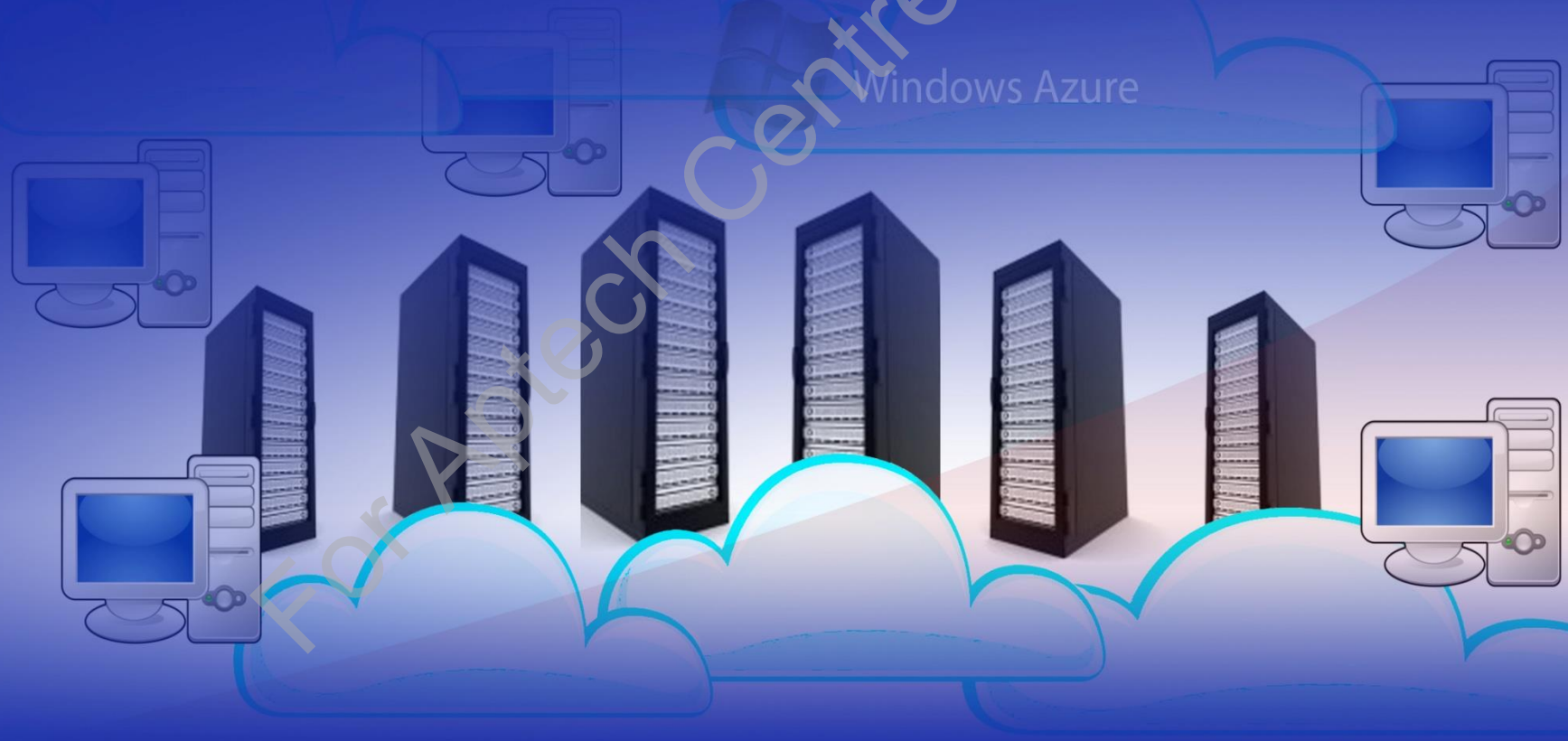# Enterprise Application Development Using Windows Azure and Web Services

Session 1

Introduction to Windows Azure

Learning Objectives

- Define and describe cloud computing

- Describe cloud computing delivery models

- Define and describe the Windows Azure platform

- Differentiate between Windows Azure platform and Windows Azure Operating System (OS)

- Define and describe Web Services

- Describe the Windows Azure Portal

- Explain the process to create, deploy, and test a cloud service using the portal

Overview of Cloud Computing 1-2

Cloud computing is an approach that enables convenient and on-demand access through the Internet to computing capabilities and resources.

Overview of Cloud Computing 2-2

Data resides on servers hosted across different datacenters spread over geographical locations.

Consumers of applications and services use the data over the Internet from a service provider.

Datacenter: A dedicated storage space to store computers and various related components.

Definition of Cloud Computing 1-2

❑ **Cloud computing approach**:

- – Helps small, medium, and big enterprises.
- – Helps them to build, deploy, and manage any kind of services from creating a small and simple Website to working with large server workloads.

❑ **The key advantages of cloud computing**:

- – In the IT department, the application performance does not affect as there are no hardware limitations.

