Module 1

Introduction to Microsoft Azure

Module Overview

- Cloud technology overview
- Overview of Azure
- Managing Azure with the Azure portals
- Managing Azure with PowerShell
- Managing Azure with Azure CLI
- Overview of Azure deployment models

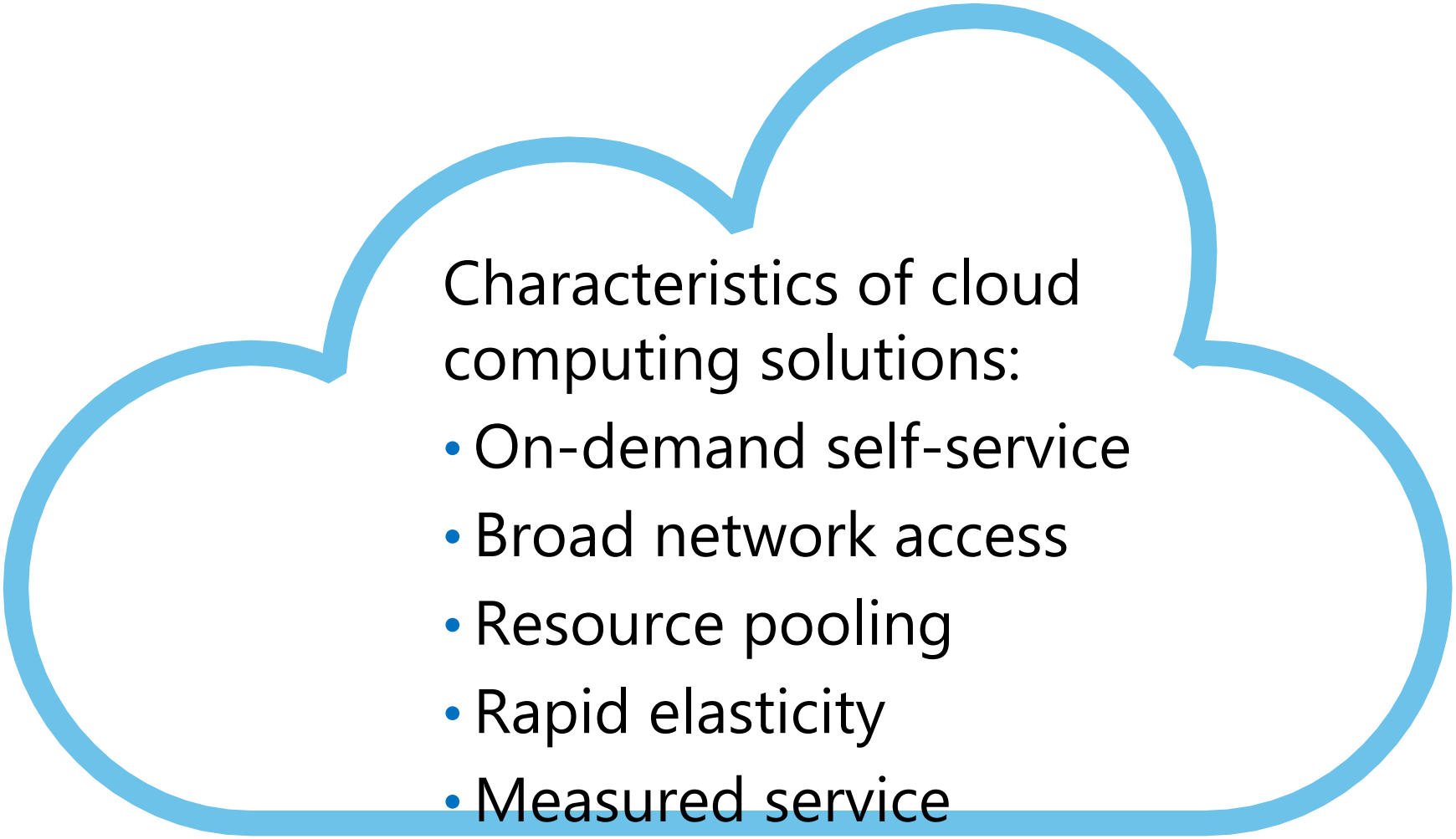
Lesson 1: Cloud technology overview

- Demonstration: Preparing the lab environment
- Introduction to cloud computing
- Types of cloud services

