



Building and Connecting to Cloud Services by Using Windows Azure

...for Java developers

Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2011 Microsoft Corporation. All rights reserved.

Microsoft is either a registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Introduction

As a Java developer, you may have created a cloud-based application and deployed it to Google App Engine (GAE) or to a virtual machine that Amazon hosts by using Amazon Elastic Compute Cloud (Amazon EC2). Windows® Azure™ is the Microsoft cloud platform that provides on-demand, cloud-based hosting services for your applications.

Cloud-based computing offers you a viable way to create robust and scalable applications that are hosted on servers that are running in a data center that a third party manages, instead of deploying applications to servers in your own data center. By deploying your applications on a dedicated cloud platform, you can focus on creating your business application and rely on the third party to manage the data center, maintain the infrastructure, and provide a standardized runtime environment for your applications.

The Windows Azure hosted service model makes it easy to scale up your application on demand by adding additional instances of your services. It is just as easy to scale back when the additional capacity is no longer required. The various services that are available through the Windows Azure platform are metered, so you only pay for what you use.

The Windows Azure SDK provides a comprehensive set of application programming interfaces (APIs) that your applications can use to interact with the many services that the [Windows Azure platform](#) offers. Windows Azure is an open platform, so you're not limited to using .NET languages and tools. There are Java APIs for interacting with Windows Azure and tools for Eclipse. You can also run Java code on Windows Azure.

The Windows Azure platform currently consists of several elements:

- Windows Azure compute services, which can run application code.
- Windows Azure storage services, which can store application data.
- A collection of other cloud-based services such as Microsoft® SQL Azure™ and Windows Azure AppFabric, which provide additional data services and support for security.

Contents

Introduction	ii
Contents	ii
Windows Azure	3
The Web Role	5
The Worker Role	5
Windows Azure Storage	6
SQL Azure	7
Windows Azure AppFabric	7
The AppFabric Service Bus	7
AppFabric Access Control	8
Summary	8

Windows Azure

Windows Azure offers a robust and scalable environment to host and run your applications in the cloud. You can build complex business applications by using the Microsoft .NET Framework tools and then deploy the application to a Windows Azure data center from where it is globally accessible. Microsoft has constructed data centers in several locations around the world. The runtime environment is standardized, so if you move your application to a different Windows Azure data center, you can be sure that it will behave in exactly the same way.

After you have deployed your application, you can manage it remotely over the Web by using a Web-based console, or through a collection of Web services that Microsoft provides.

A typical Windows Azure application has three key elements:

- A Web role
- A worker role
- Storage

This is a little different from other cloud platforms. For example, if you use Java to create an application to deploy to GAE, your application consists of servlets and JavaServer Pages (JSP) pages that are running in a custom Web container. GAE does not have a direct equivalent of a worker role. In GAE, you run asynchronous tasks in the Web container, but it only supports tasks that have a duration of up to 10 minutes. If you deploy a Java application to a virtual machine that Amazon EC2 manages, you are likely to be using a standard Java Platform, Enterprise Edition (JEE) server installed on that virtual machine, so you will be using standard Java APIs. It is also your responsibility to design the application for scalability in the Amazon EC2 environment. The Windows Azure platform handles far more of the scalability issues for you in the Web and worker roles.

Figure 1 illustrates the relationships between the Windows Azure elements. These elements are summarized in the sections that follow.



Did you know?

Windows Azure now also supports virtual machine roles that enable you to run custom virtual machines that contain your own installation of Windows Server® 2008 and any other software that you require. This approach is similar to the way in which the Amazon EC2 service enables you to run custom Amazon Machine Images on Amazon servers in the cloud. When you take this approach, it is up to you to keep the operating system and installed software in the virtual machine up to date.

