Deploy Weather Service on Azure

2015/1/7 NEW

# Introduction

Computing the forecasts is a CPU intensive task. The service makes use of the industry standard MPI library to distribute the computation across processors, and it utilizes the new Windows HPC Pack to manage the distribution of executing forecast jobs across the HPC Cluster. Azure Blob and Table storage are used to keep track of jobs and store intermediate and final results.

This document demonstrates how to deploy the Weather Service and the HPC Cluster on Azure.

In this topic:

* [Prerequisites](#_Prerequisites)
* [Create a HPC Cluster](#_Create_a_HPC)
* [Configure Weather Service Website](#_Configure_Weather_Service)
* [Configure Weather Service on HPC HeadNode](#_Configure_Weather_Service_1)
* [Configure Weather Service for HPC ComputeNode](#_Configure_Weather_Service_2)
* [Run a Weather Forecasting job](#_Run_a_Weather)
* [Auto grow & shrink](#_Auto_grow_&)

# Prerequisites

The following things are required before we start.

**● Microsoft Azure subscription**

You must obtain or have access to a Microsoft Azure subscription.

* To create a Microsoft Azure subscription, go to the [Microsoft Azure](http://www.windowsazure.com/) site.
* To access an existing subscription, go to the [Microsoft Azure Management Portal](https://manage.windowsazure.com/).

**● A storage account**

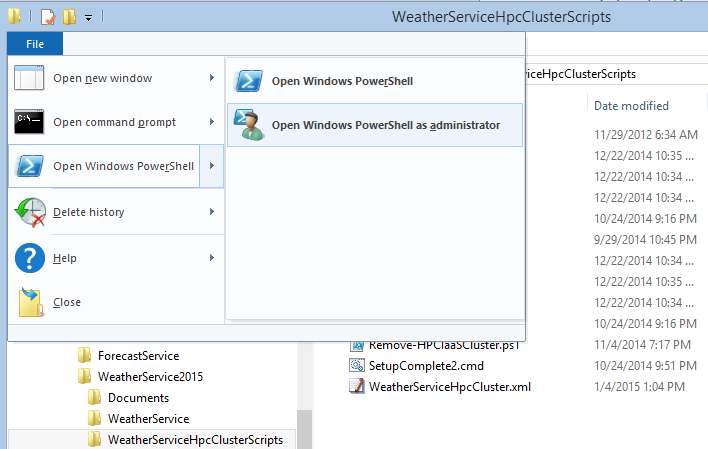
The storage account will be used to store your weather data which is created & used by the Weather Service.

**● Tools and scripts**

* You can download this document from [GitHub](https://github.com/coolmay/weatherservice).
* You can download all scripts for creating the HPC Cluster from [here](http://dannyea.blob.core.windows.net/weatherservice/WeatherServiceHpcClusterScripts.zip).
* You can download all tools and scripts used by Weather Service from [here](http://dannyea.blob.core.windows.net/weatherservice/WeatherService.zip).

# Create a HPC cluster

We will deploy a HPC Cluster on Azure by using the HPC IaaSClusterScript which is released with Microsoft HPC Pack 2012 R2 Update 1.

1. Open the folder *WeatherServiceHpcClusterScripts* in File Explorer and open a Windows PowerShell client as administrator.  
   
2. Type the following cmdlet in the PowerShell client. It will lead you to the Azure publishsettings file download page. And download the publishsettings file to your disk with the credentials for your Azure account.

Get-AzurePublishSettingsFile

1. Import the publishsettings file.

Import-AzurePublishSettingsFile <publishsettings-filepath>

1. Type the following cmdlet to create a new HPC Cluster.  
     
   **ConfigFile:** Specifies the file path of the configuration file which describes the HPC cluster.  
   **AdminUserName:** Specifies the user name. It is the local administrator user name for all VMs as well as the domain administrator user name if the domain forest is created by the script. If the domain forest already exists, specify the domain user name to install HPC Pack.  
   **AdminPassword:** Specifies the administrator’s password. If not specified in the command line, the script will prompt the user to input the password during execution.  
   **LogFile**: (optional) Specifies the deployment log file path. If not specified, the script will create a log file in the user’s temp directory.  
   **Force: (**optional)Suppresses all the confirmation prompts if specified.  
     
   Note:   
   you can find more details about this cmdlet in Manual.rtf.

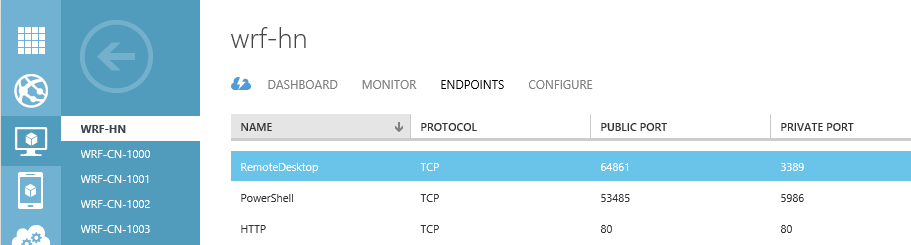
New-HPCIaaSCluster.ps1 -ConfigFile “.\WeatherServiceHpcCluster.xml” -AdminUserName “<adminname>” -AdminPassword “<password>” -LogFile “.\deploy.log” -Force

1. Wait till the script completes. You will see some information like below.  
     
   Write down the HPC compute node image name. We will use it in later steps.

