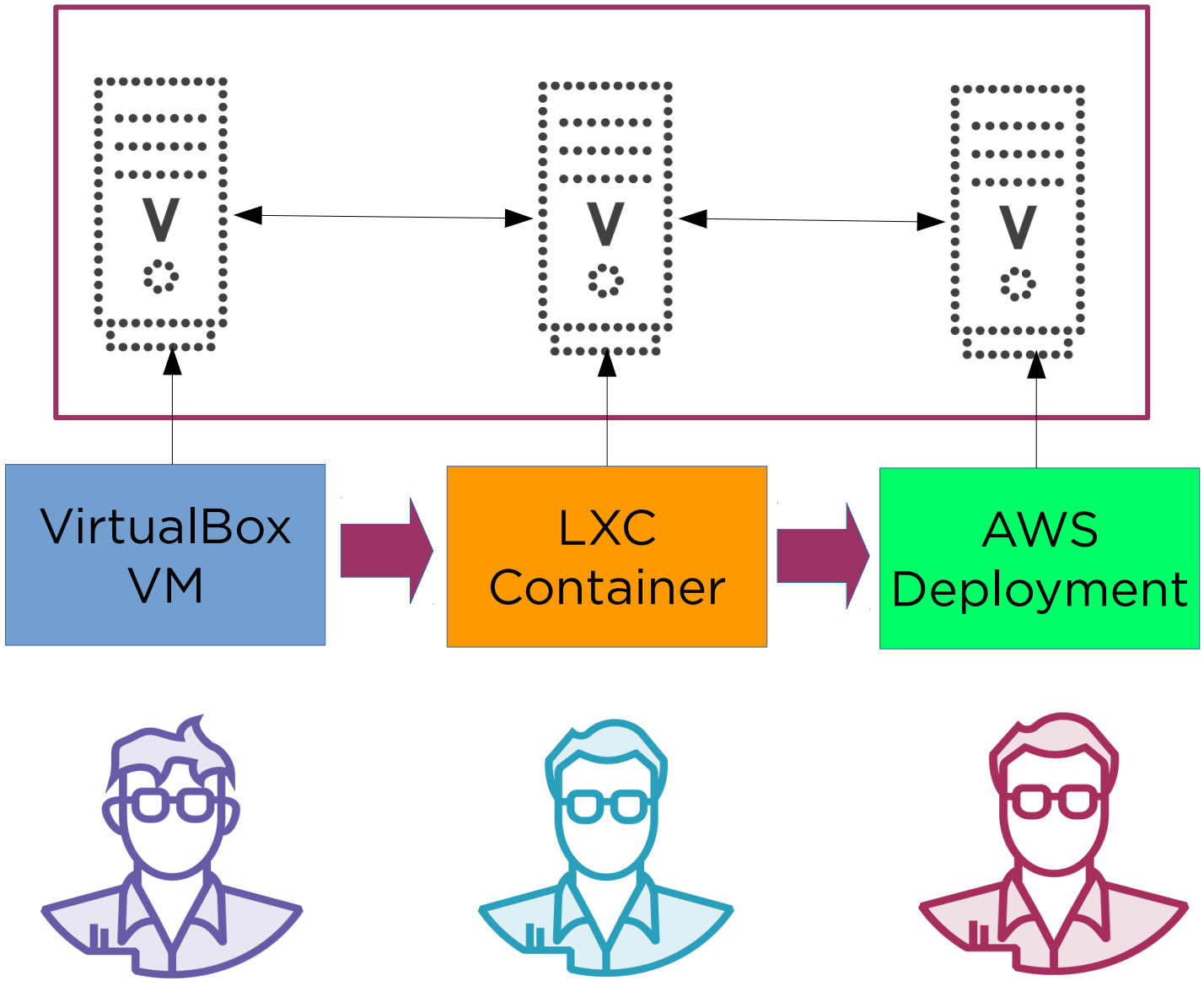
Vagrant and Packer

David Clinton
LINUX SYSTEM ADMINISTRATOR

@davidbclinton | www.bootstrap-it.com | www.bootstrap-it.com/blog

Packer

The Packer/Vagrant Process



Packer template.json file

```
{
  "variables": {
    "aws_access_key": "",
    "aws_secret_key": ""
  },
  "builders": [{
    "type": "amazon-ebs",
    "access_key": "{{user `aws_access_key`}}",
    "secret_key": "{{user `aws_secret_key`}}",
    "region": "us-east-1",
    "source_ami": "ami-fce3c696",
    "instance_type": "t2.micro",
    "ssh_username": "ubuntu",
    "ami_name": "packer-example {{timestamp}}"
  }]
}
```