Demonstration: Preparing the lab environment

In this demonstration, you will see how to prepare the lab environment

Introduction to cloud computing



Introduction to cloud computing

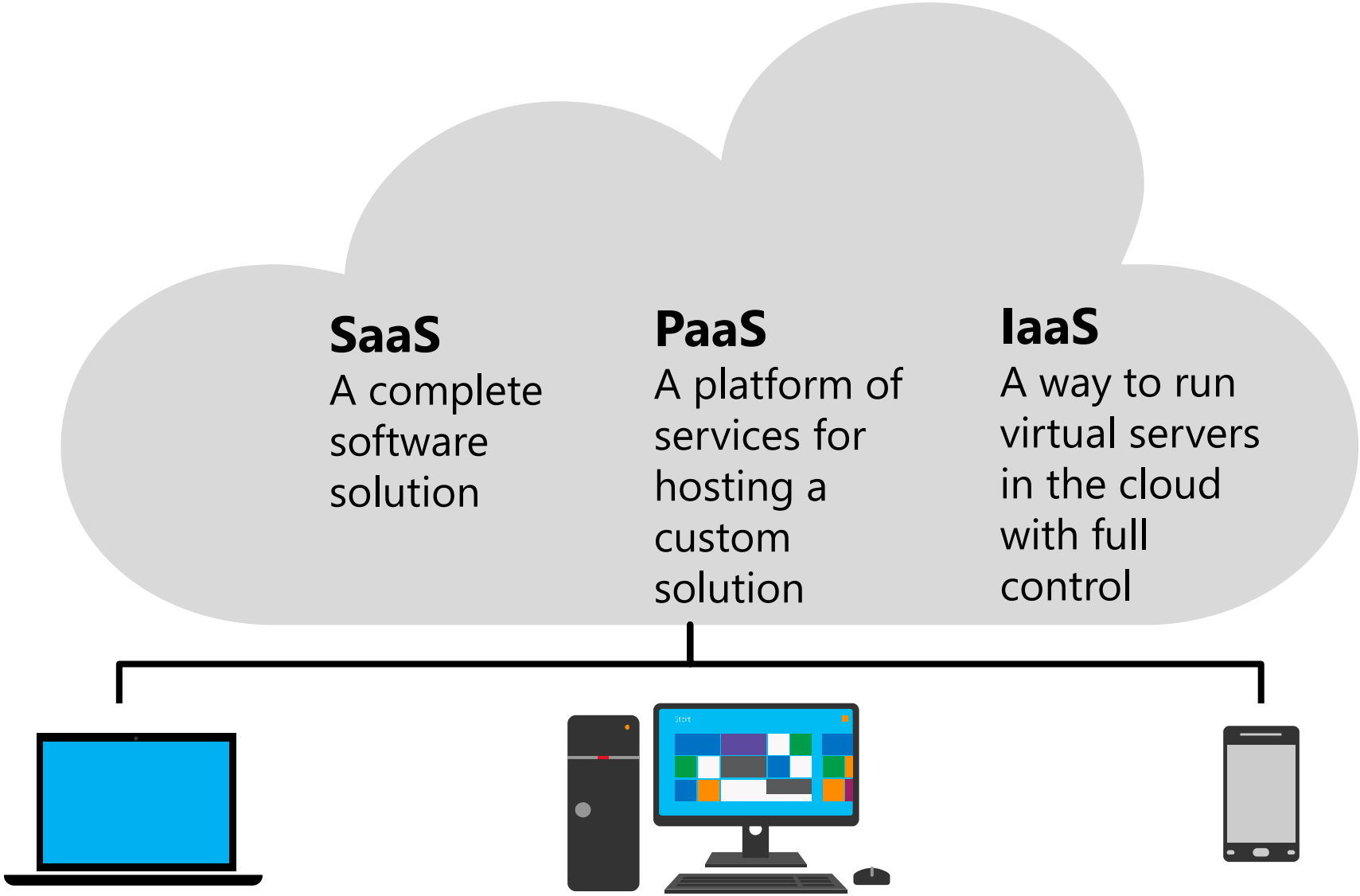
Public, private, and hybrid clouds

A public cloud is an infrastructure, platform, or application service that a cloud service provider delivers for access and consumption by the public

