Ektron in the Azure Cloud

Doc. Rev. 1.6 (Mar. 2012)

Ektron in the Azure Cloud

Ektron 8.5

Doc. Rev. 1.6 (Mar. 2012)

Ektron, Inc.

Ektron, Inc.
542 Amherst St.
Nashua, NH 03063
Tel: +1 603-594-0249
Fax: +1 603-594-0258
Email: sales@ektron.com
http://www.ektron.com

For the latest version of this manual, go to http://www.ektron.com/Resources/Product-Documentation/

Trademarks

Ektron, the Ektron logo and Ektron, are trademarks of Ektron Inc.

Other company and product names may be trademarks of their respective owners.

Table of Contents

(This page intentionally blank.)

Setting Up Ektron in the Azure Cloud

This document contains step-by-step instructions to set up and run Ektron applications in Windows Azure Cloud.

Ektron in the Azure Cloud has the following benefits:

- Simplifies the process of creating, deploying and managing ASP.NET Web sites in Azure
- Provisions new sites within hours
- Connects your On Premises infrastructure with the cloud
- Manages content changes easily through a true hybrid cloud deployment
- Unlimited scalability
- Reduces time-to-scale
- Significant cost reductions

Before You Install to the Cloud

Before you install or use Ektron in the Windows Azure Cloud, EktronAdministrators and Developers should read this document. First build the system for On Premises use and then follow the instructions in this documentation to get it running in cloud. You also should understand the following:

- Windows Azure SDK 1.6, and Azure Libraries 1.6 (November 2011) on a development machine or on-premise server.
- You should have a valid subscription from the Windows Azure portal.
- Run the security configurator and generate certificates before you deploy Ektron to the Azure cloud.
- Use Azure deployment wizard for proper configurations.
- Instances must be started before running database or template synchronizations.
- Ektron eSync deploys the data to SQL Azure and file resources to BLOB storage.
- Connecting SQL Azure through enterprise manager requires SQL Server 2008R2 for SQL Management studio (latest cumulative update).
- You can use both Web site and Application model as of this printing.
- SQL Azure licensing limits the maximum database size.
- The following are not supported as of this printing:
  - Active directory
  - eCommerce
  - multi-site
  - Web Alert
  - Notification Services
  - Diagnostic Services

Configuring the Portal

The following sections describe the steps you take to configure your portal.

1. *Sign Up* below
2. *Install a Cloud Server Certificate* below
3. *Create a Hosting Service on the facing page*
4. *Create a Storage Account* on page 8
5. *Create a CDN Endpoint* on page 8
6. *Create a Database Server* on page 9
7. *Create a Storage Account* on page 8
8. *Create a Namespace in AppFabric* on page 10

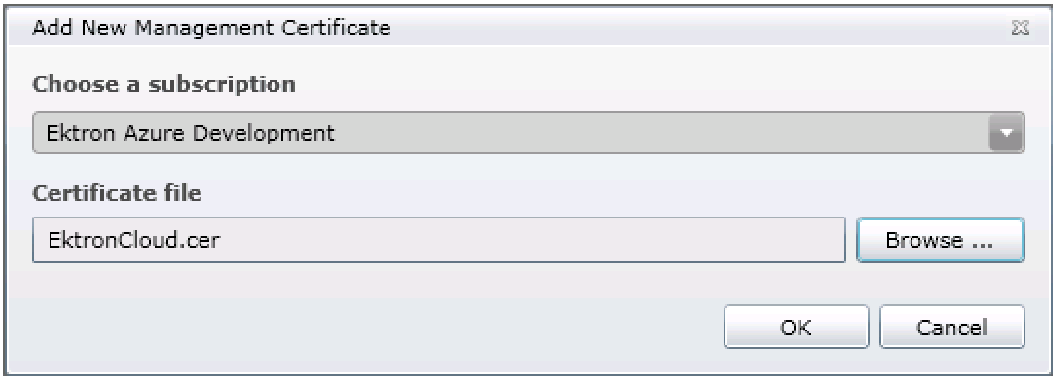
Sign Up

Windows Azure hosting requires Windows Live-ID sign on using your email address and subscription, which can be purchased from Microsoft at http://www.microsoft.com/windowsazure/. When you have the subscription, go to http://windows.azure.com and log in using your Live-ID and password.

Install a Cloud Server Certificate

To install a cloud server certificate:

1. From the home page of http://windows.azure.com, click **Hosted Services, Storage Accounts & CDN**.
2. Click the **Manage Certificates** folder.
3. Click **Add Certificate**. The Add New Management Certificate dialog box appears.



Add New Management Certificate

Choose a subscription: Ektron Azure Development

Certificate file: EktronCloud.cer

4. Choose a subscription from the drop-down list.
5. Click Browse and choose `EktronCloud.cer` from the following path:
   - 64-bit system:

     ```
     ~\Program Files (x86)\Ektron\CMS400v85\AzureCerts
     ```

   - 32-bit system:

     ```
     ~\Program Files\Ektron\CMS400v85\AzureCerts
     ```

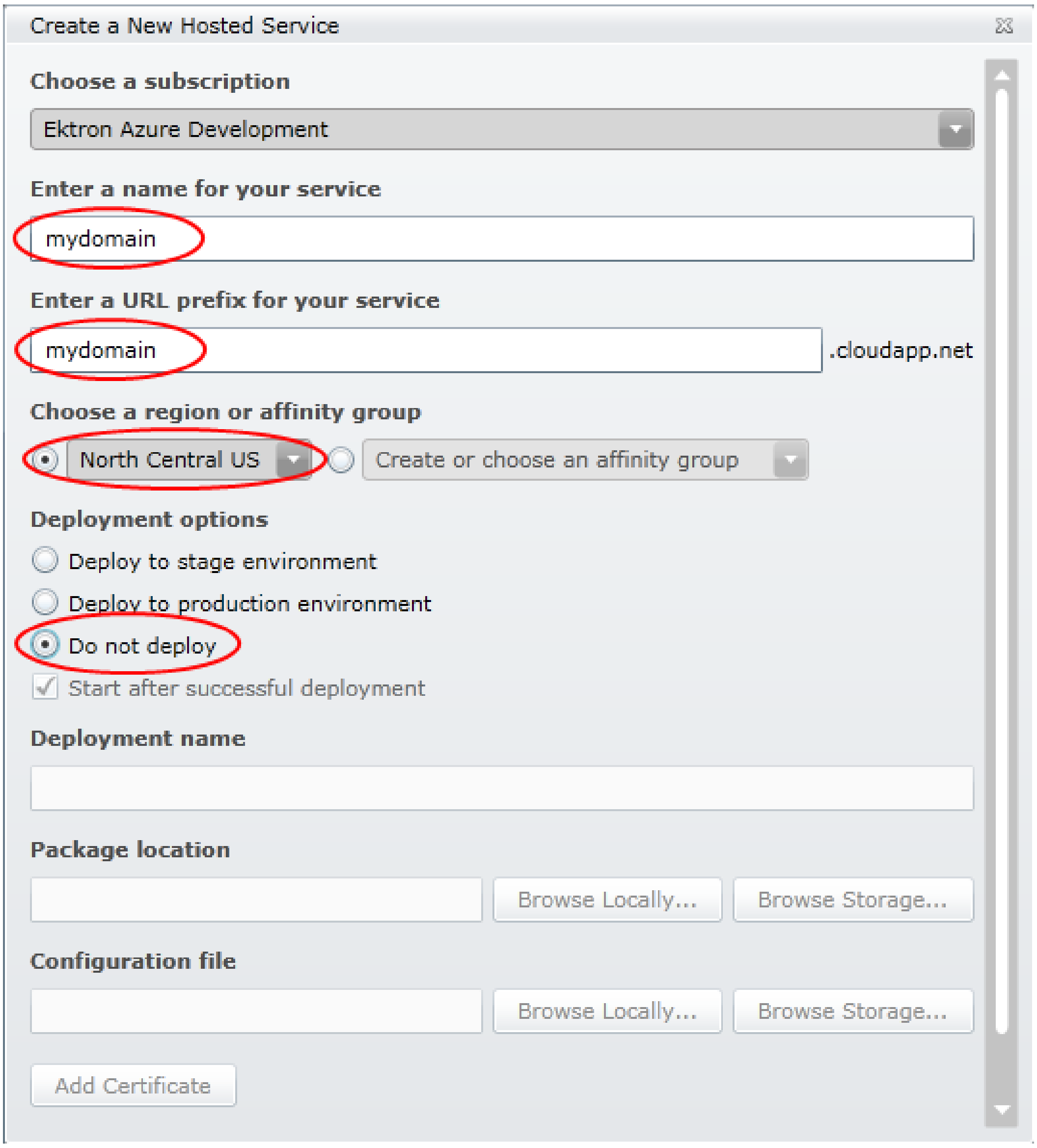
6. Click **OK**.