Packer template.json file

```
{
  "variables": {
    "aws_access_key": "",
    "aws_secret_key": ""
  },
  "builders": [{
    "type": "amazon-ebs",
    "access_key": "{{user `aws_access_key`}}",
    "secret_key": "{{user `aws_secret_key`}}",
    "region": "us-east-1",
    "source_ami": "ami-fce3c696",
    "instance_type": "t2.micro",
    "ssh_username": "ubuntu",
    "ami_name": "packer-example {{timestamp}}"
  }]
}
```

Packer template.json file

```
{
 "variables": {
  "aws_access_key": "",
  "aws_secret_key": ""
 },
 "builders": [{
  "type": "amazon-ebs",
  "access_key": "{{user `aws_access_key`}}",
  "secret_key": "{{user `aws_secret_key`}}",
  "region": "us-east-1",
  "source_ami": "ami-fce3c696",
  "instance_type": "t2.micro",
  "ssh_username": "ubuntu",
  "ami_name": "packer-example {{timestamp}}"
 }]
}
```

Packer template.json file

```
{
 "variables": {
  "aws_access_key": "",
  "aws_secret_key": ""
 },
 "builders": [{
  "type": "amazon-ebs",
  "access_key": "{{user `aws_access_key`}}",
  "secret_key": "{{user `aws_secret_key`}}",
  "region": "us-east-1",
  "source_ami": "ami-fce3c696",
  "instance_type": "t2.micro",
  "ssh_username": "ubuntu",
  "ami_name": "packer-example {{timestamp}}"
 }]
}
```

Packer template.json file

```
{
  "variables": {
    "aws_access_key": "",
    "aws_secret_key": ""
  },
  "builders": [{
    "type": "amazon-ebs",
    "access_key": "{{user `aws_access_key`}}",
    "secret_key": "{{user `aws_secret_key`}}",
    "region": "us-east-1",
    "source_ami": "ami-fce3c696",
    "instance_type": "t2.micro",
    "ssh_username": "ubuntu",
    "ami_name": "packer-example {{timestamp}}"
  }]
}
```

Packer template.json file

```
{
 "variables": {
  "aws_access_key": "",
  "aws_secret_key": ""
 },
 "builders": [{
  "type": "amazon-ebs",
  "access_key": "{{user `aws_access_key`}}",
  "secret_key": "{{user `aws_secret_key`}}",
  "region": "us-east-1",
  "source_ami": "ami-fce3c696",
  "instance_type": "t2.micro",
  "ssh_username": "ubuntu",
  "ami_name": "packer-example {{timestamp}}"
 }]
}
```

Vagrant

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

Review

Packer: artifacts and boxes

Packer scripts: builders, provisioners, post processors

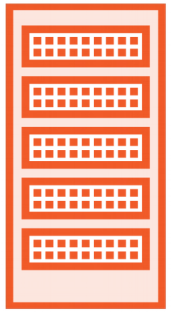
```
packer build example.json
packer push -name myAtlasName/mynewexample example.json
```

Vagrant

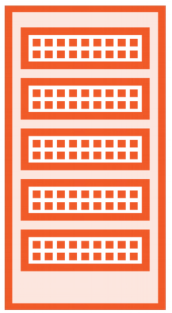
```
vagrant init fgrehm/precise64-lxc
vagrant up --provider=lxc
vagrant ssh
```

