A Report On

DISTRIBUTED COMPUTING

Using Parallel computation in different VMs in a Public Cloud

Submitted in requirement for the course

CLOUD COMPUTING (CSN-520)

of Bachelor of Technology in Computer Science and Engineering

by

Gurwinder Singh (Enroll No.14114026)

Saurabh Goyal (Enroll No. 14114051)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, ROORKEE

ROORKEE- 247667 (INDIA)

Spring, 2018

Objective

Test any compute intensive application on any public cloud platform like AWS.

Problem Statement

Given two square matrices A and B of n*n dimension, multiply them.

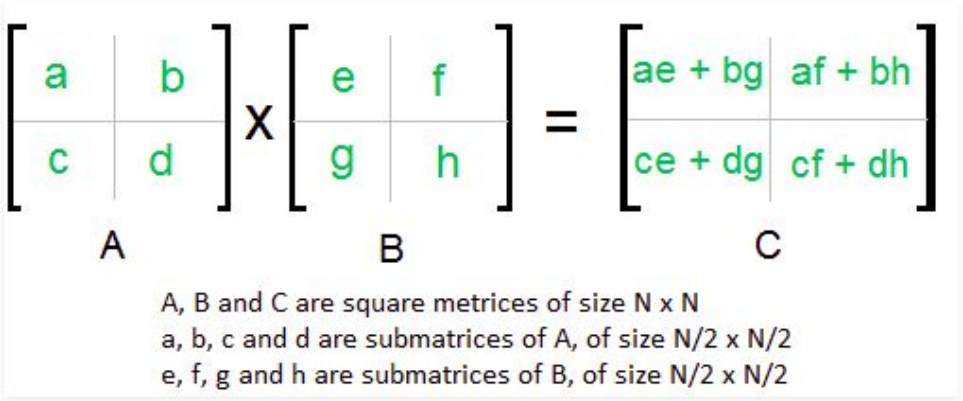
Approach:
We have used Divide and Conquer approach to multiply the matrix parallely on 4 VMs

Divide and Conquer

Following is simple Divide and Conquer method to multiply two square matrices.

1) Divide matrices A and B in 4 sub-matrices of size N/2 x N/2 as shown in the below diagram.

2) Calculate following values recursively. ae + bg, af + bh, ce + dg and cf + dh.



VM 1 :- ae+bg
VM 2 :- af+bh
VM 3 :- ce+dg
VM 4 :- cf+dh

Services / Infrastructure Used

Following services provided by Amazon Web Services (AWS) have been used:

- **Amazon Simple Queue Service (SQS) - Amazon AWS**
    - Amazon Simple Queue Service (SQS) is a fully managed message queuing service that makes it easy to decouple and scale microservices, distributed systems, and serverless applications. Building applications from individual components that each perform a discrete function improves scalability and reliability, and is best practice design for modern applications.

    - https://aws.amazon.com/sqs/

- **Amazon EC2 - Amazon AWS**
    - Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction.

    - https://aws.amazon.com/ec2/

Requirements/Setup

Instances created

4 instances of Amazon EC2 service are used.
VM1:- Both master and slave
VM2,3,4:- slaves

2 queue for processing requests:
qinfo: queue for picking up input from
qresult: queue in which VMs push results

Software requirements

Each EC2 instance (Ubuntu Server 16.04 LTS) is installed with the following :
python 2.7
AWS sdk for python - boto3
aws-cli
numpy
pickle

Process

1 User enters the matrix dimension

2 Matrix in which numbers from 0 to n^2-1 are arranged is generated

3 Matrix B is same as Matrix A

4 Matrix A is divided into 4 parts. (a,b,c,d)

5 Similarly B into (e,f,g,h)

6 The concatenated matrix (a,e,b,g) is passed to qinfo.

7 VM1 picks this and similar other VMs picks other inputs.

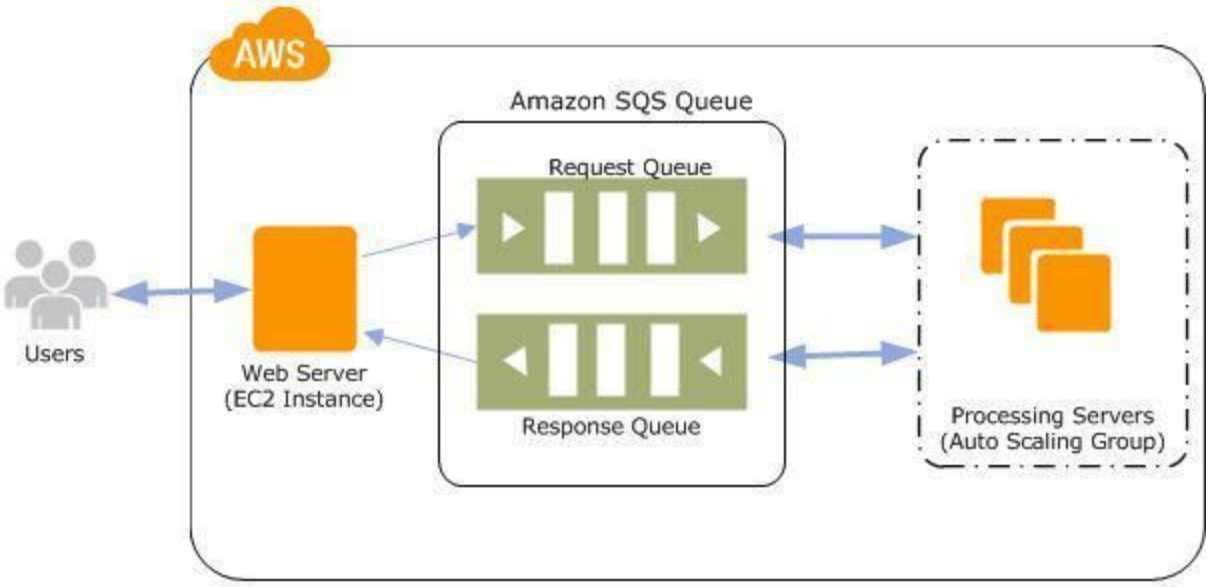
8 VM1 picks up this matrix computes a.e +b.g and pushes result in the qresult along with the key which identifies the position of this submatrix in the final matrix.