Create a Hosting Service

NOTE: This is an optional step. If you do not create the hosted service, Ektron will create it for you.

To create a hosting service:

1. From the home page of http://windows.azure.com, click **New Hosted Service**. The Create a New Hosted Service dialog box appears.



Create a New Hosted Service

Choose a subscription: Ektron Azure Development

Enter a name for your service: mydomain

Enter a URL prefix for your service: mydomain .cloudapp.net

Choose a region or affinity group: ● North Central US ○ Create or choose an affinity group

Deployment options
○ Deploy to stage environment
○ Deploy to production environment
● Do not deploy
☑ Start after successful deployment

Deployment name

Package location

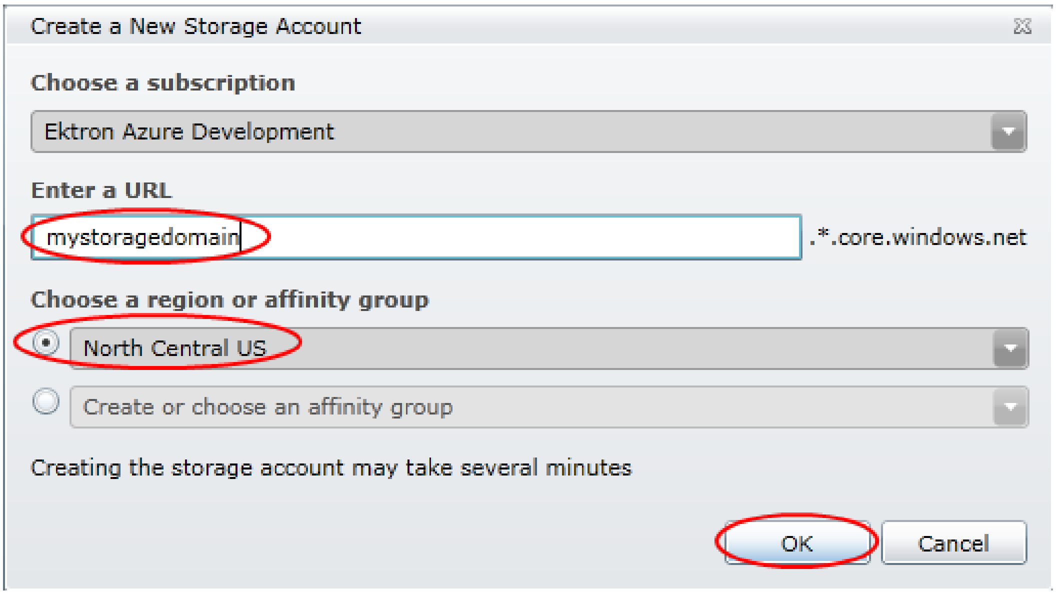
Configuration file

2. Enter the name of your service.
3. Enter the URL prefix for your service. If your host name is `mydomain` then your URL will be `mydomain.cloudapp.net`, which is used to configure the CNAME rec and location.
4. Choose a region.
5. Click **Do not deploy**.
6. Click **OK**.

Create a Storage Account

You need a storage account to store site resources in the cloud storage service, and to store and synchronize assets, private assets, uploaded files, uploaded images and asset library. To create a storage account:

1. From the home page of http://windows.azure.com, click **New Storage Account**. The Create a New Hosted Service dialog box appears.



Create a New Storage Account

Choose a subscription

Ektron Azure Development

Enter a URL

mystoragedomain .*.core.windows.net

Choose a region or affinity group

● North Central US

○ Create or choose an affinity group

Creating the storage account may take several minutes

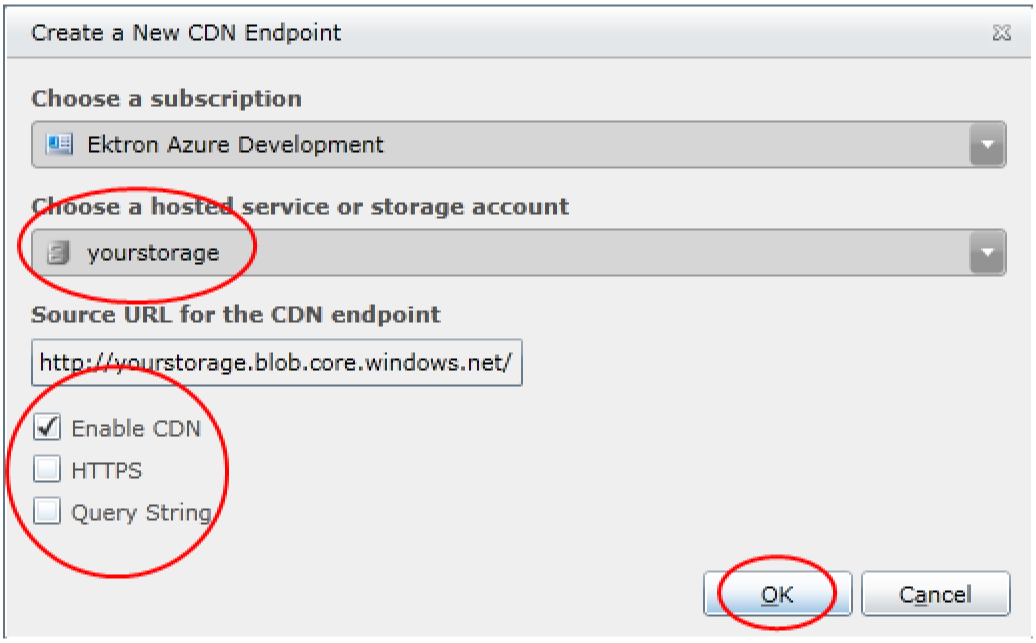
2. Enter your URL.
3. Choose a region.
4. Click **OK**.