A private cloud is a privately owned and managed cloud that offers benefits similar to those of a public cloud, but is designed and secured for use by a single organization

A hybrid cloud is a technology that binds two separate clouds—public and private—together for the specific purpose of obtaining resources from both

Types of cloud services



Lesson 2: Overview of Azure

- Understanding Azure datacenters
- Understanding the Azure service model
- Locating Azure-related information and resources
- Demonstration: Locating Azure-related resources
- Understanding Azure services
- Understanding Azure compute-hosting options
- Azure deployment models
- Azure management tools

Understanding Azure datacenters

Azure datacenters are located in the following geographic areas

This graphic has been supplied by the Microsoft Azure team. Please check the following link for any updates to the graphic: https://aka.ms/Dpabwy

Understanding Azure datacenters

- Global presence
- Managed by Microsoft
- Modular architecture:
  - Clusters of thousands of servers in pluggable units
  - Full power redundancy and contingency
  - High-speed, redundant intra-datacenter networks
  - High-speed inter-datacenter and Internet connectivity
  - Triple-redundant data storage and geo-replication
- Highly efficient power and water usage
- Distributed management service
- Paired with another Azure region in the same geographical area

Understanding the Azure service model

- Azure is a pay-per-use, multitenant service
- Organized into accounts and subscriptions:
  - Accounts:
    - Billing and reporting functionality
    - Container of subscriptions
    - Managed by the Account Administrator
    - Accessible via the Account portal
  - Subscriptions:
    - Billing and administrative boundary
    - Container of resources (subject to quotas)
    - Managed by the Service Administrator and Co-Administrators, and via delegation through RBAC
    - Accessible via the Azure portal

Azure billing and pricing

- The purchase options are:
  - Pay-As-You-Go
  - Microsoft reseller
  - Enterprise Agreement
  - Azure Hybrid Benefit
  - Azure Reserved VM Instances
  - MSDN, Partner, BizSpark
- The support options are:
  - Developer
  - Standard
  - Professional Direct
  - Premier
- Azure pricing:
  - Per-minute/per-second compute charges
  - Estimate available via Azure pricing calculator