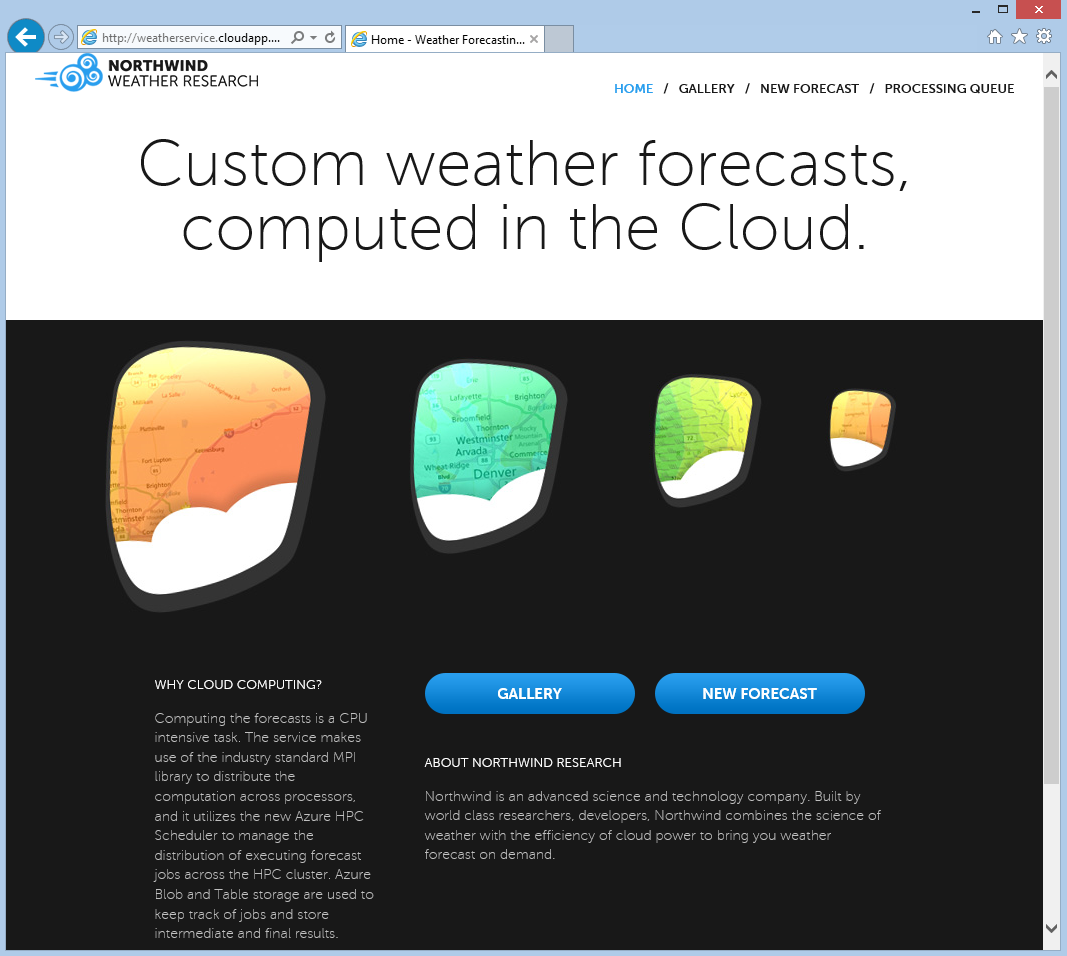
The HPC cluster deployment completed. Total cost time: 36.56 minutes.  
You can use the image name(s) as below to add more HPC Pack nodes later:  
HPC compute node image : HPC-Pack-ComputeNode\_20141229-160427.7541495

# Configure Weather Service Website

A website is included in the Weather Service System to show weather forecast information to users. In the scenario of this document, we will deploy the website on HPC HeadNode. Alternatively, you can deploy the website on a separated VM or as an Azure website.

1. Log into the [Microsoft Azure Management Portal](https://manage.windowsazure.com/).
2. Select the HeadNode VM from **VIRTUAL MACHINES** tab.
3. Add a HTTP 80 endpoint to the HeadNode.  
   
4. Connect to your HeadNode with Remote Desktop Connection.
5. Copy the entire folder *WeatherService* to *C:\* on HeadNode. The website is under folder *WeatherService\WeatherServiceWebSite.*
6. Navigate to *C:\WeatherService\WeatherServiceWebSite* on HeadNode.
7. Set desired value for **ConnectionString** in *Web.config.*  
   Note:   
   The weather-data-storage is the storage account you prepared for Weather Service to store weather data in [Prerequisites](#_Prerequisites).

<add key="Microsoft.WindowsAzure.WFTable.ConnectionString" value="DefaultEndpointsProtocol=https;AccountName=<weather-data-storage-account>;AccountKey=<weather-data-storage-accountkey>"/>

1. Setup a Website binding to *C:\WeatherService\WeatherServiceWebSite* and start the Website in IIS Manager.
2. You can visit the Weather Service Website through *http://<headnode-cloudservice-name>.cloudapp.net*.  
   

