud-centric world:

Task	On-premises function	Cloud function
Health monitoring	Use various tools such as Microsoft System Center to monitor applications and provide Root Cause Analysis (RCA) of failures	Embed with developers to monitor in real time the applications and rapidly understand impact of (perhaps daily or even hourly) updates (such as DevOps)
Data backup	Use on-premises tools such as Microsoft System Center Data Protection Manager (DPM) to	Use DPM for IaaS VMs or Azure Backup Services for PaaS to create online (optionally geo-replicated)

	create disk- or tape-based data backups	backups
Scalability	Add and provision additional hardware instances (servers) in the datacenter; ensure proper operation and network connectivity	<i>Configure</i> scale up/out options to automatically respond to spikes by enabling scale, reliability, and resiliency
Business continuity/disaster recovery testing	Use custom scripts to failover to alternate datacenters	Turn on tools such as Azure Site Recovery to perform script-driven orderly failover and recovery of applications

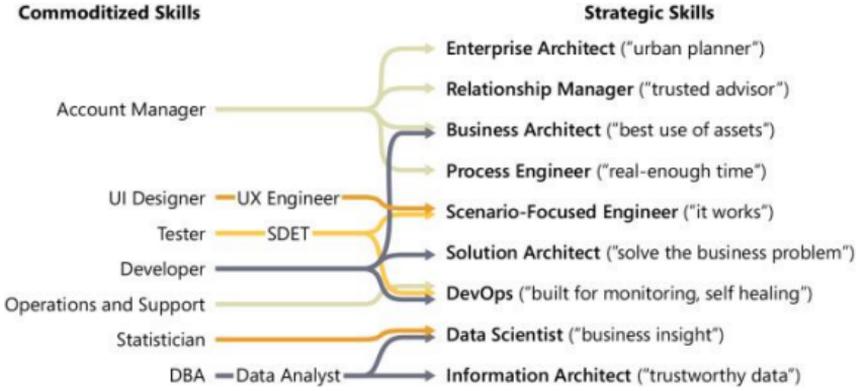
		and storage
Network configuration and optimization	Use various tools to analyze and optimize network performance, discover router loops, and so on.	Ensure hybrid network connections such as V-Nets and MPLS routers ("ExpressRoute") are appropriately tuned and load balanced
Identity provisioning and deprovisioning	Maintain user director (for example, Active Directory), ensure appropriate user access to resources, enable/enforce single sign-on (SSO)	Extend directory to cloud and possibly utilize alternate forms of authentication for specific applications and resources

This list is by no means exhaustive or conclusive; rather, it is illustrative of the types of issues an operations staff will want to address.

The operations staff, in addition, typically maintains a Configuration Management Database (CMDB) for all of its hardware assets. There is much in the CMDB that is relevant for the cloud migration process. As we will discuss later, the CMDB can provide information such as the size of servers required for a given application, the typical number of VM instances, what storage is being used, and so on. This information in combination with the portfolio management system will provide the raw data used to prioritize application migration.

## Human resources and the evolution of roles

Migration to the cloud will force the roles and responsibilities of IT professionals to evolve. Much has been written about how the cloud will eliminate IT jobs. Our experience is that this is not the case; instead, IT roles change (see Figure 4-6), and become less about rote IT functions and more about high-value contributions to the business of the enterprise.



**Figure 4-6:** The evolution of IT roles in the cloud era

Existing IT skills will remain but become of less value than the newer, cloud-centric skills. Enterprise architects, evolving from senior technologists, solution architects, and, in some cases, relationship managers, will maintain the portfolio as a whole, understanding how to extract the most business value from large collections of applications and people. In a sense, they are the "urban planners" of the organization. Business architects, using quantitative models and working closely with their partners in the actual business units, examine technical assets and business processes in various business domains and plan their evolution into the future. Process engineers optimize business processes such that they run in real time wherever appropriate, and "real-enough" time (where appropriate) elsewhere. Six

Sigma as well as other quality methodology skills are useful here.

With the cloud comes greater reach, and with greater reach comes the essential requirement to create applications that are both productive and pleasing for the user. User interface (UI) design has evolved from simply creating menus and dialog boxes to ensuring that the entire experience of performing a task online from end to end is efficient and, in this era of Facebook, YouTube, and Twitter, enjoyable.

Solution architects focus on envisioning and enhancing an application or set of applications focused on a particular domain, such as Finance, and work closely with their counterparts in business architecture (BA) and EA. The solution architects provide oversight and direction to the development of new features and capabilities within the applications in their space. They typically are very technical individuals.

Perhaps one of the most interesting—and talked about—evolutions in cloud migration is the merging of two communities, development and operations, that were previously separate. This is now called the *DevOps* movement. As applications move to the cloud and the ability to deploy applications quickly and repeatedly

(sometimes adding new features each week, or even more often, by using agile methodologies) is recognized, the traditional boundaries between developers, testers, and operations staff begin to blur. Developers will test their applications in staging areas in the cloud. Testers will necessarily be as conversant in cloud technologies as others and often write cloud-based automation scripts or applications in the cloud, making them cloud developers, as well. And, operations personnel will less and less manage hardware assets such as servers and networks, and more and more handle creating automated configuration, deployment scripts, insight portals, monitoring scripts, and orchestration flows, or using those provided by the cloud or tools vendor.

Lastly, the information architect will ensure both the consistency of data models across the enterprise and their lifecycle. A well-designed, documented, and maintained set of models—for example, for “customer” and “product” data entities—ensure ease of system integration and consistency of reporting, among other benefits.

The human resources team should work with the relevant leaders to build readiness and training plans for the affected individuals. Nearly all roles in IT will evolve. Many will require specialized

training; for example, in new tools or new processes.

## Applications teams

You will need to consult applications teams on a range of topics. Typically, it is these teams that will provide the required information for the application catalog or portfolio management system (discussed later) that will help prioritize application migration.

In addition, discuss with them the technical implications of running their applications in the cloud. If an application is “chatty” in the datacenter (meaning it sends and receives a lot of messages to accomplish a task) it’s possible that the latency inherent in moving to an off-premises cloud datacenter will amplify delays. To ameliorate this, application teams might either want to update the application or recommend using a high-speed dedicated line to provide additional bandwidth. If you’re using a cloud database, it might impose certain size restrictions, but this can be addressed by using specific approaches such as database *sharding* (a database shard is a partition of data in a database; each shard is commonly hosted on a separate database server instance).

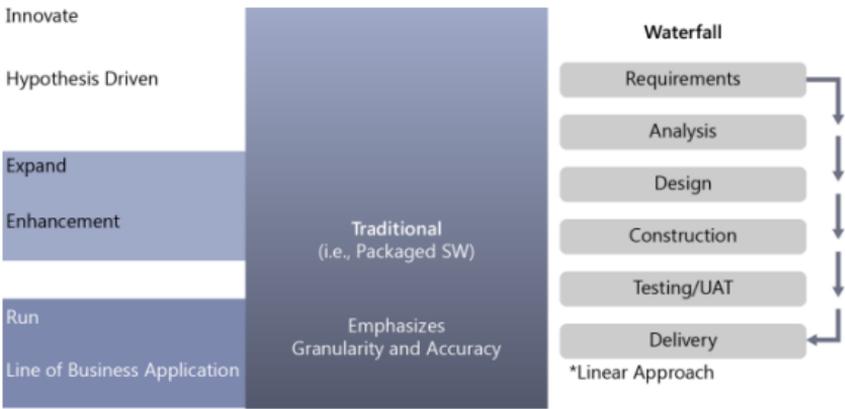
Applications teams should know the longer-term possibilities of a cloud-centric application. For example, redesigning an application to be PaaS or to be a collection of so-called “microservices” (discussed in more detail in Chapter 5) will require awareness and training.

From a methodology perspective, applications teams should consider if using a traditional *waterfall* approach (as shown in Figure 4-7) is appropriate, or if an *agile* methodology, incorporating many short development sprints with feedback and potential course correction can be used. For certain types of applications—for example, financial accounting, for which strict regulatory requirements can essentially dictate the functional specification—a waterfall approach might be used. Waterfall-based projects usually include a detailed, comprehensive requirements document that project managers can validate.

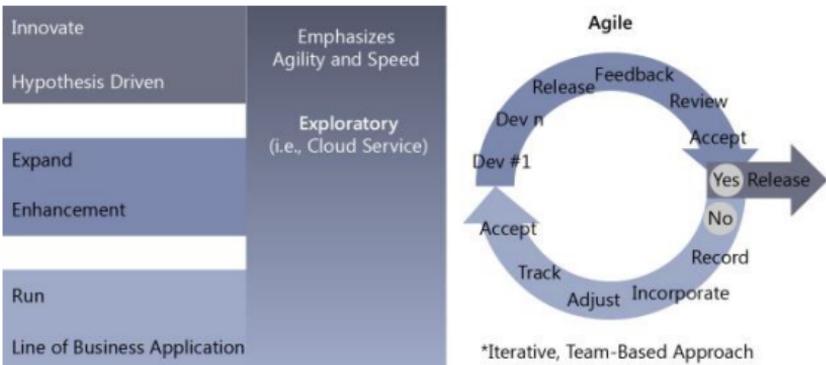
However, fewer applications today require this amount of rigor and most actually benefit by short amounts of development followed by user testing and feedback. In this way, users can get a sense of the application, request new features, suggest others be removed, and so on, and in many cases the agile methodology leads to a

solution that meets users' needs far better than waterfall.

Having this discussion is important because the cloud accommodates much faster development/deployment cycles and thus lends itself very well to agile. Figures 4-7 and 4-8 give you a view of traditional versus cloud software development.



**Figure 4-7:** Traditional software development



**Figure 4-8:** Cloud software development

## Business units

You should also consult business units. Some will embrace and champion the change; others might initially resist because such a change introduces risk, as we discussed earlier. Providing an understanding in nontechnical terms of how you will address those risks will go a long way toward easing their fears. Of course, describing the expected benefits in terms of cost savings, increased reach, and quicker deployment times will, hopefully, whet their appetites. Partnering with your champions early and building real applications that demonstrate the benefits will sway the others.

Take note of their business calendars. Often business leaders have important times of the year when their systems must be available, such as at critical financial reporting periods or, for e-commerce functions, around holidays such as so-called “Black Friday” in the United States. You can then plan around these times.

# Building the catalog

How then do you prioritize the migration of applications to the cloud?

To understand what applications should be moved, when, and how, it's important to create a well-attributed catalog of applications managed by IT. Then, the relative importance of each attribute (say, business criticality or amount of system integration) can be weighted and the prioritized list can be built.

There might be many attributes ranging from document classification types to server counts to protocols, and so forth. It is often useful to roll these up into management sets of overall attributes, such as is shown in Figure 4-9. Here, the top-level criteria include performance, architecture, financial attributes, risk, operations, and security and compliance.

Performance	Architecture	Financial	Risk	Operations	Security and Compliance
Elasticity	User Interface	Operating Cost	Organizational	Business Continuity	Jurisdiction
Scalability	Access Points (Mobile or Offline)	Business Value	Business Criticality	Tools/ Integration	Regulation
Resource Intensiveness	Application		Technical	Deployment	Privacy
Latency	Complexity		Resource		Encryption
Throughput	Size		Contractual		
	Application Life Expectancy		Audit		
	Data				
	Structured Magnitude				
	Unstructured Requirements				
	Complexity				
	Infrastructure				
	Hardware Life Expectancy				

**Figure 4-9:** Evaluation criteria

Many enterprises already have a portfolio management system where such a list is maintained, and they can usually use or extend these systems for cloud purposes. Others might need to use an ad hoc tool such as Microsoft Excel. Either can be effective.

It can be useful to think about application characteristics, or attributes, from two perspectives, the business (“top-down”) and technical (“bottom-up”) models, because the data comes from different constituencies. The top-down approach asks where each application or workload *should* go; the bottom-up approach

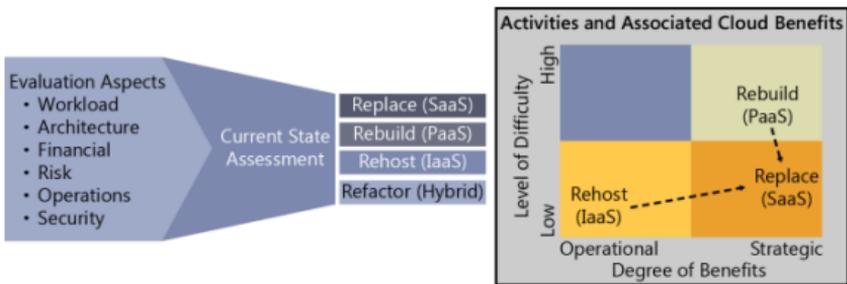
will describe where each *can* go. The following sections describe each and the attributes they capture.

## Top-down portfolio analysis

So far, we have discussed the migration process as a systematic approach, examining objective and subjective metadata to determine where applications or workloads should go. This is a top-down assessment method, which provides a strategic approach, driven by planning and your detailed analysis and modernization needs.

As shown in Figure 4-10, the top-down assessment first evaluates the security aspects previously mentioned, such as the categorization of data (high, medium, or low business impact), compliance and sovereignty, and security risk requirements. Then, it assesses the current complexity interface, authentication, data structure, latency requirements, and coupling and application life expectancy of the architecture. Next, top-down assessment measures the operational requirements of the application, such as service levels, integration, maintenance windows, monitoring, and insight among others. When all of those aspects have been analyzed and taken into consideration, the result is a score that reflects the relative difficulty

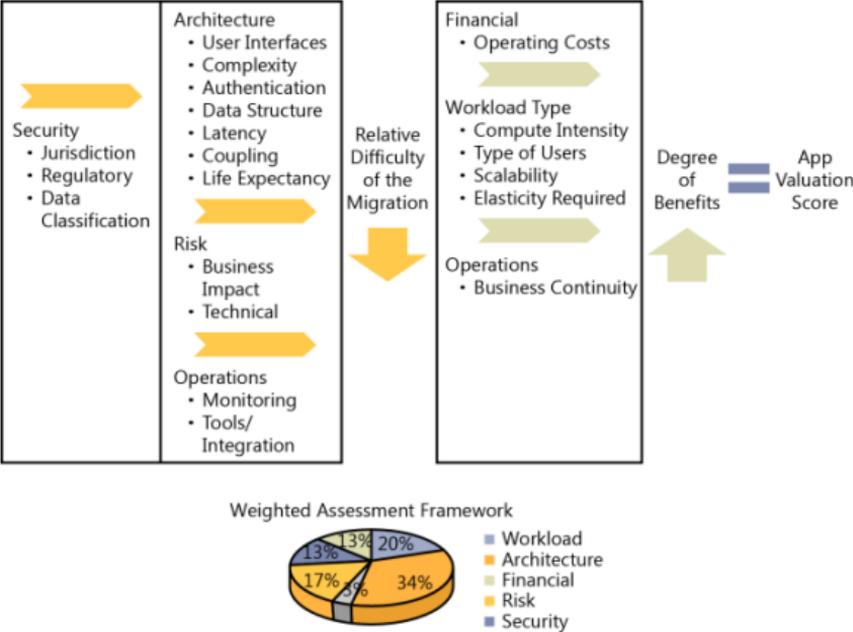
to migrate this application to each of the cloud platforms (IaaS, PaaS, and SaaS).



**Figure 3-10:** Top-down current state application assessment

Second, the top-down assessment evaluates the financial benefits of the application such as operational efficiencies, total cost of ownership (TCO), return on investment, or any other appropriate financial metrics. In addition, the assessment also examines the seasonality of the application (are there times of the year when demand spikes) and overall compute load. Also, it looks at the types of users it supports (casual/expert, always logged on/occasionally logged on, and so on) as well as the consequent required scalability and elasticity required. Finally, the assessment concludes by examining the business continuity and resiliency requirements that the application might have as well as dependencies to run the application if a disruption of service occurs.

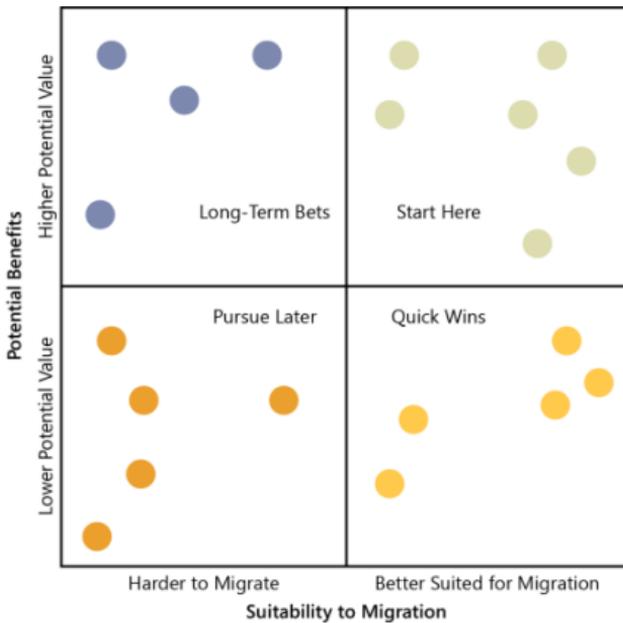
The two parts of the process result in an application valuation score that reflect the balance resulting from the difficulty to migrate to each platform versus the potential benefit gained by it. You can see the entire process in Figure 4-11.



**Figure 4-11:** Top-down assessment process

With the results from the top-down assessment method, you can then map out which applications have the highest potential value and are better suited for migration and start there, and you might even be able to combine that list with the quick wins which are based on lower potential value applications that are also better

suited for migration. After your organization has gained the appropriate experience, built the right set of tools and processes, and gained confidence in its methods, it then will be time to move to the harder-to-migrate applications that have a high potential value, leaving for last the applications that are more difficult to migrate that have a low potential value. This can be easily visualized in Figure 4-12.



**Figure 4-4** Application migration suitability versus potential benefit

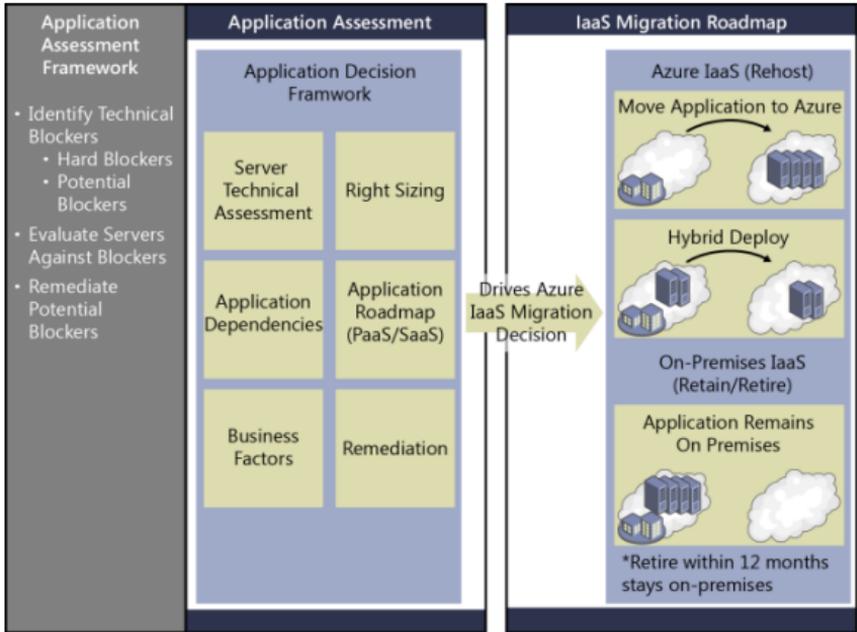
## Bottom-up portfolio analysis

There is a complementary approach that is more tactical and technical, and is focused more on

requirements. As we mentioned, the top-down approach analyzes where an application *should* go; here we are asking where a particular workload *can* go, based on its purely technical requirements.

The bottom-up approach occurs simultaneously to your top-down planning and is aimed at providing a view into the eligibility, at a technical level, of an application to migrate. As mentioned earlier, we typically can pull much of this information from a CMDB. Enterprises use this approach to provide additional insight to the top-down approach.

The type of requirements evaluated by the bottom-up assessment cover the application or service required: maximum memory, maximum number of processors (CPU cores), maximum operating system disk space, maximum data disks, network interface cards (NICs), IPv6, network load balancing, clustering, version of the operating system, version of the database (if required), domains supported, and third-party components or software packages, among others.



**Figure 4-13:** Example of bottom-up assessment for IaaS

Each day, it seems, cloud platforms are becoming more and more capable of handling different application profiles. Massive amounts of memory and storage are available, and 6- and 32-core servers have become commonplace. Still, you might find applications that for one technical reason or another you cannot move at this time,<sup>10</sup> or you should wait until cloud capabilities are further extended.

<sup>10</sup> For example, applications that depend on deprecated features in operating systems, run on

As part of the bottom-up planning, catalog the technical aspects of your applications, including its operating system type, version, number of processors required, memory required, disk space and number of drives needed, and so on. Knowing the size of an application's database, and its data types, will help inform decisions such as to whether to use, for example, a cloud-centric relational database such as Azure SQL Database, Microsoft SQL Server, or Oracle in a VM, or perhaps even a NoSQL database.

Your existing integration systems will be affected by cloud migrations, at least temporarily, so you should spend time documenting the potential impact on these systems. You will need a clear understanding of which applications connect to which; is there an order of precedence for integration operations; how much data is moved and how frequently; what is the architecture of your Extract, Transform, and Load (ETL) tools? The complexity of your integration operations should be an important factor in prioritizing cloud migration goals.

---

operating systems not supported in the cloud, or rely upon poor programming practices such as hard-coded IP addresses.

Moreover, many cloud vendors now implement options for business continuity and disaster recovery, including failover to alternate datacenters, redundant disk storage, and online backup. You need to understand their offerings and their capabilities and how they map to the needs of your applications.