Locating Azure-related information and resources

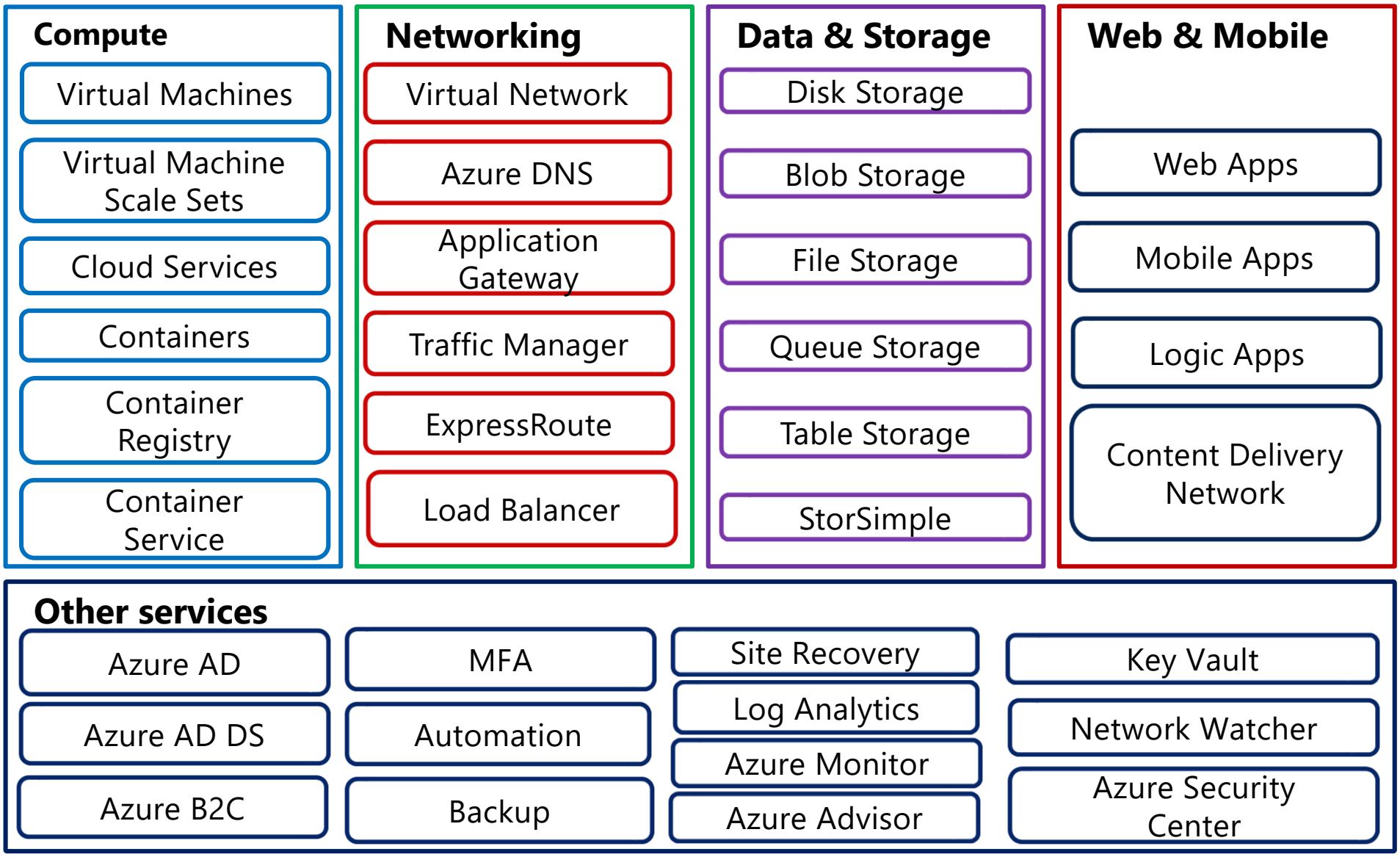
- Microsoft Azure (https://azure.microsoft.com):
  - Documentation of Azure products and services
  - Description of Azure solutions
  - Azure pricing and billing
  - Azure training and certification resources
  - Documentation of Azure Marketplace
  - Azure partner directory
  - Azure support knowledge base
  - Azure-related blogs
  - Azure Trust Center
- GitHub: APIs, SDKs, and open source projects

Demonstration: Locating Azure-related resources

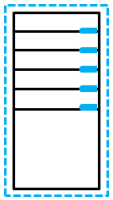
In this demonstration, you will see how to:

- View resources in the Azure Marketplace
- View Azure-related information on GitHub
- View information in Azure Trust Center