Create a CDN Endpoint

Content Delivery Network (CDN) serves files faster from distributed servers. For better performance, Ektron recommends that you define the endpoint for your storage accounts. To create a CDN endpoint:

1. From the home page of http://windows.azure.com, click **Hosted Services, Storage Accounts & CDN**.
2. Click on **CDN** in the left navigation panel.

3. Click **New Endpoint**. The Create a New CDN Endpoint dialog box appears.



Create a New CDN Endpoint

Choose a subscription

Ektron Azure Development

Choose a hosted service or storage account

yourstorage

Source URL for the CDN endpoint

http://yourstorage.blob.core.windows.net/

☑ Enable CDN

□ HTTPS

□ Query String

4. Choose the hosted service from the drop-down menu.
5. Optionally check **Enable CDN**, **HTTPS**, and **Query String**.
6. Click **OK**.

Create a Database Server

To create a database server:

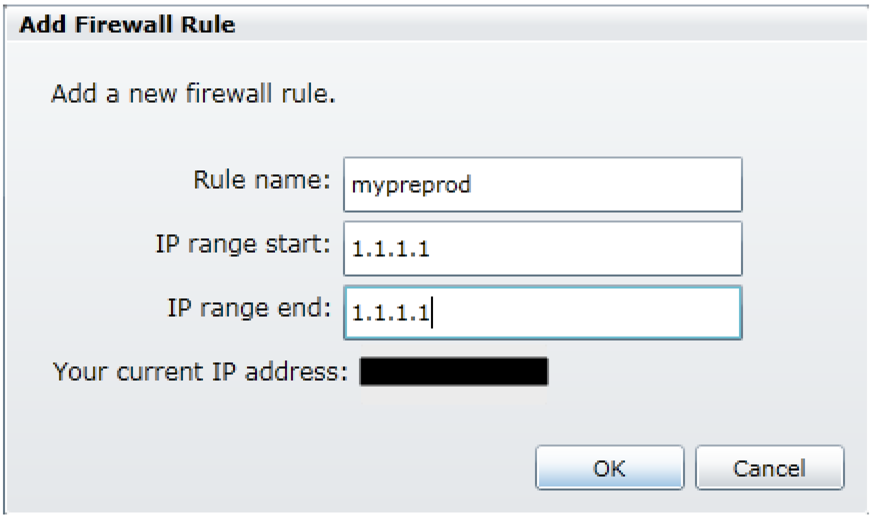
1. From the home page of http://windows.azure.com, click **Database**.
2. Click the subscription in which you want to create the server.
3. Click **Create** in the Server button bar. The Create Server dialog box appears.
4. Select a region where the SQL Azure server will be hosted and click **Next**.
5. Specify the Administrator Login and Password to use with the SQL server and click **Next**.
6. Click **Add**. The Add Firewall Rule dialog box appears. (See *Create a Firewall Rule in SQL Azure* below.)
7. Check the box next to *Allow other Windows Azure services to access this server*.
8. Click **Finish**.

Create a Firewall Rule in SQL Azure

To set up a firewall rule in SQL Azure, including the IP Address from your pre-production and development machines:

1. From the home page of http://windows.azure.com, click **Database**.
2. Choose **Subscriptions > Ektron Azure Development > *yourServerName***.

3. Under Firewall Rules, click **Add**. The Add Firewall Rule dialog box appears.



Add Firewall Rule

Add a new firewall rule.

Rule name: mypreprod

IP range start: 1.1.1.1

IP range end: 1.1.1.1

Your current IP address: [redacted]

4. Specify the rule name, and the starting and ending IP range.
5. Click **OK**.

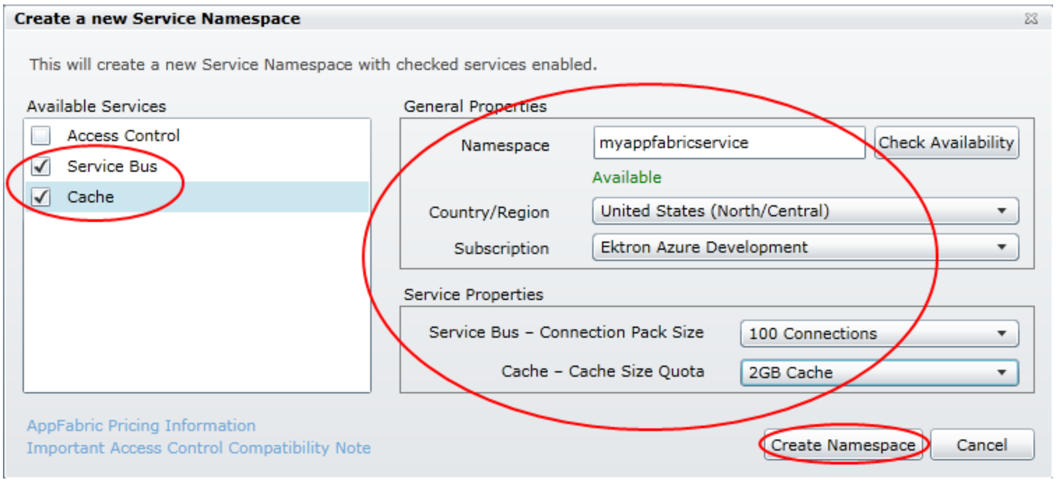
Create a Namespace in AppFabric

You need to create a namespace in Appfabric for service bus and caching services. Azure does not support sticky sessions, so you need to use the AppFabric caching service for your caching for more than one instance or load balancing.

Use the service bus to connect to On Premises services for search or any custom-driven services, because there are no search servers in the cloud. Ektron's hybrid deployment model lets you make changes to and run searches on On Premises servers. You can use eSync to deploy the change data to the cloud.

To create a namespace in AppFabric:

1. From the home page of http://windows.azure.com, click **Service Bus, Access Control & Caching**.
2. Choose **AppFabric > Cache**.
3. Click **New**. The Create a new Service Namespace dialog box appears.



Create a new Service Namespace

This will create a new Service Namespace with checked services enabled.