A large number of tools that can help you assess the current state of your applications exist. As mentioned, a portfolio management system will be of great utility in listing and attributing your applications. The Microsoft System Center suite includes a CMDB as well as a host of monitoring and health management capabilities that will help you to ascertain the state of your systems.<sup>11</sup> The Microsoft Assessment and Planning (MAP) Toolkit is a solution accelerator that provides a large feature set for assessing existing IT environments. MAP facilitates the automated inventory and assessment of applications to determine basic suitability and VM sizing requirements.<sup>12</sup> You can also use a number of

---

<sup>11</sup> <http://www.microsoft.com/en-us/server-cloud/products/system-center-2012-r2/>

<sup>12</sup> <http://technet.microsoft.com/en-us/solutionaccelerators/gg581074>

third-party applications and utilities to perform similar functions.

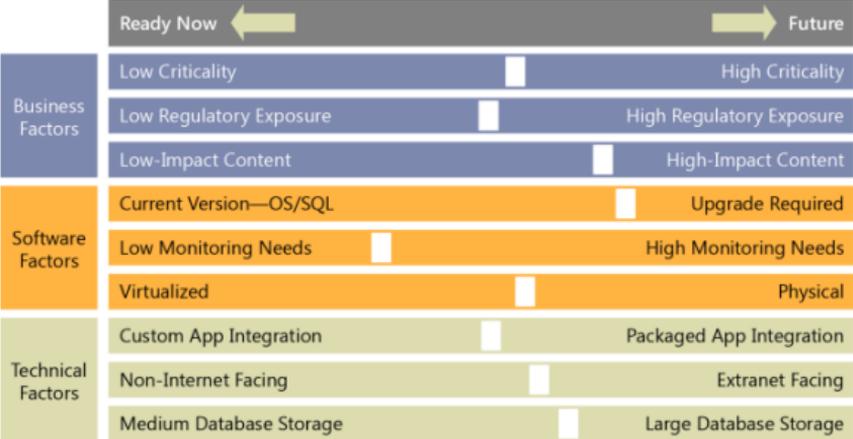
The type of information you collect with these instruments will address objective factors concerning hardware/VM eligibility and application/workload eligibility for cloud migration. These tools are useful for automated metadata collection about your applications and operating system instances. We will look at the manual and more subjective evaluation factors in the following section, which looks at mapping and modernization.

## The cloud migration plan

Your mapping exercise leading from the current state to the desired state is the root of your cloud migration plan. The migration plan takes the map and adds specifics such as priorities and sequencing.

You should set priorities within your plan based upon a combination of business factors, hardware/software factors, and other technical factors. Your business liaison team should work with the operations team and the business units involved to help establish a priority listing that is

widely agreed upon. Figure 4-14 illustrates principles that you might use in establishing prioritization guidelines.



**Figure 4-14:** Migration priority strategy example

For sequencing the migration of your workloads, you should begin with less-complex projects and gradually increase the complexity after the less-complex projects have been migrated. As with running a pilot project, you will gain valuable experience while migrating applications with lower complexity and lower business risk, which can help prepare you for the more complex and more business-critical migrations.

Your cloud migration plan will be more of a process than a static plan document. In its essentials, your plan will actually be a compilation of a number of smaller plans that deal with the migration of each departmental

workload, based upon the sequence you establish. The particulars of each migration will generally follow this pattern:<sup>13</sup>

- 1. Analysis** This is the process illustrated earlier in the discussion of mapping the current state to the desired state. This process will help you to identify the gaps between what you currently have and what it will take to migrate that workload to the cloud. Those gaps might involve changes to the architecture of the workload or might require a complete rewrite of the program. (See the section “Evolution of the five R’s of modernization” in Chapter 2) Additionally, many legacy programs will require significant work to make them more performant and scalable, and you should identify this work during your analysis of the workload.
- 2. Application migration** When you determine that a particular workload should be moved to the cloud, *it is a best practice to create a version of the workload with a minimal amount of data in order to get the application working on the cloud or to build a*

---

<sup>13</sup> <https://msdn.microsoft.com/en-us/library/azure/jj156161.aspx>

*new version of the application there.* If the application is already running on a VM, it might be possible to simply migrate the VM to the cloud without further changes. In general, many on-premises applications can run on Microsoft Azure with minimal or no changes, but this does not mean that the application will be optimized for performance, scalability, and security. So, you might need to redesign and rebuild the application, to some degree, by using modern service-oriented principles.

- 3. Data migration** This is somewhat similar to the application migration in that the data structure can be moved as-is to either a relational (Azure SQL Database, SQL Server in Azure VM) or nonrelational (blob, table, queue, Azure DocumentDB, and so on) location on the cloud. Several of these kinds of migrations are extremely easy, and you can conduct them with the help of a wizard such as the SQL Server Azure Migration Wizard. However, you might want to consider rebuilding the data model as a new Azure SQL Database to gain performance, scalability, resiliency, and security improvements. If you need to synchronize data between on-premises and SQL

Database or between different SQL Database servers, set up and configure the SQL Data Sync service. In addition, *it is a best practice that you set up and configure a data recovery plan in case of user errors or natural disasters.*

- 4. Optimization and testing** After you migrate your application and data to Azure, perform functional and performance tests. At this phase, test your application in the cloud and confirm that it works as expected. Then, compare performance results between on-premises and Azure. After that, resolve any feature, functionality, performance, or scalability issues in your cloud application.
- 5. Operation and management** After the testing and optimization phase, set up and implement application monitoring and tracing with the Azure Application Insights, which enables you to collect and analyze telemetry from your application. You can use this data for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis and capacity planning, and auditing.

You can use the Microsoft Operations Management Suite (OMS) to manage applications running both on premises and

off premises. OMS provides a single view of all your applications, regardless of where they are hosted.

These five phases of migration will be conducted for each workload that you want to migrate. However, there is also an iterative process that is greater than any one migration, by which you can begin moving applications that meet your initial minimum standards, based on priority and sequence. When the initial group is migrated, you can then begin to work on making more applications and hardware eligible by upgrading operating system/SQL versions, getting current with all security patches, moving applications on physical machines to VMs, addressing issues caused by multiple IP addresses, and so on.

## Microsoft IT's experience

When Microsoft IT began its cloud migration journey in 2009, it followed a similar process. First, it cataloged its operating system instances and application workloads. This assessment included both quantitative data that was mostly retrievable by tools as well as qualitative data that was partially retrievable by tools and also required examination by both the operations

team and the business liaison team. This latter category of metadata included relationships, dependencies, and integration points.

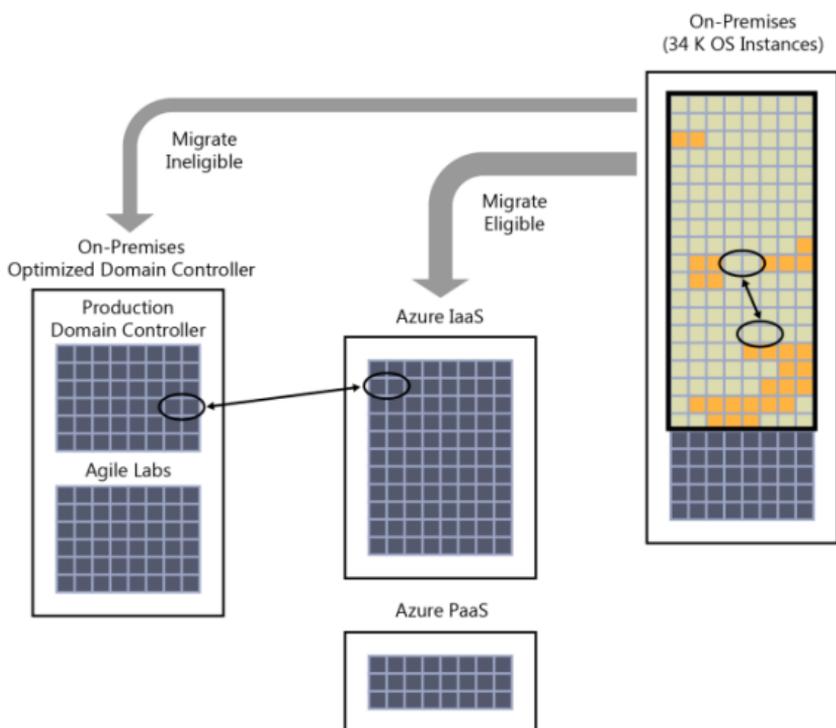
Microsoft IT then identified the initial prioritized eligible operating system instances and initial prioritized eligible workload/applications. Next, these initial migration candidates were reduced by removing any business-critical systems, which would be moved after more experience had been gained. Then, this initial list of candidates was prioritized and sequenced with less-complex applications placed before more-complex applications, and applications running on updated VMs prioritized over those running on physical machines or legacy VMs. Some applications were identified as ineligible for various reasons (most of which limitations no longer exist in 2015) and these were migrated to an optimized on-premises datacenter.

After the initial set of migrations was completed, Microsoft IT completed work to make less-eligible operating system instances (OSIs) and workloads more eligible. For example, OSIs with older operating systems or database versions were updated, more applications on physical machines were moved to VMs, and more mission-critical applications were deemed eligible. Applications and workloads that were

identified as requiring a major overhaul were rebuilt as services on Azure.

Here's the process, which you can see illustrated in Figure 4-15:

1. Identify eligible hardware (OSIs) per Azure compute, storage, and RAM limits.
2. Identify eligible applications, remove HBI apps, sequence critical and complex apps for later, and right-size to include more apps.
3. Increase eligible hardware and applications by doing the following:
  - Virtualizing more servers
  - Expanding to more regions
  - Including extranet-facing apps
  - Including HBI apps
  - Getting current (OS, SQL)
  - Increasing Azure VM limits
4. Build new applications as services for SaaS IT.



**Figure 4-5:** Implementing the plan

## Cloud governance

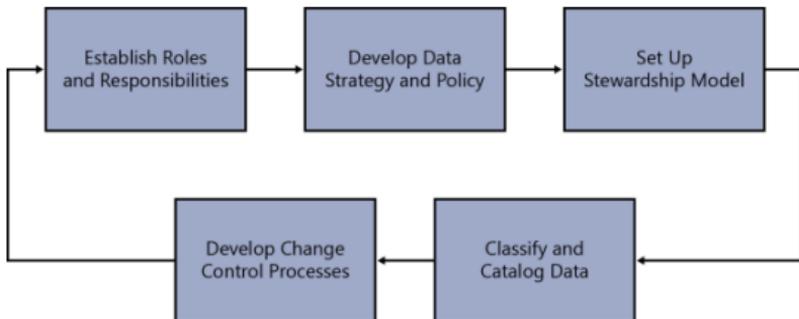
Governance, broadly speaking, can be defined as providing the oversight to ensure that any change to the environment neither causes any degradation of function nor adds any new risks. Like any major activity in IT, managing cloud assets requires an effective governance function.

In general, cloud governance is not normally a “net-new” function but rather extends existing governance activities. Governance professionals

should therefore fully understand the implications that the cloud has for their areas and extend existing practices.

## Data governance

Data governance in IT has long been a critical function, and its essential nature existed long before the cloud. Creating and ensuring adherence to common data models, providing extensibility where needed, managing changes, ensuring regular and controlled taxonomy updates, use of master and reference data, data classification, formal processes around data retention and destruction: all of these activities have been part of the IT governance function for decades. Figure 4-16 depicts the process involved in data governance.



**Figure 4-16:** Data governance process

The cloud, however, adds some new dimensions to data governance. First, many countries have,

or are developing, laws governing *where* information about their citizens can reside. This is called *data sovereignty*, and the concerns are, in a nutshell, that if your data leaves your country, it will be easier for foreign government agencies to obtain it. To be clear, as of this writing the legal elements of data sovereignty are still evolving, but nevertheless it is important to design an appropriate governance strategy.

To address these concerns, know what data your applications are keeping in the cloud, and know as well what the laws of your country are with regard to data sovereignty. Some potential actions you might want to take include the following:

- Not placing any individual or customer data in the cloud
- Encrypting key PII such as email addresses or physical addresses prior to moving data to the cloud
- Disabling geo-replication to other geographies

Much has been written on this topic and (as we have mentioned) the laws continue to evolve, so it is a best practice to stay on top of the emerging legislation and case studies.

Now that applications are in the cloud, additionally, it is far easier to access Internet-resident data sources. Many governments now place very large quantities of data in the cloud (in the United States, for example, at <http://www.data.gov>; in the United Kingdom, <http://www.data.gov.uk>; <http://data.gouv.fr> in France; and so on). Other companies make data available over the web for a fee, and such data, for example, can augment or even replace on-premises master data sources or can provide additional marketing insights. Use of such data, including analyses of their semantics to ensure alignment with enterprise data sources, should be governed by the data governance organization.

## Financial governance

We have already described the fairly significant changes to IT finance implied by the cloud: the change from a capital expense model to an operational expense, or subscription model. Financial governance ensures that the financial changes are managed in a methodical and predictable fashion, including:

- Actual cloud costs are in line with predicted cloud costs

- Capital expenses are declining in line with expectations
- Cloud billing is consolidated and no “rogue” credit card accounts are allowed
- Appropriate chargeback mechanisms are created or extended to support cloud computing
- Quarterly or annual budgeting shows the appropriate changes
- Reporting systems accurately reflect current spend on IT

## Security and compliance

It is important that you have clarity with respect to your responsibilities versus those of the cloud providers, because running a VM would entail very different responsibilities of what the cloud provider does for you versus a full-fledged SaaS offering such as a CRM or an ERP system.

Figure 4-17 describes security and compliance responsibilities by application type (on-premises, IaaS, PaaS, and SaaS). Existing security and compliance controls should be modified and/or extended in order to support the new application types.

On-Premises Security Dependencies	IaaS Infrastructure as a Serv.	PaaS Platform as a Service	SaaS Software as a Service
1. Security Strategy, Governance, and Operationalization: Provide clear vision, standards, and guidance for organization			
2. Administrative Control: Defend against loss of control of your cloud services and on-premises systems			
3. Data: Identify and protect most important information assets			
4. User Identity and Device Security: Strengthen protection for accounts and devices			
5. Application Security: Ensure application code is resilient to attacks			
6. Network: Ensure connectivity, isolation, and visibility into anomalous attacks			
7. OS and Middleware: Protect integrity of hosts			
8. Private or On-Premises Environments: Secure the foundation			

**Figure 4-17:** Cloud security governance responsibilities

## Change management

Most IT organizations have a program management office (PMO) of one form or another. The PMO's function is to ensure that changes entail minimal risk and disruption to the IT function. In moving to the cloud, the PMO will need to manage new change management functions. These include the following:

- Operational readiness, to ensure the operations or DevOps function is ready to manage a cloud-resident application
- User readiness, in the case of functional changes to applications
- Organizational readiness, to ensure (for example) that dependent applications

continue to function and that all security, compliance, and financial requirements are completed

- Application and ecosystem readiness, to ensure applications moving to the cloud and applications that are remaining but integrated with the cloud applications are fully tested and ready, and that all issues are known in advance.

There are other aspects of governance (for example, supplier management). However, it should now be clear that governance in the cloud, by and large, extends existing functions, and professionals in each of these areas should consider the impact of cloud applications to their space.

Migrating applications to the cloud is an important and significant activity, requiring changes to how both businesses and IT operate. In this chapter, we have described how to form and use a Cloud Strategy Team to drive the migration; how to involve the many organizational stakeholders; how to prioritize application migration; and how to extend existing governance activities.

Of equal importance are the transformational aspects of the cloud, which should be examined and performed concurrently with migration. In Chapter 5, we outline what we mean by transformative innovation and the opportunities afforded by the cloud.

## Information Technology Infrastructure Library and the cloud

Many IT organizations rely on the Information Technology Infrastructure Library (ITIL) framework for service management and operations. Over the years the ITIL has proven a useful set of practices for IT Service Management (ITSM) and for aligning IT investments and operations with business goals. Among its benefits, advocates and practitioners of ITIL point to increased reliability, uptime, and predictable costs.

ITIL fundamentally concerns itself with IT services; that is, the functions and processes that the IT organization provides to the business. A service is something—an application, a set of applications, information, people—that a business user consumes in order to perform a business function. In general, the cloud as a

technology does not change the goals of ITIL; however, the cloud can dramatically change how services are delivered, as we have shown.

ITIL consists of five key strategic areas:

- **IT Service Strategy** ITIL's Service Strategy provides a set of frameworks for determining what services are delivered, how their value is measured, how to measure cost and provide a measure of return on investment (ROI), and how to manage the IT relationship with its business partners. Earlier in this chapter, we described how to set up a strategy effort that defines the overall goals—technical, financial, and organizational—of the cloud migration effort.
- **IT Service Design** In IT Service Design, design of processes and how they relate to one another, service level agreements (SLA), capacity and availability management, business continuity management, security, and supplier management are covered. We discussed these topics as well earlier in this chapter; patterns for backup and business continuity are provided in Appendix B.

IT Service Design also notes the need for a service catalog, of which the portfolio management and configuration management systems described earlier are key parts.

- **IT Service Transition** Service Transition governs how services are delivered and deployed. Such areas as change management, release and deployment management, and service evaluation are typically part of the transition phase. The goal, of course, is that new services and changes to existing services are deployed with minimum impact to the overall IT ecosystem.

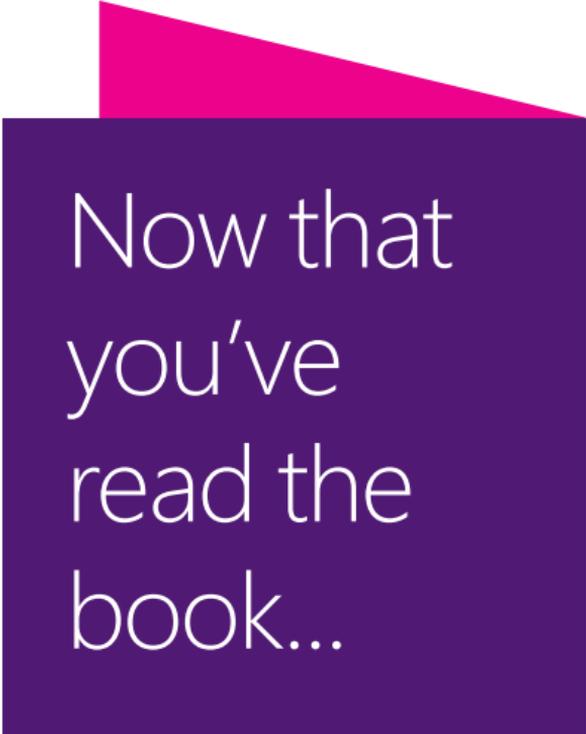
Whereas the structure of service transition remains the same, the actual tasks when deploying a service to the cloud change significantly as we have described. In particular, the emergence of DevOps and its associated methodologies means that the processes and tools associated with deployment are new and different. In addition, IT departments might want to think about such areas as SLA measurement

differently, considering that there might be additional latency to the cloud, for example. Similarly, IT departments should set up a test cloud environment mirroring the production environment in order to allow user acceptance testing (UAT), load and penetration testing, and integration testing with other applications prior to full production deployment.

- **IT Service Operation** Service Management covers the management and monitoring of services, and how issues are managed and resolved. Key to the Service Management component is the notion of a Service Desk, the primary point of contact for service incidents and events. The service desk as well as the call center and help desk, if separate, will need to be trained to support cloud-based services.
- **IT Continual Service Improvement** In Continual Service Improvement (CSI), IT personnel and business teams work together to ensure services can quickly meet new and emerging business requirements. CSI is heavily data driven and relies upon

operational statistics as well as business insights to determine where focus should be placed.

In general, cloud migration will force organizations to change some of the mechanisms and processes by which they implement ITIL, although the basic structure of ITIL is generally technology-independent. However, organizations should also consider how to extend their own processes to be more agile than ITIL might suggest; given that experimentation and prototyping (as we have discussed earlier) are quick, think about how to do them as part of the strategy and design phases of ITIL.



Now that  
you've  
read the  
book...

Tell us what you think!

Was it useful?

Did it teach you what you wanted to learn?

Was there room for improvement?

**Let us know at <http://aka.ms/tellpress>**

Your feedback goes directly to the staff at Microsoft Press, and we read every one of your responses. Thanks in advance!

# Transformation

During your migration to the cloud, some—perhaps the majority—of your applications will be moved more or less intact as virtual machines (VMs). This Infrastructure as a Service (IaaS) style of migration has a number of advantages, as discussed in previous chapters. After they are in the cloud, applications can take advantage of the numerous services available to them, quickly giving those applications far more function and return on investment. This chapter first looks at how you

can redesign your applications to better take advantage of the underlying cloud framework, and then how you connect them to services in the cloud to rapidly expand their features and functions.

## Platform as a Service architecture

So far, we've have talked primarily about migrating applications in a fairly simplistic way; that is, by simply copying VMs from an on-premises datacenter to the cloud provider—the IaaS model. Of course, IaaS carries with it a number of advantages, such as passing responsibility for the datacenter to the cloud provider. To really *transform* to a cloud-centric model, designing applications specifically for the cloud is the next step.

IaaS has certain limitations: you are still responsible for maintaining the system software, operating system, and database for your application, including items such as periodic patches and software upgrades. In fact, we can say that IaaS is only the *first step* in fully taking advantage of the cloud.

In *Platform as a Service* (PaaS) models, you only need to maintain *your application*, whereas the system software is provided by and maintained by the cloud provider. In addition, PaaS offerings typically add seamless scalability and resiliency by providing scale-out and data replication, and PaaS can interact with cloud services such as Microsoft Azure Active Directory for robust identity management.

Azure App Service Web Apps, for example, provides a way to rapidly provision a scalable website in the cloud with a minimum of effort. Microsoft provides the underlying web infrastructure (operating system, networking stack, storage, language support, and scalability features) that remove much of the systems overhead of managing a large-scale web application. It is straightforward to configure scalability, backup, and monitoring capabilities into a Web Apps application. Web Apps also

connects to all the other services offered by the cloud for rich applications (more on this later).

Azure Cloud Services are a cloud analog to the “three-tier” line of business applications of a decade ago. In Cloud Services, an application consists of three components: a *web role*, effectively a web front end, scalable independently from other parts of the application; a *worker role*, providing background computation and processing (analogous to the business logic layer in the three-tier model); and *persistent storage* using an Azure-enabled version of SQL Server (Azure SQL Database). Although it requires some redesign to take an existing application to Cloud Services, this will be relatively straightforward because the model is intentionally similar to three-tier.