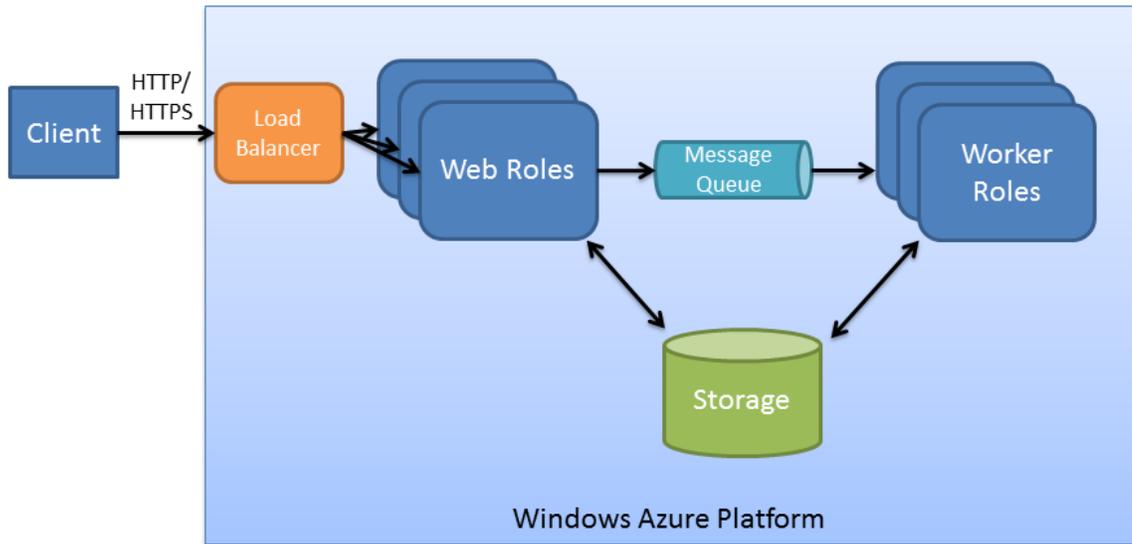


Figure 1. Architecture of a typical Windows Azure hosted application

Figure 2 illustrates the equivalent elements in a Java application that is hosted in Google App Engine.

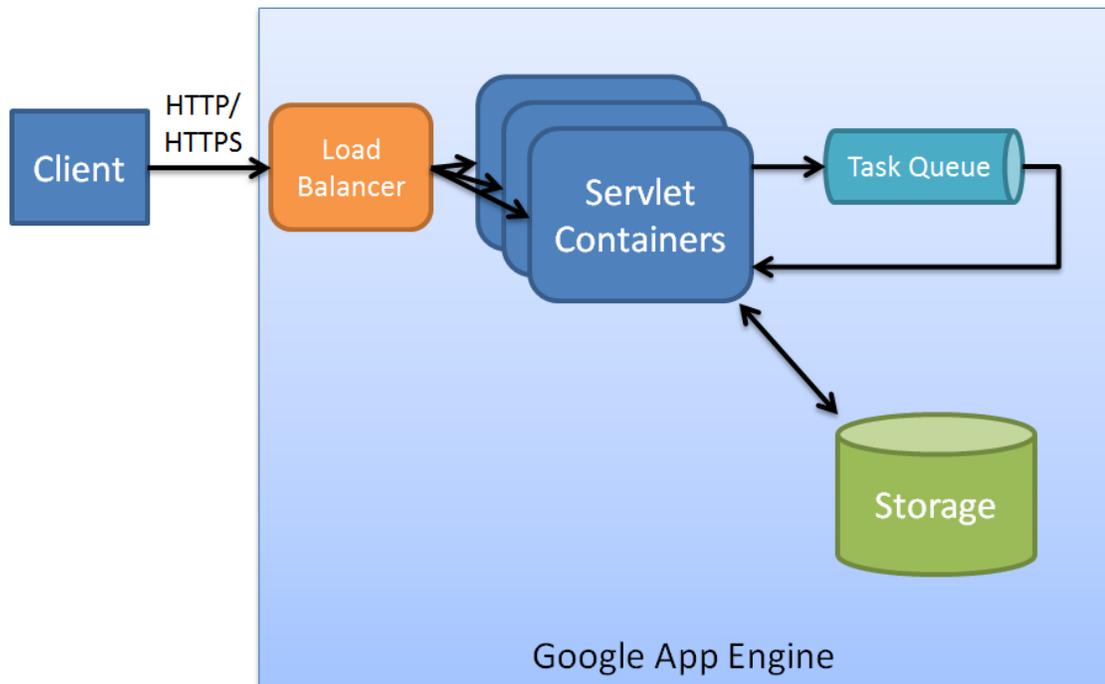


Figure 2. Architecture of a typical Google App Engine hosted application

Figure 3 illustrates the equivalent elements in a Java application that is deployed by using Amazon EC2.