Available Services

□ Access Control

☑ Service Bus

☑ Cache

General Properties

Namespace: myappfabricservice

Available

Country/Region: United States (North/Central)

Subscription: Ektron Azure Development

Service Properties

Service Bus – Connection Pack Size: 100 Connections

Cache – Cache Size Quota: 2GB Cache

Check Availability
AppFabric Pricing Information
Important Access Control Compatibility Note
Create Namespace
Cancel

4. Check **Service Bus** and **Cache**.

5. Enter a name (such as `myappfabricservice`) in the Namespace field.
6. Select a region from the Country/Region drop-down menu.
7. Select a subscription from the Subscription drop-down menu.
8. Optionally, select the number of connections you want in the Service Bus - Connection Pack Size drop-down menu.
9. Select the cache size that your application needs from the Cache - Cache Size Quota drop-down menu.
10. Click **Create Namespace**.

Creating and Deploying a Site Package to the Cloud

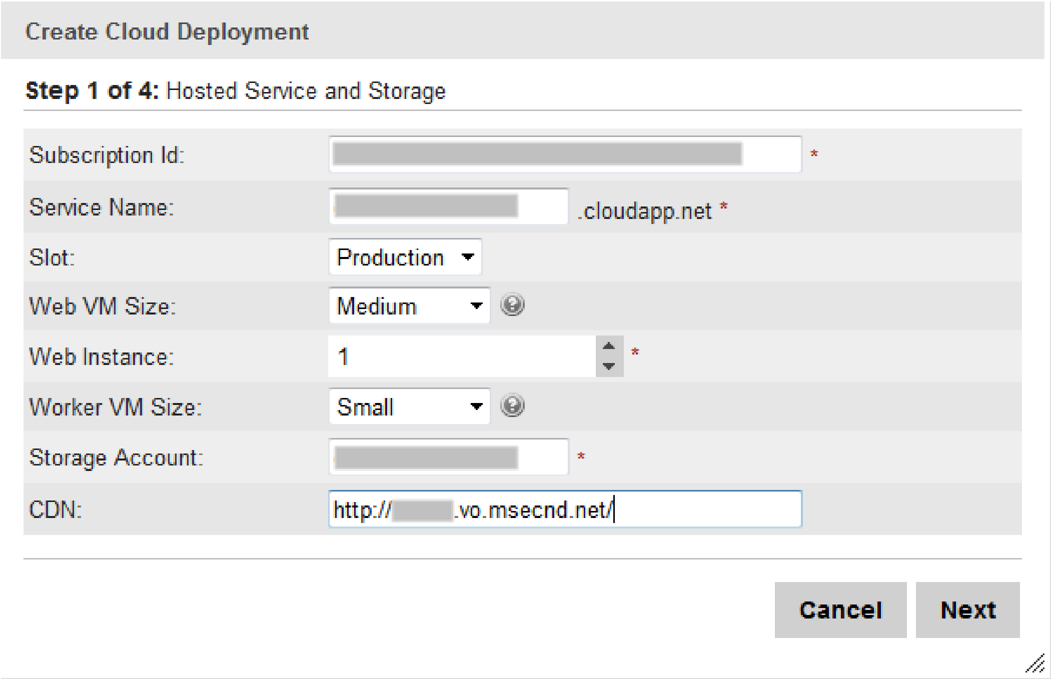
After you configure the portal, the following sections show the steps you take to get your code running in the Cloud.

1. *Create the Initial Deployment* below
2. *Synchronize the Database and Templates* on page 15
3. *Manage the Cloud Instances* on page 16
4. *Configure Ektron Storage* on page 17
5. *Configure Ektron Caching* on page 17
6. *Set the SQL Azure Connection String* on page 18
7. *Set the BLOB Redirect Handler* on page 18
8. *Create a Cloud Package* on page 18

Create the Initial Deployment

To create the initial deployment:

1. In the Ektron Workarea, choose **Settings > Configuration > Windows Azure Deployment**.
2. Choose **Create Azure Hosted Service Deployment**. A wizard appears (Step 1 of 4: Hosted Service and Storage). Fields with a red asterisk (*) are required.



Create Cloud Deployment

Step 1 of 4: Hosted Service and Storage

Subscription Id:	[redacted] *
Service Name:	[redacted] .cloudapp.net *
Slot:	Production
Web VM Size:	Medium
Web Instance:	1 *
Worker VM Size:	Small
Storage Account:	[redacted] *
CDN:	http://[redacted].vo.msecnd.net/

a. **Subscription ID**—You can get this from http://windows.azure.com.
   1. Click on **Hosted Services, Storage Accounts & CDN**.
   2. Click the **Storage Account** folder.
   3. Copy and paste the Subscription ID from the Properties panel.

b. **Hosted Service Name**—Specify the name of the server.

c. **Deployment Environment**—Choose either **Production** (Friendly URL) or **Staging** (GUID-based URL).

d. **Web Virtual Machine Size**—Choose **Small**, **Medium** (default), **Large**, or **Extra Large** for your web virtual machine. This is your website and you can have as many instances as you want. Each option is described at *How to Configure Virtual Machine Sizes* (http://msdn.microsoft.com/en-us/library/windowsazure/ee814754.aspx).

e. **Web Instances**—Choose the number of servers on which the load will be balanced.
   - 1—no load balancing.
   - more than 1—load balance equally among the number of servers you choose.

f. **Worker Virtual Machine Size**—Choose **Small** (default), **Medium**, **Large**, or **Extra Large** for your windows service virtual machine. You can have only 1 instance of this VM.

g. **Storage Account**—Specify the name of the storage account that you created in *Create a Storage Account* on page 8.

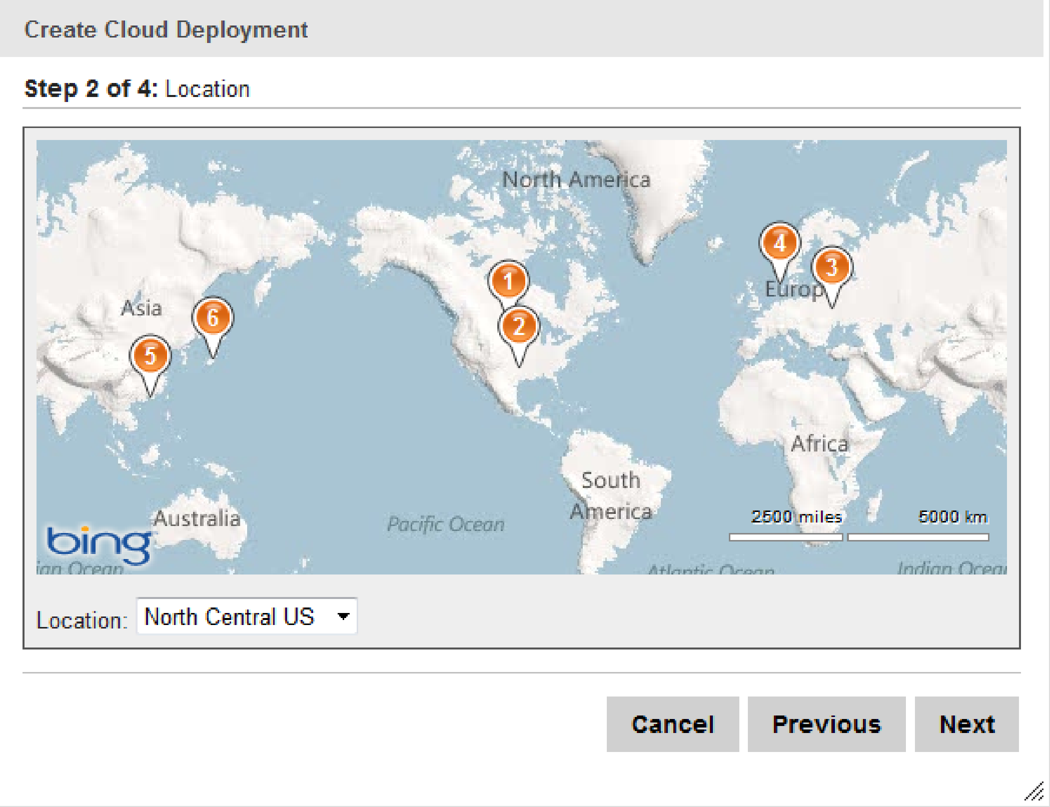
h. **CDN**—Specify the URL with the Content Delivery Network (CDN). You can get this from http://windows.azure.com.
   1. Click **Hosted Services, Storage Accounts and CDN**.
   2. Click the **CDN** folder.
   3. Click on the server name with a CDN endpoint type.