## Containers and microservices

A new technology that has emerged in the past year is called *containers*. This refers to the ability to create an application that runs with a strictly defined subset of operating system resources and is fully isolated from other applications and from the operating system. Pioneered by

Docker,<sup>14</sup> containers are highly portable across environments ranging from on-premises bare-metal systems through cloud environments. Docker containers now run in most cloud environments, including Azure.

Another development in cloud computing is the emergence of the *actor* model. Actors, which are small, highly concurrent objects, are actually a relatively old idea first introduced in the 1970s. A niche technology until recently, actors have found great application in gaming and in Internet of Things (IoT) scenarios, in which a very large number of small objects—representing (for example) users or sensors that require the ability to communicate with one another—make up the application. For example, a small piece of code might control the operation of a valve in a pipeline autonomously: and there can be thousands of such things, each of which operates and reports status to some central database.

Finally, so-called *microservices*, another recent development, are a software methodology that proposes that cloud applications are made up of many independently maintained components. So, an e-commerce application might comprise a

---

<sup>14</sup> <https://www.docker.com/>

catalog microservice, a payment microservice, a payments microservice, and so on, all connecting through APIs. A fabric management system (such as the forthcoming Microsoft Service Fabric) controls the concurrency and maintains the state of the microservices, providing a new form of application where logic and state are in the same layer, providing a more distributed and scalable application than the typical three-tier web-application design. This means, for example, that middle tiers, traditionally stateless because of scale limitations, can now be *stateful*, scaling out across the cloud as needed.

## Storage

Only a short few years ago storage options were relatively limited; but recently, both the number and the types of available storage have exploded. Whereas relational databases continue to support the highest data integrity and high performance, the table-based metaphor of relational database management systems (RDBMS's) has been augmented by other forms of storage including binary large objects (BLOBs), simple key-value tables, document databases, graph-based databases, so-called "Big Data" storage, and others. Application developers have a wide variety of data stores to choose from;

increasingly, many cloud-based applications use a combination of several types of storage, a pattern which has been termed *polyglot persistence*.

For example, a typical e-commerce application will usually have a transactional relational database to track purchases and sales because an RDBMS of this sort has superb integrity; its transactions are *ACID* (atomic, consistent, isolated, and durable). However, these capabilities are not required for every aspect of the application: a simple file will suffice for maintaining logs, a Hadoop-based system can be used for log analysis, a document database can be used for maintaining unstructured data such as a product catalog, and so on. These “NoSQL” data stores can support transactions but typically are not as robust (for this purpose) as a relational database. On the other hand, they are ideal for quickly deriving insights from very large amounts (petabytes or more) of unstructured data, and the cloud offers vast quantities of storage space that are typically very expensive for enterprises to provide on-premises.

## Relational databases in the cloud

There are many options for relational database functionality in the cloud, and they serve different purposes.<sup>15</sup>

**SQL Server in Azure Virtual Machines** gives you the ability to migrate existing databases to an IaaS infrastructure. When using SQL Server in a VM, you can either bring your own SQL Server license to Azure or use one of the preconfigured SQL Server images in the Azure preview portal. SQL Server on a VM is optimized for extending existing on-premises SQL Server applications to Azure in a hybrid scenario, deploying an existing application to Azure in a migration scenario, or creating a development/test scenario. An example of the hybrid scenario is keeping secondary database replicas in Azure via Azure Virtual Network. With SQL Server in Virtual Machines, you have the full administrative rights over a dedicated SQL Server instance and a cloud-based VM. It is a perfect choice when an organization already has IT resources available to maintain the VMs. With SQL Server in Virtual Machines, you can build a highly customized

---

<sup>15</sup> Read “Understanding Azure SQL Server Database and SQL Server in Azure VMs” at

<http://aka.ms/azsqldb>

system to address your application's specific performance and availability requirements.

SQL Server running in Virtual Machines is perfect when your existing and new applications require access and control to all features of a SQL Server instance, and when you want to migrate existing on-premises applications and databases to the cloud as-is. Because you do not need to change the presentation, application, and data layers, you save time and budget on redesigning your existing solution. Instead, you can focus on migrating all your solution packages to the VMs and doing some performance optimizations required by the Azure platform.

You can run many other commonly used relational databases such as Oracle or IBM's DB2 as VMs in the cloud.

**SQL Database** is a relational database-as-a-service that makes it possible for you to create new applications using database services in a PaaS environment. With SQL Database, you can develop directly on the service by using built-in features and functionality. When using SQL Database, you pay as you go, with options to scale up or out for greater power. SQL Database is optimized to reduce overall costs to the minimum for provisioning and managing many

databases because you do not need to manage any VMs, operating system, or database software including upgrades, high availability, and backups.

SQL Database is the right solution for cloud-designed applications when developer productivity and fast time-to-market are critical. With programmatic DBA-like functionality, it is perfect for cloud architects and developers because it lowers the need for managing the underlying operating system and database. It helps developers understand and configure database-related tasks. For example, you can use the REST API and PowerShell cmdlets to automate and manage administrative operations for thousands of databases. With elastic scale in the cloud, you can easily focus on the application layer and deliver your application to the market faster.

## NoSQL (nonrelational) storage

The NoSQL arena has many options, ranging from simple object storage to complex document and graph-based data stores.

**Azure Storage** has several component features that provide the flexibility to store and retrieve large amounts of unstructured data such as

documents and media files with Azure Blobs, structured NoSQL-based data with Azure Tables, reliable messages with Azure Queues, and Server Message Block (SMB) protocol file shares with Azure File Service. The following is a brief look at how to differentiate each of these component features:

- **Azure Blob Storage** is designed to store data in essentially any format. Blobs (which stands for “Binary Large Object” and is somewhat of a misnomer, given that text is equally at home in a blob) are analogous to files on a server or client machine. Blobs can hold text, images, media, comma-separated-value (CSV) files, databases—virtually anything.

Like Azure Tables (described next), blobs are triple redundant, meaning their contents are always replicated to two other physical stores in Azure, thus minimizing the possibility of data loss in the event of hardware failure.

- **Azure Tables** provides a simple but performant key/value store. Azure Tables lets an application store properties of various types, such as strings, integers, and dates. An application can then retrieve a group of

properties by providing a unique key for that group. Although complex operations such as joins aren't supported, tables offer fast access to typed data. They're also very scalable, with a single table able to hold as much as a terabyte of data. And, matching their simplicity, tables are usually less expensive to use than relational storage.

You should consider Azure Tables if you need to create an Azure application that needs fast access to typed data, maybe lots of it, but does not need to perform complex SQL queries on this data. For example, imagine that you're creating a consumer application that needs to store customer profile information for each user, and you expect to have a large number of users, but you won't do much with this data beyond storing it and retrieving it in simple ways. This is the kind of scenario for which Azure Tables makes sense.<sup>16</sup>

- **Azure Queues** is a service for storing large numbers of messages that users can access from anywhere in the world via

---

<sup>16</sup> Ibid.

authenticated calls by using HTTP or HTTPS. A single queue message can be up to 64 KB in size, and a queue can contain millions of messages, up to the total capacity limit of a storage account. Common uses for Queues storage include creating a backlog of work to process asynchronously and passing messages from an Azure Web role to an Azure Worker role.

- **Azure Files** provides file storage accessible through the SMB protocol using a `\\Server\share` format. Applications running in Azure can use it to share files between VMs using familiar file system APIs like `ReadFile` and `WriteFile`. In addition, the files can also be accessed at the same time via a REST interface, which allows you to access the shares from on-premises when you also set up a virtual network. Azure Files is built on top of the Blob service, so it inherits the same availability, durability, scalability, and geo-redundancy that's built in to Storage.<sup>17</sup>

---

<sup>17</sup> <https://azure.microsoft.com/en-us/documentation/articles/storage-dotnet-how-to-use-queues/>

There are several common scenarios for which Azure Files would be a good storage solution:<sup>18</sup>

- *Migrating existing apps to the cloud* It's easier to migrate on-premises applications to the cloud that use file shares to share data between parts of the application. Each VM connects to the file share, and then it can read and write files just like it would against an on-premises file share.
- *Shared application settings* A common pattern for distributed applications is to have configuration files in a centralized location where they can be accessed from many different VMs. You can store these configuration files in an Azure File share where they can be read by all application instances. You can also manage the settings via the REST interface, which allows worldwide access to the configuration files.

---

<sup>18</sup> <https://azure.microsoft.com/en-us/documentation/articles/fundamentals-introduction-to-azure/#data>

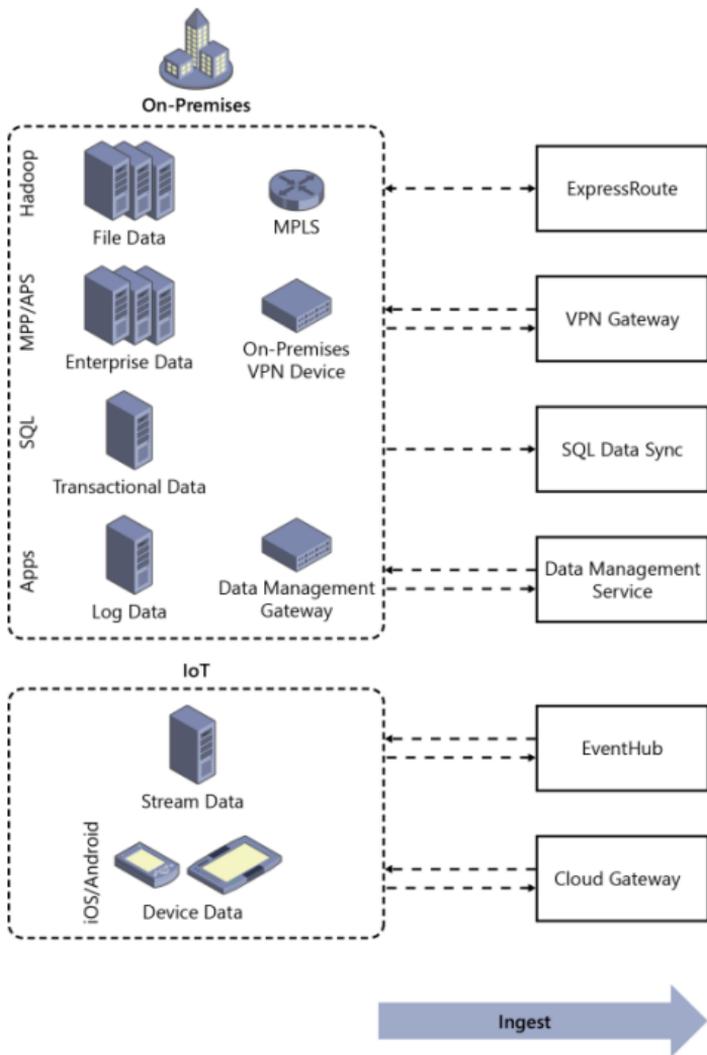
- *Diagnostic Share* You can save and share diagnostic files such as logs, metrics, and crash dumps. Having these files available through both the SMB and REST interface makes it possible for applications to use a variety of analysis tools for processing and analyzing the diagnostic data.
- *Dev/Test/Debug* When developers or administrators are working on VMs in the cloud, they often need a set of tools or utilities. Installing and distributing these utilities on each VM is time consuming. With Azure Files, developers or administrators can store their favorite tools on a file share and connect to them from any VM.

**Azure DocumentDB** is a NoSQL document database service designed from the ground up to natively support JavaScript Object Notation (JSON) directly inside the database engine. It's the right solution for applications that run in the cloud when predictable throughput, low latency, and flexible query are crucial.

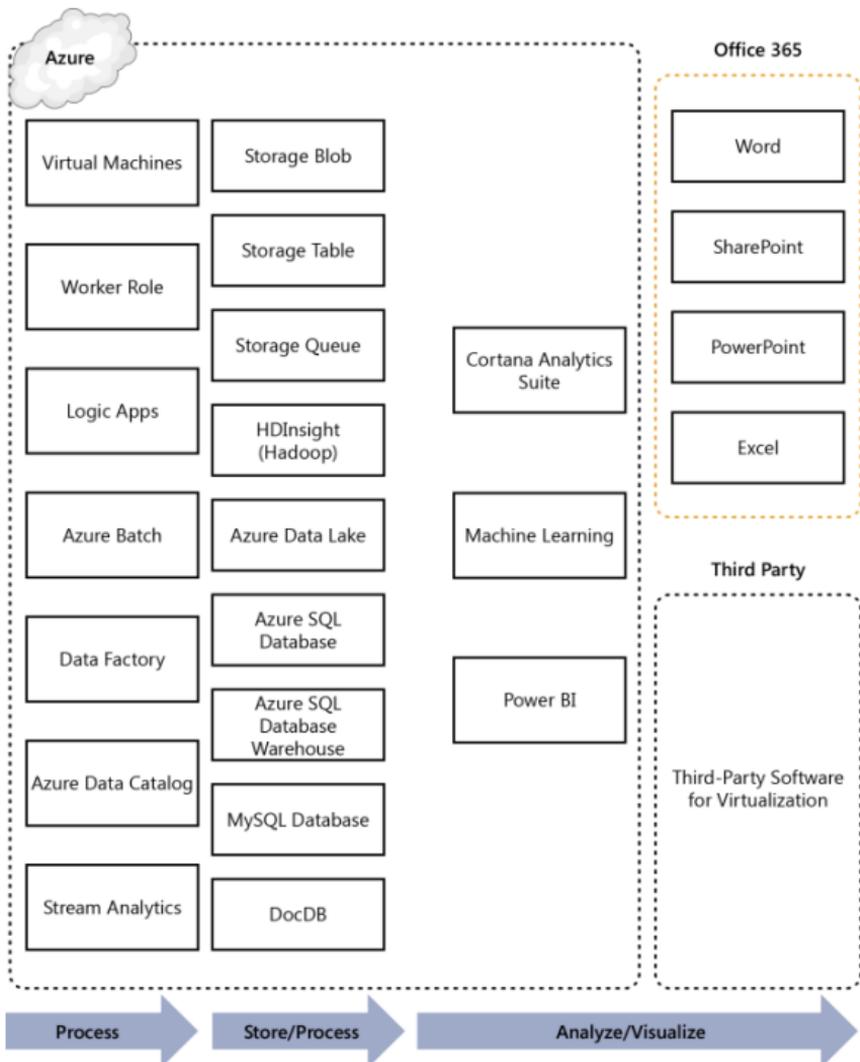
A common problem for developers is that application schemas constantly evolve.

DocumentDB automatically indexes all JSON documents; adds them to the database, including all fields; and lets you use familiar SQL syntax to query them without specifying schema or secondary indices up front.

**Other data options** include a wide range of relational and nonrelational applications from other vendors, including IBM and Oracle, as well as popular open-source packages such as MongoDB and others. Figures 5-1 and 5-2 show just some of the possibilities available as of this writing; new ones are constantly being added. The Azure Marketplace provides a convenient way for architects and applications developers to find and try out different options.



**Figure 6:** Data platform landscape in the cloud (part 1 of 2)



**Figure 7:** Data platform landscape in the cloud (part 2 of 2)

# Analysis

Flexibility and a variety of options also characterize analytics in the cloud. There are many new possibilities with analytical analysis, but the most exciting possibility is predictive analytics; that is, the ability to predict future performance from historical data. The possibilities are almost limitless, but consider the use of Big Data for predictive analysis in the following:

- Churn analysis
- Social-network analysis
- Recommendation engines
- Location-based tracking and services
- IT infrastructure and web application optimization
- Weather forecasting for business planning
- Legal discovery and document archiving
- Equipment monitoring
- Advertising analysis
- Pricing analysis

- Fraud detection
- Personalized everything

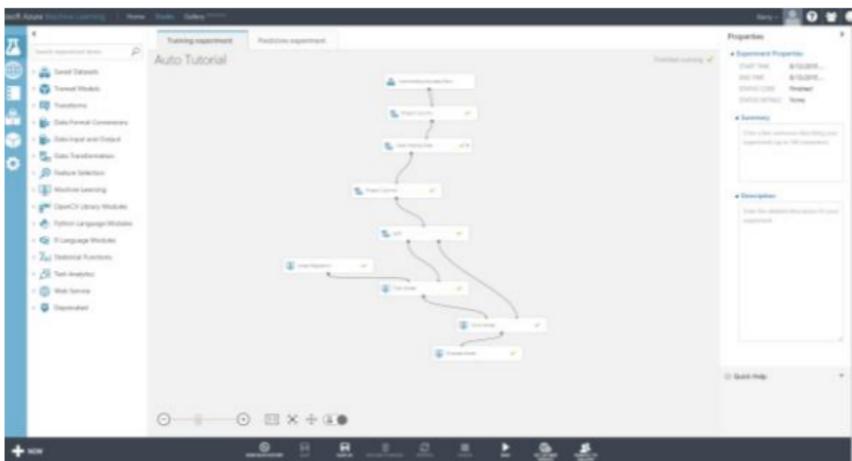
The tools available in the cloud to help you with analytics include HDInsights, Machine Learning, Stream Analytics, Search, and business intelligence (BI) and reporting. With these tools you can explore how to turn today's noise into tomorrow's insights.

**Azure HDInsight** is a Hadoop-based service that brings an Apache Hadoop solution to the cloud. You can use it for managing data of any type and any size. HDInsight can process unstructured or semi-structured data from web clickstreams, social media, server logs, devices and sensors, and more. This makes it possible for you to analyze new sets of data that uncovers new business possibilities to drive your organization forward.

**Azure Machine Learning** offers a streamlined experience for all data-science skill levels, from setting up with only a web browser to using drag-and-drop gestures and simple data flow graphs to set up experiments. Machine Learning Studio features a library of time-saving sample experiments, R and Python packages, and best-in-class algorithms from Microsoft businesses

like Xbox and Bing. Azure Machine Learning also supports R and Python custom code, which you can drop directly into your workspace and share across your organization.

Figure 5-3 shows a simple example of the drag-and-drop user interface for Azure Machine Learning. In this case the experiment is to ingest data about cars, including engine size, mileage, size of the car, and so on, and then to train a model to predict a car's price based on those parameters. Data scientists can pick from any one of many prebuilt algorithms to train the model, or they can supply a custom R module. At the end of the training, a Web Service can then be created (see the bottom panel of the screen) that can subsequently be used in an application in which an end user supplies data and Azure Machine Learning provides a predicted price.



## Figure 5-3: Azure Machine Learning

**Azure Stream Analytics** gives you the ability to rapidly develop and deploy a low-cost, real-time analytics solution to uncover real-time insights from devices, sensors, infrastructure, and applications. It opens the door to various opportunities, including IoT scenarios such as real-time remote management and monitoring or gaining insights from devices like mobile phones or connected cars.

Stream Analytics provides out-of-the-box integration with Azure Event Hubs to ingest millions of events per second. Stream Analytics will process ingested events in real time, comparing multiple real-time streams or comparing real-time streams together with historical values and models. You can use this to detect anomalies, transformation of incoming data, to trigger an alert when a specific error or condition appears in the stream, and to power real-time dashboards.

**Azure Search** is a fully managed cloud service with which developers can build rich search applications by using .NET SDK or REST APIs. It includes full-text search scoped over your content, plus advanced search behaviors similar to those found in commercial web search

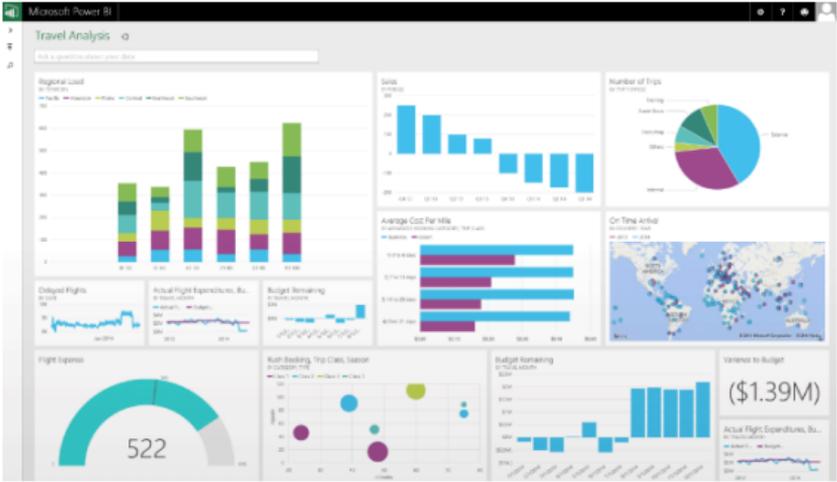
engines, such as type-ahead query suggestions based on a partial term input, hit-highlighting, and faceted navigation. Natural language support is built in, using the linguistic rules that are appropriate to the specified language.

Search is an API-based service for developers and system integrators who know how to work with web services and HTTP. Search takes the complexity out of managing a cloud search service and simplifies the creation of search-based web and mobile applications.

**Azure BI and Reporting;** The Microsoft Azure Virtual Machine gallery includes images that contain SQL Server installations that you can use to easily set up SQL Server Reporting Services on the cloud. You can create an Azure Virtual Machine that runs Microsoft SQL Server Business Intelligence (BI) features and Microsoft SharePoint 2013.

**Microsoft PowerBI** (at <http://www.powerbi.com>) is a SaaS application with which you can quickly build visually appealing, interactive dashboards. An ever-increasing number of connectors gives you the ability to bring data from cloud data sources and other Microsoft and third-party SaaS applications into PowerBI.

Figure 5-4 shows an example of a PowerBI dashboard.



**Figure 8:** PowerBI sample dashboard

Whereas data is dramatically increasing in volume, velocity, and diversity, actionable analytics is challenging. You need interoperable tools and systems to maximize your existing investments in analytics, and provide the flexibility to evolve on your own terms.