Definition of Cloud Computing 2-2

❑ An example of a commonly used cloud computing application:

 Twitter

❑ The application is run from the servers hosted on the cloud and the data is also maintained on the cloud servers.

❑ Other such applications are:

| Facebook | Gmail | Flickr | DropBox |

Why Use Cloud? 1-3

❑ **The word 'cloud'**:

> Is used as a representation of the Internet, because a pictorial representation of the Internet has always been the cloud.

> Refers to cloud computing in the programming world, which is derived from the concept of utility computing.

❑ **Utility Computing**:

- Is a concept that allows you to use-and-pay computing resources.
- Is a concept of using and paying resources on a metered basis that gave birth to the concept of cloud computing over the last few decades.

Why Use Cloud? 2-3

❑ Three core reasons that have driven the organizations to use cloud computing are:

Economical

Scalability

Deployment

Why Use Cloud? 3-3

Economical

– Cloud computing is more economical and cheaper than hosting applications on local infrastructure in an organization.

– It does not require purchase of any hardware.

– It is managed by the service providers, such as Amazon and Microsoft.

Scalability

– Cloud computing offers unlimited scalability.

– The IT department of an organization can scale hardware such as storage within a few minutes, if needed.

– The hardware can be scaled without impacting the application or causing the down time.

Deployment

– Cloud computing offers quick deployment.

– It allows the developers to use similar hardware resources so that deployment does not fail.

Cloud Services Models 1-5

❑ Cloud computing platforms provide different kinds of services, depending on the delivery model that they use to deploy.

❑ The commonly available cloud service models are:

Infrastructure-as-a-Service (IaaS)

Platform-as-a-Service (PaaS)

Software-as-a-Service (SaaS)

Cloud Services Models 2-5

Infrastructure-as-a-Service (IaaS)

– In this cloud service model, the lower level services such as hardware and infrastructure are provided.

– Examples:
  o Network attached storage
  o Virtual machines
  o Load-balancer settings

– Amazon is a popular IaaS provider with services such as Elastic Cloud 2 (EC2) and S3.

Cloud Services Models 3-5

Platform-as-a-Service (PaaS)

– In this cloud service model, the platform is provided over the Internet wherein developers can write their required code.

– The application runs on a specialized environment.

– Examples:

| Windows Azure	Google App Engine (GAE)
• It is a cloud offering by Microsoft and is used for building, deploying, and managing services and applications.	• In this engine, developers can write code in a subset of Python. • This code is then executed inside a custom hosting environment. • In this case, Google is providing the platform for writing the code along with storage and servers.

Cloud Services Models 4-5

Platform-as-a-Service (PaaS)

– The following figure describes the working of GAE:
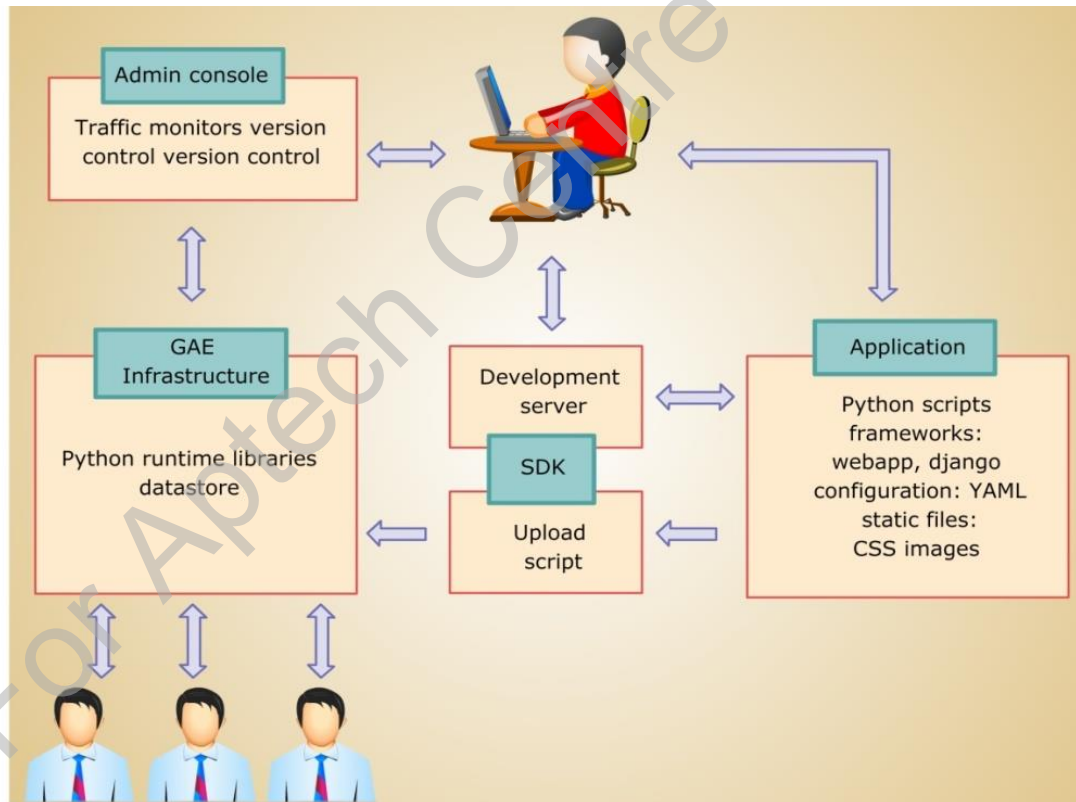
Cloud Services Models 5-5

Software-as-a-Service (SaaS)

- In this type of model, the software subscription is leased to the consumer.