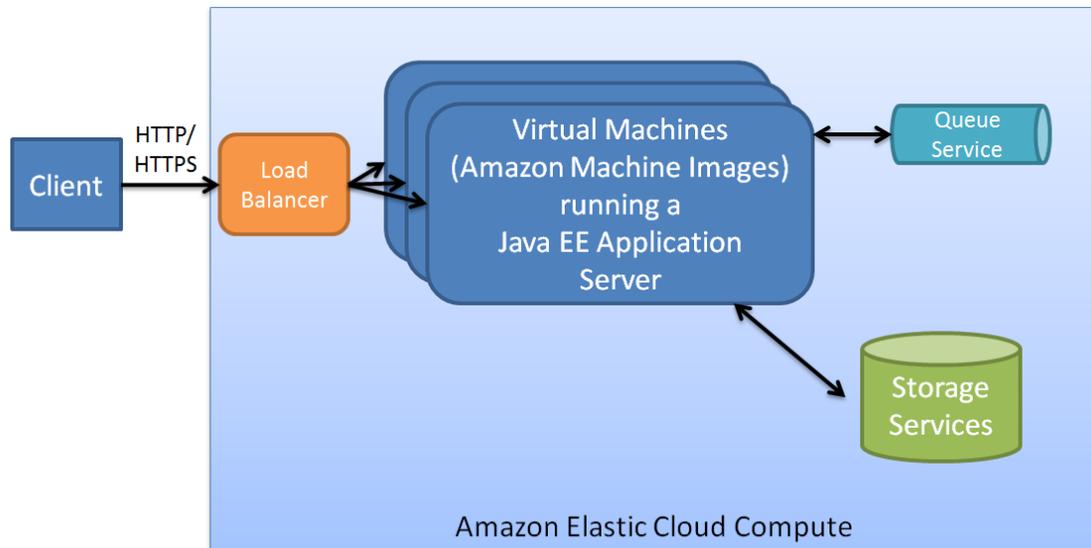


Figure 3. Architecture of a typical Amazon EC2 hosted Java EE application

The Web Role

You can think of the Web role as being your Web server in the cloud where you can deploy your ASP.NET Web applications and Windows Communication Foundation (WCF) services. This is similar to a GAE instance where you can host JSP pages or servlets in the cloud in a custom Web container. However, the Windows Azure Web role is much less restrictive about the APIs that are available to your code than GAE. You use the Web role to handle incoming HTTP or HTTPS requests. The reply that a Web role can return might be a Web page or a serialized SOAP or REST response message.

The Worker Role

A worker role in Windows Azure is a general-purpose, hosted container for executing long-running, non-interactive tasks. These tasks often run asynchronously. A worker role



Did you know?

You can run applications such as Apache Tomcat in a worker role, or even run your own Java code.

typically performs the background processing that is specified by messages that are sent from a client application to a Web role. The Web role can use a queue to reliably pass the necessary information to run a task in a worker role. Windows Azure manages and monitors both the worker role and the Web role.

Windows Azure Storage

Figure 4 shows the two main types of storage that Windows Azure provides for applications to use. These are:

- **Windows Azure Binary Large Object (BLOB) storage.** Follows a similar approach to Amazon Simple Storage Service, enabling you to store large objects, such as media files, in the cloud.
- **Windows Azure Table storage.** Uses a non-relational storage model that is optimized for the cloud, much like the Amazon SimpleDB data store.

