Datacenter in a box: test your SDN cloud-datacenter controller at home

José Teixeira‡, Gianni Antichi*, Davide Adami†, Alessio Del Chiaro†, Stefano Giordano†, Alexandre Santos‡

*Computer Lab, University of Cambridge
†Dept. of Information Engineering, University of Pisa
‡Engineering school, University of Minho

EWSDN’13
Contents

• Introduction
• Datacenter in a box: our framework
  • Overview
  • Modules
• Use Case
  • Understanding VM allocation policies dynamics
• Performance Evaluation
• Conclusions
Introduction

Cloud Datacenters

- Unlike traditional DCs
  - Services bounded to physical servers
- Cloud Datacenters
  - Consist of virtualized resources
  - VMs can migrate between physical servers resulting in:
    - increasing both scalability and reliability – Better DC resources usage

Issues

- DC Management
  - Becomes harder – Separation of
  - Isolation and connectivity of VMs
- Performance degradation

- Systems administration (i.e. servers, VMs)
- Network administration (i.e. physical switches)
- Resource fragmentation
- Rigidity Intra-DC network architecture
Introduction

Cloud service providers
• Demand a new generation of Cloud DCs
  • Efficient – high server utilization
  • Agile – Fast network response to server/VMs provisioning
  • Scalable – Consolidating and migrating VMs based on application requirements
  • Simple

Software Defined Networking
• Promising way to satisfy DC network requirements
• Decouples control plane (routing decisions) from data plane (traffic forwarding)

Openflow
• Set forwarding rules into OF-Compliant switches
• Centralized intelligence - Controller
Introduction

• SDN cloud-DC controller
  • Fulfills DC requirements
    • Both IT and network resources

• Novel architectural solutions
  • Test campaigns must be performed
    • Experimental environments

• A Novel Framework
  • Enhances Mininet & POX
  • Develop and assess novel controllers
  • Compare performance of control and management strategies
Related Work

• Cloudsim – Estimate performance;

• Cloudsim Extension - topology generator & flow-based approach (collecting delay)

• Other DC emulators
  • icancloud[12] greencloud[10] and groudsim[18]

SDN solutions

• FPGA emulation platform

• Meridian – Create and manage logical network topologies
  • M. Banikazemi et al. Meridian: An sdn latform for cloud network services. Communications Magazine, 2013
**Main Purpose:** provide a full package for the development and test of DC SDN Controller

**Single Environment**

**DC topology emulator**
- Starting point – Mininet
  - API to reproduce custom topologies
  - Assess OF controllers before deployment
- Lacks tools
  - Correctly emulate DC behavior

- How to easily generate and configure typical DC topologies?
- How to simulate VMs allocation requests?
- How to emulate the inter and in/out DC traffic?

**DC oriented controller**
- POX
  - Python OF controller
  - Ready-to-use modules
- Not high level enough API
  - Implement DC controller
  - Prevents rapid development

- Provides necessary abstraction level
  - Still dynamic

---

**Datacenter in a box: our framework**
Datacenter in a box: our framework
Datacenter in a box: our framework

Mininet Environment

- **Topology Generator**
  - Support for tree and fat-tree topologies
    - Gateways, core, aggregation, edge, servers, links

- **Traffic Generator**
  - Correctly emulate DC behavior
  - Fully customizable traffic emulation
    - D-ITG, a distributed traffic generator
    - Large spectrum of network traffic profiles (e.g., Poisson distribution, DNS, VoIP, etc.)

- **Mininet DC Configuration**
  - Fully configurable
    - DC-architecture
    - Host resources
    - Set per-VM D-ITG configuration

**Note**: in Emulated Environment VMs are not really allocated
Datacenter in a box: our framework

Virtual Machine Requests

- **WEB Platform**
  - Monitoring tool
  - End-user GUI for requesting VM

- **VM Requester**
  - Act as external user requiring resources (VM)
  - Communicates with SDN Cloud DC controller
    - CPU, RAM, Disk, Network & VM lifetime

- Lifetime expires
  - Traffic and rules automatically handled
Datacenter in a box: our framework

SDN Cloud DC Controller

• VM Request Handler
• OF Rules Handler
• Topology Discovery
  • Intra-DC network – multi-layer hierarchical infrastructures
  • Host_tracker and discovery
• Statistics Handler
  • Automatically collects and saves statistics
• User-defined Logic
  • Define desired controller functionalities
  • Focus only on the algorithm
Datacenter in a box: our framework

Emulation Working flow

1. VM Request
2. Statistics Request
3. Statistics Reply
   - Run User-defined logic
     • Which rules to install
     • Where to put the VM
4. Install rules
5. VM Reply
6. Generate traffic
Validation and Tests

Validation

• Best fit VS Worst fit
  • Best fit
    • Fill each server with VMs
  • Worst fit
    • Spread the VMs in all servers

DC Architecture

• 16 servers (hosts)
• Up to 3 VMs per server
• 1 request per second
• first host link saturates at the 33–th second
Validation and Tests

Performance Evaluation
Conclusions

• Novel SDN Framework (Mininet & POX)
• Addresses issues in testing SDN Cloud DC controllers
• Validated with well-known VM scheduling algorithms
• Evaluated performance (scalability and stability)

Work still in progress...

• Insert more features
  • VM Migration and storage
• Extend tests
  • More algorithms and bigger DCs
Thank you

José Teixeira‡, Gianni Antichi*, Davide Adami†,
Alessio Del Chiaro†, Stefano Giordano†,
Alexandre Santos‡

*Computer Lab, University of Cambridge
†Dept. of Information Engineering, University of Pisa
‡Engineering school, University of Minho