- For example:

  - Microsoft provides Office 365 as a SaaS model.
  - A popular SaaS service is that of Gmail where an e-mail client is offered as an Internet service without requiring any local installation.
  - Anyone, anywhere, can use Gmail, provided he/she has an account.

Characteristics of Cloud Computing 1-6

❑ Most modern cloud computing platforms including Windows Azure have the following distinct characteristics:

Access to infinite resources

Scale on demand

Pay-for-play

High availability and agreements

Geographically distributed datacenters

Characteristics of Cloud Computing 2-6

Access to infinite resources

Scale on demand

Pay-for-play

High availability and agreements

Geographically distributed datacenters

- The cloud computing platforms provide an illusion of providing infinite capacity of computing and storage of resources.

- Need not plan much about storage or usage of your computing or infrastructural resources.

- Once a company or organization deploys its own storage on a cloud computing platform, it can leverage large datacenters that can span across the globe.
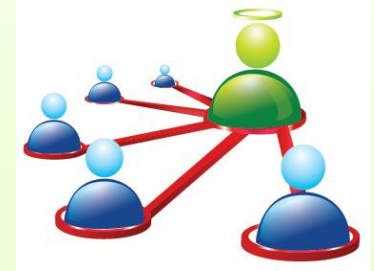
Characteristics of Cloud Computing 3-6

Access to infinite resources

Scale on demand

Pay-for-play

High availability and agreements

Geographically distributed datacenters

- Cloud computing allow you to add resources only when you need them.

- It helps you to economize on your cost and time to acquire resources.

Characteristics of Cloud Computing 4-6

Access to infinite resources

Scale on demand

Pay-for-play

High availability and agreements

Geographically distributed datacenters

- All initial investment, setup fees, and reservation are minimized and you incur only the software and hardware fees.

- Cloud computing allows you to reduce the upfront Capital Expenditure (CapEx) costs and incur only the Operating Expenditure (OpEx) cost.

- Your expenditure is only for what you use.

Characteristics of Cloud Computing 5-6

Access to infinite resources

Scale on demand

Pay-for-play

High availability and agreements

Geographically distributed datacenters

- Cloud computing platform providers:

  o Will create a Service Level Agreement (SLA) for storage and other computing outages.

  o Will guarantee a set level of uptime and if they cannot meet the SLA, they will then provide you a refund.

  For example, the SLA that Windows Azure platform (which is a cloud platform from Microsoft) takes care of both its hosting and its storage.

Characteristics of Cloud Computing 6-6

Access to infinite resources

Scale on demand

Pay-for-play

High availability and agreements

Geographically distributed datacenters

- Cloud computing spreads your data in datacenters globally located in different geographies.

- You can make maximum use of:
  - Load balancing
  - Network latency
  - Edge caching

Introduction to Windows Azure

❑ **Windows Azure**:

– Was first unveiled at the Professional Developers Conference (PDC) on October 27 2008.

– Became commercially available only from 2010.

– Designed as a platform along with the following components in its initial launch:

SQL Azure (later renamed to SQL Database)

Cloud OS

Microsoft .NET Services
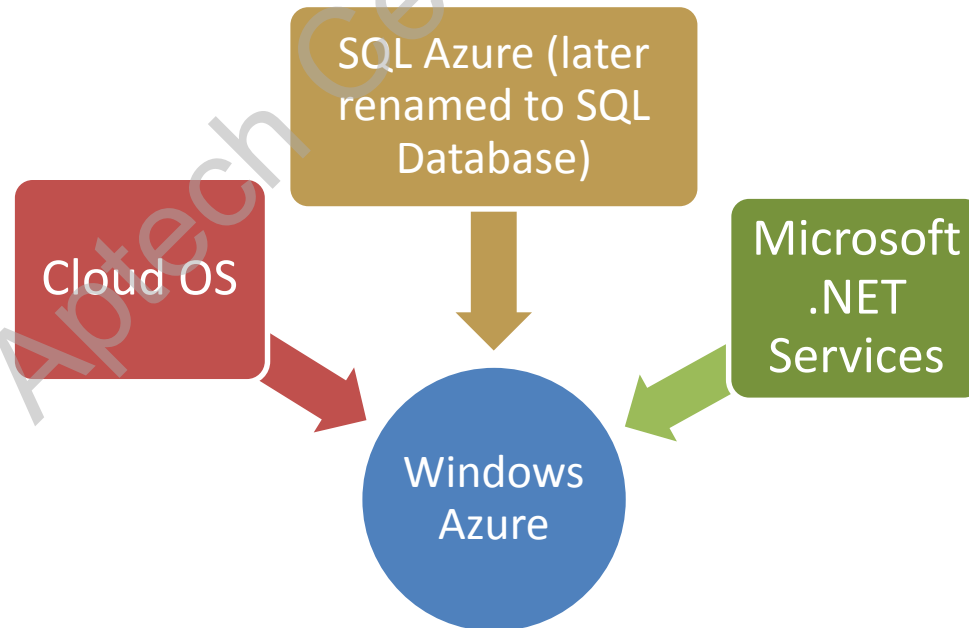
Windows Azure

Windows Azure-Definition

❑ **Windows Azure**:

> Is a cloud computing platform created by Microsoft and is used for building, deploying, and managing services and applications.

> Uses a network of datacenters connected globally to accomplish such tasks.

> Provides PaaS as well as IaaS services, supports various frameworks, programming languages, and tools, which are Microsoft specific as well as third party software.

Need for Windows Azure 1-2

❑ The need for Windows Azure arose because:

There were hardly any PaaS products that could be beneficial for companies.	It helps simplify IT based management and reduces upfront and regular expenses.	It is used to prepare, allocate, and upgrade Web applications instead of using expensive on site resources.

Need for Windows Azure 2-2

❑ Windows Azure OS works as an integral part of the Azure Services Platform.

❑ Windows Azure OS, as the Azure Services Platform:

   – Covers different and separate application, storage, desktop environment, security, and so on.

   – Supports Microsoft standards, programming languages, platforms, and protocols.
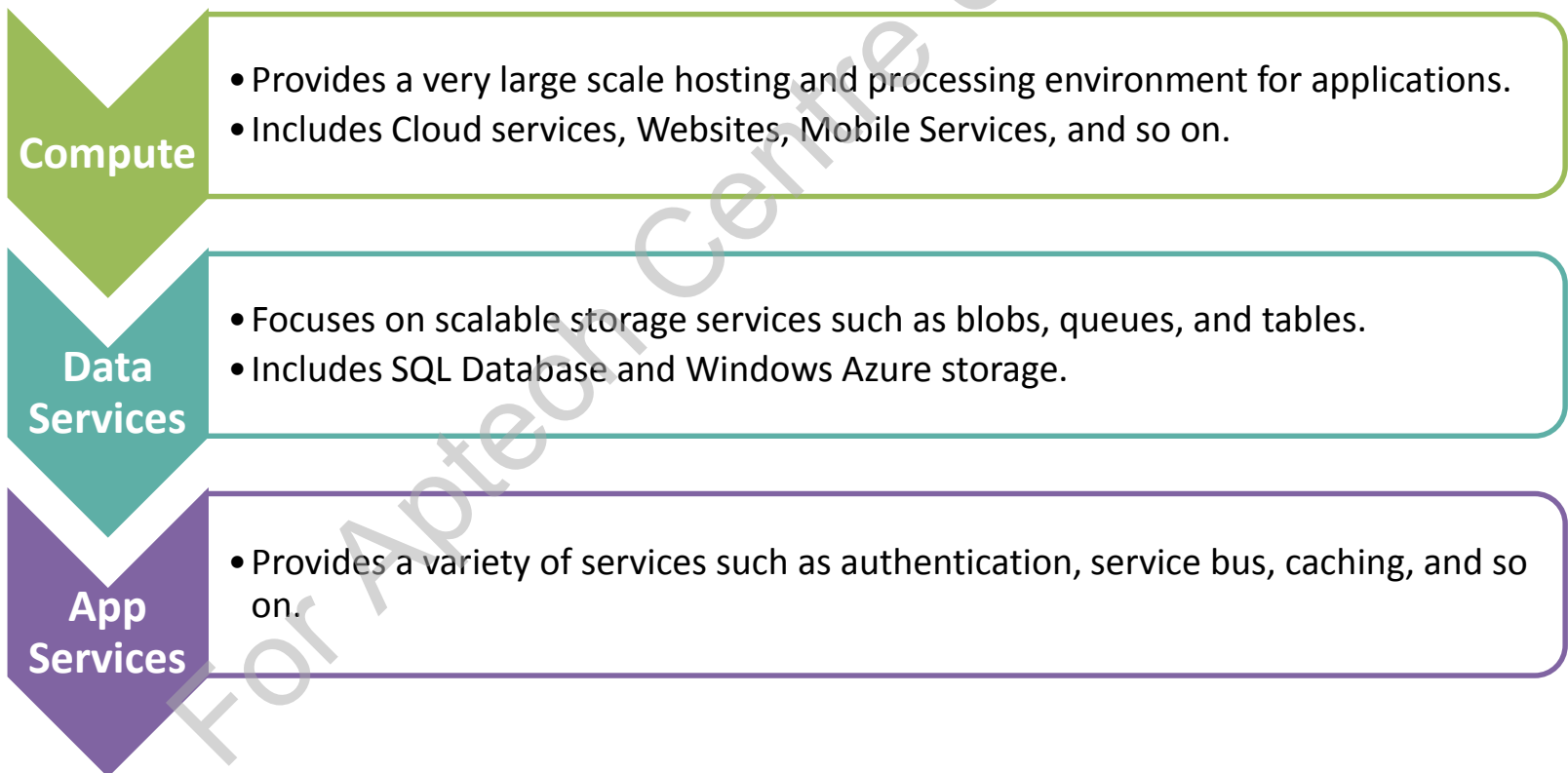
Azure Platform and Cloud OS 1-2

❑ Windows Azure was conceptualized as part of the cloud OS.