9 Master finally concatenates this submatrices and the result is obtained.

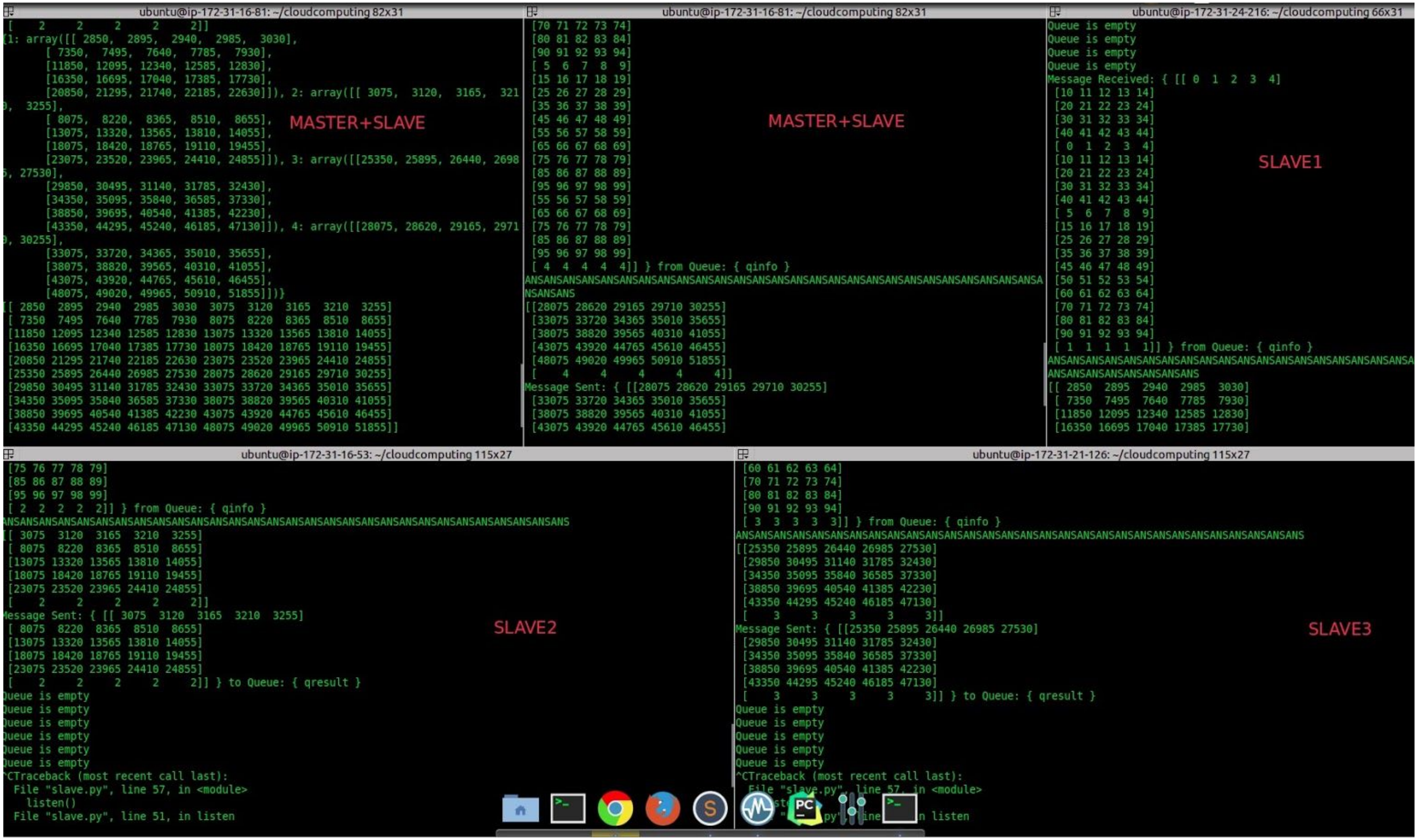
Schema

Producer (Web Server) : EC2 instance SQS Queues : Request/Response queues

Consumers: EC2 instance(s)



Screenshots


ubuntu@ip-172-31-16-81: ~/cloudcomputing 82x31
MASTER+SLAVE
ubuntu@ip-172-31-16-81: ~/cloudcomputing 82x31
MASTER+SLAVE
ubuntu@ip-172-31-24-216: ~/cloudcomputing 66x31
SLAVE1
ubuntu@ip-172-31-16-53: ~/cloudcomputing 115x27
SLAVE2
ubuntu@ip-172-31-21-126: ~/cloudcomputing 115x27
SLAVE3