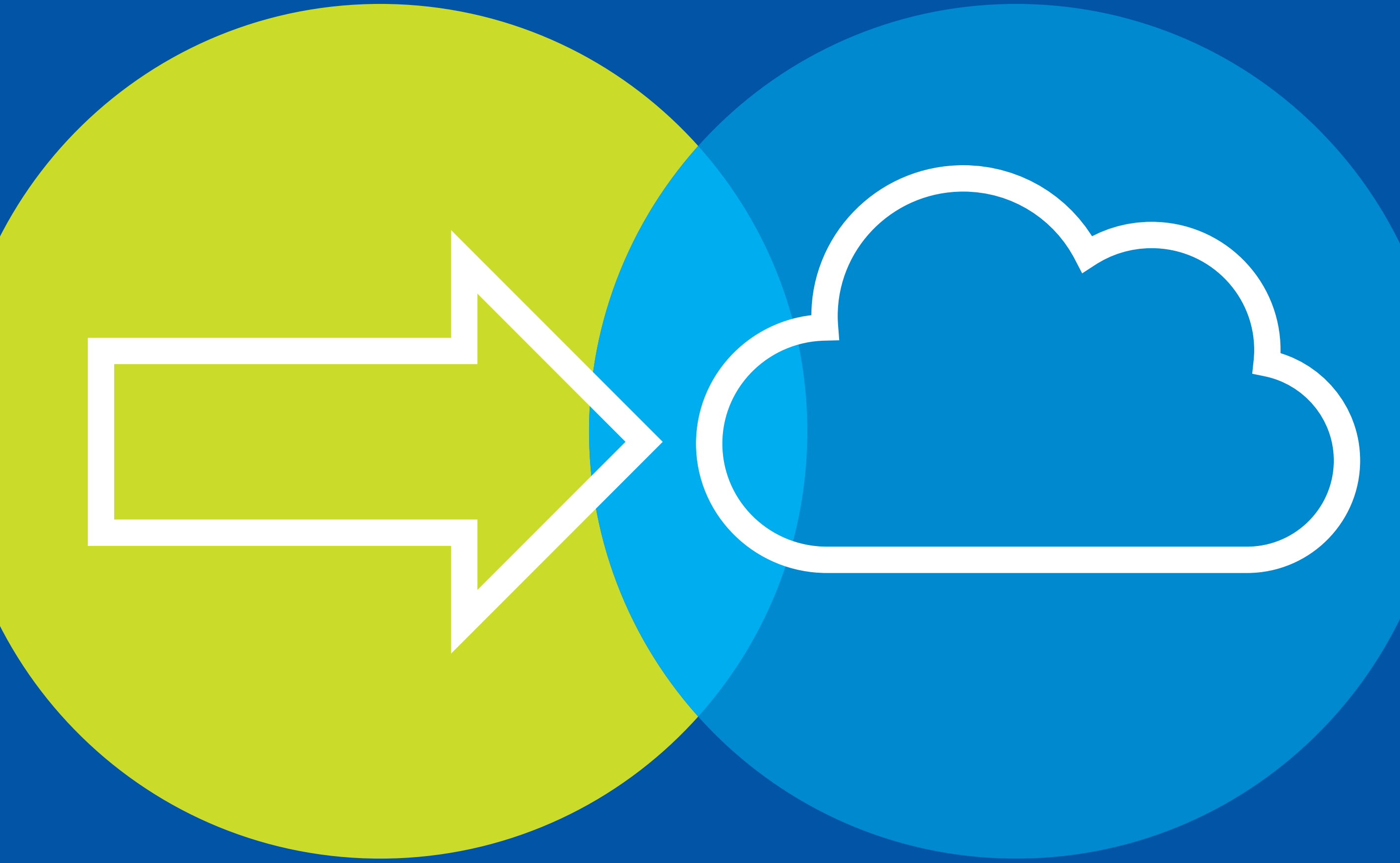
Microsoft

Cloud Architectural Blueprints



Barry Briggs and Eduardo Kassner

PUBLISHED BY
Microsoft Press
A division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Control Number:

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. 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.