4. Copy and paste the **Default HTTP Endpoint** from the Properties panel. Be sure to start the URL with `http://` and place a trailing slash (`/`) on the end.

i. **License Key**—

3. Click **Next**.
4. Click a number (or select the location from the Location field), then click **Next.**

NOTE: Step 2 of 4 of the wizard may show only the Location field (no map), unless you already defined the following *Bingmap* key value in `web.config` before starting this wizard.
`<add key="BingMapCredentials" value="YourValue" />`

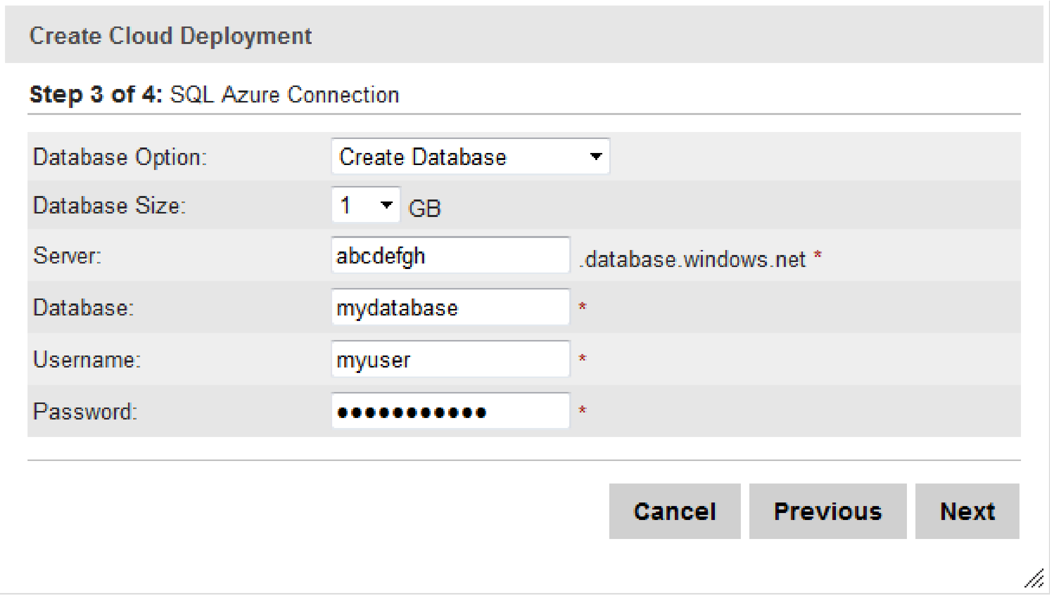


Create Cloud Deployment

Step 2 of 4: Location

Location: North Central US

5. Specify the fields on the Azure SQL Database Connection dialog box. Fields with a red asterisk (*) are required.


Create Cloud Deployment

Step 3 of 4: SQL Azure Connection

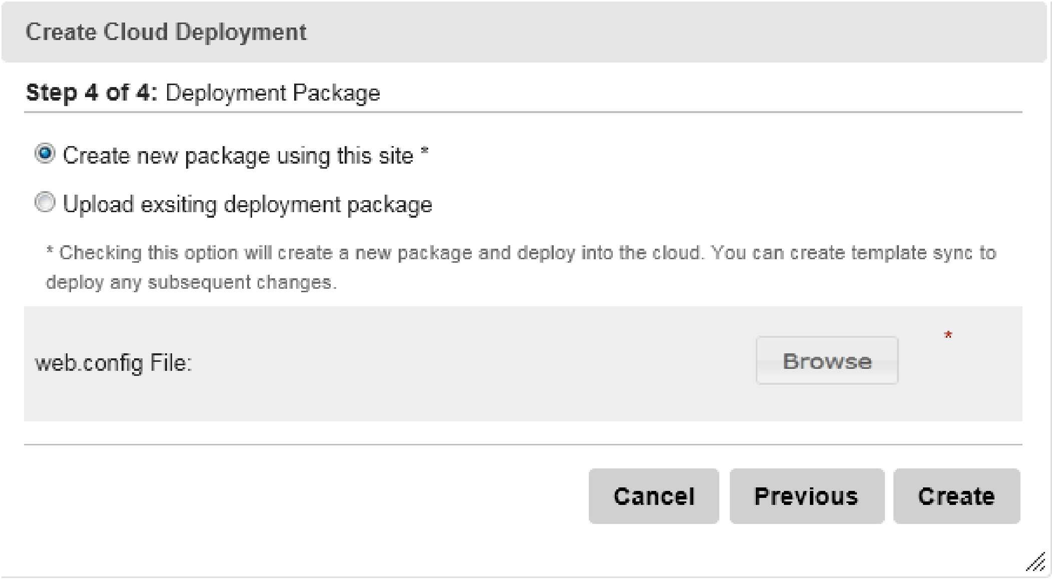
Database Option:	Create Database
Database Size:	1 GB
Server:	abcdefgh .database.windows.net *
Database:	mydatabase *
Username:	myuser *
Password:	●●●●●●●●●●● *

- **Database Option—Create SQL Database** is the default database option. If you already have your database set up, choose **Do not create SQL Database**.
- **Database Size**—Select the size of your database.
- **SQL Database Server**—You can get the server name from the http://windows.azure.com portal: click **Database** and look under the Subscriptions folder for the server name.
- **Database Name**—Specify the name of the database.
- **Administrator Login Name**—Create the database administrator account name.
- **Password**—Specify the password for the account.

6. Choose to create a new package or upload an existing package to deploy to the cloud. The newly created deployment package is stored at:

```
~\sync\azuredeploy\{subscription-
id}\{servicename}\package\{servicename}.cspkg
```

   - If you choose to create a new package, you can optionally choose another configuration file or click **Create** to use the default.



