

# Integration project to deploy and use Apache Mesos on a CloudStack based Cloud

## Abstract:

The project aims to bring cloudformation[1] like service to cloudstack. One of the prime use-case is cluster computing frameworks on cloudstack. A cloudformation service will give users and administrators of cloudstack ability to manage and control a set of resources easily. The cloudformation will allow booting and configuring a set of VMs and form a cluster. Simple example would be LAMP stack. More complex clusters such as mesos or hadoop cluster requires a little more advanced configuration. There is already some work done by Chiradeep Vittal at this front [5]. In this project, I will implement server side cloudformation service for cloudstack and demonstrate how to run mesos cluster using it.

## Mesos:

Mesos is a resource management platform for clusters [2]. It aims to increase resource utilization of clusters by sharing cluster resources among multiple processing frameworks(like MapReduce, MPI, Graph Processing) or multiple instances of same framework. It provides efficient resource isolation through use of containers. Uses zookeeper for state maintenance and fault tolerance.

## What can run on mesos ?

Spark: A cluster computing framework based on the Resilient Distributed Datasets (RDDs) abstraction. RDD is more generalized than MapReduce and can support iterative and interactive computation while retaining fault tolerance, scalability, data locality etc.

Hadoop: Hadoop is fault tolerant and scalable distributed computing framework based on MapReduce abstraction.

Begel: A graph processing framework based on pregel.

and other frameworks like MPI, Hypertable.

## How to deploy mesos

Mesos provides cluster installation scripts [7] for cluster deployment. There are also scripts available to deploy a cluster on Amazon EC2 [8]. It would be interesting to see if this scripts can be leveraged in anyway.

## Deliverables:

1. Deploy CloudStack and understand instance configuration/contextualization
2. Test and deploy Mesos on a set of CloudStack based VM, manually. Design/propose an automation framework.
3. Test stackmate and engage chiradeep (report bugs, make suggestion, make pull request)
4. Create cloudformation template to provision a Mesos Cluster
5. Compare with Apache Whirr or other cluster provisioning tools for server side implementation of cloudformation service.

## Architecture and Tools:

The high level architecture I propose is as follows:

It includes following components:

1. CloudFormation Query API server:

This acts as a point of contact to and exposes CloudFormation functionality as Query API. This can be accessed directly or through existing tools from Amazon AWS for their cloudformation service. It will be easy to start as a module which resides outside cloudstack at first and I plan to use dropwizard [3] to start with. Later may be the API server can be merged with cloudstack core. I plan to use mysql for storing details of clusters.

2. Provisioning:

Provisioning module is responsible for handling the booting process of the VMs through cloudstack. This uses the cloudstack APIs for launching VMs. I plan to use preconfigured templates/images with required dependencies installed, which will make cluster creation process much faster even for large clusters. Error handling is very important part of this module. For example, what you do if few VMs fail to boot in cluster ?

3. Configuration:

This module deals with configuring the VMs to form a cluster. This can be done via manual scripts/code or via configuration management tools like chef /ironfan/knife. Potentially workflow automation tools like rundeck [4] also can be used. Also Apache whirr and Provisionr are options. I plan to explore this tools and select suitable ones.

## API:

Query API will be based on Amazon AWS cloudformation service [9]. This will allow leveraging existing tools for AWS [10].

## Timeline:

1-1.5 week : project design. Architecture, tools selection, API design

1-1.5 week : getting familiar with cloudstack and stackmate codebase and architecture details

1-1.5 week : getting familiar with mesos internals

1-1.5 week : setting up the dev environment and create mesos templates

2-3 week : build provisioning and configuration module

Midterm evaluation: provisioning module, configuration module

2-3 week : develop cloudformation server side implementation

2-3 week : test and integrate

#### **Future Work:**

##### **1. Auto Scaling :**

Automatically adding or removing VMs from mesos cluster based on various conditions like utilization going above/below a static threshold. There can be more sophisticated strategies based on prediction or fine grained metric collection with tight integration with mesos framework.

##### **2. Cluster Simulator :**

Integrating with existing simulator to simulate mesos clusters. This can be useful in various scenarios, for example while developing a new scheduling algorithm, testing autoscaling etc.

#### **References**

- [1] <http://aws.amazon.com/cloudformation/>
- [2] <http://incubator.apache.org/mesos/>
- [3] <http://dropwizard.codahale.com/>
- [4] <http://rundeck.org/>
- [5] <http://cloudierthanhou.wordpress.com/2013/04/26/stackmate-execute-cloudformation-templates-on-cloudstack/>
- [6] <http://siel-iiith.github.io/HadoopStack/>
- [7] <https://github.com/apache/mesos/blob/trunk/docs/Deploy-Scripts.textile>
- [8] <https://github.com/apache/mesos/blob/trunk/docs/EC2-Scripts.textile>
- [9] [http://docs.aws.amazon.com/AWSCloudFormation/latest/APIReference/API\\_Operations.html](http://docs.aws.amazon.com/AWSCloudFormation/latest/APIReference/API_Operations.html)
- [10] <http://aws.amazon.com/developertools/AWS-CloudFormation>