Understanding Azure services



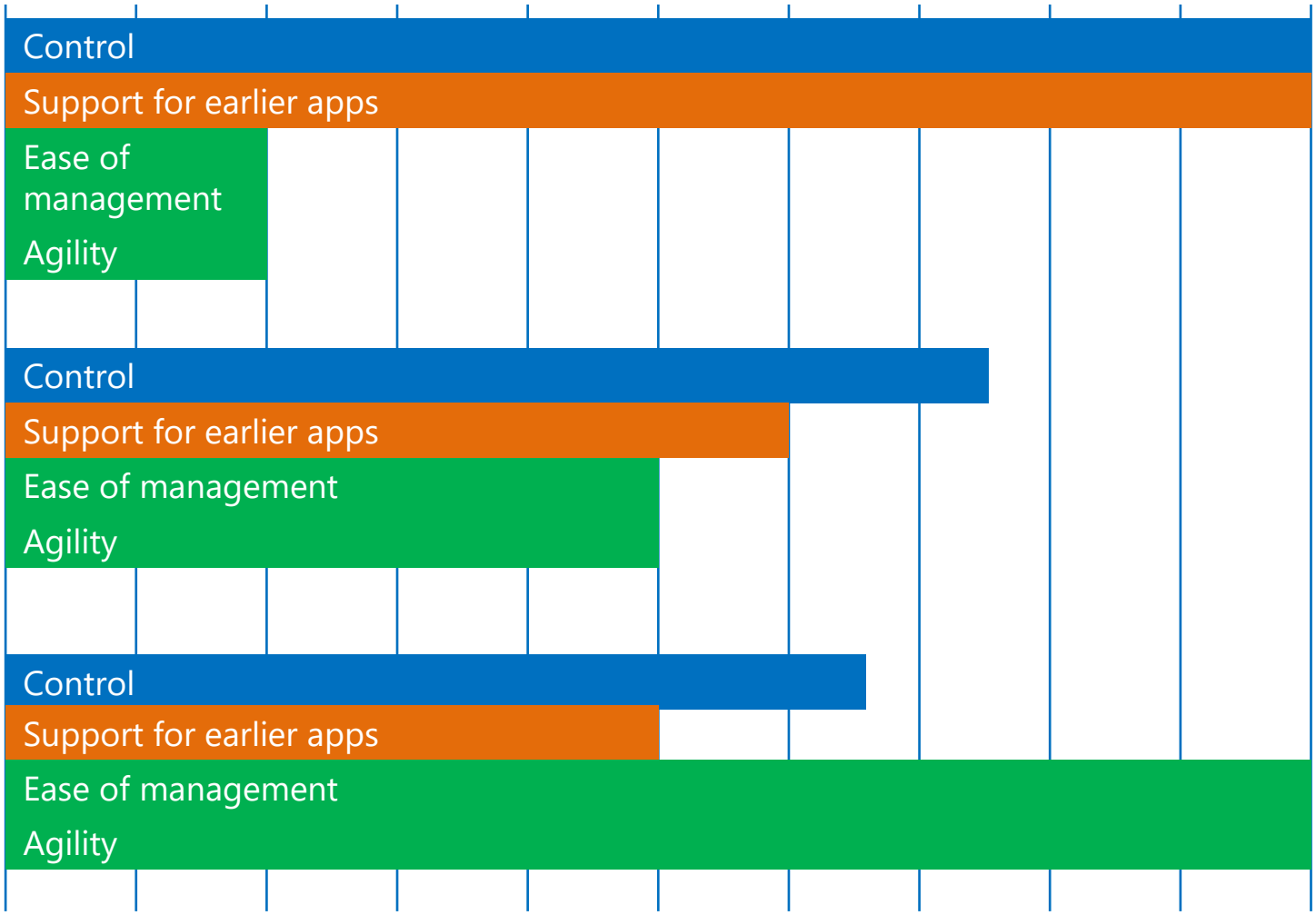
Understanding Azure compute-hosting options



Azure Virtual Machines

Azure Cloud Services

App Service



Azure deployment models

Deployment models determine underlying API for provisioning and managing Azure services:

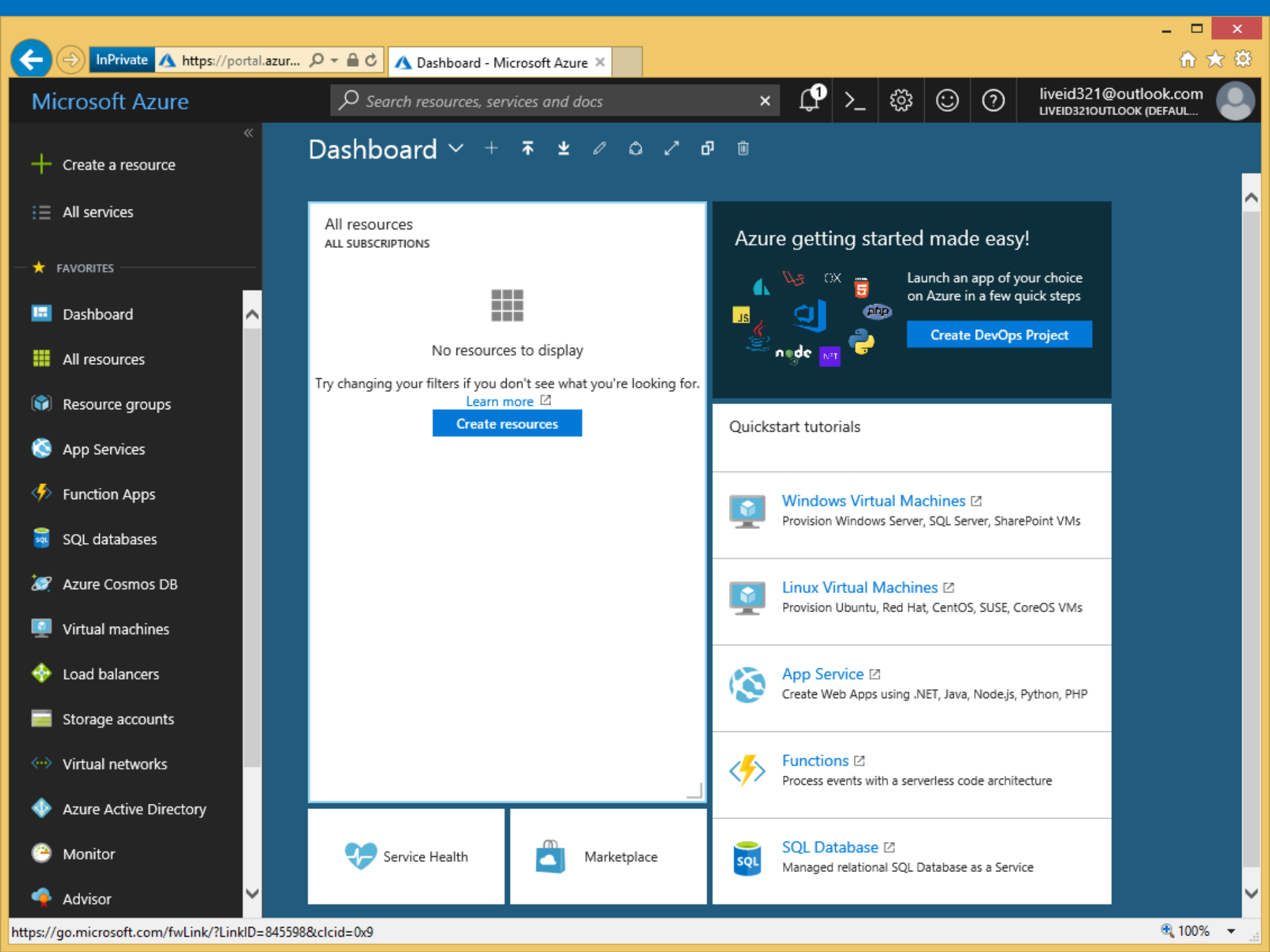
- Classic (Azure Service Management)
- Azure Resource Manager:
  - Based on the concept of resources and resource groups
  - Strongly recommended for any current and future deployments

Azure management tools

- Azure portal
- Azure PowerShell
- Azure CLI
- Cloud Shell
- Visual Studio
- Visual Studio Code

Lesson 3: Managing Azure with the Azure portals

- Using the Azure portal
- Managing account subscriptions with the Azure portals
- Demonstration: Using the Azure portals


InPrivate
https://portal.azur...
Dashboard - Microsoft Azure
Microsoft Azure
Search resources, services and docs
liveid321@outlook.com
LIVEID321OUTLOOK (DEFAUL...
Create a resource
All services
FAVORITES
Dashboard
All resources
Resource groups
App Services
Function Apps
SQL databases
Azure Cosmos DB
Virtual machines
Load balancers
Storage accounts
Virtual networks
Azure Active Directory
Monitor
Advisor
Dashboard
All resources
ALL SUBSCRIPTIONS
No resources to display
Try changing your filters if you don't see what you're looking for.
Learn more
Create resources
Azure getting started made easy!
Launch an app of your choice on Azure in a few quick steps
Create DevOps Project
Quickstart tutorials
Windows Virtual Machines
Provision Windows Server, SQL Server, SharePoint VMs
Linux Virtual Machines
Provision Ubuntu, Red Hat, CentOS, SUSE, CoreOS VMs
App Service
Create Web Apps using .NET, Java, Node.js, Python, PHP
Functions
Process events with a serverless code architecture
SQL Database
Managed relational SQL Database as a Service
Service Health
Marketplace
https://go.microsoft.com/fwLink/?LinkID=845598&clcid=0x9
100%

Managing account subscriptions with the Azure portals

- The **Subscription** and **Billing** blades at https://portal.azure.com
  - View cost by resource and burn rate charts
  - Perform cost analysis
  - View billing information
- The **subscriptions** page at http://account.azure.com/subscriptions
  - Manage payment methods
  - Download usage details
  - Edit subscription details
  - Edit partner information
  - Change subscription address
  - Cancel subscription

Demonstration: Using the Azure portals

In this demonstration, you will see how to:

- Use the Azure portal
- Use the Azure account portal

Lesson 4: Managing Azure with PowerShell

- Azure PowerShell modules
- Authenticating to Azure by using Windows PowerShell
- Azure PowerShell cmdlets for Azure classic deployment model and Azure Resource Manager
- Demonstration: Using Azure PowerShell

Azure PowerShell modules

- PowerShell modules for Azure include:
  - Azure PowerShell (Azure Resource Manager)
  - Azure Service Management PowerShell (classic)
  - Azure Active Directory PowerShell
  - Azure Information Protection PowerShell
  - Azure Service Fabric PowerShell
  - Azure ElasticDB PowerShell
- Available via:
  - Web Platform Installer
  - PowerShell Gallery
  - MSI packages (GitHub)

Authenticating to Azure by using Windows PowerShell

- Azure AD user authentication:
  - A Microsoft account
  - A work or school account
  - An Azure AD service principal (**Add-AzureRmAccount**)
- Certificate-based authentication:
  - Azure Resource Manager:
    - Obtain a certificate (self-signed or CA-issued)
    - Create a service principal associated with the certificate
    - Delegate permissions to the service principal through RBAC
  - Classic:
    - Obtain a certificate (self-signed or CA-issued)
    - Store the private key on the local computer and the public key in Azure
    - The certificate grants full permissions to the subscription

Azure PowerShell cmdlets for Azure classic deployment model and Azure Resource Manager

Functionality	**Classic**	**Azure Resource Manager**
Sign in to Azure	**Add-AzureAccount**	**Add-AzureRmAccount**
Create a virtual machine	**New-AzureVM**	**New-AzureRmVM**
Create a web app	**New-AzureWebsite**	**New-AzureRmWebapp**

Demonstration: Using Azure PowerShell

In this demonstration, you will see how to use Azure PowerShell to:

- Create a resource group
- Create a storage account
- Delete a resource group with its resources

Lesson 5: Managing Azure with Azure CLI

- Azure CLI versions
- Installing Azure CLI
- Using Azure CLI to access your Azure subscription
- Demonstration: Using Azure CLI

Azure CLI versions

- Available in two versions:
  - Azure CLI 1.0:
    - Supports Azure Resource Manager and classic deployment model
    - **Azure** commands
    - Based on Node.js
  - Azure CLI 2.0:
    - Supports Azure Resource Manager deployment model only
    - **Az** commands
    - Based on Python
- Can be installed on Windows, Linux, and OS X
- Integrates with Linux shell scripting tools
- Included in the Azure Cloud Shell

Installing Azure CLI

- Installation process depends on:
  - The azure CLI version
  - The target operating system
- Prerequisites:
  - Azure CLI 1.0: Node.js https://nodejs.org/en/download/
  - Azure CLI 2.0: Python https://www.python.org/downloads/
- Installation:
  - Azure CLI 1.0
    - Node package manager - **npm install –g azure-cli**
    - Docker host - **docker run it microsoft/azure-cli**
    - GitHub-based installers
  - Azure CLI 2.0
    - Ubuntu, Debian, Bash on Windows - apt-get tool
    - Linux and Mac OS – curl http://aka.ms/InstallAzureCli
    - Windows, Linux, Mac OS – GitHub-based installers

Using Azure CLI to access your Azure subscription

- To authenticate:
  1. run:
     a. **az login** (Azure CLI 2.0)
     b. **azure login** (Azure CLI 1.0)
  2. type autogenerated code at http://aka.ms/devicelogin
- To select a target subscription, run:
     1. **az account set** (Azure CLI 2.0)
     2. **azure account set** (Azure CLI 1.0)
- To switch between deployment models (Azure CLI 1.0), run:
  - **azure config mode arm** (default)
  - **azure config mode asm**

Demonstration: Using Azure CLI

In this demonstration, you will see how to:

- Create a resource group
- Create a storage account
- Delete a resource group with its resources

Lesson 6: Overview of Azure deployment models

- Core concepts of Azure Resource Manager deployment model
- Managing resources and resource groups
- Azure Resource Manager deployment methodologies
- Introduction to Azure Resource Manager templates
- Exploring the syntax of Azure Resource Manager templates
- Deploy Azure Resource Manager templates
- Demonstration: Viewing and deploying a GitHub Azure Quickstart template
- Core concepts of the Azure classic deployment model