Create Cloud Deployment

Step 4 of 4: Deployment Package

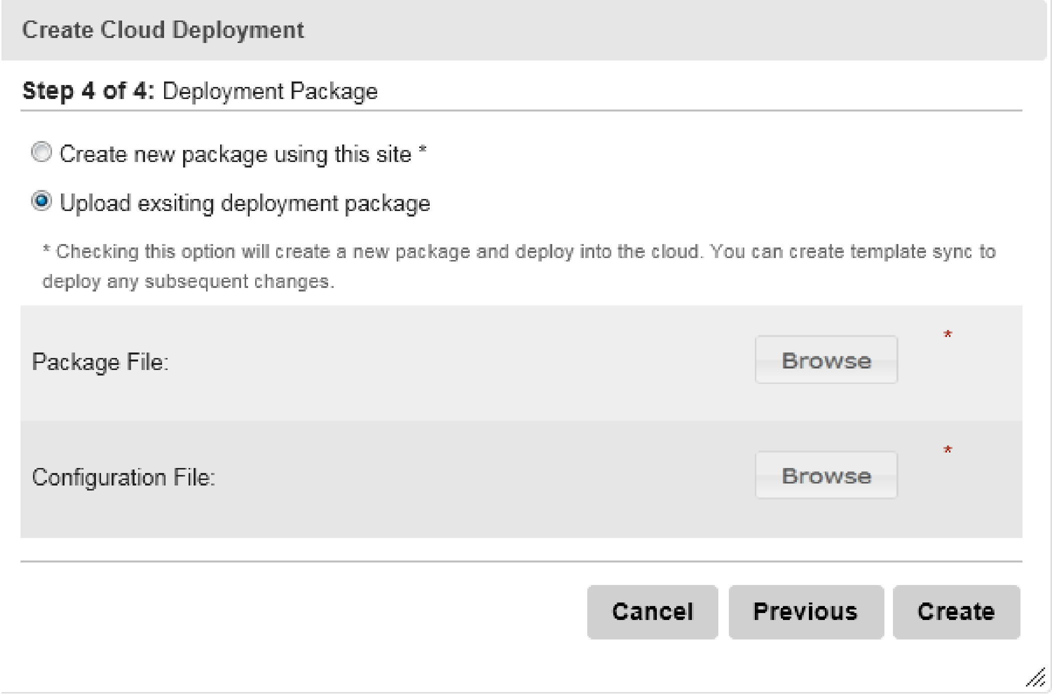
● Create new package using this site *

○ Upload exsiting deployment package

* Checking this option will create a new package and deploy into the cloud. You can create template sync to deploy any subsequent changes.

web.config File: Browse *

- If you choose to upload your files, click the **Browse** buttons to select the files.



Create Cloud Deployment

Step 4 of 4: Deployment Package

○ Create new package using this site *

● Upload exsiting deployment package

* Checking this option will create a new package and deploy into the cloud. You can create template sync to deploy any subsequent changes.

Package File: Browse *

Configuration File: Browse *

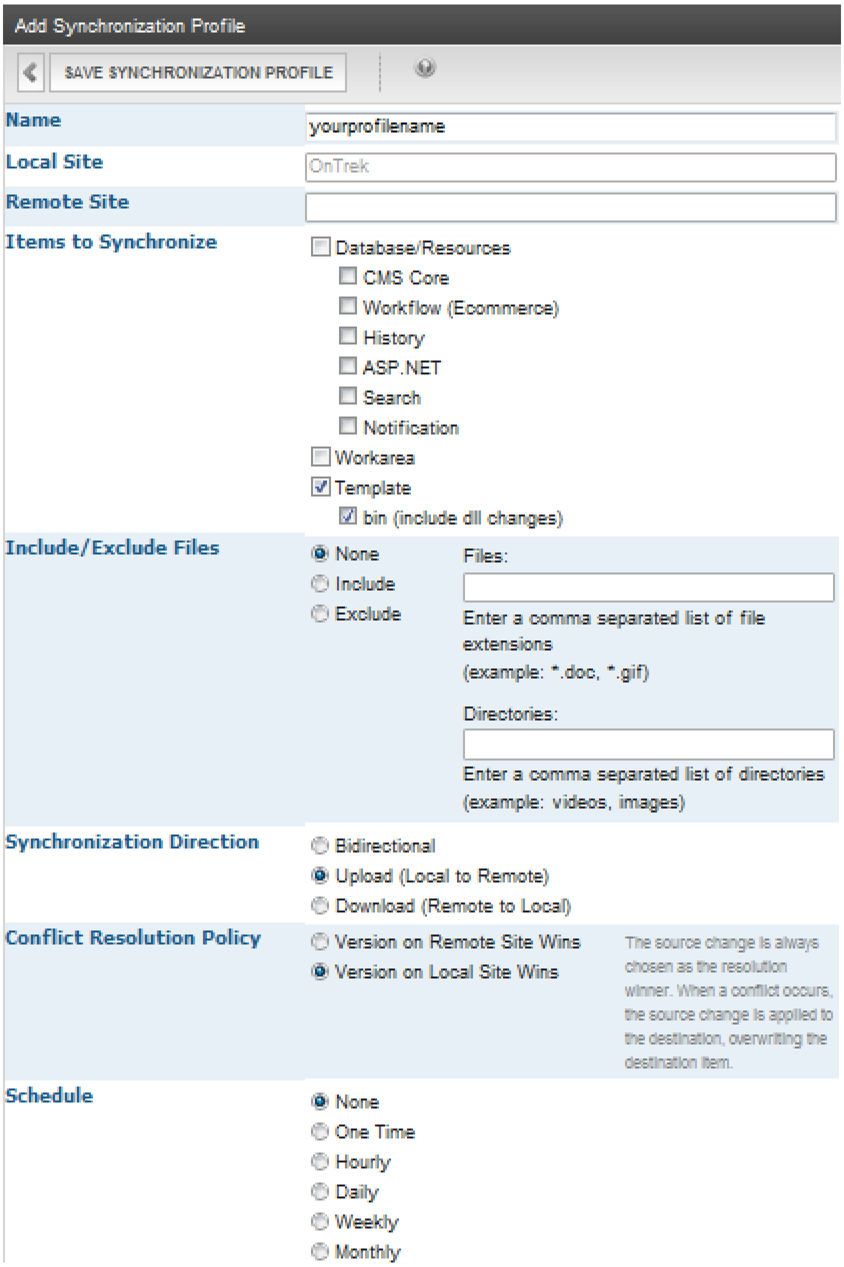
An advanced user can copy the `Web.Cloud.config` file from Ektron Windows service (`~\Program Files (x86)\Ektron\CMS400v85\AzureConfig`).

7. When you click **Next**, Ektron creates a new min database, provisions a storage container, creates the deployment package, and deploys them into a hosted cloud service. A confirmation message appears.
8. Click **Finish** to close the wizard.