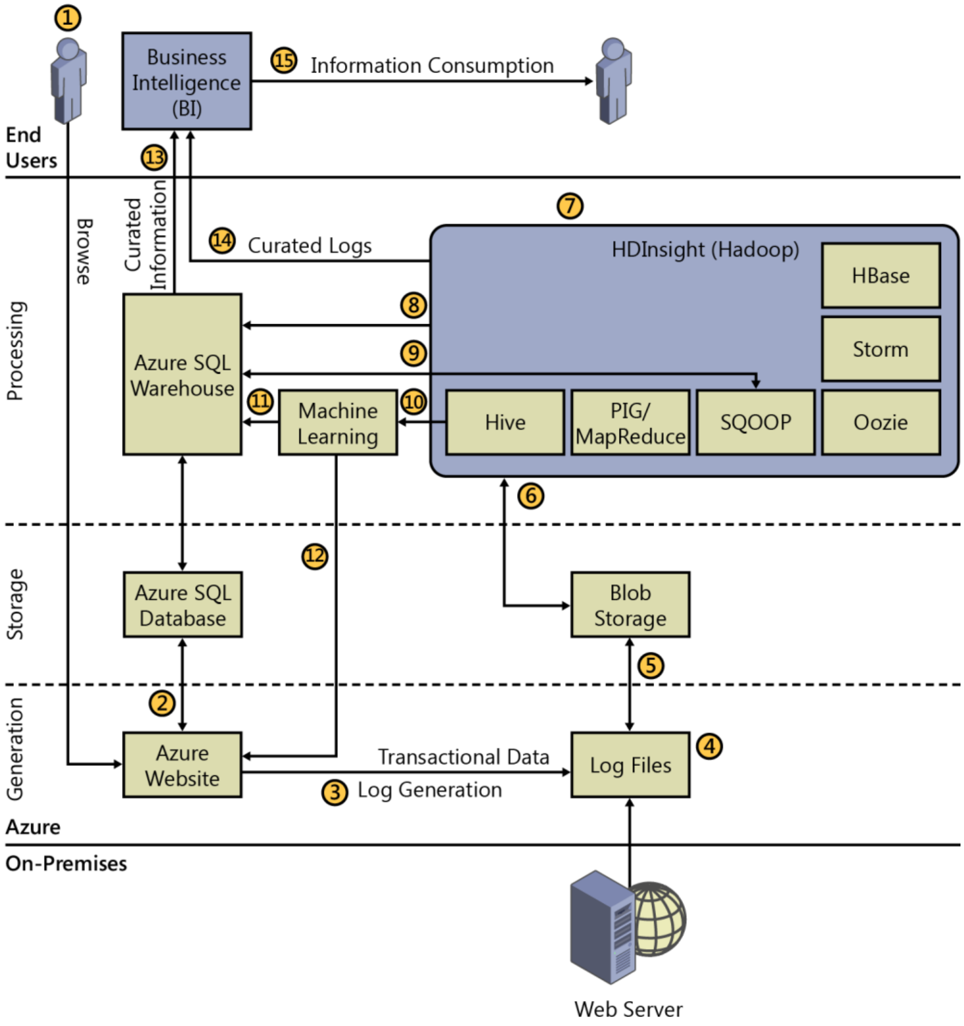
Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

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

Cloud architectural blueprints

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.

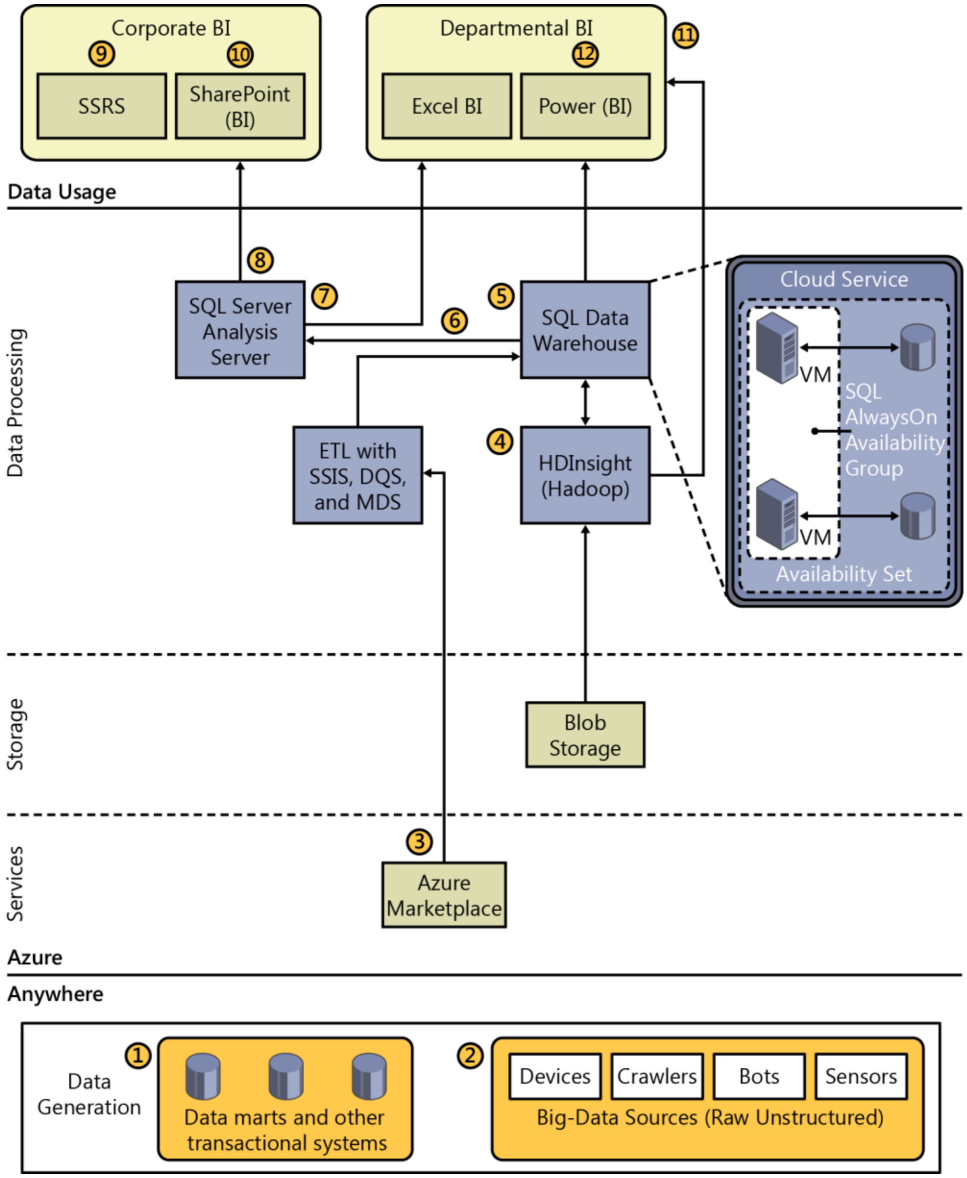
13. The data warehouse sources data into business intelligence (BI) systems such as Power BI, SQL Server Analysis Services, and/or custom apps to create data models.