Core concepts of Azure Resource Manager deployment model

Azure Resource Manager core concepts:

- Resources:
  - Individual building blocks of Azure-based solutions
  - Managed by resource providers
- Resource group:
  - Custom collection of resources
  - Typically represents common lifecycle of its resources
  - Commonly used to delegate permissions to its resources
  - Aggregate billing data and auditing events of its resources
  - Each resource belongs to only one resource group
- RBAC
- Tagging
- Templates
- Policies and locks

Managing resources and resource groups

- Resource groups:
  - Serve as logical groupings of resources
  - Support moving resources
- Considerations when moving resources:
  - Azure region must remain the same
  - Source and target resource groups are locked during move
  - For cross-subscription moves:
    - Subscriptions must be associated with the same Azure AD tenant
    - All dependent resources must be moved together
    - The target subscription must be registered for providers of resources in the scope of the move
  - Not all resources support the move operation

Azure Resource Manager deployment methodologies

- Imperative:
  - Based on Azure PowerShell or Azure CLI scripts
  - You explicitly specify each provisioning step
- Declarative:
  - Based on Azure Resource Manager templates:
    - JSON-formatted
    - Describe the end state
    - Idempotent
    - Support versioning
    - Allow for OS configuration via VM Agent extensions
  - Azure determines the optimal provisioning steps
    - You can affect the provisioning sequence by specifying resource dependencies in the template

Introduction to Azure Resource Manager templates

- When creating and working with resource templates, consider:
  - Which resources you are going to deploy
  - Where your resources will be located
  - Which version of the resource provider API you will use
  - Whether there are dependencies between resources
  - When you will specify values of resource properties
- The template consist of the following sections:

```
{
  "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "",
  "parameters": {  },
  "variables": {  },
  "resources": [  ],
  "outputs": {  }
}
```

Exploring the syntax of Azure Resource Manager templates

- Azure Resource Manager template components:
  - Parameters
  - Variables
  - Resources
  - Outputs

- Azure Resource Manager template function types:
  - Numeric
  - String
  - Array
  - Deployment value
  - Resource

Deploy Azure Resource Manager templates

- Deploy a template by using:
  - Azure PowerShell

```
New-AzureRmResourceGroupDeployment `
    -Name <DeploymentName> `
    -ResourceGroupName <ResourceGroupName `
    -TemplateUri <TemplateUri>
```

  - Azure CLI

```
azure group deployment create
```

  - **Deploy to Azure** link on GitHub
  - **Custom deployment** blade in the Azure portal
  - Visual Studio and Visual Studio Code
- Visualize a template by using:
  - Azure Resource Manage Template Visualizer

Demonstration: Viewing and deploying a GitHub Azure Quickstart template

In this demonstration, you will see how to:

- Visualize an Azure Resource Manager template
- Deploy an Azure Resource Manager template from GitHub

Core concepts of the Azure classic deployment model

- Classic VMs:
  - Always part of a cloud service:
    - A logical container hosting one or more VMs
    - Public IP address and automatically assigned DNS name
    - Accessible via endpoints
    - Built-in load balancer and NAT
  - Can reside on a virtual network (optional)
- Classic resources:
  - Always part of a resource group
  - Support for moving resources depends on the resource type:
    - VMs: yes (as long as moved with its cloud service)
    - VNets: no
  - Support for migration to Azure Resource Manager

Lab: Managing Microsoft Azure

- Exercise 1: Using the Azure portals
- Exercise 2: Using the Azure Resource Manager features in the Azure portal
- Exercise 3: Using Azure PowerShell
- Exercise 4: Using Azure CLI

Logon Information
Virtual machine: **20533E-MIA-CL1**
User name: **Student**
Password: **Pa55w.rd**

Estimated Time: 50 minutes

Lab Scenario

Adatum Corporation wants to expand their cloud presence by taking advantage of the benefits of Azure. Your task is to explore and compare the available IaaS features by using the Azure portal, Windows PowerShell, and Azure CLI.

Lab Review

- Why did you use Azure PowerShell cmdlets that contained Rm in the lab?

Module Review and Takeaways

- Real-world Issues and Scenarios
- Tools