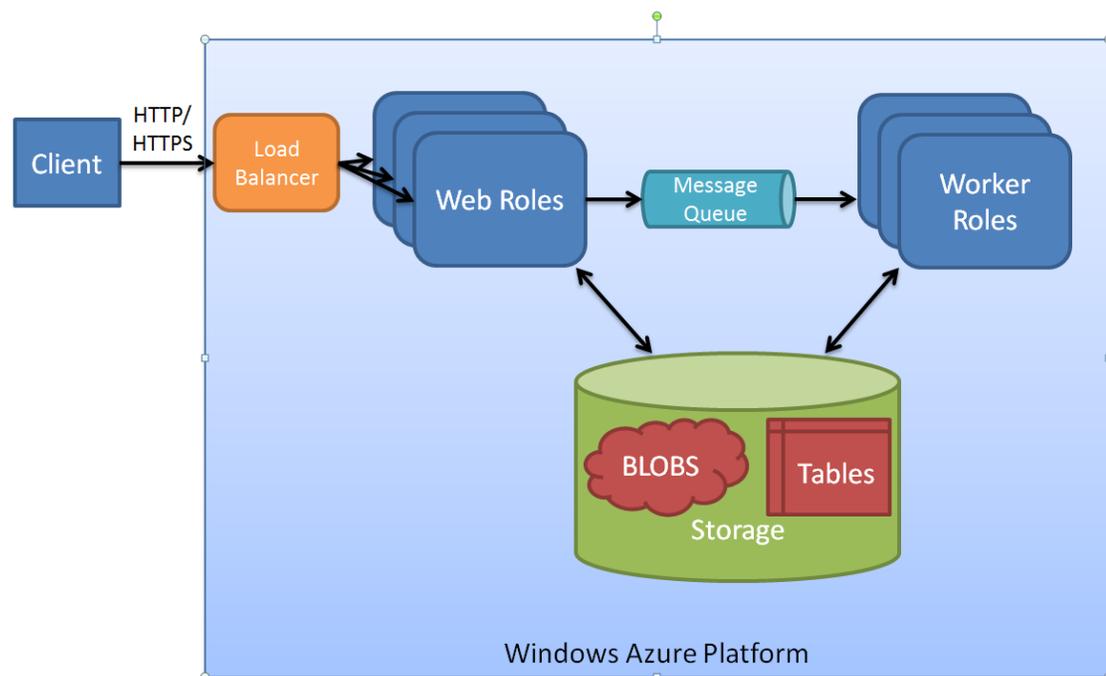


Figure 4. Windows Azure storage options

You can access Windows Azure BLOB and Table storage from both worker and Web roles, and directly from your on-premises applications, by using a REST API. If you are building Java applications that need to access data that is stored in Windows Azure storage, you can use the [Windows Azure SDK for Java](#). Under the covers, Windows Azure storage, Amazon SimpleDB, and the GAE data store all



Did you know?

Windows Azure Table and BLOB storage are optimized for the cloud environment. However, if your application needs access to standard file system-type storage, you can also use a Windows Azure drive that is formatted as an NTFS file system (NTFS) volume.

use REST-based APIs, making it easy to access them from any programming language.

SQL Azure

Although Windows Azure includes Table storage services for holding simple, structured data, there may be times when you need a full, relational database in the cloud. For example, you may be making extensive use of joins and complex queries, you may already have a relational database design, or you may want to use features such as Microsoft SQL Server® Reporting Services. [SQL Azure](#) is simply SQL Server (an enterprise-class relational database management system) running in the cloud in the same way that Amazon Relational Database Service is MySQL running in the cloud.

Windows Azure AppFabric

[Windows Azure AppFabric](#) is a collection of additional cloud-based services that applications that are running in Windows Azure, or on-premises applications, can use. The [AppFabric SDK for Java Developers](#) is useful for creating Java client applications that can take advantage of the Windows Azure AppFabric services. The following sections describe the services that Windows Azure AppFabric currently provides.



Did you know?

There are [AppFabric SDKs](#) for Ruby and PHP developers in addition to tools for Eclipse.