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.



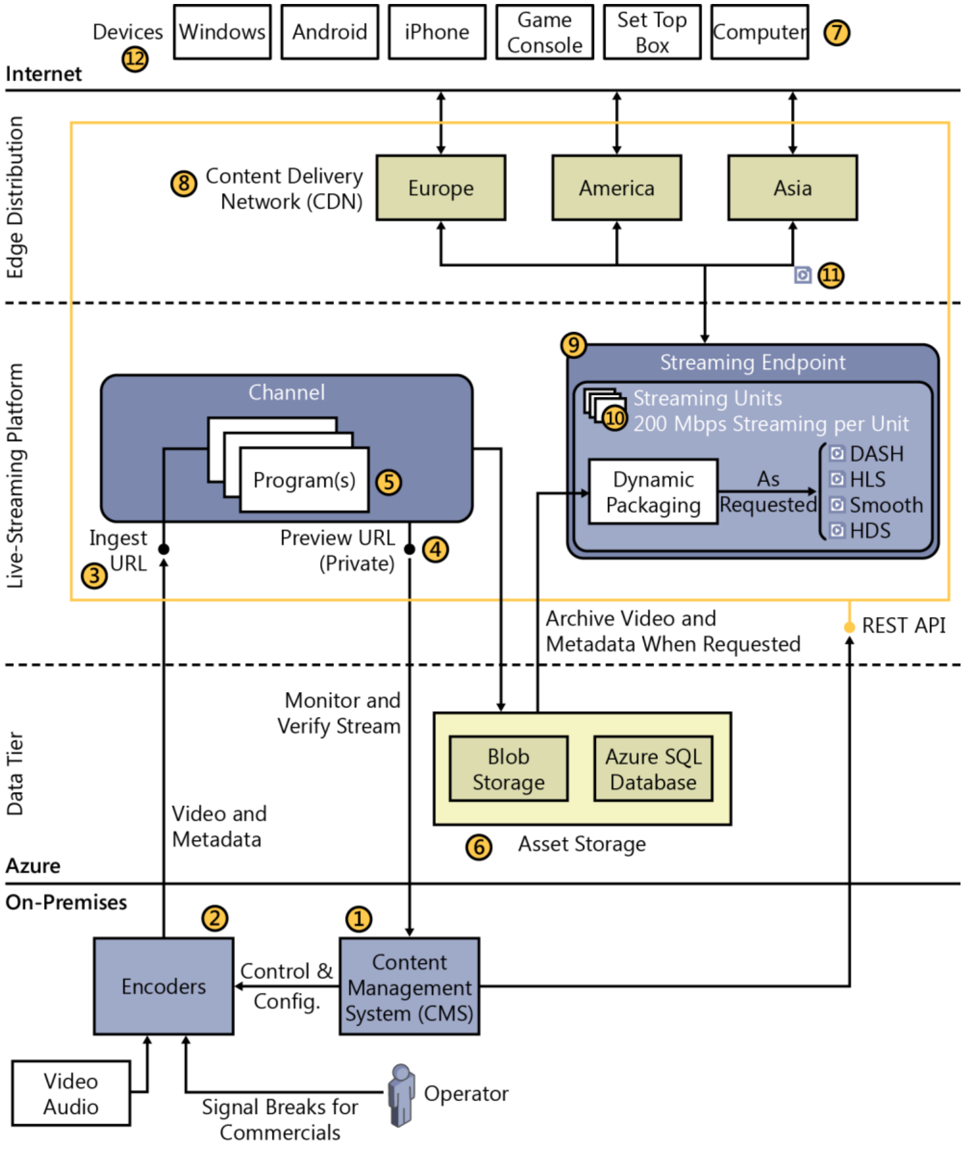
As shown in the illustration:

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.