Synchronize the Database and Templates

To synchronize the database and templates:

1. In the Workarea, choose **Settings > Configuration > Windows Azure Deployment**. The view panel displays the instances of databases.
2. Click **Add Profile** (⊕) to create a profile for the database and template. The Add Synchronization Profile screen appears.



Add Synchronization Profile

SAVE SYNCHRONIZATION PROFILE

Name	yourprofilename
Local Site	OnTrek
Remote Site	
Items to Synchronize	□ Database/Resources □ CMS Core □ Workflow (Ecommerce) □ History □ ASP.NET □ Search □ Notification □ Workarea ☑ Template ☑ bin (include dll changes)
Include/Exclude Files	● None ○ Include ○ Exclude Files: Enter a comma separated list of file extensions (example: *.doc, *.gif) Directories: Enter a comma separated list of directories (example: videos, images)
Synchronization Direction	○ Bidirectional ● Upload (Local to Remote) ○ Download (Remote to Local)
Conflict Resolution Policy	○ Version on Remote Site Wins ● Version on Local Site Wins The source change is always chosen as the resolution winner. When a conflict occurs, the source change is applied to the destination, overwriting the destination item.
Schedule	● None ○ One Time ○ Hourly ○ Daily ○ Weekly ○ Monthly

3. Specify a name, uncheck **Database/Resources** and **CMS Core**, and check **Template** and **bin**.
4. Click **Save Synchronization Profile**. The profile is saved and the View All Cloud Deployment screen appears.
5. In the Action section of the profile you just saved, click **Sync** (🔄) to synchronize the database and templates.

Manage the Cloud Instances

In the View All Cloud Deployment screen (**Workarea > Settings > Configuration > Windows Azure Deployment**), you can modify cloud instances in the following ways:

- ▶ **Start the instance**—If the icon is green, the instance is stopped; click to start it.
- ● **Stop the instance**—If the icon is red, the instance is running; click to stop it.
- ⚙ **Configure the instance**—Click this icon to modify the number of instances. One (1) instance means no load balancing; more than 1 instance balances the load equally among instances.
- ⇄ **Swap the instance**—If you have a production and staging server, click to swap the instance (to staging and production).
- ⊖**Remove the instance**—Removes the instance from the list.

Configure Ektron Storage

Ektron uses the `unity` section of your cloud `web.config` file to configure a storage destination. Follow these steps:

1. Add the following to the `configSections` section:

```
<section  name="unity.storage"
  type="Ektron.Unity.Configuration.UnityConfigurationSection,
  Ektron.Unity.Configuration, Version=2.0.414.0, Culture=neutral,
  PublicKeyToken=1e97f90fc2b998db"/>
```

2. Define `unity.storage` as follows:

```
<unity.storage>
  <assembly name="Ektron.Storage"/>
  <assembly name="Ektron.Azure"/>
  <namespace name="Ektron.Azure.Storage"/>
  <namespace name="Ektron.Storage"/>
  <container name="storageContainer">
    <register type="IFileService" mapTo="CloudFileService"/>
    <register type="IDirectoryService"
       mapTo="CloudDirectoryService"/>
  </container>
</unity.storage>
```

Configure Ektron Caching

NOTE: Caching configuration is required only if you run more than 1 instance (load balancing). You can get the section values from the portal.

In your cloud `web.config` file, do the following:

1. Add the following to the `configSections` section:

```
<section name="dataCacheClients"
type="Microsoft.ApplicationServer.Caching.DataCacheClientsSection,
  Microsoft.ApplicationServer.Caching.Core"
  allowLocation="true" allowDefinition="Everywhere"/>
```

2. Define `dataCacheClients` as follows:

```
<dataCacheClients>
  <dataCacheClient name="default">
    <host><host name="[yournamespace].cache.windows.net"
```

```
      cacheport="22233" /></hosts>
    <securityProperties mode="Message">
      <messageSecurity authorizationInfo="[yoursecretkeyinfo]"/>
    </securityProperties>
  </dataCacheClient>
  <dataCacheClient name="SslEndpoint">
    <host>
      <host name="ektron.cache.windows.net" cacheport="22243"/>
    </hosts>
    <securityProperties mode="Message" sslEnbaled="true">
      <messageSecurity authorizationInfo="[yoursecretkeyinfo]"/>
    </securityProperties>
  </dataCacheClient>
</dataCacheClients>
```

3. Configure the sessions to use Appfabric server:

```
<sessionState mode="Custom"
  customProvider="AppFabricCacheSessionStoreProvider">
  <providers>
  <add name="AppFabricCacheSessionStoreProvider"
    type="Microsoft.Web.DistributedCache.
      DistributedCacheSessionStateStoreProvider,
      Microsoft.Web.DistributedCache"
      cacheName="default"
      useBlobMode="true"
      dataCacheClientName="default" />
  </providers>
</sessionState>
```

Set the SQL Azure Connection String

In your cloud `web.config` file, modify the `Ektron.DbConnection` section to have an SQL Azure connection string. You can get this value from the portal.

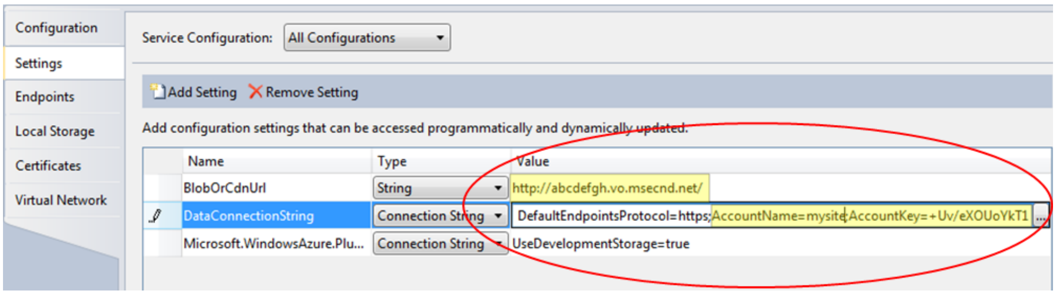
Set the BLOB Redirect Handler

In your cloud `web.config` file, insert the BLOB redirect handler into the `module` section:

```
<add name="EkBlobModule"
  type="[assemblyname].BlobRedirect.Module,[assemblyname]"
  precondition="integratorMode"/>
```

Create a Cloud Package

1. Open the cloud solution from Ektron Developer's SDK.
2. Configure the storage account details in the service configuration file. Use *Visual Studio 2010* (or edit the XML-based configuration file) to have the following details:



Service Configuration: All Configurations

Add configuration settings that can be accessed programmatically and dynamically updated.

