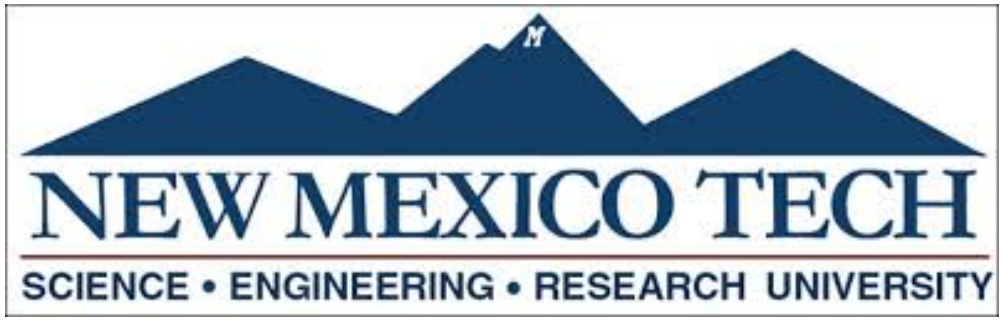

Ad hoc Cloud-based Computing Clusters for Big Data Processing

Abdelmounaam Rezgui

rezgui@cs.nmt.edu

NSFCloud Workshop on Experimental Support for Cloud Computing, Arlington, VA, Dec. 11-12, 2014

Description of Research

- Research Area
  - Volunteer Cloud Computing to Support Big Data Science
- Research Problem
  - Build efficient, distributed cloud "clusters" on-the-fly to support big data processing

Research Questions

- Can a computing cluster be **dynamically** composed from **donated** cloud-based resources to run a **specific** big data workload?

- if the answer is yes, how can we speed up big data workloads by **interleaving the processes** of cluster formation, data loading and data processing?

Why is it important?

- Enable big data science over volunteered cloud computing resources.

- Lower the entry barrier to big data processing so that scientists may conduct big data research without making huge investments

Challenges

- Availability of volunteered resources is **unpredictable**.
- Large number of **alternatives**
- How to dynamically build an **optimal cluster** taking into account:
  - The specific big data application
  - The user's location
  - The location of the donated resources

Experiment 1

- **Optimal cluster dimensioning**
  - Manually build distributed cloud clusters on the testbed
  - Compare their performance with what the optimization framework suggests in terms of cluster size (number of nodes) and topology (locations of nodes).

Experiment 2

- **Prediction of the cost of the dynamic creation of big data processing clusters**
  - Purpose: Assess the accuracy of our model for predicting the time needed to set up a computing cluster for a given big data workload.
  - Run code that automatically builds a cluster of (possibly geographically distant) computing nodes
  - Measure the time needed before the cluster is entirely functioning and ready to run the given big data workload.

Experiment 3

- **Determining the best choreography**

- Three tasks :
  - cluster formation,
  - data loading, and
  - data processing

- Purpose: determine which interleaving scheme is the best.

Capabilities 1

- **Application Programming Interface**
  - **create** a computing cluster with a given size and/or given topology,
  - **select specific sites** for nodes
  - dynamically **resize** a cluster,
  - **transfer data** to and from nodes

- **Automatic Virtual Machine Deployment**
  - configure a VM according to requirements of experiment
  - automatically (possibly concurrently) deploy it on one or more of the testbed's nodes.

Capabilities 2

- **Location**
  - expose the geographical location of cluster's nodes to users
- **Transparent Fault Tolerance**
  - Failures should be transparent to applications and experiments
- **Reproducibility**
  - Accurately reproduce the same conditions that characterized the execution of a given script.
  - This would make it meaningful to compare the results of two executions of the same script.