❑ The Windows Azure cloud computing platform:

Is built as an integrated platform that can cater to the building, deployment, and management of applications.

Is an open and flexible cloud platform and can also cater to workloads hosted on a global network of Microsoft managed datacenters.

Azure Platform and Cloud OS 2-2

❑ Following are the most important components of the Windows Azure platform architecture:

Compute
- Provides a very large scale hosting and processing environment for applications.
- Includes Cloud services, Websites, Mobile Services, and so on.

Data Services
- Focuses on scalable storage services such as blobs, queues, and tables.
- Includes SQL Database and Windows Azure storage.

App Services
- Provides a variety of services such as authentication, service bus, caching, and so on.

Different Features of Azure 1-2

❑ The table lists some of the features of Windows Azure:

Feature	Description
Service Hosting	• Allows you to create your own server-side applications, such as Websites, computation services, and host them by using Windows Azure. • In its current version, a code that requires administrative privileges on the machine will not be supported by Azure.
Service Management	• Allows you to use the Windows Azure in-built fabric controller that helps you deal with application monitoring and management. • Monitors and maintains software automatically and hardware related upgradations or failures.

Different Features of Azure 2-2

Feature	Description

Storage	• Allows you to store data through Azure's three services:
	    ◦ Semi-structured tables
	    ◦ Binary Large Object (BLOB) storage (for storing raw data)
	    ◦ A queue service
	
	• Backup is prepared in case of failures, but you pay only for the storage that you use.

Windows Server	• Allows you to use the same code in Windows Azure that has been used in Windows Server. • No separate code or framework is needed. • The regular .NET Framework as well as ASP.NET code can be used. • You can also use any other framework that supports FastCGI.

Development Tools	• Allows you to use many in-built development tools, such as:
	    ◦ APIs for logging and reporting errors
	    ◦ Tools for deploying applications to cloud simulator
	    ◦ Other tools to read and update service configuration files

Advantages of Windows Azure 1-2

❑ Following are the advantages of Windows Azure:

Pricing	Scalability
• Offers the pay-per use pricing. • The consumer is charged depending upon the usage of the cloud infrastructure. • Each component in cloud environment, such as storage and bandwidth, is charged separately and billing is done on the actual usage.	• Offers agility in the IT infrastructure. • It helps organizations to gain application performance. • The scalability can be achieved within a few minutes by scaling the IT hardware in the cloud environment. • The businesses pay only for what they use.

Advantages of Windows Azure 2-2

Availability

- Provides assured availability with 99.95% up time.

Manageability

- Uses Fabric Controller that maintains the instances on which an application is running.
- Performs a number of tasks, such as updating the operating system with the updates and patches and recovering the instance in case of a crash.

Integration

- Allows the developers to integrate data and users from their local infrastructure to the Windows Azure applications using the Connect feature.

Overview of Web Services

❑ **A Web service is**:

> A software component that can be accessed by other applications (such as a client, a server, or another Web service) through protocols including HTTP.

> An object-oriented class that runs on the Web server and allowing the remote clients to invoke its methods.

❑ Web services facilitate most flexible infrastructure for the development of distributed cloud applications.

Distributed Applications 1-3

❑ **Distributed Application**:

- Helps users access information, but users are unaware where the information is coming from.

- Allows adding more systems as and when required. A single and consistent environment is provided to the users.

- Is typically used in a client/server network where the client interacts with the servers with specific requests.

- Key example of distributed application is a collaboration application.

Distributed Applications 2-3

❑ Following are the types of distributed applications:

General	• Are general applications that are used on systems. • Consider an application that enables a user to login at the local machine and get authenticated through the remote domain controller.
Collaboration	• Are applications that allow users to work on specific portions of an application at the same time. • Each user is assigned a specific task.
Real-time	• Are the applications in which real-time information exchange takes place. • Example: Chat application in which a user is interacting with another user.
Computational	• Are applications in which processing of code takes place on the server. • Example: A central code repository, which multiple users access at the same time.

Distributed Applications 3-3

❑ Following technologies were used to create distributed applications prior to Web services:

Remote Method Invocation (RMI)	Remote Procedure Calls (RPC)	Distributed COM (DCOM)

❑ These technologies were found to be difficult to use or both client and server were required to be using the same technology.

❑ As a solution to these issues, the concept of Web services was conceived.

