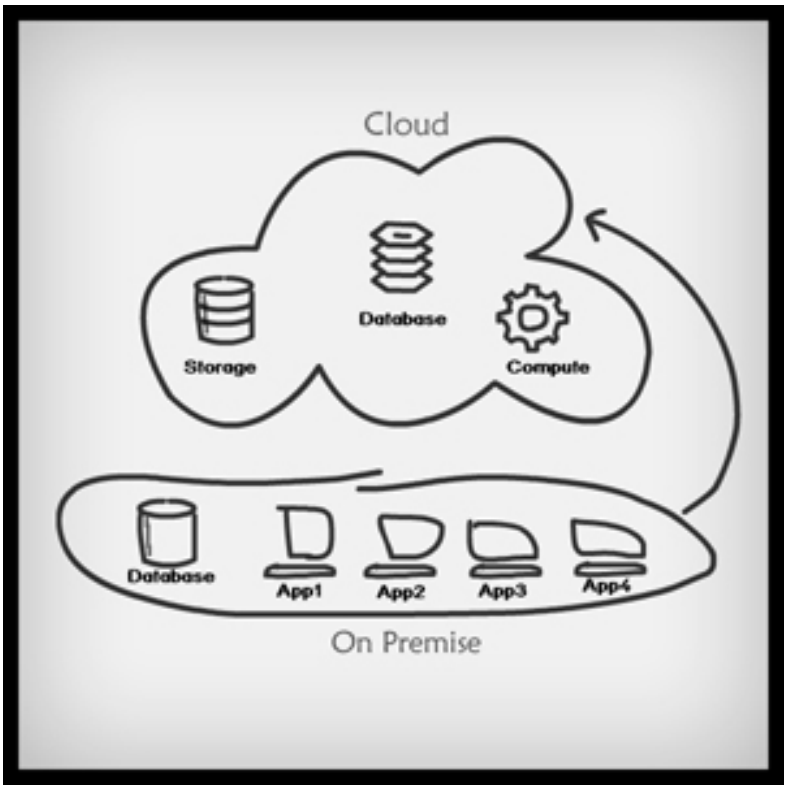
Migrating Your Application to the Cloud



The migration process to Windows Azure is actually quite straightforward. Here's the generic process we used:

1. Verify that the application is running correctly in the development environment.
2. Migrate the SQL Server back end to SQL Azure using the SQL Azure Migration Wizard.
3. Update the local application to work with the SQL Azure database.
4. Convert the application into a Web Role project.
5. Validate that the application runs on the local development fabric.
6. Package up the Web Role and deploy it to Windows Azure.
7. Validate that the application runs from Windows Azure.

With all the talk about the cloud, and an increasing understanding of its value and importance, there are elementary steps that must be taken before migrating your application to the cloud.

July 19, 2011

Migration Basics

When deciding to migrate an application from on-premises to the cloud (or to create a new application on a cloud service), there are several aspects of the application architecture that need to be considered:

- Application management
- Application security
- Application compatibility
- Database compatibility

Application Management

No matter whether your application is running on-premises or in the cloud, the operations management team needs data that will enable them to make effective decisions. The issues you'll need to consider include service-level agreements, capacity planning, customer billing, auditing, application monitoring, traffic analysis and managing costs (knowing when to scale up or down). These need to be resolved before the application is deployed to production—and for best results, often before the application is created. By utilizing the Windows Azure Diagnostics API provided in the Windows Azure SDK (Microsoft.WindowsAzure.Diagnostics), customers are able to expose application crash dumps, failed request tracing, Windows event logs, IIS logs, Windows Azure logs and performance counters.

Application Security

A top concern of any organization moving to the cloud is security. Most companies have invested a substantial amount of time, money and engineering into designing and developing a security model and it's important that they're able to leverage existing investments such as identity stores, single sign-on solutions and firewalls.

Fundamentally, Windows Azure must provide confidentiality, integrity, and availability of customer data, just like any other application hosting platform. It must also provide transparent accountability to allow customers and their agents to track administration of applications and infrastructure, by themselves and by Microsoft.

Confidentiality ensures that a customer's data is only accessible by authorized entities. Windows Azure provides confidentiality via the following mechanisms:

- Identity and Access Management - Ensures that only properly authenticated entities are allowed access.
- Isolation - Minimizes interaction with data by keeping appropriate containers logically or physically separate.
- Encryption - Used internally within Windows Azure for protecting control channels and is provided optionally for customers who need rigorous data protection capabilities.

Application Compatibility

Windows Azure is an application platform, so it's important to understand the types of applications suited to the Windows Azure platform. While you have the ability to run native code and you can run applications with full trust, you must package your application before deploying it to the cloud, which means it's important to evaluate your application to see if it's a good fit.

Database compatibility

First, it's important to check the size of your database and how it fits within the database allowances used by SQL Azure. Currently, SQL Azure offers Web Editions in 1GB and 5GB sizes and Business Editions in 10, 20, 30, 40 and 50GB sizes. You need to check your database and make sure it isn't larger than 50GB. If your database is larger than 50GB, then you'll need to examine your database and see if it can be broken down into smaller databases (in other words, sharding your database) or moving large data to blobs.

SQL Azure supports only SQL Authentication, so you'll need to consider whether changes are needed to the authentication scheme used by your application. On top of that, SQL Azure has a resource throttle that limits connection time.

ASP.Net application Migration to Windows Azure

Migration of existing ASP.Net web application to Windows Azure involves manual work as there is no automated tool available. This also requires you to look at several aspects of your application:

- Web Site to Web Application conversion
- App.config/Web.config changes
- .Net Full trust configuration
- Caching / State Management Migration
- Forms Authentication Migration
- Send Email from Cloud Application
- File System in Azure
- COM Support in Azure
- Migrate WCF service to Cloud
- Migrate only SQL Server to SQL Azure
- ADO.Net Data Services and Entity Framework Support in Azure
- Data Access Layer Migration
- Background Job Migration
- Migrating Database Schema
- SQL Server Data Migration