# Configure Weather Service on HPC HeadNode

In this section, we will do some configuration of the Weather Service on the HPC HeadNode.

1. Connect to your HeadNode with Remote Desktop Connection if you haven’t done so.
2. Copy the entire folder *WeatherService* to *C:\* on HeadNode if you haven’t done so.
3. Navigate to *C:\WeatherService\WeatherServiceOnPremisesScripts* on HeadNode.
4. Set desired values in *SetupServer.bat* & *SetupClient.bat.  
   SetupServer.bat  
     
   SetupClient.bat*

rem The storage account you prepared for Weather Service to store weather data.

set WEATHER\_ConnectionString= DefaultEndpointsProtocol=https;AccountName=<weather-data-storage-account>;AccountKey=<weather-data-storage-accountkey>

rem The folder path to store trace files.

set WEATHER\_TraceFiles=C:\WeatherServiceTraceFiles

rem The folder path to store data files.

set WEATHER\_DataFiles=C:\WeatherServiceDataFiles

rem The folder path to store forecast files.

set WEATHER\_ForecastFiles=C:\WeatherServiceForecastFiles

rem The folder path of Weather Service.

set WEATHER\_AppRoot=C:\WeatherService

rem The HPC Cluster name.

set WEATHER\_ClusterName=localhost

rem The user of the HPC Cluster who will submit weather forecast jobs.

set WEATHER\_AdminAccount=<adminname>

rem The password for the user.

set WEATHER\_AdminPassword=<password>

rem The storage account you prepared for Weather Service to store weather data.

set WEATHER\_ConnectionString= DefaultEndpointsProtocol=https;AccountName=<weather-data-storage-account>;AccountKey=<weather-data-storage-accountkey>

rem The folder path to store trace files.

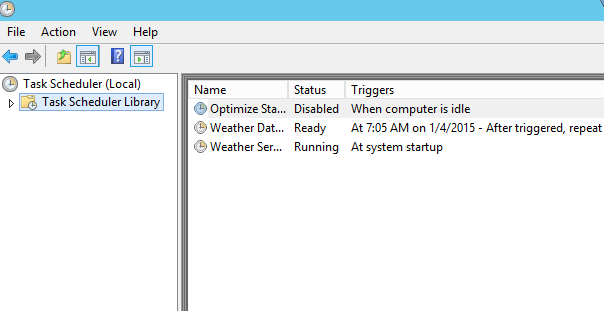
set WEATHER\_TraceFiles=C:\WeatherServiceTraceFiles

rem The folder path to store data files.

set WEATHER\_DataFiles=C:\WeatherServiceDataFiles

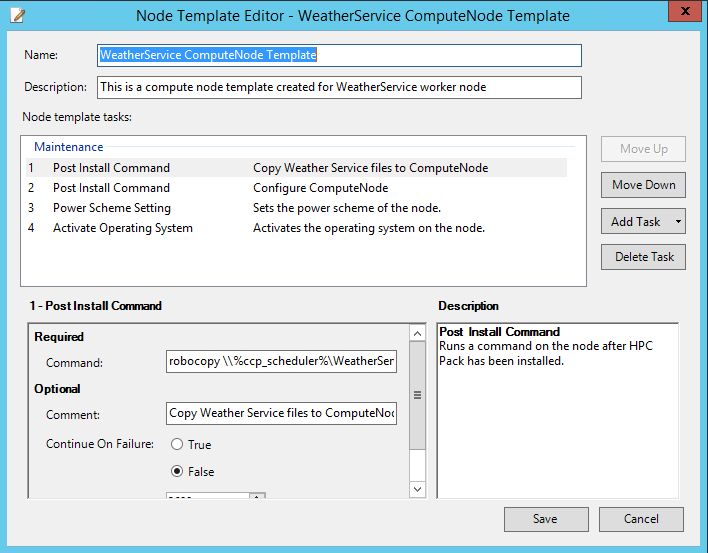
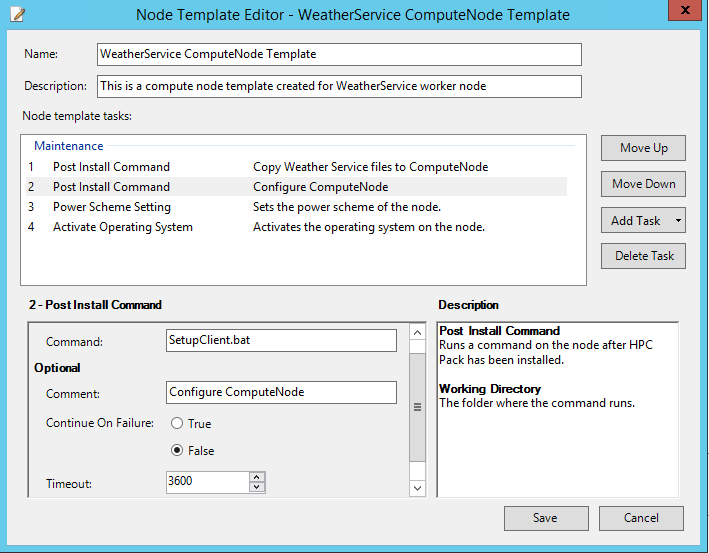
rem The folder path to store forecast files.

set WEATHER\_ForecastFiles=C:\WeatherServiceForecastFiles

1. Share the folder *C:\WeatherService* to Everyone. The ComputeNodes need to access this folder to get Weather Service files.
2. Run *SetupServer.bat.*
3. Wait till the script completes. In Task Scheduler, you can see 2 new tasks have been created – Weather Data Preprocess & Weather Service.  
   