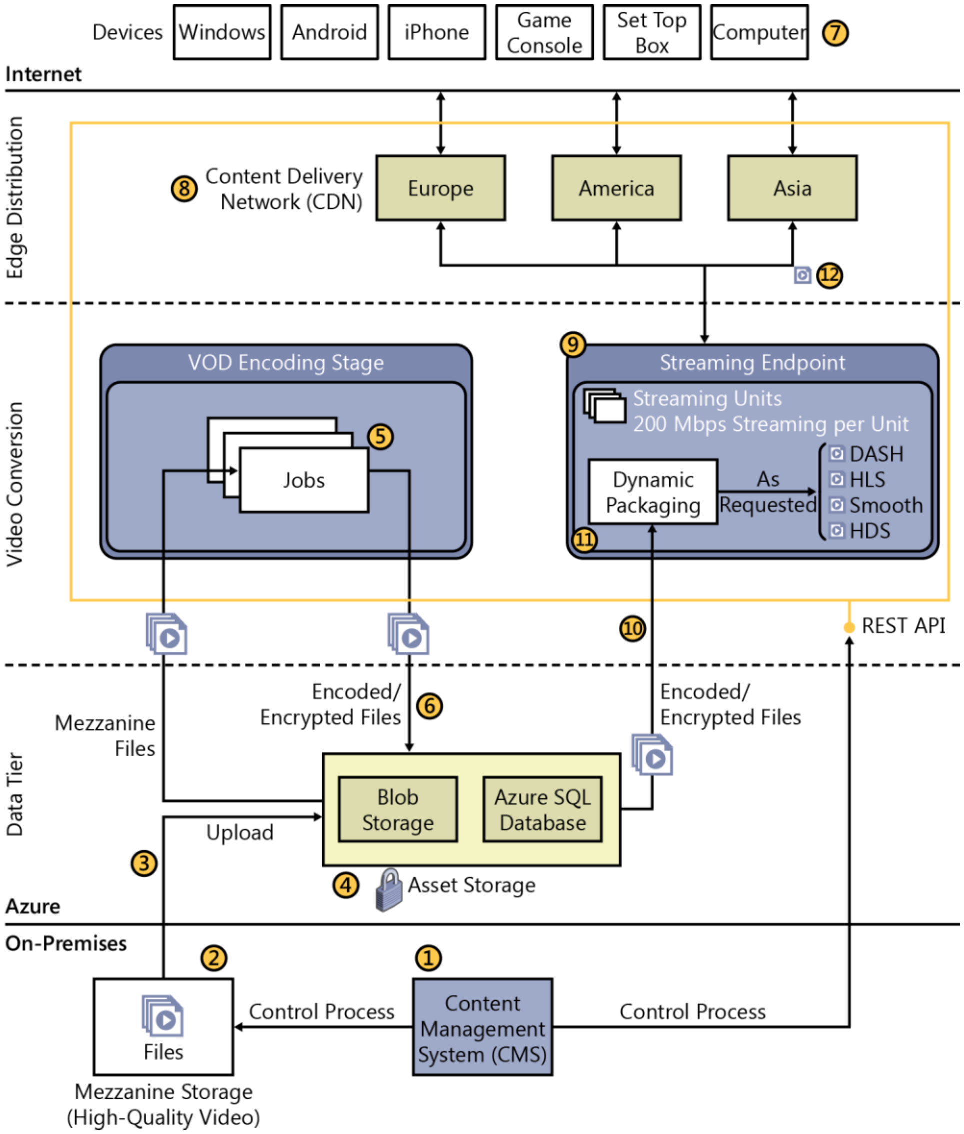
As shown in the illustration:

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.



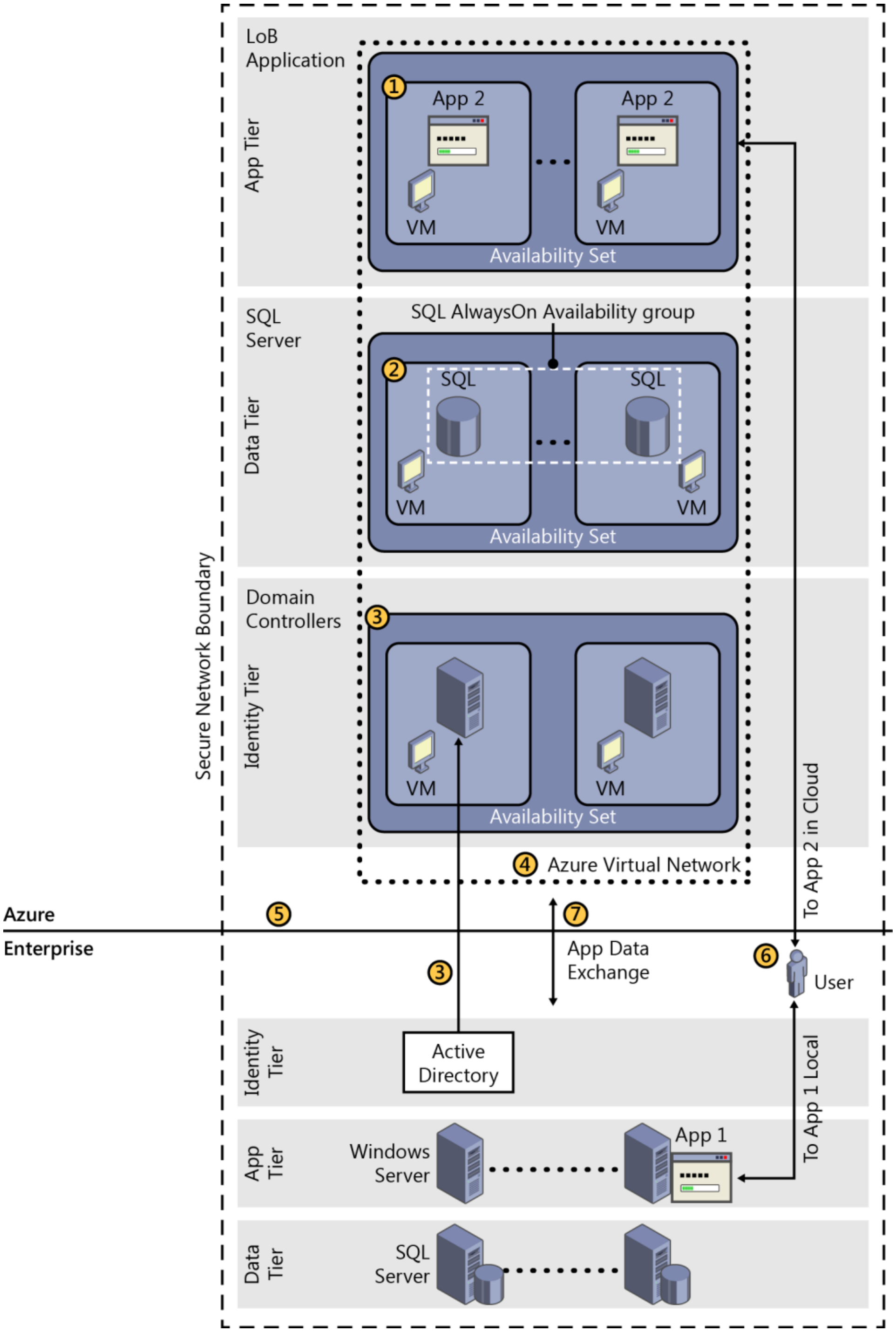
The illustration shows America:

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).



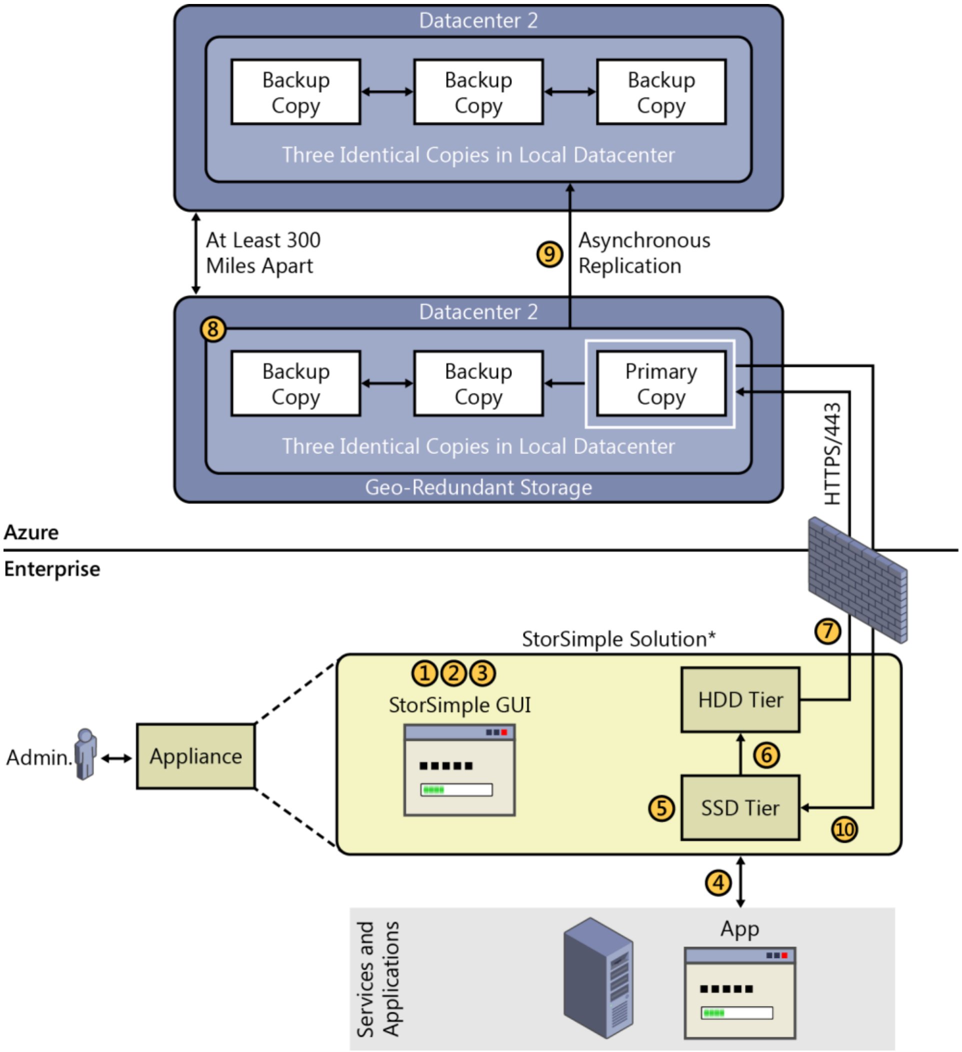
As shown in the illustration:

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.