Concept of Service Oriented Architecture (SOA) 1-2

❑ **SOA** is an architectural design that:

Helps to build or extend modular systems in a flexible and reusable manner.	Provides loose coupling among the modules of an application.

❑ SOA defines three key roles:

Service Provider (Server)	Service Consumer (Client)	Service Broker (Middleware)

Concept of Service Oriented Architecture (SOA) 2-2

❑ Entities interact with each other with the help of three key components:

Service	Message	Dynamic Discovery
• A self-contained piece of functionality. • It can be remotely located and invoked on a request. • It can be consumed by a remote client from any geographical location, using any operating system and language. • A service provider such as a Web server makes the service available to consumers.	• Is a means of communication between service providers and service consumers. • Can be in the form of a request from a consumer or as a response from the service provider. • Are defined in Extensible Markup Language (XML).	• Is implemented by the directory service. • Helps providers to register themselves along with their services. • Helps to find the valid provider for invoking the required service.

Introduction to Web Services 1-3

❑ Following are the features of Web services:

They can be defined as application components

They can use open protocols for communication

They can be reused by multiple applications

They use HTTP and XML as the base

Introduction to Web Services 2-3

❑ Following are the main objectives behind the development of Web services:

| The Web service and the client that will consume the service can use completely different operating systems or programming languages. | The client and Web service are allowed to work remotely from each other. | The Web service is made available through firewalls (allow port 80 and block other ports). |

❑ These objectives can be accomplished through the use of XML by client and Web services to call services and get back the results.

Introduction to Web Services 3-3

❑ Following are the characteristics of a Web service:

- A Web service contains only classes that are used by the clients, but does not contain a user interface.

- It uses only port 80 to make the Web service available to client.

- It can be called using HTTP Get, HTTP Post, or using Simple Object Access Protocol (SOAP).

- The clients of a Web service can be a Web browser, Web application, or desktop application.

- All the clients using Web service require a proxy, which gathers the parameters and results to invoke a Web service.

Need for Web Services

❑ Following are the key purposes of Web services:

Function reusability	• A developer can expose the functions of an application over the network. • The exposed functions can then be reused by different applications.
Interoperability of different types of applications	• A developer can integrate services and data of applications that are developed using different programming languages. • A developer does not need to make changes to the existing applications.
Use of standard protocols	• Web services use standard protocols that allow wide adoption across different applications.
Less expensive	• Since Web services use standard HTTP protocols, there is no extra cost involved in designing the applications.

Applications of Web Services 1-2

❑ Web services:
- – Have been utilized for a large number of applications that people use on day-to-day basis.
- – In the backend often perform the functions, but end-users are unfamiliar about it.

❑ People use mobile phones, tablets, and computers to interact with online applications.

❑ For example:
- – A user at home orders a pizza on an online portal from a pizza joint.
- – The user is unaware that the software of the portal makes use of Web services in ordering the pizza.

Applications of Web Services 2-2

❑ Some common Web services on a mobile phone, a tablet, or a computer are:

Receiving online score updates from a sports site

Making travel reservations

Getting online weather updates

Receiving a road travel map

Web Services in .NET 1-3

❑ Following are the various supported Web services in .NET:

ASP.NET Web Services

Windows Communication Foundation (WCF)

ASP.NET Web API Services

Web Services in .NET 2-3

ASP.NET Web Services

- Are also called as XML Web services.
- Were used in .NET Framework versions prior to 3.0.
- Are still considered as legacy services and not recommended for use in new applications.

ASP.NET Web API Services

- Used to build flexible and extensible Web services that can support a wide range of operating systems and modern devices.
- Is a framework that helps build HTTP-based services on top of the .NET Framework.
- Can also be used for building APIs while HTTP is used for building Web pages.
- Since most platforms these days have a HTTP library, creating HTTP-based services helps in reaching out to different clients such as desktop applications, browsers, mobile devices, and Smart phones.

Web Services in .NET 3-3

Windows Communication Foundation (WCF)

- Was included in the .NET Framework 3.0, a first framework.

- Has been included into the later releases of .NET Frameworks.

- Is a unified programming model that is primarily used for service-oriented applications.

- Helps to build service-oriented applications, which use message to communicate.

- Helps to build services that has number of benefits when compared with ASP.NET Web services.

Following are the benefits of WCF:

- o **Network Protocols**: Can use multiple network protocols, such as HTTP for sending messages.
- o **Switching**: Can switch network protocols without any effort for sending messages.
- o **Web Service Standards**: Supports latest Web services standards, such as SOAP 1.2 and WS-*.
- o **Format**: Can send messages in formats such as SOAP.

Web Services Terminologies

❑ Following are Web services terminologies:

EndPoint	Contract	Consumer	Host	SOA
• This is the Uniform Resource Locator (URL) that is used to communicate with the Web service. • A client application uses this URL for communication.	• This is the service agreement by a service provider and the client.	• It is the end user that uses the Web service.	• It is the server that contains the Web service and the application.	• It defines the how the Web services will be used to support the user requirements.

Understanding the Windows Azure Portal

❑ **Windows Azure Portal**:

- Defines the access to the components based on your subscription.