## Integration

Of course, no application in enterprise IT exists as an island; every application communicates in one form or another with others. Applications can receive or send real-time updates to others through direct messaging, queues, or publish-and-subscribe techniques; can receive events

from external sources such as sensors; and/or can receive bulk, batch updates (“extract, transform, and load” or ETL) from others.

**Azure Service Bus** is a messaging system in the cloud for connecting applications, services, and devices to one another through a variety of protocols, including topic-based, message-based, and publish-and-subscribe. Service Bus supports a variety of protocols (REST, AMQP, WS-\*), and you can use it to connect cloud applications to one another and to on-premises applications, as well.

**Azure Event Hubs** provides a massively scalable event ingestion service. Also supporting a variety of protocols, Event Hubs can scale out to support thousands, millions, or even billions of events per day and is designed for small or very large IoT applications.

**Azure Logic Apps** gives developers a means to quickly create applications in a stepwise fashion by connecting applications such as SQL Database or Twitter visually. With Logic Apps, you can rapidly and graphically develop workflow apps with connectors and triggers.

**EDI** (Electronic Data Interchange) is one of the oldest data integration standards, and its use in

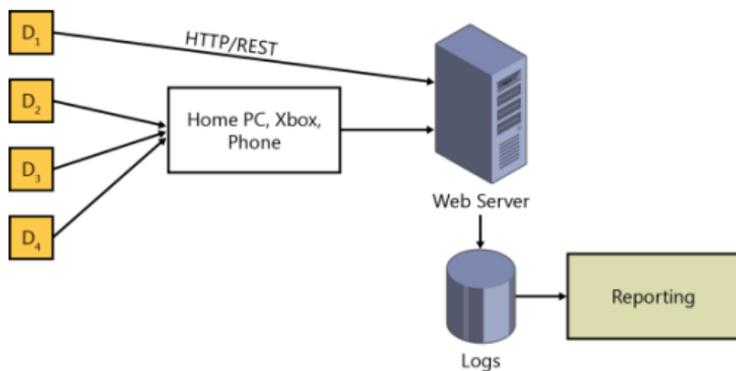
electronic commerce is widespread. **Azure BizTalk Services** provides a cloud-based means for connecting EDI applications together with support for EDI, X.12, EDIFACT, and AS2.

Of course, this list of services does not represent the full panoply of capabilities available. We encourage you to frequently review the Microsoft Azure website ([www.microsoftazure.com](http://www.microsoftazure.com)) for updates, new features, and new services.

## Using services to create rich end-to-end applications

How do you put all of this (perhaps bewildering) array of capabilities together to form an application?

Recently, we had a conversation with a startup that is creating an IoT application; in this case, capturing output from a home medical device over the Internet. The architecture of the application was quite simple, as is illustrated in Figure 5-5.



**Figure 5-5:** Simple IoT architecture

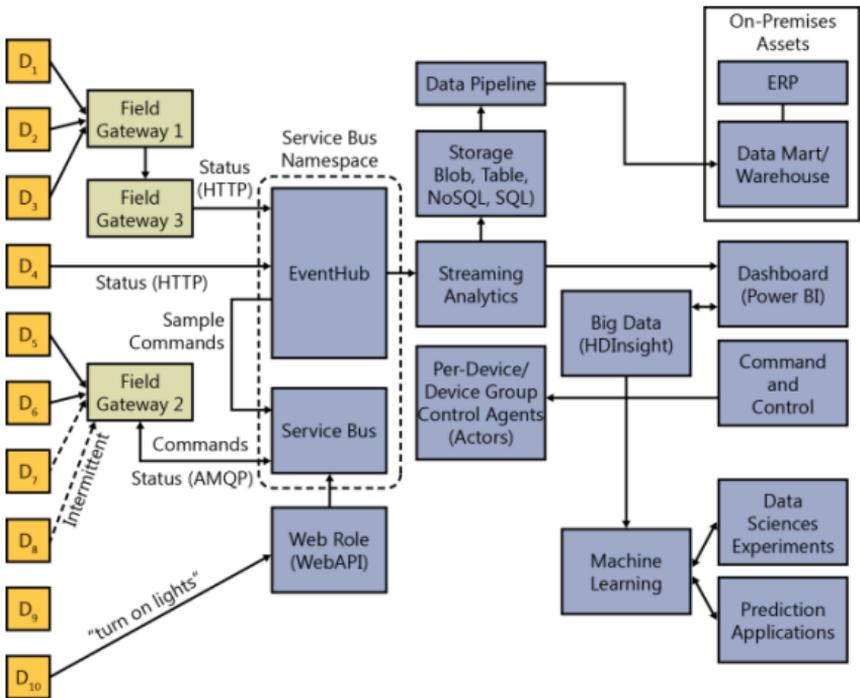
As Figure 5-5 shows, the devices send data over the REST protocol either directly to a web server or through a router to the server. The server, in turn, did some processing and stored the data in a database where it was subsequently displayed in a homegrown reporting application.

We asked some simple questions:

- What happens when the company becomes wildly successful and must support tens of thousands of devices online, at any given time?
- How would the company support resiliency?
- How could it do preventative or predictive maintenance?
- How could it discover its most unreliable suppliers for its devices?

- How could the company add new reports quickly?

We suggested that the company integrate with capabilities in the cloud. In the proposed architecture, shown in Figure 5-6, the startup simply needs to connect its devices to various services.



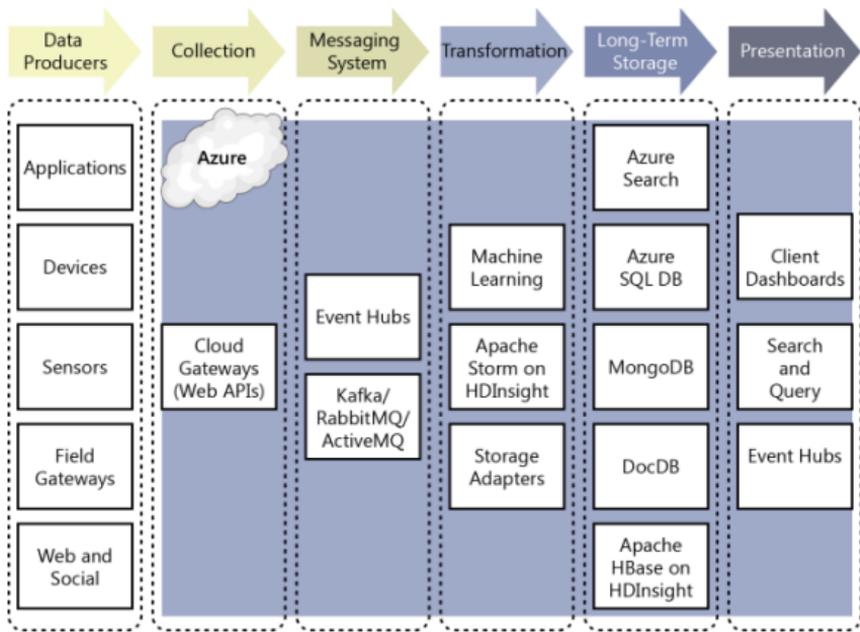
**Figure 5-6:** A rich end-to-end IoT application

Using Event Hubs, which supports massive throughput in event ingestion, the startup's application can easily scale to as many devices as it needs. The data received is stored in Azure

Tables which automatically keeps two replicas, ensuring that no data is ever lost. The data can be analyzed by large scale MapReduce programs in HDInsight; and actors can provide real-time programmatic command and control. Machine Learning applications can be written to predict upticks in sales or parts failures, and all of the data can be visualized in an intuitive and visually pleasing dashboard—all with a *minimum of coding*.

In short, what was once a fairly limited application very rapidly became one that was innovative, insight-rich, and transformational.

In Figure 5-7, we provide a more generic example of a Big Data flow for an IoT scenario to demonstrate the components that come to life in this type of solution.



**Figure 9:** Big Data flow for an IoT scenario

The simple example presented in Figure 5-7 shows the wealth of capability available in the cloud—but it certainly does not cover all of Azure’s features. For more examples, see the “patterns” section in the Appendix, or sample solutions on the Microsoft Architecture site (<http://aka.ms/msarch>).

In many ways, applications created by connecting services together as we have shown represent one of the most innovative, perhaps even revolutionary aspects of the cloud. Remarkable, rich applications taking advantage of years of development by experienced subject-

matter experts can be created by wiring together services in a matter of hours or minutes—not by coding them from scratch over years. This may well turn into a new paradigm for application development.

## Conclusions

The impact of cloud computing on the enterprise, and on business generally, cannot be overestimated. Indeed, nearly every function in enterprise IT is affected, from the way development and testing is done to improving the levels of resiliency and backup the IT ecosystem has, to entirely new ways of building and managing applications. Indeed, it can be said that cloud computing not only affects everyone in the IT department, but everyone in the enterprise as a whole.

We hope that we have shown that utilizing the cloud—for everything from simple application hosting to using the componentry available in the cloud for innovative new applications—is of huge benefit to the enterprise. We hope that we have shown you the way to get there, beginning with early stage experimentation, moving to large-scale migration of applications to the

cloud, and then finally transforming your enterprise by taking advantage of all the features now available in the cloud. Without question, there is effort involved, but as we have said, the rewards are significant, ranging from real cost savings to the ability to reach entirely new customer segments, to gain new insights into customer behavior, and to model and predict your business's performance in the future.

We believe, as we have stated, that cloud adoption is now an inevitability for all organizations. We hope that by reading this document you have learned how you can quickly realize all the benefits of the cloud

Enjoy your cloud journey!

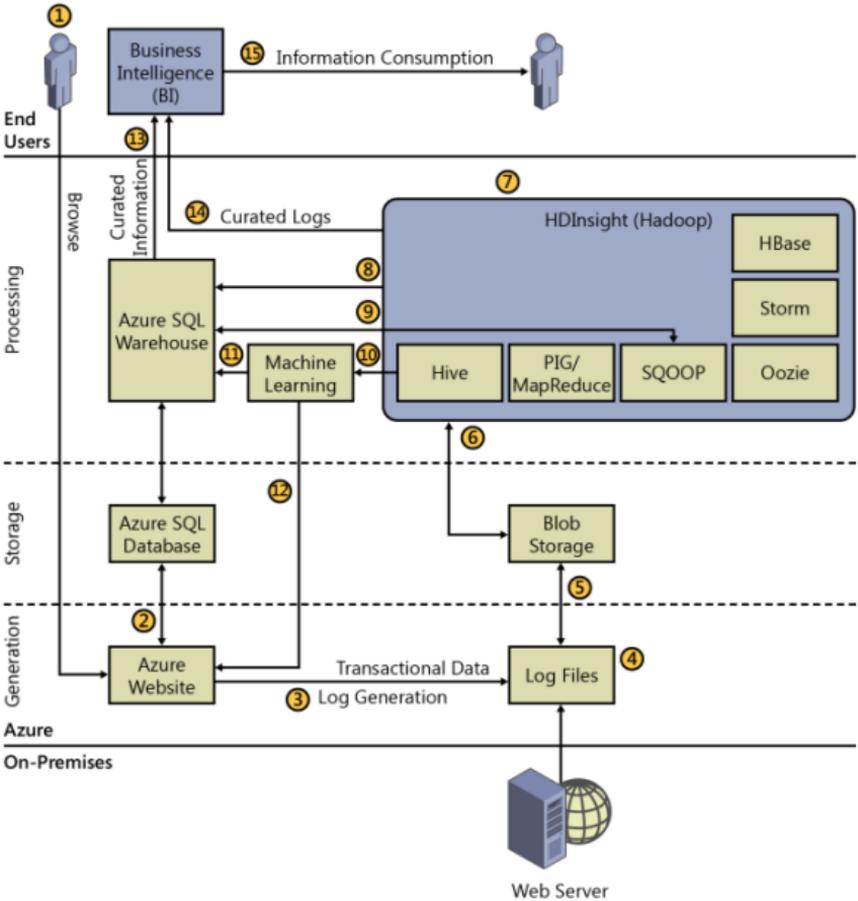
# Cloud architectural blueprints

We encourage you to visit the [www.microsoft.com/architecture](http://www.microsoft.com/architecture) website to view the latest cloud architecture blueprints. We have included some in this appendix to illustrate the potential of the cloud architectures that we have discussed throughout this book.

## Data analytics

Log analysis is an important part of any website or web server to gain more insight into usage behavior. The size, velocity, and variety of log files are akin to “Big Data” and are not suitable for processing directly into some traditional systems. Microsoft Azure with services such as

Azure HDInsight, Azure Blob storage, and Azure Machine Learning make this task a lot easier.



As shown in the illustration:

1. A user accesses an Azure website, leaving behind a trail of data to mine and information to extract.
2. The Azure website has underlying structured data hosted on Azure SQL Database.

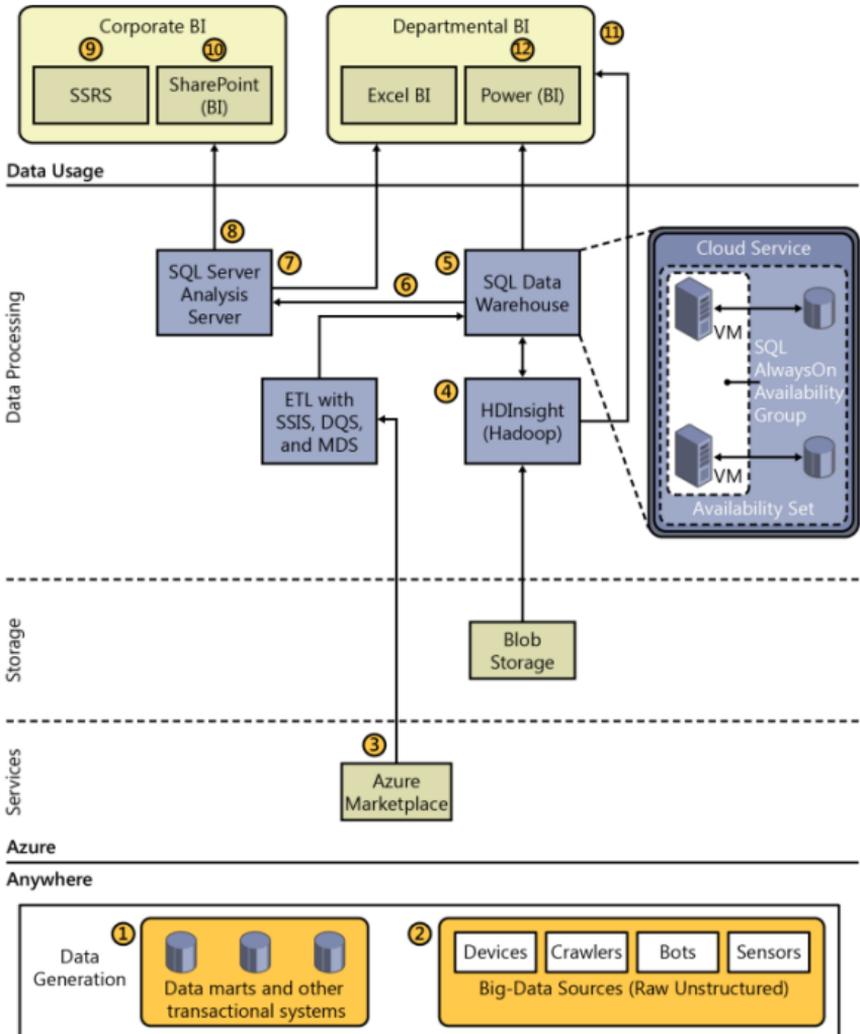
3. The Azure website generates clickstream data and usage logs.
4. The users' actions in cloud or on-premises servers generate logs (for example, server, security, or audit logs).
5. This unstructured/semi-structured data is stored in Blob storage for access from various systems.
6. HDInsight processes unstructured/semi-structured data by using Hadoop tools such as Hive, Pig, and Mahout. It can also orchestrate the data movement by using Sqoop and Oozie. For real-time capabilities, you can use Storm and HBase.
7. HDInsight supports Windows PowerShell to automate tasks like creating and deleting clusters, running MapReduce programs, running Hive commands, and more.
8. SQOOP interacts with structured data stores such as SQL Database to import or export data in and out of the HDInsight (Hadoop) clusters.
9. Data processed in HDInsight is fed into existing data warehouses via common

Extract, Transform, and Load (ETL) channels to enrich the information contained in them.

10. Machine Learning accesses processed data to Hive tables to discover and predict future trends and/or assist with security and performance risks.
11. Machine Learning feeds prediction information into the existing data warehouse.
12. Machine Learning also feeds prediction information to the Azure website directory for immediate customer-centric action.
13. The data warehouse sources data into business intelligence (BI) systems such as Power BI, SQL Server Analysis Services, and/or custom apps to create data models.
14. HDInsight sends processed and curated log information to the BI system.
15. The BI systems publish information models into dashboards and reports by using tools such as Microsoft PowerView, PowerMap, SQL Server Reporting Services, and SharePoint BI for end user consumption.

# BI and analytics

Organizations collect a huge amount of data during day-to-day operations. Azure can deliver an end-to-end solution to convert this data into actionable information.



As shown in the illustration:

1. Data from various transactional systems, data marts, third-party APIs, and so on, is streamlined by using EIM tools such as Microsoft SQL Server Integration Services, Master Data Services, and Data Quality Services.
2. Various devices and sensors are fed into Big Data systems such as HDInsight.
3. Data from external services, such as those hosted on Azure Marketplace, can also be accessed.
4. HDInsight helps curate, process, and analyze unstructured or semi-structured data with MapReduce, Hive, Pig, Sqoop, Hbase, and so on, and format it for consumption or further analysis.
5. SQL Data Warehouse running on Azure virtual machines (VMs) host all the data from a variety of sources. When set up with AlwaysOn, these provide high availability and disaster recovery options for any failure situation.
6. The data warehouses send data to SQL Server Analysis Services for processing.

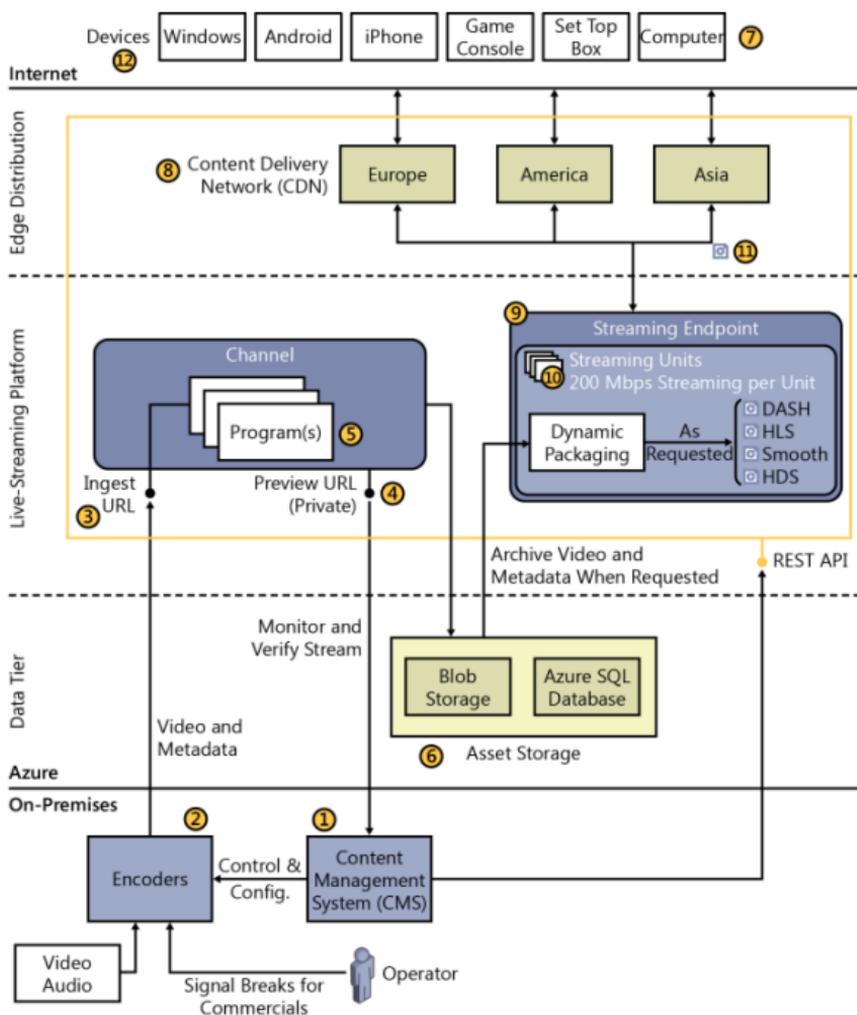
7. SQL Server Analysis Services makes it possible for you to create Online Analytical Processing (OLAP) models to churn the data coming in from the Data Warehouse and can also accommodate merging with external data.
8. The processed model feeds information into reporting tools such as Microsoft SQL Server Reporting Services or SharePoint Dashboards, and consumed by self-service BI tools like Microsoft Excel and Power BI.
9. SQL Server Reporting Services creates reports for business consumption, provides integration extensions for enterprise apps such as Microsoft Dynamics CRM, and can be embedded into custom apps.
10. SharePoint BI provides capabilities to design interactive dashboards and reports for users to consume. It also includes self-service capabilities such as Power View, PowerPivot, and Excel Services.
11. Departmental BI, which comprises Power BI and Excel BI capabilities, consumes data from all the data sources available such as SQL Server Analysis Services models, Data Warehouse, HDInsight clusters and/or Azure

Marketplace. This makes it possible for users to utilize the analysis done and build upon it, if required, in their familiar tool, Excel.

12. Power users can share and collaborate by using their models and reports in Power BI sites, with capabilities such as Q&A and Data Catalog.

## Live media streaming

Live events such as sports or performances are viewed all over the world on different devices. The bandwidth of each device varies depending on proximity to a tower and other factors. Adaptive rate streaming means bit rates vary for every device, from low-resolution to hi-resolution. A streaming endpoint in the cloud reaches all.



As shown in the illustration:

1. A Content Management System (CMS), such as Ooyala, controls functions of the live broadcast. The system manages all of the elements of the live workflow such as encoder configuration, asset management, advertisement signals, and subclips for

highlight reels. It also can manage telemetry and system health.