# Configure Weather Service for HPC ComputeNode

We will configure the ComputeNodes automatically by creating a ComputeNode Template in HPC Cluster Manager.

1. Connect to your HeadNode with Remote Desktop Connection if you haven’t done so.
2. Open HPC Cluster Manager. In **Configuration** tab, select **Node Templates** and create a new ComputeNode Template (Without operating system).
3. Double-click the template to open the Node Template Editor.
4. Add 2 Post Install Command Task to the template.  
   For the first task, enter the following content:  
     
     
   For the second task, enter the following content:  
     
   

**Command**:   
robocopy \\%ccp\_scheduler%\WeatherService c:\WeatherService /E

**Timeout**:   
3600

**Command**:   
SetupClient.bat

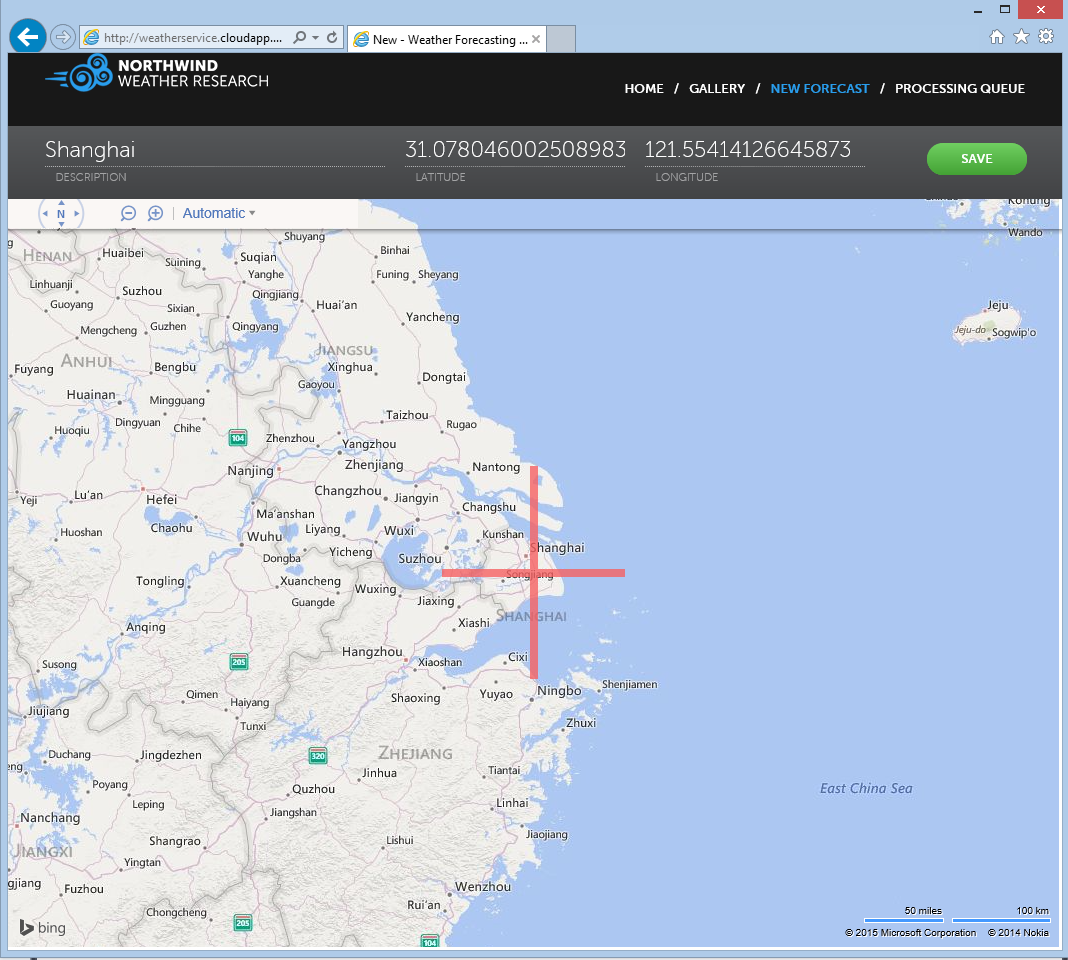
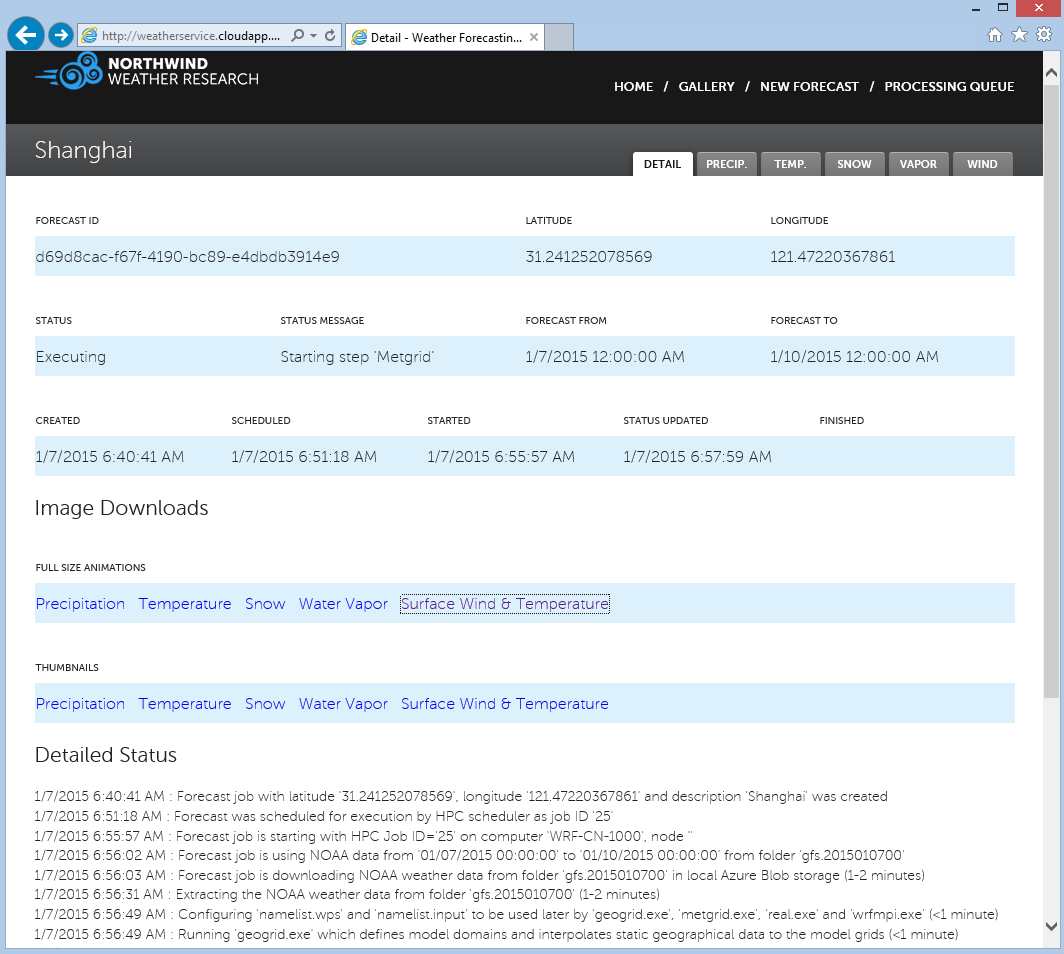
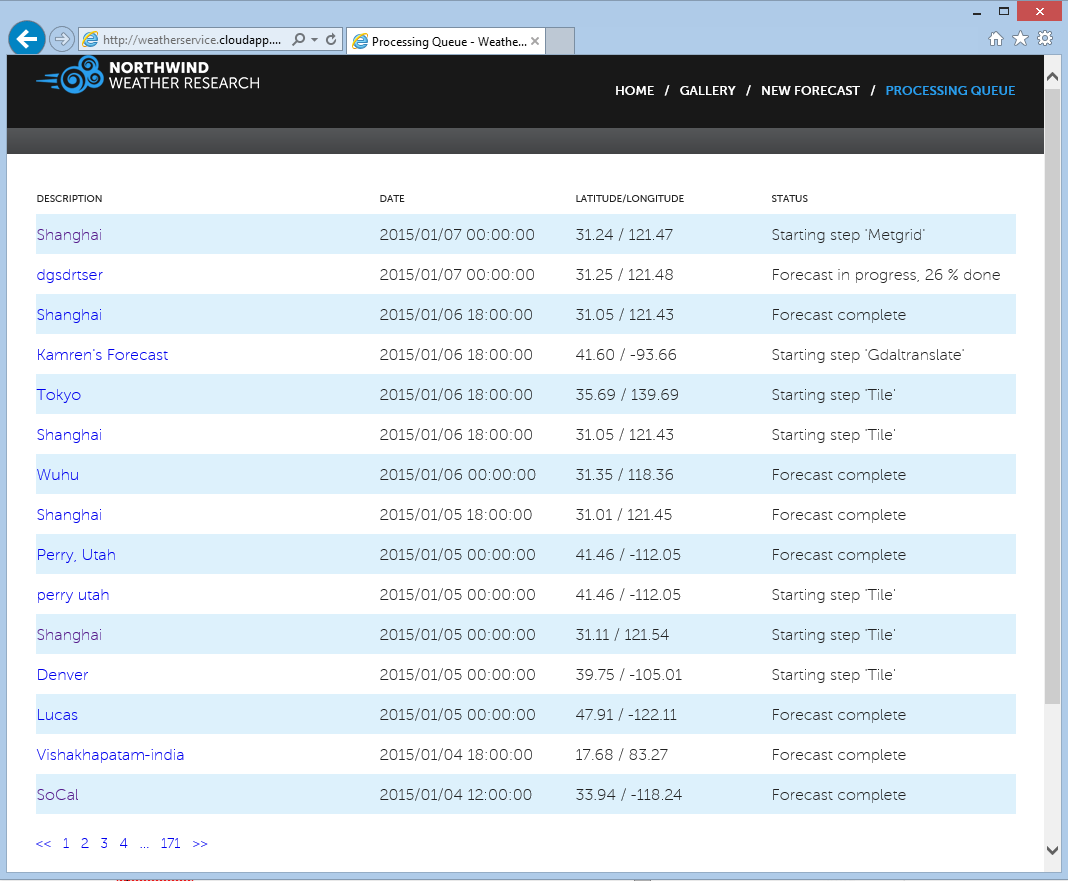
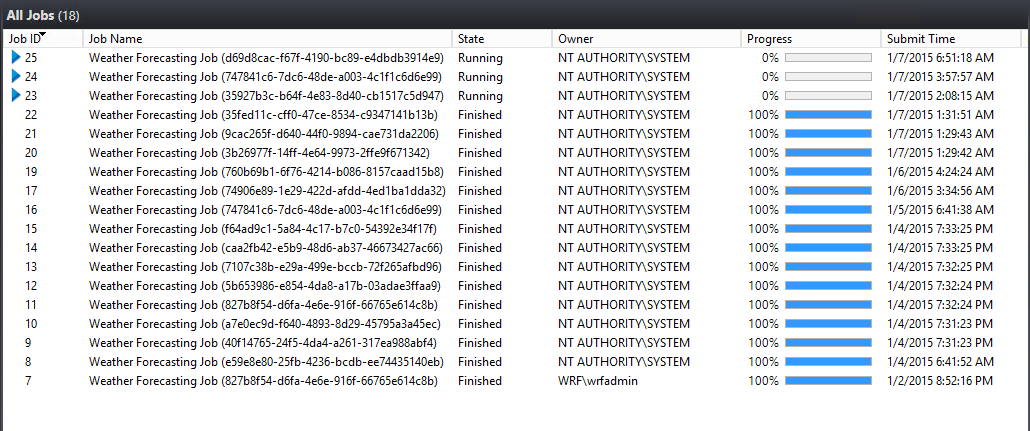
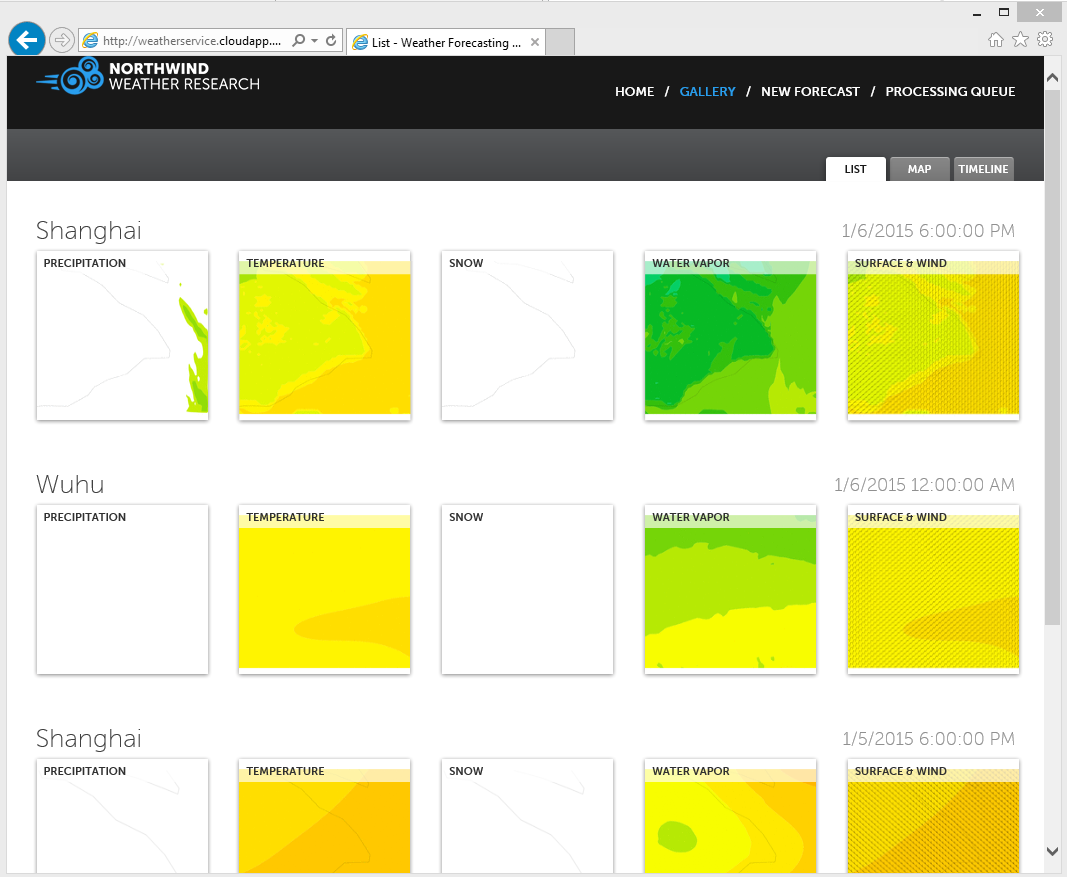
**Timeout**:   
3600

