

# AWS-Style Regions

## Regions

### 1 Background

Currently, CloudStack only supports Amazon style Availability zones (AZ). This feature is to allow the CloudStack infrastructure hierarchy to support Amazon style Regions. Regions would provide the following benefits:

- **Higher availability of the services:** users can deploy services across AZs and even if one of the AZ goes down the services are still available to the end-user through VMs deployed in other zones.
- **Higher availability of the MS:** Since each MS Cluster only manages a single Region, if that MS Cluster goes down, only that particular Region is impacted. Admin should be able to access all the other Regions.
- **Scalability:** The scalability limit of CloudStack dramatically improves, as the scalability limit of MS Cluster is limited to a single Region.
- **Object Store:** With Regions construct, CloudStack would also allow users to define Object Store (Secondary Storage) across AZs. This helps users easily deploy VMs in different AZs using the same template, offerings.
- **Geographical Grouping:** Regions allow admins to group AZs (that have low latency and are geographically located nearby) into a broader region construct.

### 2 Requirements

- Admin should be able to create multiple Regions within a CloudStack cloud. I.e., CloudStack should manage multiple regions within a cloud. The following operations should be supported:
  - Create Region
  - Delete Region
  - Update Region
  - List Regions
- Admin should be able to create multiple zones with a Region. Management server cluster manages all the zones within the region. Following Zone operations should be supported on per region basis:
  - Create Zone
  - Delete Zone
  - Update Zone
  - List Zones
- Management Server cluster for each region should only control the infrastructure within that region. In other words, if a management server cluster fails, then only the region managed by that server cluster should be inaccessible for admin and users. It should not have any impact on the other Regions in the cloud
- Each Region should have access to an object store. The object store should be common to all Zones within the Region. I.e., any object stored from a Zone should be visible across all the other zones within the region
- EIP function should be available across both basic zones and across advance Zones within a Region
- ELB function should be available across both basic zones and across advance Zones within a Region
- The administrative hierarchy (Users, Accounts, Domains, Projects) - should be valid across all the regions. I.e., if admins create domains, accounts and users etc. in one region it should be reflected in other regions as well.
- Authentication credentials – username, password, keys – should be valid across all regions
- Switching between Regions should not require the user to sign-on again (SSO should be supported).
- Resource management should be extended to Region level
  - Available compute, storage, network (IPs, VLANs etc.) resources that are currently tracked should be aggregated and displayed at region level
  - Appropriate global parameters should be available at Region level while the remaining would be available at Zone level
- Usage: All the (per account) usage records should be aggregated to Regions level
- Global Configurations: All of the administrative hierarchy (Domains, Accounts, Users, Projects) related Global Configurations should have consistent values across all regions and has to be propagated when a user modifies a configuration on one of the region.
- Each region should maintain a list of all other regions and their region endpoints.

Note:

- Users should be required to deploy all the Management Servers of a particular region in one zone.
- Sync of User data can be made optional. If sync of user data is not supported, make sure necessary events are published for a customer to write their own tool to sync user data.

### 3 UI / UX Requirements

- User/Admin should be able to view all Regions by logging into a MS of any of the Regions. User then should be able to select a specific Region to view details of that Region.
  - Reports on the dashboard should be aggregated to Region level
- On first UI access (after MS installation), user should be able to connect to another CloudStack MS (in another Region) or treat this instance as the first region
- Modify the start-up wizard to force users define the Region (Region Name, Description, etc) before they start creating the Availability Zones
- Users should be able to switch between various regions for UI using Single Sign-On.

### 4 Upgrade Scenarios

Following upgrade scenarios should be supported:

- Multiple Zones to Multiple Regions: Current deployments with multiple Zones should be able to move to a deployment of multiple Regions. Admins should be able to map one or more zones to a Region
- Multiple Zones to Region: Current deployments with multiple Zones should be able to move to a deployment of single Region (this is a subset of the above requirement)
- All AZs in one Region should get access to all secondary storage

## 5 Lower Priority Requirements

- Management Server limited to per Zone can be done in a future release.
- EIP and ELB functions across zones are only supported when both zones are either basic or advance. It is not required to support EIP or ELB across basic and advanced zones
- It is not required to display a dashboard/reports that are aggregated across Regions

## 6 Bugs

[Regions JIRA Issue](#)

## 7 Open Items:

- None