2. Live stream content is encoded at multiple bit rates into Smooth or Real-Time Messaging Protocol (RTMP) by using a hardware encoder. Operators mark breaks in the action for later advertisement insertion.
3. Live stream content is streamed to the channel via HTTP. Optionally, use ExpressRoute for fast private transport.
4. Preview URL monitors the health of the stream before and during broadcast.
5. Programs act on a stream to record it. Different programs can create different recordings using parts of the same stream.
6. Streams and associated metadata are saved to Assets, which use Blob storage and SQL Database.
7. Devices attempt to consume streaming content via a device-specific URL. Devices contain apps built with specific frameworks that can consume the stream.
8. If the proper streaming file already exists on the Content Delivery Network (CDN), it is

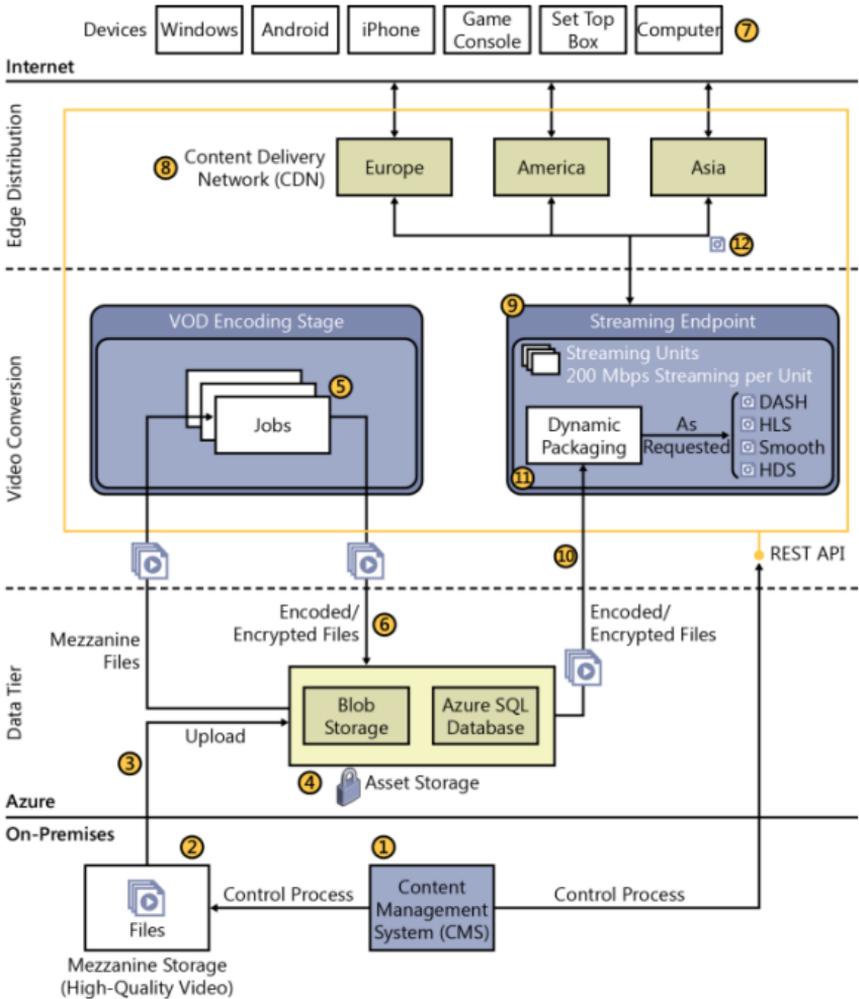
sent to the device. If it does not exist, the request is sent to the Streaming Endpoint. The device can be sent a lower-quality stream if bandwidth is low.

9. Content is dynamically encoded to the specific streaming format required by the device making the request.
10. Content is processed by using scalable Streaming Units. Each unit can deliver up to 200 Mbps. Additional units start up or shut down, based on load.
11. The Streaming Endpoint delivers the request stream to the CDN, which propagates it to all geographic regions.
12. Devices consume streaming content. Devices run apps built from player framework SDKs.

## Video on demand (VOD)

When a customer requests a video, it is delivered by using the video-on-demand process. Video is encoded into multiple bit rates and formats to accommodate an array of mobile and desktop devices. Use Azure Media Services to streamline upload, encoding, and delivery of video to

devices around the world, under changing bandwidth conditions, on demand.



The illustration shows America:

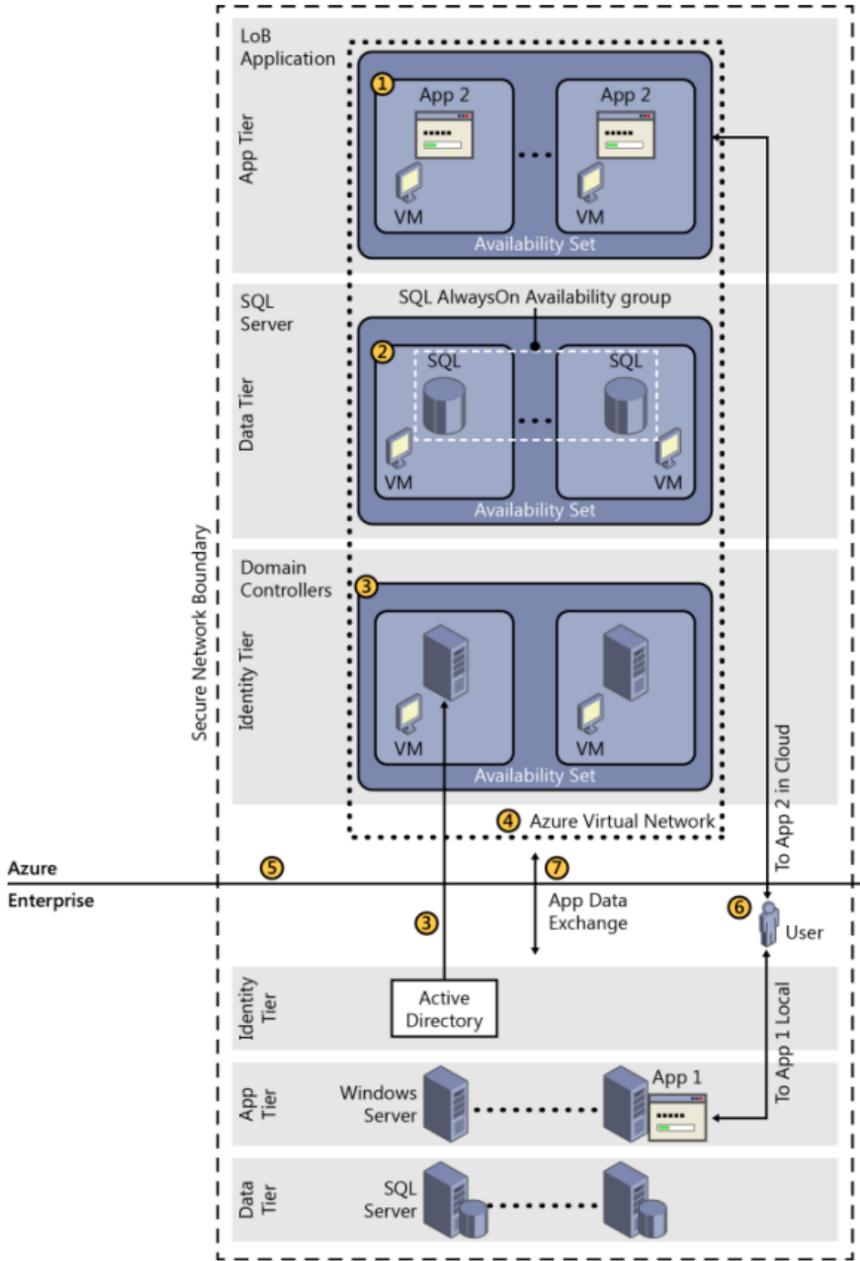
1. A Content Management System (CMS) such as Ooyala manages the upload, encoding, and delivery of video assets.

2. Video content is produced in a high-resolution format that results in large files—several megabytes for a short length. They are commonly known as “mezzanine files.”
3. The mezzanine is loaded into Blob storage by using HTTP or via User Datagram Protocol (UDP) fast-copy partners such as Aspera or Signiant. Along with its metadata, it is now known as an “asset.” A single asset can hold multiple files.
4. Assets can be encrypted at upload. They also can be protected by using a Shared Access Signature (SAS), which provides only authenticated users access to the storage account
5. Jobs encode the mezzanine assets into compressed formats; for example, an MP4 file. You can create multiple parallel jobs by using Encoding Reserved Units
6. The result is stored as an asset. A new URL that points to the file is created.
7. Devices attempt to consume VOD content via a device specific URL. Devices contain apps built with specific frameworks which can consume the stream.

8. If the proper VOD file already exists on the CDN, it's sent to the device. If not, the request is sent to the Streaming Endpoint. The device can be sent a lower-quality stream if bandwidth is low.
9. The Streaming Endpoint starts the process to dynamically package content to the specific streaming format requested by the device.
10. The Streaming Endpoint pulls the requested content (media file and metadata) from Asset storage.
11. Content is processed by using scalable Streaming Units. Each unit can deliver up to 200 Mbps. Additional units start up or shut down based on load.
12. The Streaming Endpoint delivers the requested stream to the CDN, which propagates it to all geographic regions.

# Line-of-business applications in infrastructure services

Gain operational efficiencies without redesigning your applications by moving them to the cloud by using Infrastructure as a Service (IaaS).



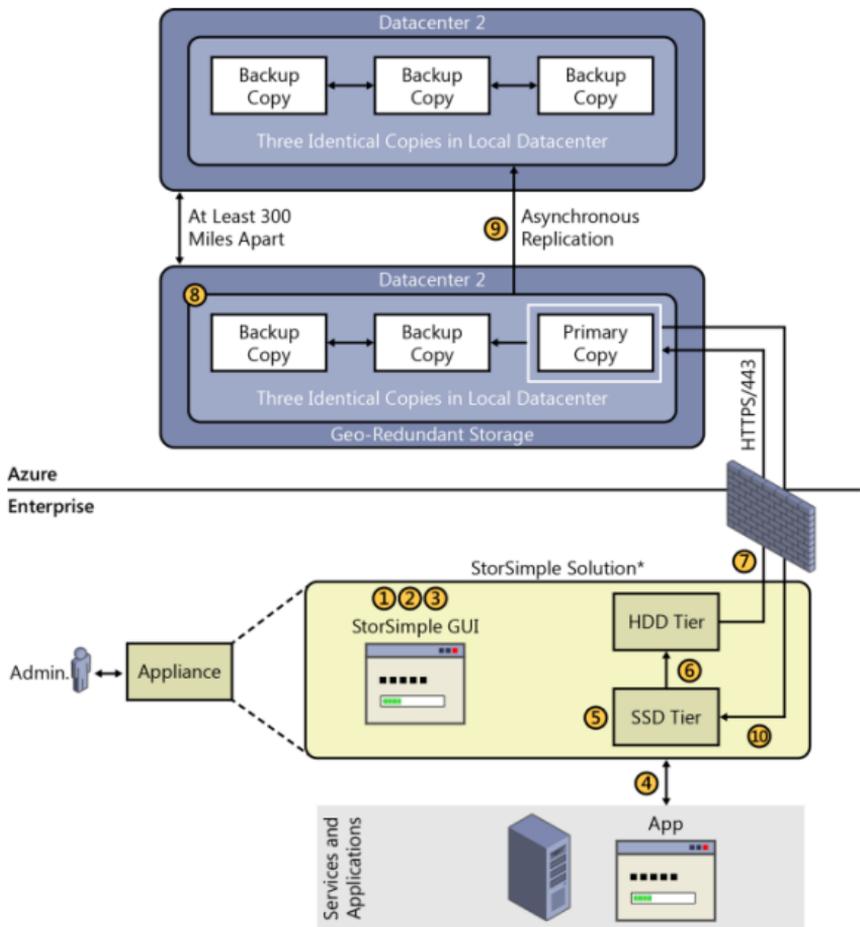
As shown in the illustration:

1. Package your application into a VM and deploy it to Azure. Run at least two copies to provide redundancy in case of failure or add more to scale out.
2. Move your data layer to the cloud for the lowest latency. Take advantage of the SQL Server 2014 AlwaysOn feature to provide redundancy and failover.
3. Run two VMs as Active Directory domain controllers and DNS servers in Azure and synchronize these services with your on-premises Active Directory domain controllers. The application can then authenticate users without the added latency of connecting to the on-premises Active Directory.
4. Connect all your VMs in the cloud to an Azure Virtual Network.
5. Connect on-premises to the cloud via a Virtual Private Network (VPN) over the Internet. For a lower-latency dedicated line, use ExpressRoute.

6. On-premises users now access their applications in the cloud with no changes to the user experience.
7. The applications in the cloud and on-premises can securely communicate and exchange data.

## Hybrid cloud storage

Affordable cloud solution that seamlessly extends on-premises capabilities for primary storage, backup, archive, and disaster recovery with no end user or application changes.



Azure StorSimple is a storage solution that uses both an on-premises appliance and Azure.

As shown in the illustration:

1. The administrator sets up an Azure storage account in the Azure management portal. The storage account credentials are configured by using the StorSimple GUI,

where you specify to the StorSimple device where to put data in Azure.

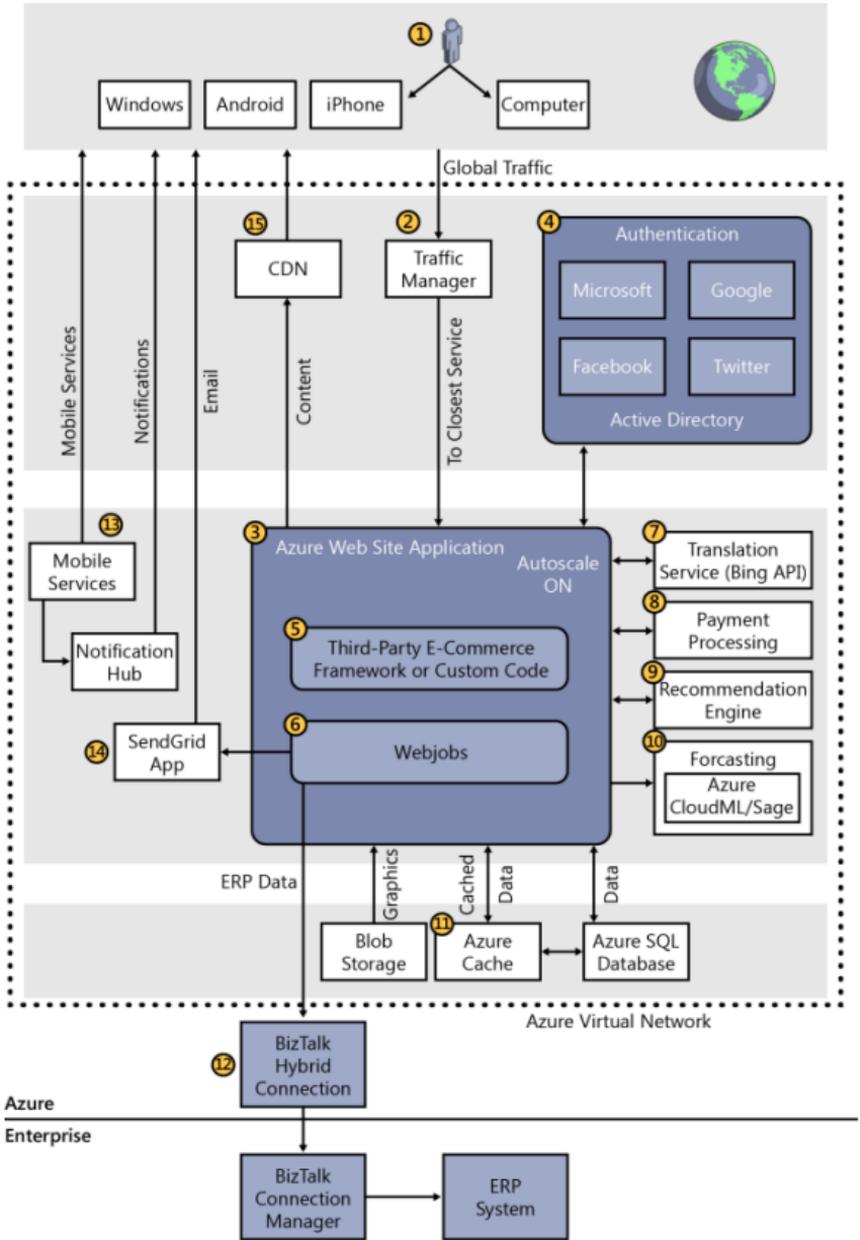
2. The administrator creates volumes on StorSimple.
3. The administrator protects data by configuring data-protection policies.
4. The on-premises file server accesses the appliance by using iSCSI.
5. StorSimple initially stores data in the fast Solid-State Drive (SSD) tier of the appliance.
6. As the SSD tier approaches capacity, the oldest data blocks are deduped, compressed, and automatically migrated to the Hard Disk Drive (HDD) tier.
7. As the HDD tier approaches capacity, the oldest blocks are encrypted and securely sent to Blob storage by using HTTPS.
8. Azure synchronously replicates the data to two other blob replicas within the same data center to insure redundancy.
9. Azure can also replicate the blobs to a secondary Azure data center at least 300 miles away using geo-redundant storage.

Three additional copies of the data are asynchronously replicated.

10. When the file server requests data stored in Azure, the data is returned without any impact to the user or his application. A copy of the requested data is then stored locally in the SSD tier.

## E-commerce website

Build a highly scalable e-commerce website with catalog, checkout, analysis, and forecasting.



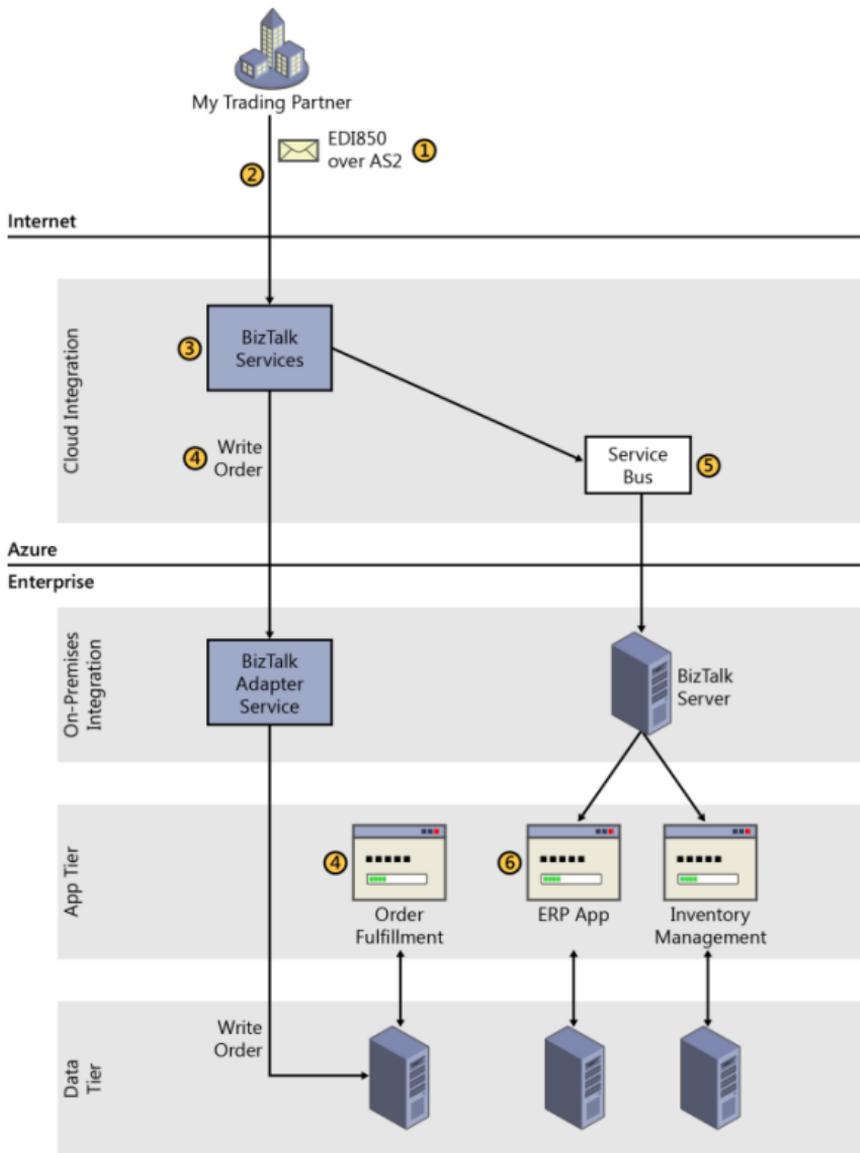
As shown in the illustration:

1. Users browse and order items from phones, tablets, and PCs by using HTML or native applications.
2. Deploy to multiple datacenters for global scale and use Azure Traffic Manager to route requests to the nearest one.
3. Azure Websites scales up and down automatically to manage spikes in customer shopping patterns.
4. Users log in to Azure Active Directory using credentials from Facebook, Google, Microsoft, Twitter, or other identity providers.
5. Save time by using third-party commerce frameworks or your own.
6. WebJobs runs in the background both submitting orders to the on-premises Enterprise Resource Planning (ERP) system and sending order confirmations.
7. Create a global website by using the translation service provided by Bing.

8. Azure is Payment Card Industry Data Security Standard (PCI DSS)–compliant for payment processing.
9. Targeted item recommendations are delivered from a Hadoop-based recommendation engine.
10. Forecast future demand of items by using cloud machine learning.
11. Azure Cache boosts performance of all data services.
12. Use Azure Hybrid Connections to send messages to on-premises databases.
13. Azure Mobile Services provides a unified back end for mobile ordering, including device authentication, data services, and notifications.
14. Send e-commerce confirmations using a third-party app.
15. Geo-distributed CDN keeps video and graphic assets closer to users.