**Working Directory**:   
C:\WeatherService\WeatherServiceOnPremisesScripts

1. In **Node Management** tab, select the compute nodes and assign the node template to them.
2. Bring the compute nodes online when they are ready.

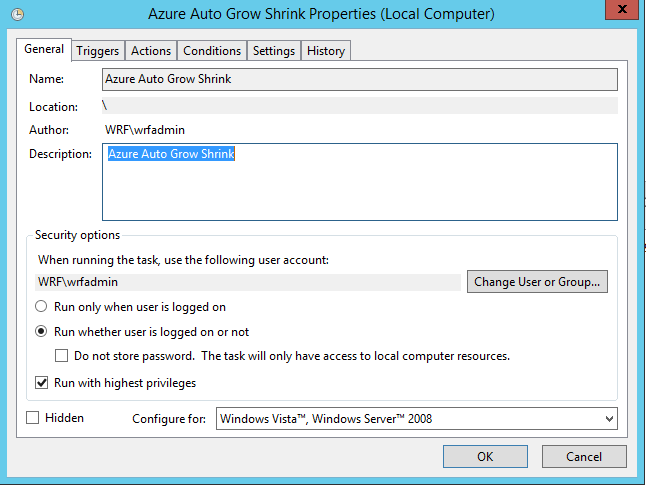
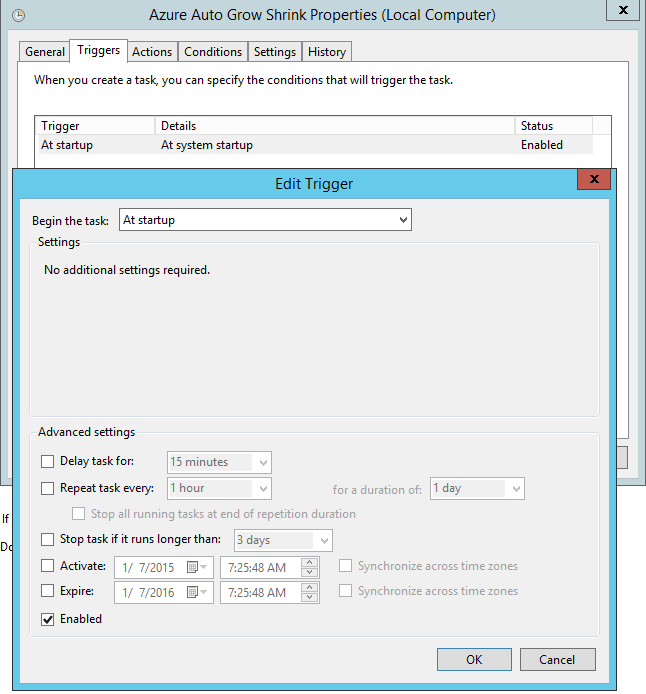
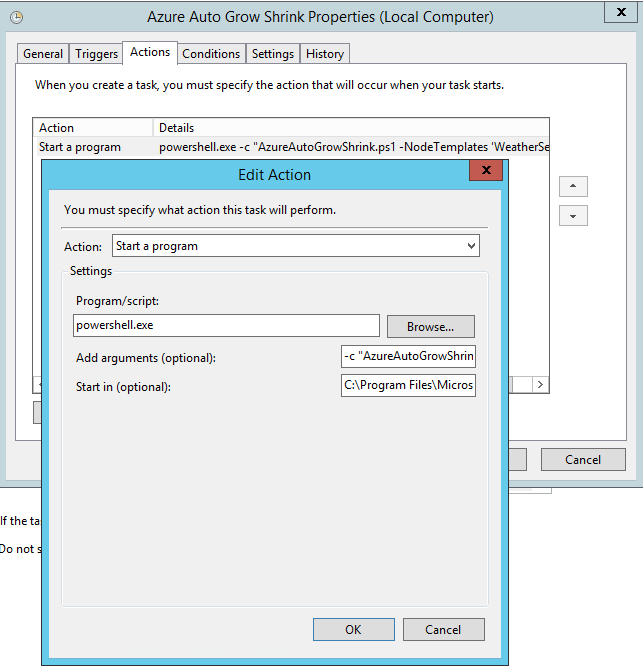
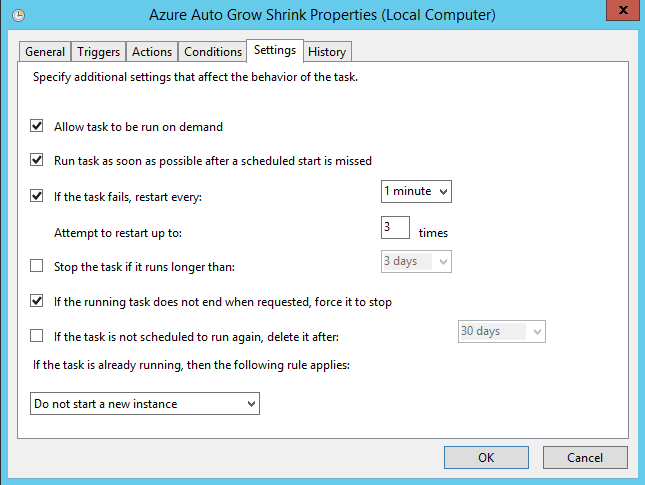
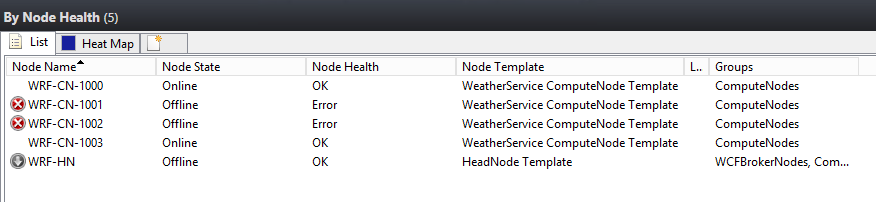
# Run a Weather Forecasting job

The deployment of the Weather Service System is all completed. Let’s run a weather forecasting job on it.