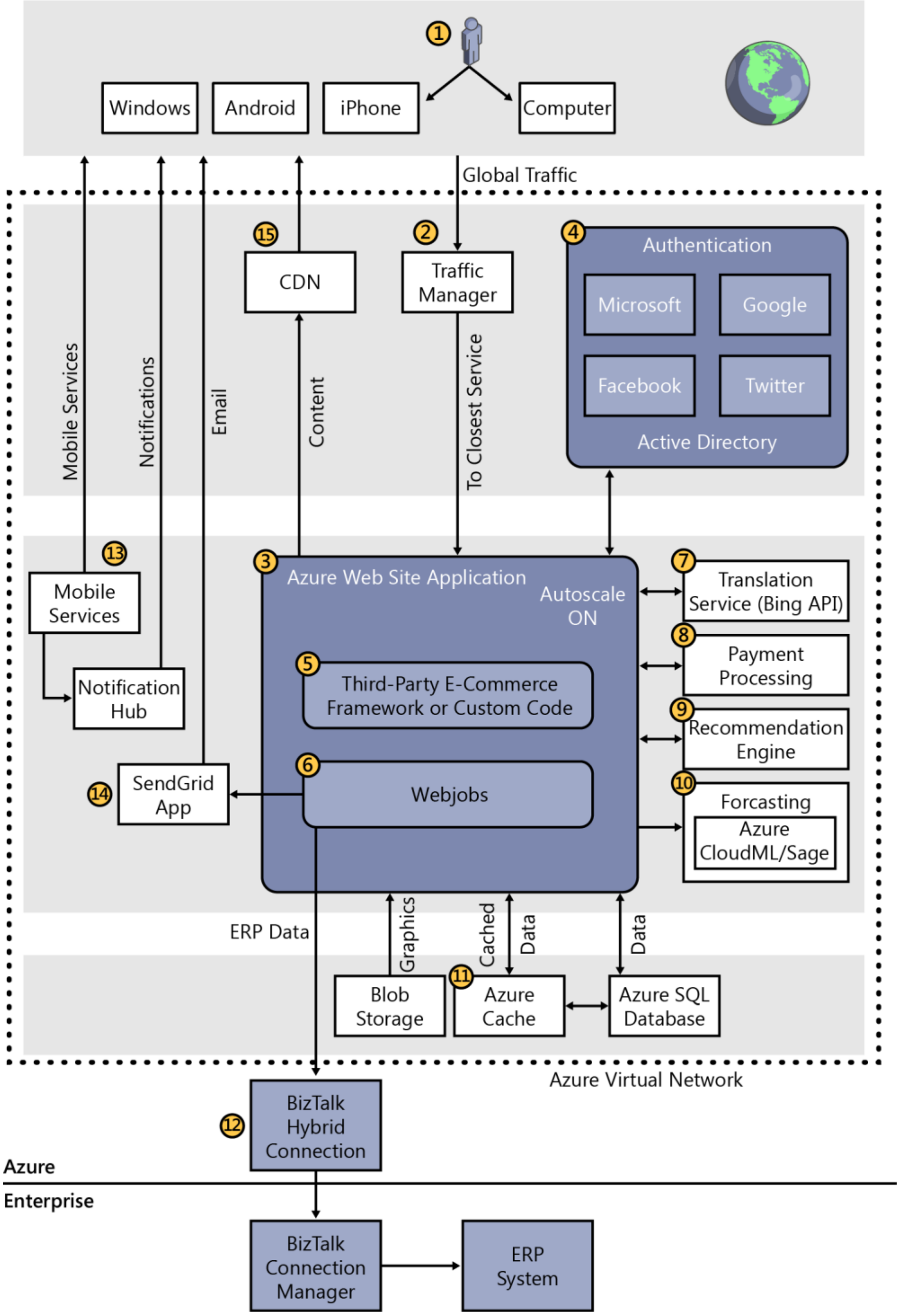
Azure StorSimple is a storage solution that uses both an on-premises appliance and Azure.

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.