# Business-to-business (B2B) e-commerce

Quickly provision and automate connections to trading partners by using EDI for real-time business.



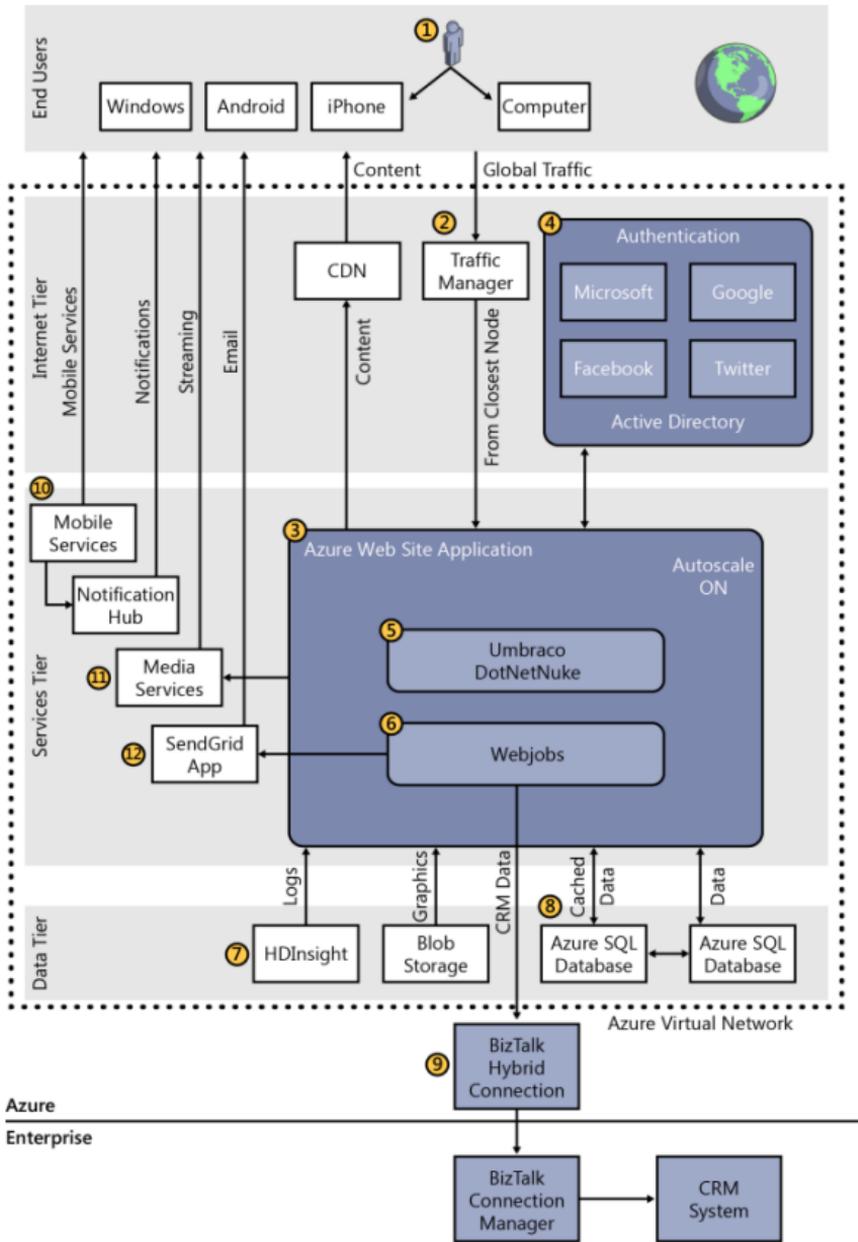
As shown in the illustration:

1. A trading partner orders from the business by using an EDI message.
2. The message is sent to Azure BizTalk Services over the Internet.
3. BizTalk Services processes the message and transforms it into a format understood by your business.
4. BizTalk Services writes the order to the on-premises order database. The order then can be viewed and processed in the on-premises order fulfillment application.
5. BizTalk Services sends another message in a different format to an on-premises BizTalk Server using Service Bus.
6. The on-premises BizTalk Server updates multiple systems of record, such as the ERP and inventory management applications.

## Multichannel marketing

This is a full-featured system that serves customers via both websites and mobile apps. Customers can also view videos on demand. Back-end services include Customer Relationship

Management (CRM) and data analysis of logs and customer data. Data is captured in SQL Database.



As shown in the illustration:

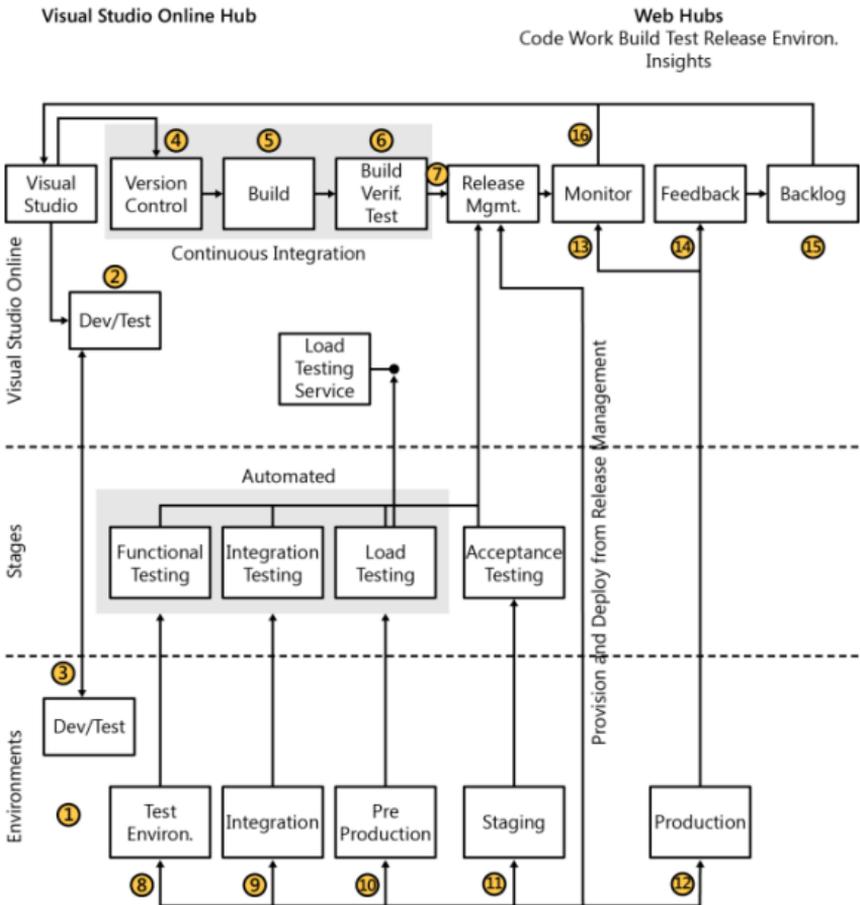
1. Users browse and order items from phones, tablets, and PCs by using HTML or native applications.
2. Deploy to multiple datacenters for global scale and use Traffic Manager to route requests to the nearest one.
3. Azure Websites scale up and down automatically to manage spikes in customer shopping patterns.
4. Users log in to Azure Active Directory by using credentials from Facebook, Google, Microsoft, Twitter, or other identity providers.
5. Save time by using third-party content management frameworks such as Umbraco or DotNetNuke.
6. WebJobs runs in the background both submitting orders to the on-premises ERP system and sending order confirmations.
7. Analyze website logs and customer data by using HDInsight, based on the Hadoop framework.

8. Azure Redis Cache boosts performance of all data services.
9. Use Hybrid Connections to send messages to on-premises databases.
10. Mobile Services provides a unified back end for mobile ordering, including device authentication, data services, and notifications.
11. Media Services supports HD-quality video encoding, live streaming, and on-demand playback.
12. Send e-commerce confirmations by using *SendGrid\**, a third-party app in the Azure Store.
13. Geo-distributed CDN keeps content closer to users.

## DevOps

Deliver value to your customers faster, improve software quality, and get feedback on performance and usage. Remove barriers between developers, operations, and users to streamline your development workflow from the backlog into production.

Visual Studio Online is a hub in the cloud that provides services to your team: agile planning, version control of your source code, building, testing, release management, and application insights. Visual Studio can be installed on physical on-premises hardware or on VMs in Azure.



As shown in the illustration:

1. VSO environments hosted in Azure. VMs are provisioned as needed for each environment. They are configured by using tools such as PowerShell Desired State Configuration (DSC) or Chef.
2. Your developers set up development machines using Azure and also any test machines that they need.
3. Set up VMs by using PowerShell DSC to easily create the environments needed for coding and testing.
4. Host your code in the cloud to access it anytime and from anywhere. Pick a centralized version control system by using Team Foundation Version Control or use Git repos for a distributed workflow.
5. Build your code by using the hosted build controller provided by Visual Studio Online.
6. Run build verification tests as part of your process to check the quality of your code before you do more testing.
7. If the tests pass, use that build to deploy your app to environments for each stage in

your release process. Manage the steps in the process with approvals for each step.

8. First deploy to a test environment for automated functional testing.
9. Deploy to an environment for integrated testing with other services that your app needs.
10. Test that your app performs under load by using cloud-based load testing provided via Visual Studio Online.
11. Use Release Management to assign approvers who must sign off before any release can be deployed to production.
12. When ready, deploy your code into production.
13. Use Application Insights to monitor your live application for performance and availability issues as well as usage.
14. The usage feedback and direct customer feedback are both added to the product backlog to be included in the agile planning process.
15. Use the product backlog to determine what the highest priority items are for your users

that your team should be working on. As the work is done, the continuous integration process checks the quality of the code.

16. If an issue occurs during production, developers can use IntelliTrace logs to debug the problem.
17. The Visual Studio Online hub makes it easy to find out everything about the status of a project in one place.

# Sample technology scenarios

In this appendix, we provide solutions to common technology problems in enterprise cloud computing. Unlike Appendix A, in which we presented full sample solutions, in this section we show “components” that you can add to any application; in fact, most applications will require many of these.

## Hybrid cloud scenarios

When creating a hybrid (a mixture of on- and off-premises computing) enterprise cloud application or set of applications, a number of

opportunities arise to both simplify operations and cut costs. Here, we show a few ways to effectively utilize the cloud for common IT operational scenarios.

## Hybrid cloud connectivity

In a hybrid cloud, some applications are hosted on-premises, whereas others reside in the cloud. Ideally, where these applications live is transparent to end-users. In other words, cloud-resident applications should appear to be within the on-premises network, with appropriate IP addressing and routing. Applications in the cloud are configured to be in the same IP range as those in the datacenter through the Microsoft Azure portal.

There are a number of approaches to achieving this type of location transparency. This section describes four separate ways to connect a datacenter to Azure:

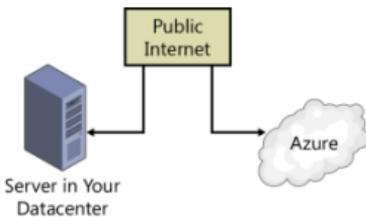
- Point-to-site connectivity
- Site-to-site connectivity
- Azure ExpressRoute (via an Exchange Provider)

- ExpressRoute (via a Network Service Provider)

The choice you make will depend on the how you calculate the bandwidth/cost tradeoff; the need, or not, to be isolated from the open Internet; and how geographically dispersed your sites are.

## Point-to-site

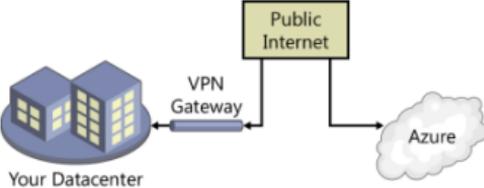
Using the Internet, you can create such a virtual private network (VPN) in two ways. The first is called *point-to-site* connectivity, in which the VPN is configured through software on individual client computers in the datacenter. The least expensive of all the options, point-to-site connections are useful when only a few machines on-premises need connectivity to the cloud, or when the connection is from a remote or branch office.



## Site-to-site

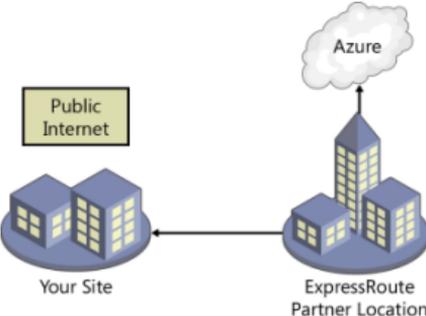
Another approach is called *site-to-site* connectivity. In this configuration, a datacenter

deploys a hardware VPN gateway to link the on-premises datacenter in its entirety with applications and data in the cloud. The hardware gateway must have a public-facing IP address and a technician must be available to perform the configuration.



## ExpressRoute via an Exchange provider

When it comes to accessing their cloud applications, many enterprises want configurable and deterministic network latency. They might also want their network traffic isolated from the public Internet. To support these requirements, a direct connection from the datacenter to Azure using a partner telecommunications carrier, called *ExpressRoute*, is provided, as depicted in the illustration that follows.

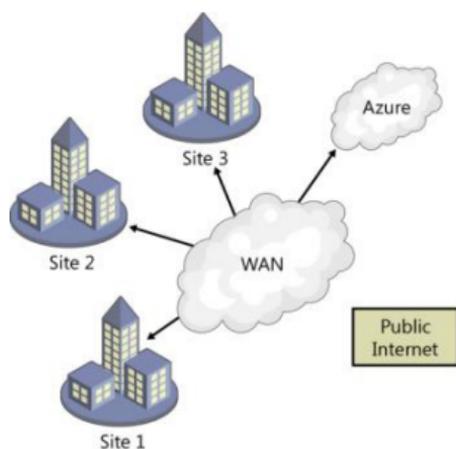


Although this is potentially a more expensive solution, ExpressRoute provides the fastest connectivity as well as isolation from the Internet, essentially by connecting via a “dedicated line.”

A full list of supported telecom providers for ExpressRoute is available on the Microsoft website at <https://azure.microsoft.com/en-us/documentation/articles/expressroute-locations/>.

## ExpressRoute via network service provider

In addition, it is possible to connect through a telecom network service provider such that Azure simply appears as another site on the enterprise’s wide area network.



As with the previous approach, by using a telecom provider as the transport, you can negotiate bandwidth with the provider and, of course, network isolation is provided. You will need to work with your telecom provider to find the best approach for your organization.

In the next few sections, we will describe a series of common application-level scenarios.

## Using the cloud for data backup and recovery

Data backup and replication is one of the most common and straightforward uses of the cloud in hybrid scenarios. Cloud storage is relatively inexpensive and, for all intents and purposes, unlimited, and these facts open up a number of useful application scenarios. In the next few sections, we will examine several such scenarios in which Azure storage complements on-premises assets.

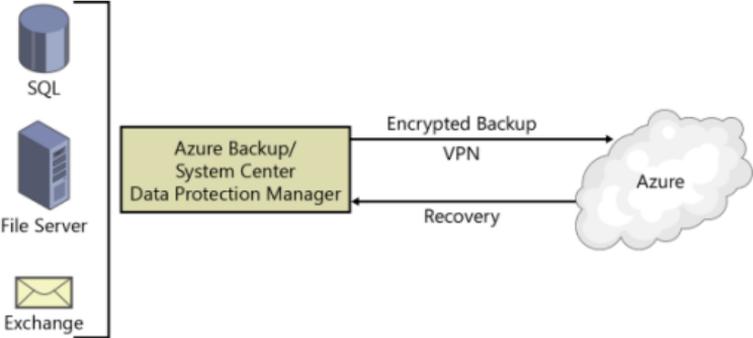
### Azure Backup

Backup, of course, although unheralded, is one of the most important functions that any IT department performs. In many cases, compliance or other legal requirements force businesses to retain data for long periods of time. Traditionally,

backup requires secondary media, a secure location to store backups, and a set of operational procedures to both carry out the backup process and recover the data in the event of a catastrophe.

By employing easy-to-use tools and the inexpensive storage available in the cloud, you can augment or replace existing backup mechanisms with Azure Backup.

Azure Backup operates in a hybrid model, utilizing a VPN tunnel to connect onsite resources to the Azure cloud. You can then use Microsoft System Center Data Protection Manager, enhanced for the cloud, to backup and restore data. Azure Backup will retain data for up to 99 years with 99.9% availability. Backed-up data is secure and encrypted, and other features such as data compression and bandwidth throttling ensure optimum use of IT and network resources.



## *Design considerations*

When designing your IT ecosystem to take advantage of Azure Backup, think about the following:

- Which applications would most benefit from offsite backup? This will help you to prioritize the applications to which you should deploy Azure Backup first.
- How much data do these applications maintain? This will help you to properly size the offsite storage.

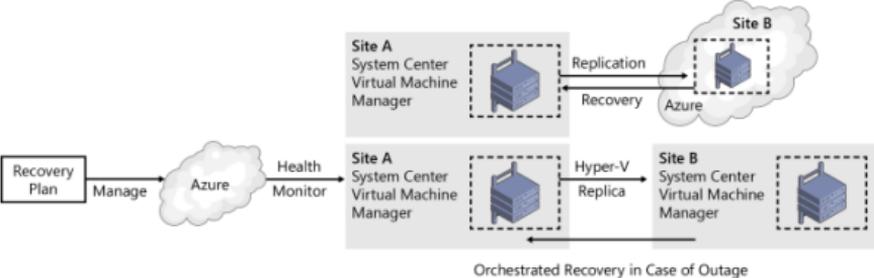
## **Azure Site Recovery**

Perhaps the greatest fear any IT manager has is downtime of the entire IT environment. When an enterprise IT environment fails—due to, for example, an on-premises datacenter outage—the results can be catastrophic for a business. Moreover, when the outage is remediated, systems often must be brought back online in a particular order to smoothly restore operations.

The cloud presents a number of new opportunities for enabling business continuity and disaster recovery. (Data backup and recovery was discussed in the previous section.) Azure Site Recovery makes it possible for workloads to be

quickly replicated to Azure, and to be restored in an orderly fashion using Orchestrated Disaster Recovery as a Service (DRaaS). Enterprise IT professionals can create recovery plans, dictating specifics such as which workloads must be run first, or only running a workload upon the successful completion of an integrity check.

Using Microsoft System Center Virtual Machine Manager, you can replicate virtual machines (VMs) as well as physical servers to the Azure cloud under the control of site-defined policies. You can also “burst” workloads to Azure when surges occur. Microsoft System Center Operations Manager will also monitor the operation of the on-premises systems from Azure, ensuring that failures are detected and managed as quickly as possible.



### Design considerations

When planning a Site Recovery deployment, consider the following:

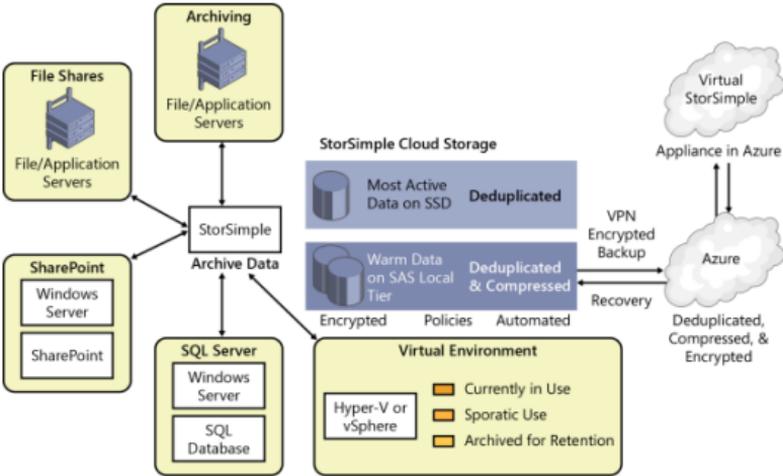
- What are your Recovery Time Objective (RTO) and Recovery Point Objective (RPO)? RTO is the time desired to bring an application or ecosystem back online; RPO describes the state of the data after the system is recovered.
- Which systems must you bring back up first, and which systems depend on others before you can restart them? For example, it might be necessary for the database server to be online prior to starting a web server or SharePoint system. Knowing this sort of information will aid in building a recovery plan.

## Archiving by using the Azure StorSimple appliance

Many businesses face situations in which they are required to retain large amounts of data for long periods of time, often with the expectation that this data will rarely or never be accessed. One way to accomplish this is through the use of a custom on-premises storage appliance designed to communicate with the cloud.

The StorSimple storage appliance manages SharePoint, SQL Service, and ordinary file share data, and “ages” out infrequently accessed data

to the Azure cloud in a highly efficient (encrypted, compressed) fashion.



In the preceding illustration, you can see the scenarios in which the StorSimple appliance is commonly used. These include the following:

- Archive
  - Archives and disaster recovery
  - Dramatic cost reduction
  - No changes to application environment
- File shares
  - File share with integrated data protection

- All-in-one primary data + backup + live archives + disaster recovery with deduplication and compression
- SharePoint
  - SharePoint storage on StorSimple and Azure
  - StorSimple SharePoint Database Optimizer
  - Improved performance and scalability
- SQL Server
  - Storage for Tier 3 – 3 SQL databases
  - Integrated backup, restore, and disaster recovery
- Virtual environments
  - Control virtual sprawl
  - Cloud-as-a-tier
  - Offload storage footprint
  - VMware Storage DRS Storage pools
  - Virtual Machine Archive
  - Regional VM Storage

The benefits of using a storage appliance that takes advantage of the Azure platform are the consolidation of primary storage with archive, backup, disaster recovery through seamless integration with Azure, cloud snapshots, deduplication, compression, and encryption. Together, these benefits result in an average reduction of enterprise storage total cost of ownership (TCO) of 60 to 80 percent.

### *Design considerations*

When planning an Azure archiving deployment using the StorSimple appliance, consider the following:

- What types of data do you want to save in the cloud using StorSimple? For example, you might choose to use Azure Backup for transactional SQL data if it is highly unlikely that your users will be querying days-old data. Alternatively, you might use StorSimple for data in SharePoint that is only occasionally referenced but needs to be accessible in real time.

- Consider StorSimple for data with legal retention requirements; for example, you must keep the data available for long periods of time.