1. Visit the Weather Service Website through *http://<headnode-cloudservice-name>.cloudapp.net*.
2. Click **NEW FORECAST** on the top-right menu bar. Select a location on the map and enter a description for it, then click **SAVE**.  
   
3. You can see a new forecasting job is created and scheduled for execution by HPC Cluster.  
   
4. You can check the processing status of the forecasting job any time by clicking **PROCESSING QUEUE** on the top-right menu bar.   
     
   And you can check them in HPC Cluster Manager as well.  
   
5. You can see your job results in **GALLERY** tab.  
   

# Auto grow & shrink

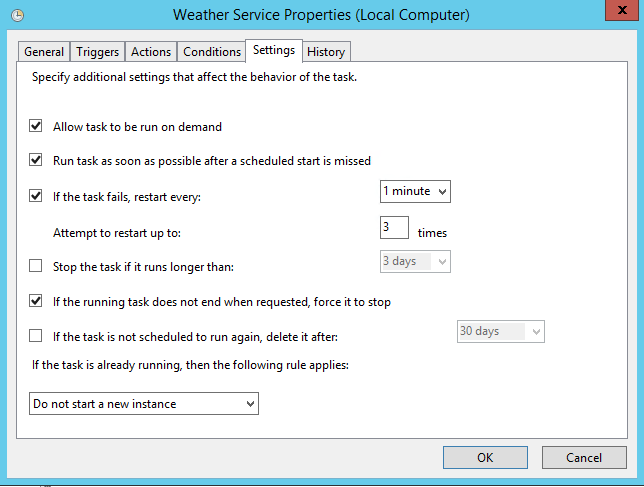
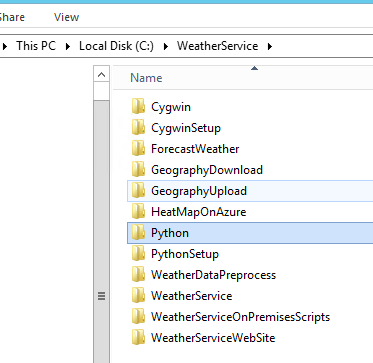
Auto grow & shrink is a new feature in HPC Pack 2012 R2 Update 1, which is used to automatically grow and shrink PaaS Azure nodes and IaaS Azure VMs in a Microsoft HPC Pack cluster based on the workload (jobs/tasks) in the cluster. We can use this feature in the Weather Service System to save money when there are less running forecasting jobs.

1. Connect to your HeadNode with Remote Desktop Connection.
2. Open Task Scheduler and create a new Task.
3. Select **Run whether user is logged on or not**.  
   
4. Add a trigger: **At startup**.  
   
5. Add an action.  
   **Action**: Start a program  
   **Program/script**: powershell.exe  
   **Add arguments**: -c "AzureAutoGrowShrink.ps1 -NodeTemplates 'WeatherService ComputeNode Template' -NodeType ComputeNodes -NumOfQueuedJobsPerNodeToGrow 1 -NumOfQueuedJobsToGrowThreshold 0"  
   **Start in**: C:\Program Files\Microsoft HPC Pack 2012\Bin  
   
6. Unselect **Stop the task if it runs longer than** checkbox.  
   
7. Click **OK** and run it.
8. The script will automatically shut down compute nodes when there is no job running on them and restart the compute nodes when there are queued jobs.  
   

Note:

For more information on Auto Grow and Shrink Azure Nodes and IaaS VMs, please refer to [The Microsoft HPC & Batch Team Blog](http://blogs.technet.com/b/windowshpc/archive/2014/12/02/automating-hpc-cluster-deployments-in-azure-iaas-part-iii-auto-grow-and-shrink-azure-nodes-and-iaas-vms.aspx).

# Known Issues

1. The scheduled tasks created by *SetupServer.bat* on HeadNode in [Configure Weather Service on HPC HeadNode](#_Configure_Weather_Service_1) will be stopped by the scheduler in 3 days by default. To make these tasks to keep running, unselect the **Stop the task if it runs longer than** checkbox in the task properties manually.  
   
2. The ComputeNode Template created in [Configure Weather Service for HPC ComputeNode](#_Configure_Weather_Service_2) installs Python on the compute nodes. But sometimes the installation may fail. You can verify this by checking whether *Python* folder has been created on the ComputeNode after provisioning.   
     
   If it fails, navigate to folder *C:\WeatherService\ForecastWeather* on the ComputeNode. And run the following script to install Python manually.

StartupTask.ps1 –Steps SetupPython