Name	Type	Value
BlobOrCdnUrl	String	http://abcdefgh.vo.msecnd.net/
DataConnectionString	Connection String	DefaultEndpointsProtocol=https;AccountName=mysite;AccountKey=+Uv/eXOUoYkT1
Microsoft.WindowsAzure.Plu...	Connection String	UseDevelopmentStorage=true

3. Compile the project with Visual Studio 2010, or use the `msbuild` command as follows:

   ```
   msbuild yourcloudproject.ccproj /p:Configuration=Release
   /t:Build;Publish
   ```

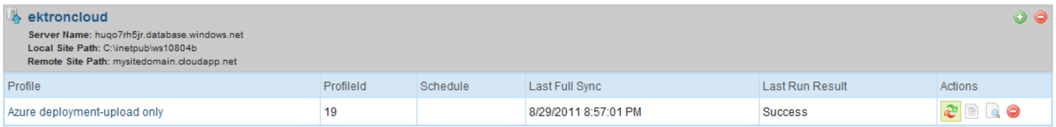
4. Log into the portal and upload *`yourprojectname`*`.cspkg` and `ServiceConfiguration.Cloud.cscfg` into your newly created hosted service.

NOTE: After you create the deployment, the set up on the cloud takes some time before the files are ready to be accessed.

Configuring eSync to a Cloud Instance

Ektron's sync technology, built on the Microsoft Sync Framework, synchronizes information between the Azure cloud and On Premises data repositories. Ektron eSync makes it easy to manage content changes through a true hybrid cloud deployment.

When you run eSync. SQL data and file system resources (such as `~/assets`, `~/privateassets`, `~/uploadedfiles`, `~/uploadedimages`, `assetlibrary`) are synched to SQL Azure Cloud BLOB Storage .



ektroncloud

Server Name: huqo7rh5jr.database.windows.net
Local Site Path: C:\inetpub\ws10804b
Remote Site Path: mysitedomain.cloudapp.net

Profile	ProfileId	Schedule	Last Full Sync	Last Run Result	Actions
Azure deployment-upload only	19		8/29/2011 8:57:01 PM	Success	

Connecting to an On Premises Search Server

To create a search in the Windows Azure Cloud environment, use Ektron's hybrid deployment model. Search requests are sent to your pre-production server, which is identical to your cloud server. You make changes on the pre-production server, and the Cloud server displays the changes. To set up a search in the Windows Azure Cloud environment:

1. In your On Premises site root's `web.config` file, modify the following settings:

   ```
   <extensions>
     <behaviorExtensions>
       <add name="ServiceRegistrySettings"
         type="Ektron.Azure.ServiceRegistry.
           ServiceRegistrySettingsElement, Ektron.Azure" />
     </behaviorExtensions>
   </extensions>
   ```

2. In your On Premises site root's `web.config` file, add the endpoint under `Ektron.Cms.Search.SearchManager`:

```
<endpoint address="sb://yourservicebus.servicebus.
  windows.net/SearchNetTcp/"
  behaviorConfiguration="sharedSecretClientCredentials"
  binding="netTcpRelayBinding"
  bindingConfiguration="NetTcpRelayEndpointConfig"
  name="RelayEndpoint"
  contract="Ektron.Cms.Search.ISearchManager" />
<netTcpRelayBinding>
  <binding name="NetTcpRelayEndpointConfig">
    <security relayClientAuthenticationType="RelayAccessToken" />
  </binding>
</netTcpRelayBinding>
<endpointBehaviors>
  <behavior name="sharedSecretClientCredentials">
    <transportClientEndpointBehavior
      credentialType="SharedSecret">
      <clientCredentials>
        <sharedSecret issuerName="owner"
          issuerSecret="yourissuersecret" />
      </clientCredentials>
    </transportClientEndpointBehavior>
    <ServiceRegistrySettings discoveryMode="Public" />
  </behavior>
</endpointBehaviors>
```

3. In your cloud `web.config` file, change the `appsettings, extension and binding` section:

```
<add key="ek_servicespath"
  value ="http://yourserver/workarea/services"/>
<add key="serviceBusNamespaceDomain" value="yourdomain"/>
<add key="serviceBusIssuerName" value="owner"/>
<add key="serviceBusIssuerSecret" value="yoursecret"/>
<add key="serviceBusPath" value="SearchNetTcp"/>
<extensions>
  <behaviorExtensions>
    <add name="ServiceRegistrySettings"
      type="Ektron.Azure.ServiceRegistry.
        ServiceRegistrySettingsElement,
        Ektron.Azure" />
  </behaviorExtensions>
  <bindingExtensions>
    <add name="netTcpRelayBinding"
      type="Microsoft.ServiceBus.Configuration.
        NetTcpRelayBindingCollectionElement,
        Microsoft.ServiceBus, Version=1.0.0.0, Culture=neutral,
        PublicKeyToken=31bf3856ad364e35"/>
  </bindingExtensions>
</extensions>
<netTcpRelayBinding>
  <binding name="RelayEndpoint" closeTimeout="00:01:00"
    openTimeout="00:01:00" receiveTimeout="00:10:00"
    sendTimeout="00:01:00" transferMode="Buffered"
    connectionMode="Relayed" listenBacklog="10"
    maxBufferPoolSize="524288" maxBufferSize="65536"
    maxConnections="10" maxReceivedMessageSize="65536">
    <readerQuotas maxDepth="32" maxStringContentLength="8192"
      maxArrayLength="16384"
      maxBytesPerRead="4096"
      maxNameTableCharCount="16384" />
    <reliableSession ordered="true" inactivityTimeout="00:10:00"
      enabled="false" />
    <security mode="Transport"
      relayClientAuthenticationType="RelayAccessToken">
      <transport protectionLevel="EncryptAndSign" />
      <message clientCredentialType="Windows" />
    </security>
  </binding>
</netTcpRelayBinding>
```

Getting Support

Ektron has resources to help you.

ektron.com	Ektron's home page has useful information, including additional contact information and services. You can find product documentation at http://www.ektron.com/Resources/Product-Documentation/.
The Dev Center	The Ektron Dev Center (http://dev.ektron.com) has resources to help you find a solution to a problem or see what you can do with CMS400.NET. The Dev Center also has Webinars for many of the features of Ektron CMS400.NET. In the product forum, you can post tips or ask other users about how they use Ektron products more effectively.
Professional Services	Ektron Professional Services (http://www.ektron.com/services/) is a complete services infrastructure that is customized to meet your specific needs.
Implementation	Using Ektron's proven Implementation (http://www.ektron.com/Solutions/Services/Implementation/), expert application engineers can implement all or part of your web site project,ensuring the success of your project.
Training	Ektron's Training Services (http://www.ektron.com/Solutions/services/training/) ensures your web project’s success by providing the right level of training to your team at the right stage of your web project’s lifecycle.
Synergy	The annual *Synergy Customer Conference* (http://synergy.ektron.com) is a great way for you to learn more about using your Ektron CMS400.NET and meet other Ektron users.
Local User Groups	Ektron management meets with Local User Groups (LUG) to bring the latest news and products from Ektron. See http://usergroups.ektron.com/ for location and dates.
Support	**Customer Support** (http://www.ektron.com/Resources/Support/) United States: 1-603-594-0249 x7002 Europe: +44 (0) 1628 509 040 Australia: +61 2 9248 7222 For additional contact information, see http://www.ektron.com/About-Ektron/Contact-Us/.