- Helps a developer can gain access to Cloud Service deployment and management tasks.

- Also has a reporting mechanism in which it displays a dashboard with status information depicting overall health of your deployments and accounts.

- Is refreshed regularly to display active status. You can create and manage various Windows Azure components and services using the portal.
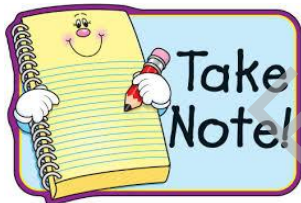
Creating a Windows Azure Account

❑ To use Windows Azure portal, you need to create an account on the portal.

❑ Following are the steps to create a Windows Azure account:

Step 1	Open _http://www.windowsazure.com_ in a Web browser.
Step 2	Go to the free trial account page and create an account. *(For this, you need to provide your credit card number and mobile phone number as an identity proof.)*

If you already have an MSDN subscription then create an account by activating your MSDN subscriber benefits from the Windows Azure Benefit for MSDN Subscribers page.

Exploring the Portal 1-2

❑ When you log on to the portal, you get to see the following portal interface with the left and right panes:

Left pane displays the services.

Right pane displays the context specific information depending on what you select in the left pane.

Exploring the Portal 2-2

❑ Following are the steps to be followed to edit or delete subscriptions:

Step 1
- Open the shortcut menu for the Windows Azure Compute node and then select **Add Deployment Environment**.

Step 2
- In the **Add Deployment Environment** dialog box, select the subscription that you want to modify, and then select the **Manage** button.

Step 3
- In the **Windows Azure Cloud Service Project Management Settings** dialog box, select the subscription that you want to modify, and then select the **Edit** button to change subscription details or the **Delete** button to delete the subscription.

Creating a Cloud Service through the Portal 1-3

❑ Perform the following steps to create a cloud service using **Quick Create** on the Portal:

Step	
Step 1	• Log on to the Windows Azure portal.
Step 2	• On the portal interface, click **New** at the bottom of the page, select **Cloud Services**, and then select **Quick Create**.
Step 3	• In the right pane, enter a sub domain public URL name for accessing cloud services in production deployments in the URL text box.
Step 4	• In **Region/Affinity Group** drop-down, select the geographic region or affinity group where you want to deploy the cloud service to.
Step 5	• Click **Create Cloud Service**.

Creating a Cloud Service through the Portal 2-3

❑ You can:

- Monitor the process as it is displayed at the bottom of the window while deploying the cloud service.

- Check the status as Created as it implies that the cloud service has been created successfully.

- Create a Web site using the WEB SITES option on the left pane.

❑ It is mandatory to upload a cloud certificate before deploying the cloud service.

❑ Once the certificate is uploaded, then the Windows applications running on role instances can access the certificate.

Creating a Cloud Service through the Portal 3-3

❑ Following are the steps to upload a certificate for the cloud service:

Step 1
- Click **Cloud Services** in the Windows Azure Management Portal. Note that you will see the services that you have created. Select a service to open its dashboard.

Step 2
- click **Certificates** on the menu bar**. The certificates page is displayed.

Step 3
- Click **UPLOAD A CERTIFICATE**. The **Upload certificate** window is displayed.

Step 4
- Click **Browse** next to the **Certificate file** text box and select the certificate (.pfx file).
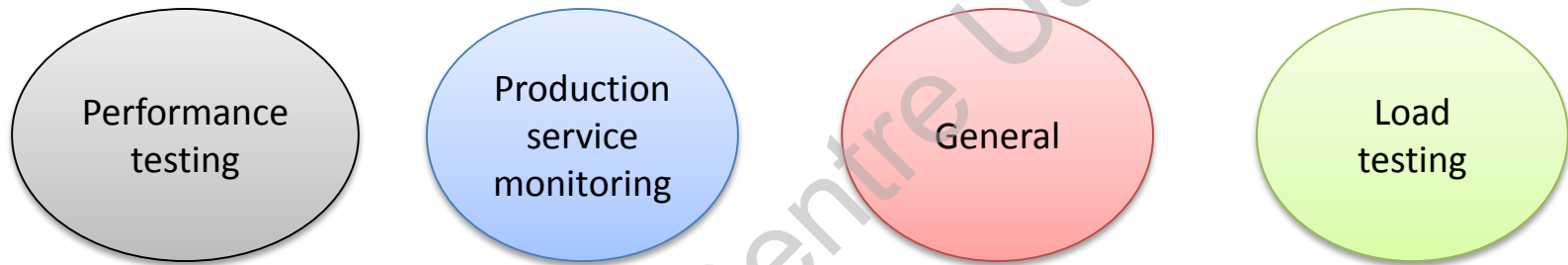
Step 5
- Enter the password for the certificate in the **Password** text box**. The password is the private key for the certificate.

Step 6
- Click **OK**. The certificate is now added.