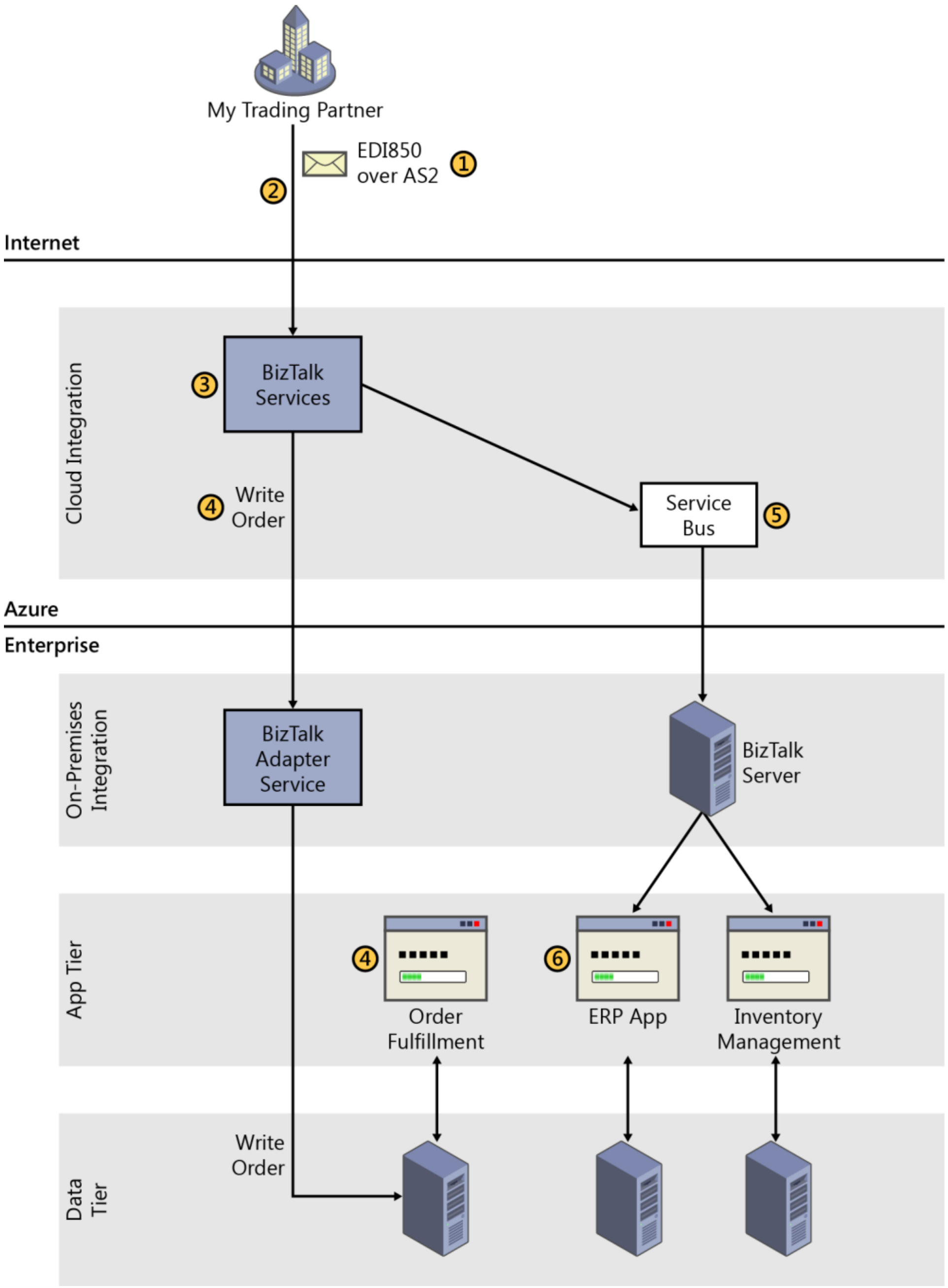
13. Azure Mobile Services provides a unified back end for mobile ordering, including device authentication, data services, and notifications.

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.



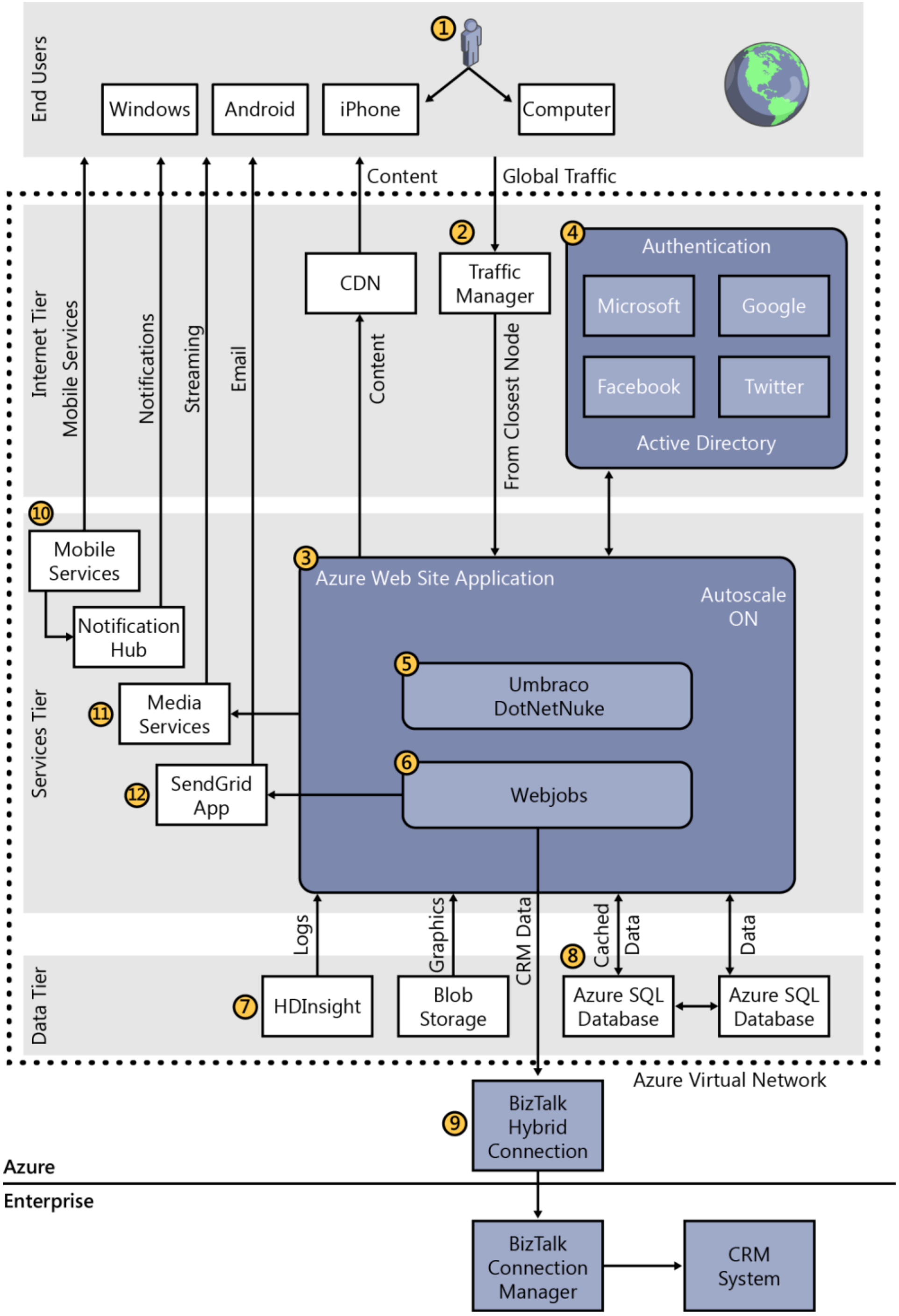
As shown in the illustration:

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.



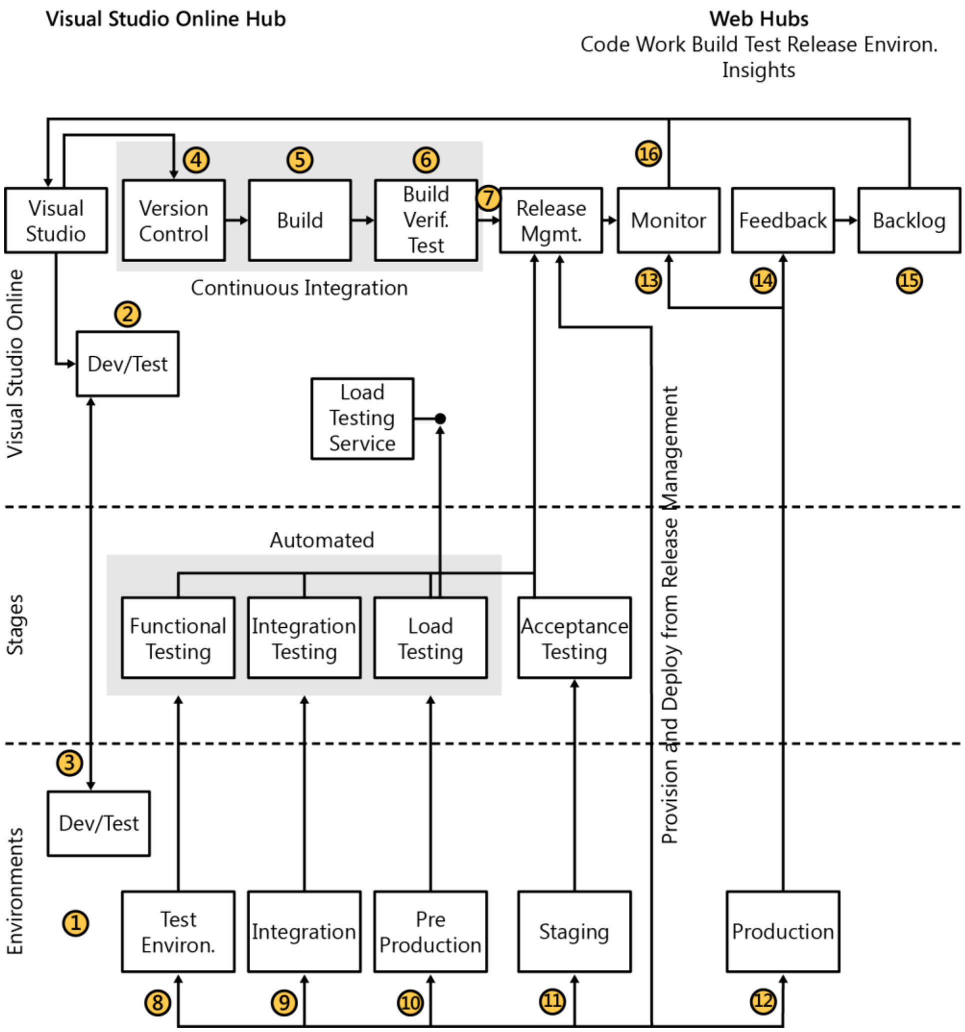
As shown in the illustration:

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.

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

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/sqldbshardingintro

- 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/Aure-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

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 20years 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.