The AppFabric Service Bus

You can build WCF services that provide communication channels between applications and enable those applications to exchange messages. This works well in many scenarios, but is difficult to implement if both the client and server are behind their own firewalls. The AppFabric Service Bus provides a solution to this problem by enabling the server application to register an endpoint that is hosted in the cloud. Then, a client running elsewhere on the Internet can discover this endpoint. The AppFabric Service Bus acts as an intermediary, enabling the exchange of messages between the client and the server. You can configure the endpoint either as a SOAP-based WCF service or as a REST WCF service, enabling interoperability with a wide range of clients, including Java applications. The functionality that the AppFabric Service Bus exposes is similar to that of Amazon Simple Notification Service.

Figure 5 illustrates how the AppFabric Service Bus enables applications to exchange messages. Application #1 could be an application that is running at a client's premises, and Application #2 could be an application that is hosting a service that is running at a separate site.

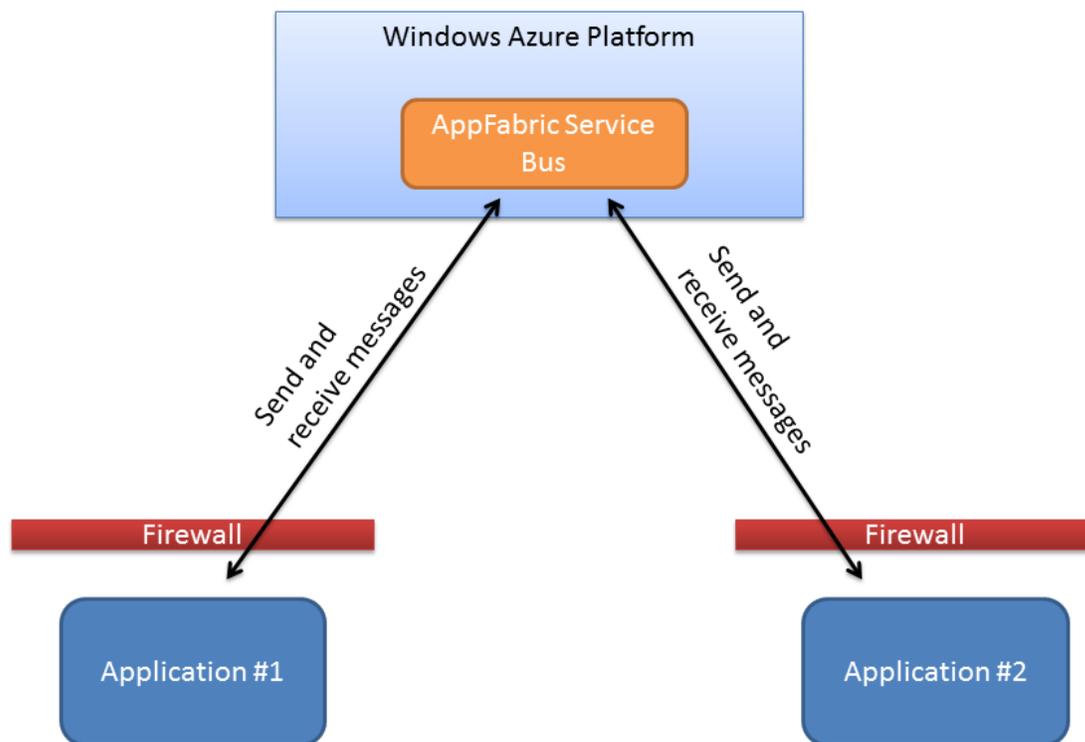


Figure 5. The AppFabric Service Bus

AppFabric Access Control

AppFabric Access Control is a service that enables your Web applications and services (whether they are hosted in Windows Azure or on your own servers) to integrate with third-party identity services such as Active Directory® Federation Services, or identity services that follow the [OAuth](#) protocol. AppFabric acts as a translator that accepts security tokens that an external identity provider issues, and converts them to a standard form (a signed Simple Web Token) that your Web application or service understands. In this way, you can easily implement single sign-on scenarios or use industry-standard authentication mechanisms with your application.

Summary

Windows Azure provides you with a platform for running your applications in the cloud. You can build your cloud application by using .NET Framework tools and languages such as C#. However, the platform is open and interoperable, so you can utilize other technologies such as Java, both for implementing your cloud services and for creating client applications that access your cloud services.