Testing the Cloud Service 1-4

❏ In organizations, testing normally involves following tests:

Performance testing

Production service monitoring

General

Load testing

❏ **Need for Cloud Testing**:

– The software testing carried out by traditional approaches incurs high cost to simulate user activity from various geographical regions.

– Testing the firewalls and load balancers involves maintenance of hardware and software.

– A number of bugs can exist in the cloud-based applications like the local environment.

– You must test the solution in a separate environment than the production environment.

Testing the Cloud Service 2-4

❑ **Limitations**:

- The test results may not be accurate due to varying performance of service providers' network and Internet.

- In many cases, service virtualization can be applied to simulate the specific performance and behaviours required for accurate and thorough testing.

- Also, the initial setup cost for migrating testing to cloud is very high as it involves modifying some of the test cases to suit cloud environment.

Testing the Cloud Service 3-4

❑ Following are some types of testing:
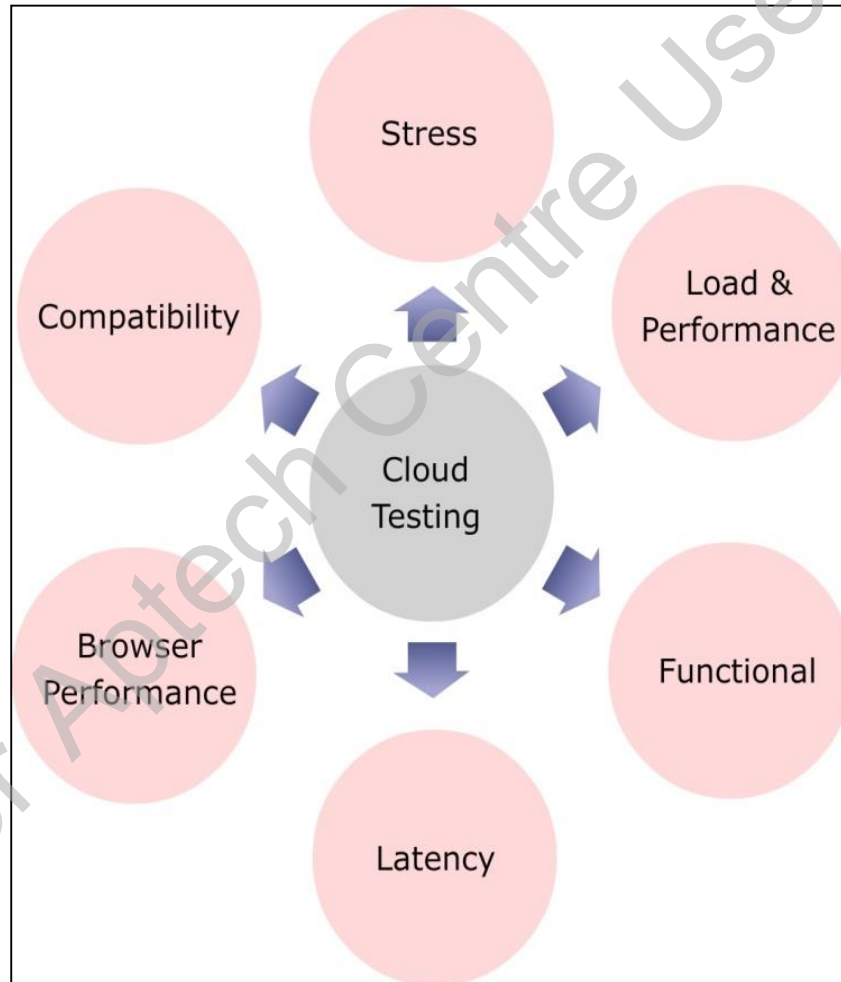
Testing the Cloud Service 4-4

❑ Some of the tests are as follows:

Load test	Performance test	Functional test	Latency testing
• This involves in creating heavy user traffic and measuring its response.	• This involves testing the performance of a working load. • By using cloud testing the traffic can be varied as required, this lessens the cost and time by simulating large number of users at various geographical locations.	• This is carried out on both Internet-based and offline applications. • It involves verification of specifications and system requirements and is carried in cloud instead of on-site software testing.	• It is mainly used to test the latency between the action performed and its corresponding response for any application after installing it on cloud.

Summary 1-2

❑ Cloud computing is an approach that enables convenient and on-demand access through the Internet to computing capabilities and resources.

❑ Cloud computing platforms provide different kinds of services, depending on the delivery model that they deploy, such as IaaS, PaaS, and SaaS.

❑ Windows Azure is a cloud computing platform created by Microsoft and is used for building, deploying, and managing services and applications.

❑ Windows Azure OS works as an integral part of the Azure Services Platform that covers different and separate application, storage, desktop environment, security, and so on.

Summary 2-2

❑ SOA is an architecture used to build or extend modular systems in a flexible and reusable manner.

❑ A Web service can be defined as an application component, can be reused by multiple applications, and uses HTTP and XML as the base.

❑ The supported types of services in .NET include ASP.NET Web Services, WCF services, and ASP.NET Web API Services.

❑ The Microsoft Windows Azure portal defines the access to the components based on your subscription.