## Hybrid database scenarios

Many enterprises have made significant investments in on-premises SQL Server. A number of features extend the functionality of on-premises SQL Server to the cloud, taking advantage of Azure's low cost and massive scale.

For example, you can synchronize an on-premises instance of SQL Server with either an instance of SQL Server running in a VM in Azure (i.e., in an IaaS instance) or with the cloud-native SQL Azure. This makes it possible, for example, for dispersed teams to do development on the on-premises instance, as shown in the diagram that follows.

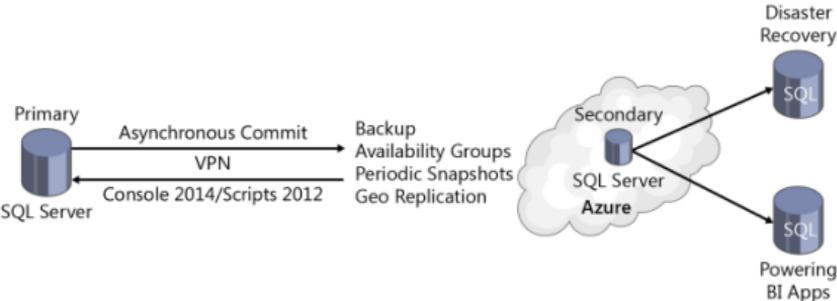


In addition, you can use an Azure instance of SQL Server as a backup target for an on-premises instance. Alternatively, for a very cost-

effective solution, SQL Server (either on-premises or cloud-based) can back up to and be restored from low-cost Azure Blob storage.



Finally, as another pattern, you can use the cloud to provide additional capabilities to an on-premises instance, lowering its load. In the example presented in the following illustration, there are two cloud replicas: one being kept as a backup for disaster recovery purposes; the other being used to power business intelligence (BI) applications.



# Development and test

All enterprise IT departments require development and test environments where developers can create and validate new

applications or new versions of existing applications prior to full production deployment. Because of its on-demand nature, the cloud provides an ideal environment to rapidly spin-up servers and storage for development and test, and then spin them down when they're no longer needed. In this way, the cloud can save IT organizations significant expense because they can replace datacenters dedicated to development and test with on-demand resources.

## Application development

Application developers and testers can quite easily make use of cloud resources. VMs to host applications can be spun-up within a matter of minutes with no need to go through a procurement cycle.

Azure provides a wide variety of operating systems (OSs) and OS versions to which you can deploy a test application. In fact, some developers host their development tools (such as Visual Studio) on a VM in Azure and use Remote Desktop Protocol (RDP) to connect. Thus, they can get to their development environment from anywhere in the world at any time and have no need to worry about data loss because their code and data are replicated.

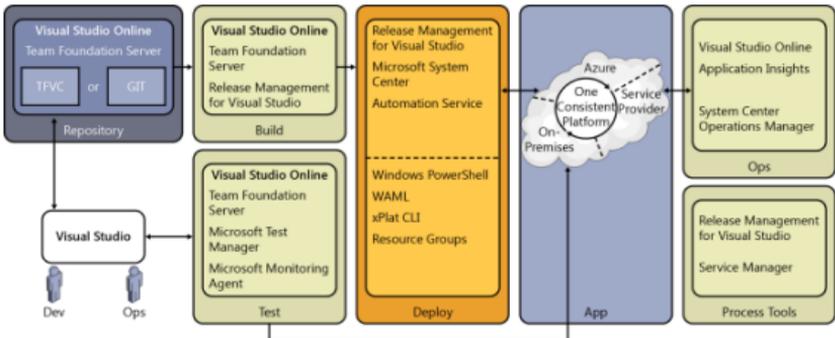
Visual Studio, moreover, can run its development, test, build, and source-code control capabilities entirely in the cloud. Increasingly, users rely on Visual Studio Online and Visual Studio Team Foundation Server to manage development workflow and backlogs.

Of course, all cloud providers are relatively agnostic regarding choice of operating system and programming language. With Azure, for example, it's as easy to write and test applications in Java (or Python, or PHP, or any number of other languages) as a .NET language, and many such applications run in Linux VMs.

With Azure DevTest Labs, IT administrators can govern how cloud resources are used for development and test. For example, with quotas and policy enforcement, the number of VMs used by a development team can be monitored and limited, thus ensuring the most cost-effective use of cloud resources. In addition, DevTest Labs lets IT organizations maintain enterprise-specific application templates, and plug-ins accommodate integration with various development and release tools.

# Microsoft SharePoint

One of the most common workloads in any enterprise IT environment is SharePoint. Like all applications, SharePoint applications must be developed and tested prior to deployment. Using the cloud as a “dev/test” environment makes both operational and financial sense. Using a VPN or high-speed connection such as ExpressRoute, it is possible to replicate on-premises SharePoint environments to the Azure cloud.



# High availability in the cloud

Many mission-critical applications require the highest availability possible, and must be resilient to hardware and network failures. Hosting applications in the cloud provides a number of capabilities, including redundancy,

fault tolerance, and resilient design that make high availability possible.

First, consider Azure Service Level Agreements (SLAs).<sup>19</sup> For example, the Azure Compute service (application services) comes with a 99.95% SLA; Azure SQL Database has a 99.9% SLA; and Azure Storage has a 99.90% SLA. Without any additional work, your application is by default guaranteed no more than 108 minutes of downtime in a month (out of 43,200 minutes).

However, much can be done to improve even these excellent numbers. Programming techniques such as durable queues and asynchronous communications make applications less tightly coupled to one another, improving chances that one failure will not cause a cascade of further failures.

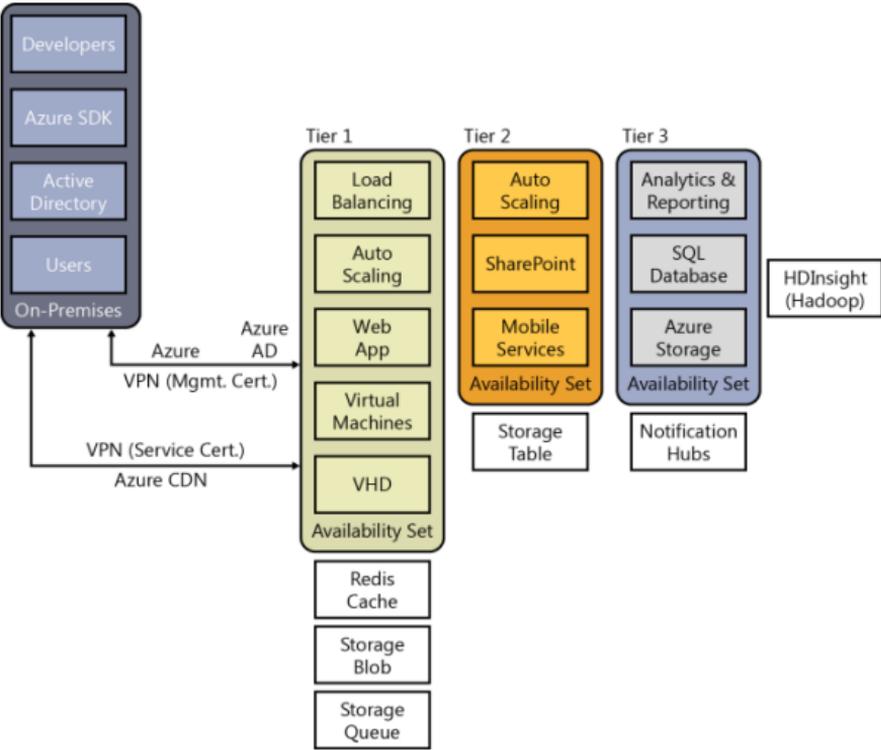
Using Azure availability sets ensures that different instances of VMs, and/or different workloads, are physically placed on different racks (different power supply, switch, and server) in an Azure datacenter. Availability sets ensure that should a planned or unplanned

---

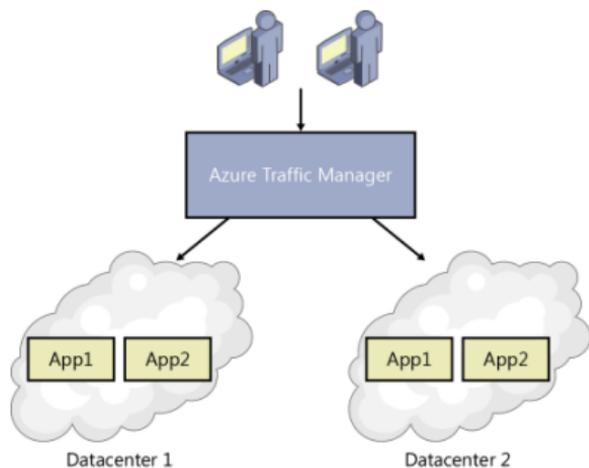
<sup>19</sup> See <https://msdn.microsoft.com/en-us/library/azure/dn251004.aspx> for more detail.

maintenance event or failure occur, at least one VM instance will be available for use.

It is also efficient to “tier” applications into availability sets. By placing all “web tier” applications into a single availability set, it becomes straightforward to reboot or upgrade the entire tier at once, with the underlying availability set logic ensuring that at least one of each application is available. In the example presented in the following illustration, a workload is spread across three tiers; each tier is associated with a different availability set.



In addition, as the next illustration shows, you can place workloads on geographically separate datacenters. You can use Azure Traffic Manager to switch operations from the primary datacenter to the backup in the event of a catastrophic failure in the primary.



### *Design considerations*

When thinking about availability of your applications or workloads, consider the following:

- Do you require an SLA greater than 99.5 percent, which is the default Azure SLA?
- How many instances of each application VM do you require?
- Which applications can make use of asynchronous and loosely coupled

programming techniques to improve their availability?

- Would geographically redundant datacenters improve your workload availability within your cost parameters?

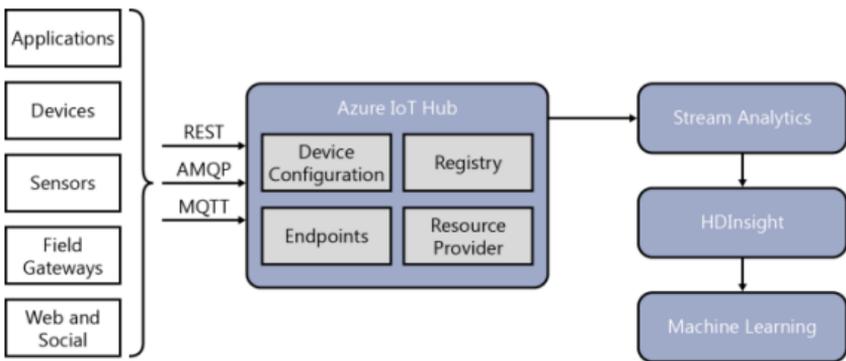
## Connected devices

The Internet of Things (IoT), as discussed earlier in the book, carries great promise on the cloud, but it also levies great demand. IoT devices range from medical sensors to manufacturing devices, to connected cars and airplanes, to building environmental sensors—and on and on. Estimates suggest that within a few years, tens of billions of such “things” will be attached to the Internet in some form. In the scenario presented in this section, the cloud receives, analyzes, and takes action on data sent by IoT devices.

Azure provides a number of services to accommodate the IoT. With the Azure IoT Hub, enterprises can create a device registry listing all allowed connected devices, and can manage, configure, and provision them. The IoT Hub makes it possible for all cloud applications to ingest very large numbers of events (billions per day, if needed) from connected devices. Then, Azure Stream Analytics can analyze these events

in real time, performing filtering operations and only passing on those events of interest (such as a device failure).

Other useful services, which you can see in the following illustration, include Azure HDInsight, which is capable of collecting very large amounts of data and running batch analytics programs (e.g., MapReduce) to find patterns; using Azure Machine Learning, you can detect anomalous patterns and predict future outages or downtime.<sup>20</sup>



Of course, with all these devices connected to your application, security must be part of the

---

<sup>20</sup> You might be interested in a free e-book on this topic. For your copy of *Microsoft Azure Essentials: Azure Machine Learning* by Jeff Barnes (Microsoft Press, 2015), go to

<http://www.MicrosoftVirtualAcademy.com/ebooks#9780735698178>.

architecture. Many IoT devices do not have the computing power to perform full public-key encryption or digital signature, so you should be familiar with and use wherever possible Shared Access Signatures (also known as SAS tokens). A SAS signature, as the name implies, is about *access*; the token includes in its query parameters the URL being requested, an expiry time, permissions, and other key data. SAS tokens provide an efficient way to guard against unauthorized access to your application from intruders.<sup>21</sup>

### *Design considerations*

When designing an application that uses Internet-connected “things,” think about the following:

---

<sup>21</sup> For more information on Shared Access Signatures, see <https://azure.microsoft.com/en-us/documentation/articles/storage-dotnet-shared-access-signature-part-1/> and <https://azure.microsoft.com/en-us/documentation/articles/storage-dotnet-shared-access-signature-part-2/>. It is important to recognize that SAS is not an IoT-only technology; you also can use it, for example, with Azure Storage to provide delegated access to data.

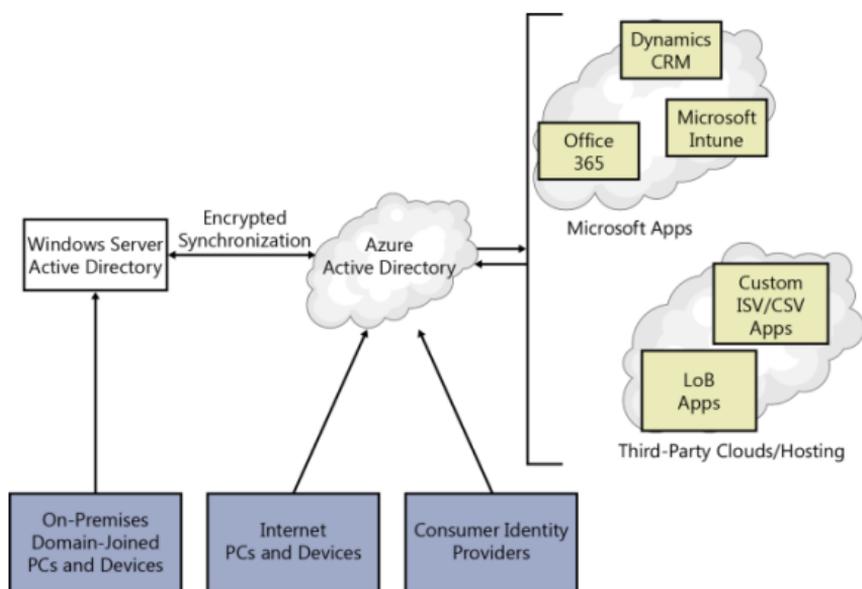
- How many devices will be connecting? How frequently will these devices be sending data and how large are the messages? This will help you to determine the scale of Azure Event Hubs, which you will need to receive and process the messages.
- What protocol (HTTP/REST, AMQP, MQTT) will they use to connect?
- What sorts of data will they send, and what of that data is useful to applications?
- Do you need to retain the data for any reason?
- How do you want to visualize the state of your devices? Do you need a “dashboard” (such as Azure Power BI) to aggregate and visualize the data coming in?

## Identity and authentication

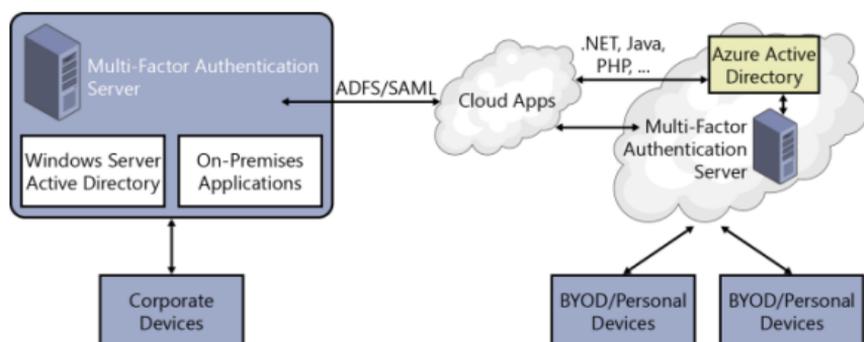
Identity management is the core of security in the cloud. A user’s identity determines to which resources that user has access, and the identity management system prevents unauthorized access where appropriate, protecting enterprise resources.

In Azure, identity management is handled by Azure Active Directory, based upon the industry standard Active Directory family of products. You can use Azure Active Directory to authenticate users to cloud applications, synchronizing with and federating to an on-premises Active Directory such that enterprise users can take advantage of single sign-on (SSO) to access both on-premises and cloud applications.

Using the OAuth/OpenID protocol, you can optionally turn on other forms of identity with Azure Activity Directory. For example, Azure Activity Directory supports Facebook, Google, Yahoo, and Microsoft Accounts as identity providers, and you can grant each of these varying levels of access. In addition, you can integrate a wide variety of SaaS applications (such as Salesforce.com and many others) with Azure Active Directory. In addition, Azure Multi-Factor Authentication ensures compliance with NIST 800-63 Level 3, HIPAA, PCI DSS, and other regulatory requirements.



Finally, Azure Active Directory supports two-factor authentication for rigorous identity management. Typically, a user first authenticates by using conventional credentials such as username/password, and then uses a physical device such as a smartphone or smartcard to complete the authentication process. You can configure Azure Active Directory to call a smartphone and request a PIN or request a badge be read, or perform a biometric authentication (for example, fingerprint).



## Design considerations

It has been said that identity management is at the core of the cloud because this controls access to its compute and data resources. With this in mind, you should consider the following:

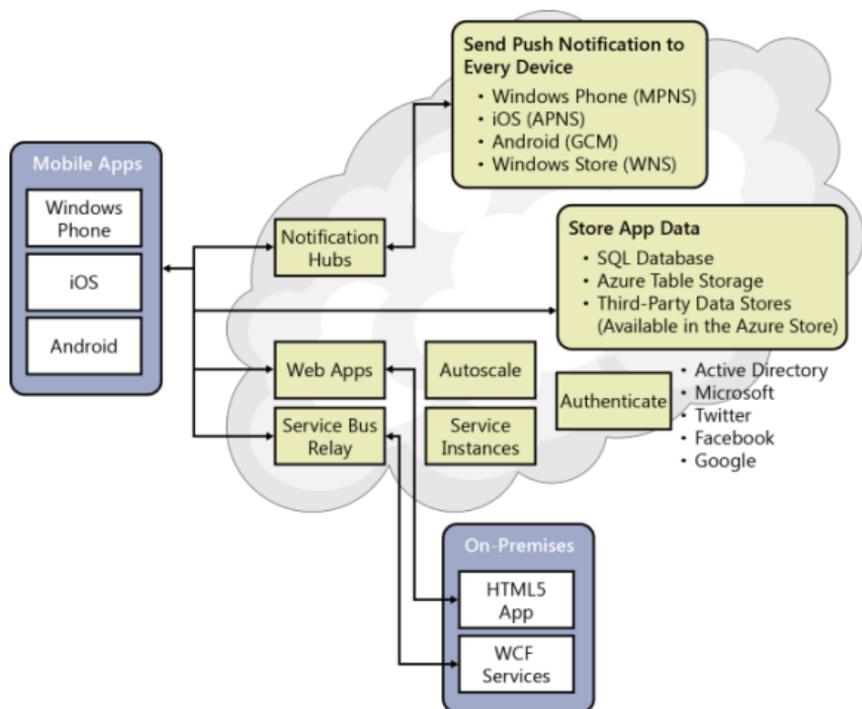
- Federating your on-premises Active Directory to Azure Active Directory to turn on SSO for cloud applications
- Turning on consumer authentication mechanisms for certain types of access (such as e-commerce customers) to your cloud applications
- Implementing two-factor authentication for the most rigorous authentication requirements

# Mobile applications

Most modern cloud applications have abilities to communicate with mobile devices. Azure provides a number of services tailored for mobile applications. Azure, of course, supports all major mobile architectures (Windows Phone, iOS, Android), including the following:

- **Notification Hubs** This is a service to push data from the cloud application to potentially millions of mobile devices; for example, breaking news or location-based information.
- **Service Bus** This is a set of services providing relay, publish-and-subscribe (topic-based), or queued communications between cloud applications and devices. For example, a mobile phone application might only be interested in traffic patterns in a certain zip code, and so could subscribe the relevant topic. Service bus uses Shared Access Signatures (based on SHA-256 hashes) as its primary security mechanism.

The mobile application scenario shown in the following illustration takes advantage of several Azure services, including Service Bus Relay and Notification Hubs.



## Design considerations

When building applications that have a mobile component, consider the following:

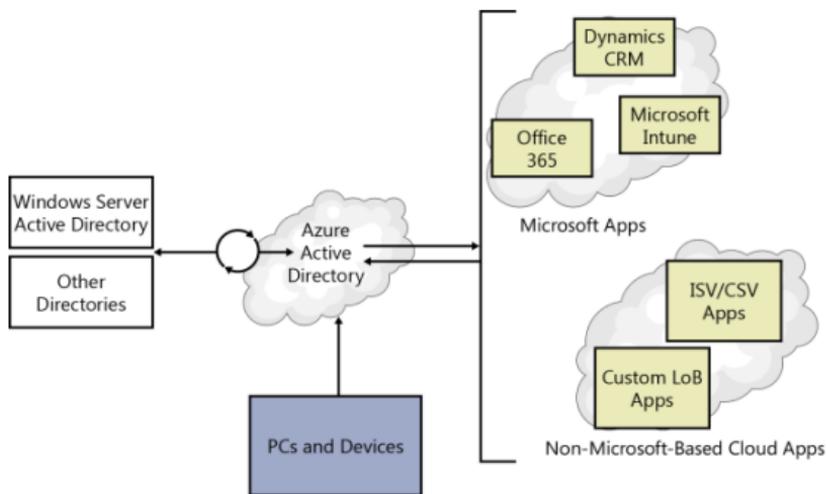
- Which mobile platforms are you targeting?
- Do you need to reformat client data for different form factors? If so, consider using the HTML5 Responsive Web Design<sup>22</sup> features to adapt the content to different size and resolution displays.

<sup>22</sup> See <https://msdn.microsoft.com/en-us/magazine/hh653584.aspx>

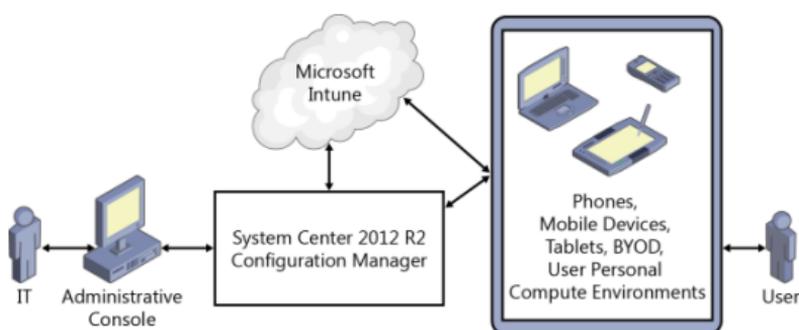
- Is the application targeted specifically at enterprise users or at the general public user population?
- Do you need push notifications or alerts to your users? If so, consider using Azure Notification Hubs.

## Enterprise mobility management

In 2014, a number of mobility-related services were bundled together to provide a cohesive mobility offering for enterprise IT departments. This bundle is called the *Enterprise Mobility Suite* (EMS), and it includes Azure Active Directory as well as additional services, including the ability to do group management and password self-service reset. It also provides preconfigured sign-on to a large number of SaaS applications and security reporting (e.g., for repeated failures, anomalous login patterns) and can accommodate two-factor authentication, described earlier. The figure that follows presents a graphic depiction.

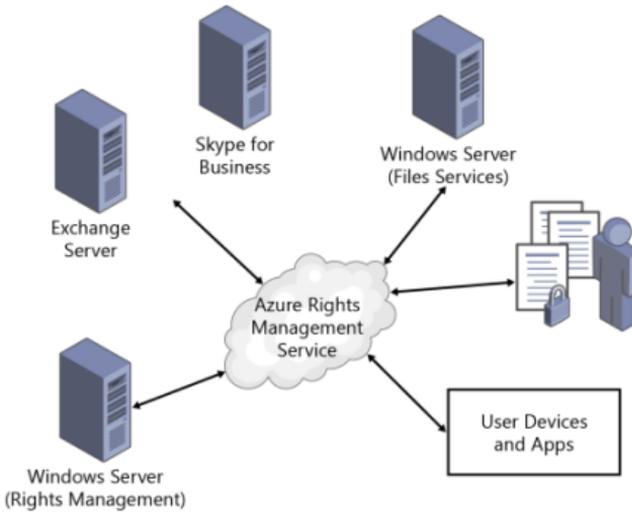


The EMS suite also includes a comprehensive Mobile Device Management (MDM) offering using Windows Intune. As is illustrated in the figure that follows, this makes it possible for IT professionals to manage mobile access to enterprise resources as well as provide abilities to perform email profile management, selective wipe, and remote lock and password reset.



Finally, EMS also comes with Azure Rights Management, providing robust document protection for both Office 365 (cloud) and on-

premises information. EMS marks great patterns that highlight hybrid identity management, mobile device and security management, mobile application management, as well as strong authentication and access-based information protection.



### *Design considerations*

Consider using the EMS if any of the following apply to you:

- You have a need to manage a variety of mobile devices.
- You want to enable users to do their own password set and reset (and thus reduce the load on your help desk).

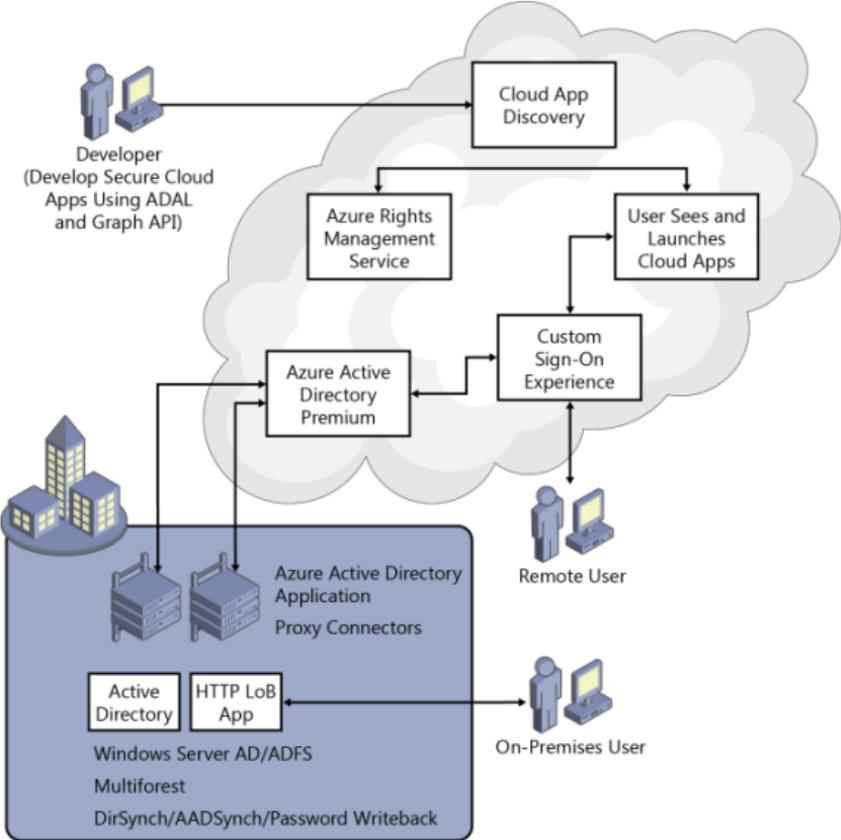
- A significant number of the mobile devices connecting to enterprise networks are actually employee-owned; for example, if your company has a Bring Your Own Device (BYOD) policy.
- You need to enforce specific data access privileges and policies for different users or classes of users.

## Websites

Using Azure Web Apps and App Service, creation and maintenance of a complex enterprise website is straightforward and inexpensive. You can build advanced HTML5-based websites using any of a number of popular web application programming languages (e.g., .NET, Java, PHP, Node.js, and Python). A wealth of tools gives your site the ability to connect to both other web assets (such as Twitter) and on-premises data assets. Developers can create secure, authenticated web applications by using Active Directory features such as Active Directory Authentication Library (ADAL) and the Active Directory Graph API, and you can secure access to documents via Azure's Rights Management Service (RMS). As mentioned earlier, you can connect to and synchronize Azure Active

Directory with an on-premises deployment of Active Directory.

The pattern presented in the illustration that follows shows website development, access, and on-premises assets.



### Design considerations

Of course, there are a plethora of design issues any time you are building and deploying a website. Here are a few that you should consider:

- Is it an intranet-only site or accessible from the broader Internet?
- How will you do content management to refresh data?
- Do you need the site to authenticate users? If so, can users authenticate with non-enterprise credentials such as Facebook, Google, or Microsoft Account? If so, do they have different access rights than enterprise users? Azure Active Directory can provide an easy, all-in-one authentication solution.
- What sorts of application integration with enterprise applications do you require? You can use BizTalk Server or Logic Apps to connect to on-premises applications such as ERP or databases.
- Do you need to perform B2B transactions on your database? Azure BizTalk Services provides the ability to connect to EDI X.12 applications elsewhere on the Internet.

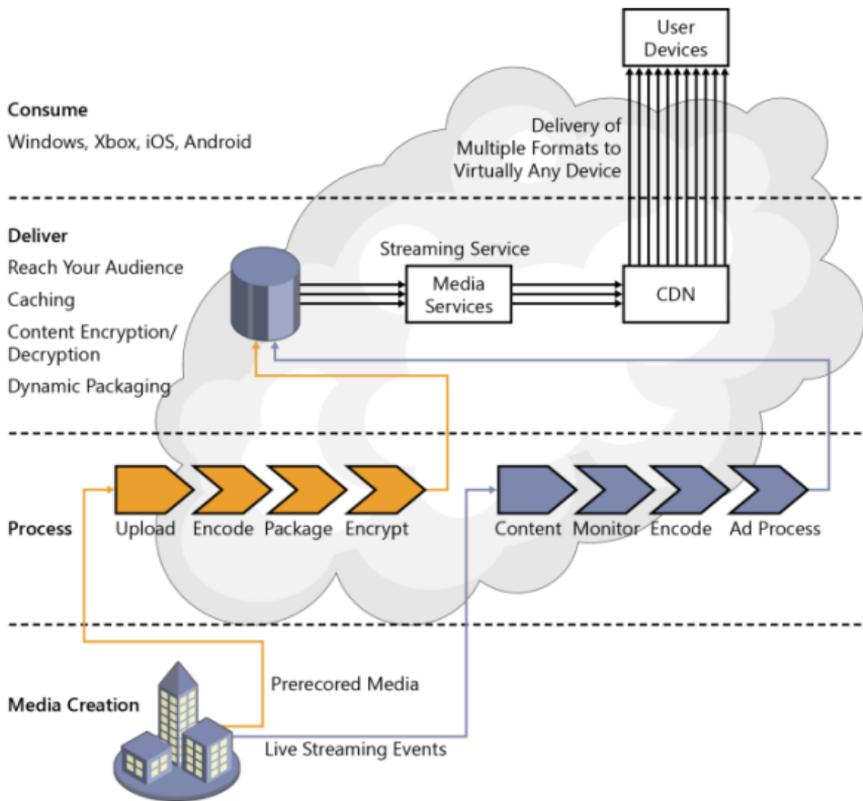
## Azure Media Services

Many applications include streaming media, for a variety of purposes (instructional, entertainment, etc.). Media applications at their core carry out four key functions:

- Upload the media to the cloud application
- Encode the media in one or more formats by using the appropriate algorithms (codecs)
- Package the encoded media in the correct format such as MPEG-DASH or Apple HLS (Azure can perform the packaging on the fly);
- Optionally, encrypt the media for secure transmission and delivery (Azure can also perform encryption on the fly)

Azure provides studio-grade encoding, with the option to include thumbnails, image and audio overlays, sub-clips and multiple-clip stitching; a Workflow Designer tool gives media professionals the ability to precisely tailor their output.

To ensure low-latency delivery, media professionals can make use of Azure's Content Delivery Network (CDN), which brings the media content to the servers and Azure datacenters closest to the consumer.



## Design considerations

When designing a cloud application that takes advantage of media, consider the following:

- Do you need to do encoding and/or encrypting at scale? Many companies can save money by using cloud resources to do media encoding rather than maintaining on-premises servers to carry out this function.
- Is the media you intend to distribute offline (stored as files) or streaming live? For saved

files you can use a variety of offline encoders and compressors to provide the data in the most efficient format possible for the user and the device.

- What is the format of the data (MP4, OGG, etc.)? Does it need to be re-encoded before transmission?
- How many users do you expect will be simultaneously viewing or listening to the media? Are they geographically dispersed? If you have a large number of users or if they are in different regions, you might want to consider using a CDN such as Azure CDN to cache the media near the users.

## Migration strategies

When contemplating the migration of IT assets to the cloud, a number of options present themselves. In this section, we present a convenient table to help categorize the various options.

In the first section of the following table are a series of application characteristics and some common examples. In the section following that are the recommended choices for migration, based on the five “R’s” presented earlier in the

book. Then, some recommended tools for performing the migration are shown, followed by the last section, which lists various Azure components that might be a part of the final cloud solution.

Migration scenario 1	Migration scenario 2	Migration scenario 3	Migration scenario 4
Legacy app operating system uses default roles and features provided by the operating system	Legacy app operating system used as a web application server	Legacy app operating system uses a third-party app on top of the operating system	Legacy app operating system used as a database server
File/application servers For example: <ul style="list-style-type: none"> <li>• Active Directory</li> <li>• File server</li> <li>• DNS</li> <li>• DHCP</li> <li>• WSUS</li> </ul>	Web Server For example: <ul style="list-style-type: none"> <li>• IIS + .NET</li> <li>• Java WAS + I2EE</li> </ul>	Third-party App Server For example: <ul style="list-style-type: none"> <li>• Custom software</li> <li>• ERP</li> <li>• CRM</li> </ul>	Database Servers <ul style="list-style-type: none"> <li>• SQL Server</li> <li>• Oracle</li> <li>• MySQL</li> </ul>
Methods:			
<ul style="list-style-type: none"> <li>• Rehost</li> <li>• Refactor</li> </ul>	<ul style="list-style-type: none"> <li>• Refactor</li> <li>• Revise</li> </ul>	<ul style="list-style-type: none"> <li>• Revise</li> <li>• Rebuild</li> </ul>	<ul style="list-style-type: none"> <li>• Refactor</li> <li>• Revise</li> </ul>

<ul style="list-style-type: none"> <li>• Replace/retire</li> </ul>	<ul style="list-style-type: none"> <li>• Replace/retire</li> </ul>	<ul style="list-style-type: none"> <li>• Replace/retire</li> </ul>	<ul style="list-style-type: none"> <li>• Rebuild</li> <li>• Replace/retire</li> </ul>
Tools for migration:			
<ul style="list-style-type: none"> <li>• Windows Server Migration Tool (WSMT)</li> <li>• Migration guides</li> </ul>	<ul style="list-style-type: none"> <li>• Web Deploy 3.5</li> </ul>	<ul style="list-style-type: none"> <li>• Follow the 9-Step Migration Process</li> <li>• Take advantage of MCS and Partners (AppZero, Vision Solutions)</li> </ul>	<ul style="list-style-type: none"> <li>• MAP Toolkit</li> <li>• SQL Server Migration Assistant (SSMA)</li> </ul>
In Azure, consider using:			
<ul style="list-style-type: none"> <li>• Active Directory Premium</li> <li>• IaaS</li> <li>• Azure Storage</li> <li>• ExpressRoute or VPN</li> </ul>	<ul style="list-style-type: none"> <li>• Active Directory Premium</li> <li>• Azure App Service Web Apps</li> <li>• IaaS</li> <li>• ExpressRoute or VPN</li> </ul>	<ul style="list-style-type: none"> <li>• Active Directory Premium</li> <li>• IaaS</li> <li>• Office 365</li> <li>• CRM Online</li> <li>• ExpressRoute or VPN</li> </ul>	<ul style="list-style-type: none"> <li>• Active Directory Premium</li> <li>• IaaS</li> <li>• Office 365</li> <li>• CRM Online</li> <li>• ExpressRoute or VPN</li> </ul>

# Recommended references

This appendix contains additional resources that can help your efforts to migrate to the cloud.

## Storage references

- Understand and Troubleshoot Storage Spaces:  
<http://microsoft.com/download/details.aspx?id=29002>
- Deploy and Manage Storage Spaces with PowerShell:  
<http://microsoft.com/download/details.aspx?id=30125>
- New D-Series Virtual Machine Sizes:  
<http://azure.microsoft.com/blog/new-d-series-virtual-machine-sizes>
- Azure's getting bigger, faster and more open:  
<http://azure.microsoft.com/blog/azures-getting-bigger-faster-and-more-open>

## Application development and insights references

- Get started with Visual Studio Application Insights:  
<http://azure.microsoft.com/documentation/articles/app-insights-get-started>
- Getting started with Application Insights Tools for Visual Studio:  
<http://channel9.msdn.com/Series/Application-Insights-for-Visual-Studio-Online/Getting-started-with-Application-Insights-Tools-for-Visual-Studio>
- Transient Fault Handling:  
[http://msdn.microsoft.com/library/hh680901\(v=pandp.50\)aspx](http://msdn.microsoft.com/library/hh680901(v=pandp.50)aspx)

## Performance best practices references

- Performance best practices for SQL Server in Azure Virtual Machines:  
<https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-sql-server-performance-best-practices/>
- Performance Guidance for SQL Server in Microsoft Azure Virtual Machines:

<http://aka.ms/performance-guidance-SQL-Server-in-Azure-VMs>

- AzureCAT Guidance: SQL Database Sharding Patterns:  
<http://channel9.msdn.com/shows/data-exposed/sqlldbshardingintro>
- Patterns and Practices for Scaling Microsoft Azure SQL Database:  
<http://www.sqlpass.org/24hours/2014/ss2014launch/sessions/details.aspx?sid-7165>
- Data Elasticity Through Database Sharding:  
<http://shauntj.wordpress.com/2015/01/01/data-elasticity-through-database-sharding>
- Introducing Elastic Scale preview for Azure SQL Database:  
<http://azure.microsoft.com/blog/introducing-elastic-scale-preview-for-azure-sql-database>
- Get Started with Elastic Scale tools:  
<http://azure.microsoft.com/documentation/articles/sql-database-elastic-scale-get-started>
- Azure SQL Database Elastic Scale on Channel 9 with Scott Kline & Torsten Grabs:  
<http://channel9.msdn.com/Shows/Data-Exposed/Azure-SQL-Database-Elastic-Scale>

## Other cloud migration references

- MSDN Architecture website:  
<https://msdn.microsoft.com/en-us/dn630665.aspx>
- MSDN Architecture blueprints:  
<https://msdn.microsoft.com/dn630664>
- Microsoft Azure:  
<http://azure.microsoft.com/en-us/>
- MSIT Stratus Business Case study:  
<https://msdn.microsoft.com/en-us/library/dn727097.aspx>
- Microsoft Assessment and Planning Toolkit:  
<https://technet.microsoft.com/library/bb977556.aspx>
- Microsoft cloud and Datacenter Solutions Hub:  
<https://technet.microsoft.com/en-us/cloud/private-cloud>
- Microsoft cloud IT Architecture Posters:  
[https://technet.microsoft.com/en-us/library/dn919927\(v=office.15\).aspx](https://technet.microsoft.com/en-us/library/dn919927(v=office.15).aspx)

## About the authors



**Barry Briggs** is an independent consultant with a long history in software and enterprise computing. He served in a number of roles during his 12-year career at

Microsoft. Most recently, he was the chief enterprise architect at the Microsoft DX (Developer Experience) Team. The DX team's job is to design and build "epic" applications with Microsoft customers that exploit new capabilities of the Microsoft stack, including both Microsoft and open-source products and frameworks.

Previously Barry served as chief architect and CTO for Microsoft's IT organization. Principal among his responsibilities were creating and leading Microsoft IT's cloud strategy team, which put in place the strategy and processes behind the migration of Microsoft's internal IT ecosystem to the cloud. In addition, he led the Enterprise Architecture practice which aligned the business strategies to technology assets for maximum impact and agility. He drove a strategic incubations unit which builds cutting-edge software designed for IT-wide impact, and technology adoption strategies, which fostered

the deep relationship Microsoft IT has with its product groups. Prior to the CTO role, Barry led the team that created the world's largest Master Data Management (MDM) solution for Microsoft. He joined Microsoft in 2003 as senior architect for Business Process and Integration Division, which built Microsoft BizTalk Server.

Prior to Microsoft, Barry served as CTO for a number of companies (Aptsoft, Wheelhouse, BroadVision, and Interleaf); before that, he spent 11 years at Lotus/IBM. There, Barry was the lead architect for Lotus' famous spreadsheet product, 1-2-3, for a number of years. In addition, he also helped develop Lotus Notes and led the technology integration of Lotus with IBM following the latter's acquisition of the former. He also created and led the team responsible for the world's first Java-based productivity suite, Lotus eSuite. In 1995, he was named a Lotus Fellow.

You can see what Barry's up to on his website at <http://www.barrybriggs.com>.



**Eduardo Kassner** is the director of cloud solution architecture in the Worldwide Enterprise and Partner Group at Microsoft. He has more

than 20 years of experience managing and designing complex IT environments, and connecting IT and business objectives in real-life scenarios. Eduardo has designed WW consulting areas, teams, and he has personally lead the strategy and the consulting engagements teams that have enabled international corporations and governments alike to build their cloud adoption strategies in a direct and no-nonsense approach. Eduardo has a proven ability to link the required technology stacks to a bottom- and top-line time-to-value equations for enterprise or government environment alike. He achieves this by designing and deploying structured frameworks with hard-earned experience that link the required technical stacks all the way to the business value frameworks.

Being one of the founding designers of the Microsoft cloud architecture roles, Eduardo then became the cloud and solutions architecture director and role owner, which is tasked to design and lead the Microsoft Cloud Solutions Architecture community worldwide. He directs the team that manages the role, bringing on strategies and methods as well as the cloud patterns, practices, and insights gained from this vast worldwide community. The Microsoft cloud architecture community has the charter to consult with our customers worldwide, and at

the deepest technical level design, and build the pilots, patterns, or prototypes, to then finally create the deployment resources, requirements, and plans to enable corporation or government to take or expand their IT environment to the cloud.

Eduardo was one of the three original authors of the Microsoft Infrastructure Optimization Models, a framework to assess the IT and operational maturity. He recently authored the Cloud First Mobile First model and is working on several cloud adoption frameworks. He has influenced and led the virtualization, private cloud, and operations management strategies for the Microsoft field technical sales and marketing communities for the past 10 years, and is a sought after speaker in forums across the world on these topics. Eduardo has spoken at more than 400 events across the globe on topics such as enterprise cloud strategy, cloud adoption strategy and best practices, public, hybrid and private cloud, virtualization, datacenter management, datacenter efficiency, IT total cost of ownership, and optimized desktop and flexible work style strategies.

Before joining Microsoft, he worked for Dell as a regional sales manager, and before that as the director of the EDS service management center

which consisted of 36 outsourced datacenters, their finances, billing, all of their resources, and service support areas, as well as the five remote print and output centers. These 36 datacenters service more than 128 customers in a full outsourcing model.



From technical overviews to drilldowns on special topics, get free ebooks from Microsoft Press at:

[www.microsoftvirtualacademy.com/ebooks](http://www.microsoftvirtualacademy.com/ebooks)

Download your free ebooks in three formats:

- PDF
- EPUB
- Mobi for Kindle

Look for other great resources at Microsoft Virtual Academy, where you can learn new skills and help advance your career with free Microsoft training delivered by experts.

Microsoft Press