IT Deployment Design

Servers



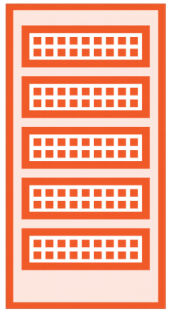
Physical
Server One



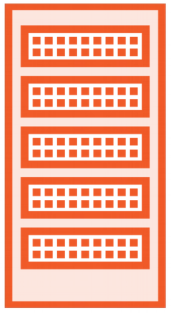
Physical
Server Two

Moodle Domains

= Moodle domain

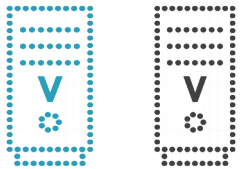


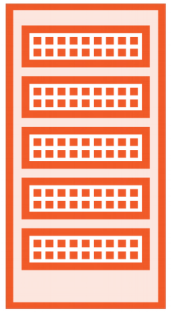
Physical
Server One



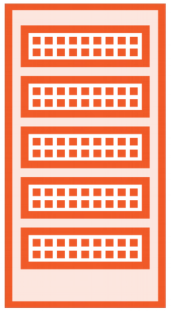
Physical
Server Two

Admin Domains



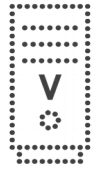


Physical Server One



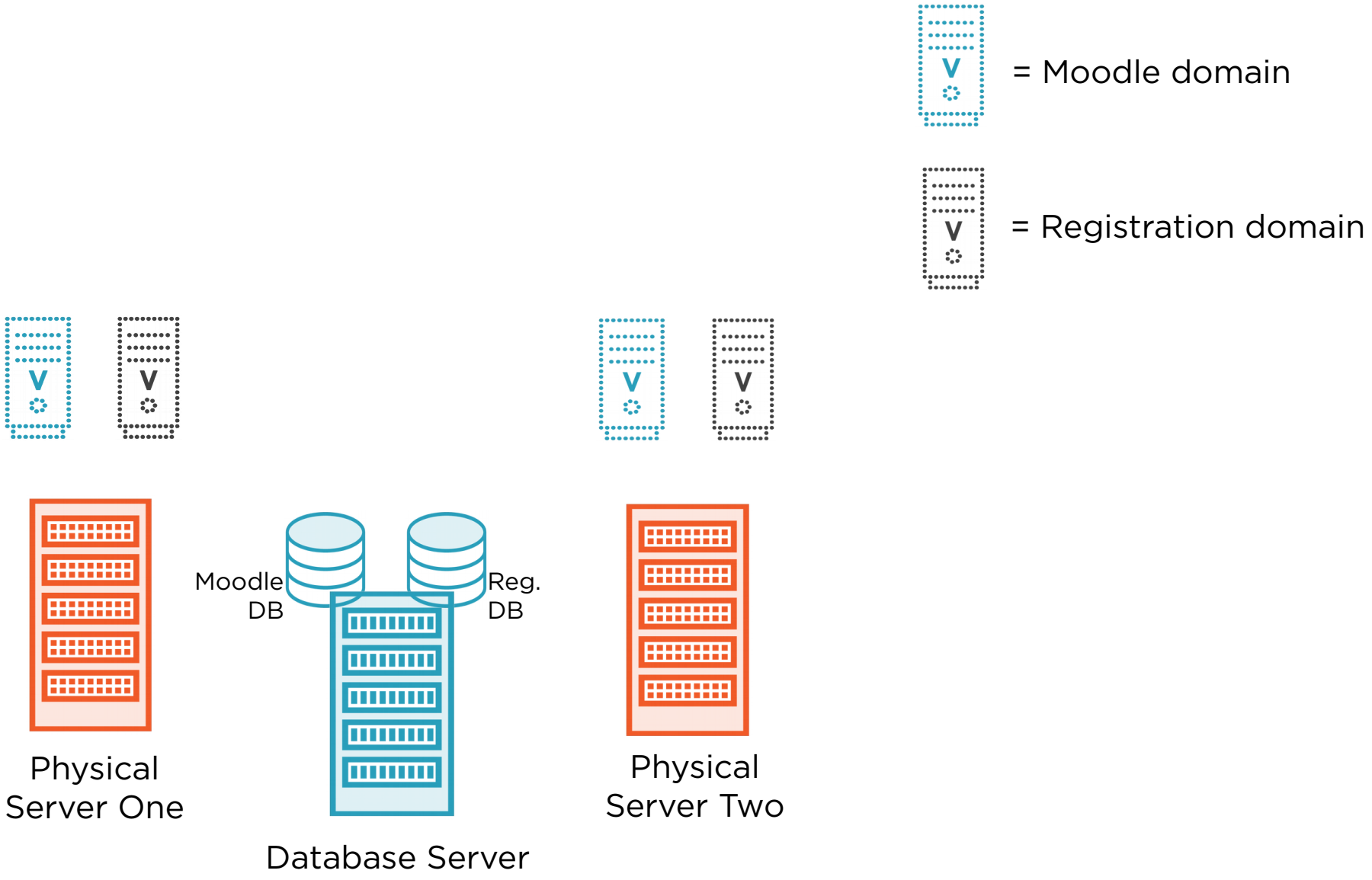
Physical Server Two

= Moodle domain

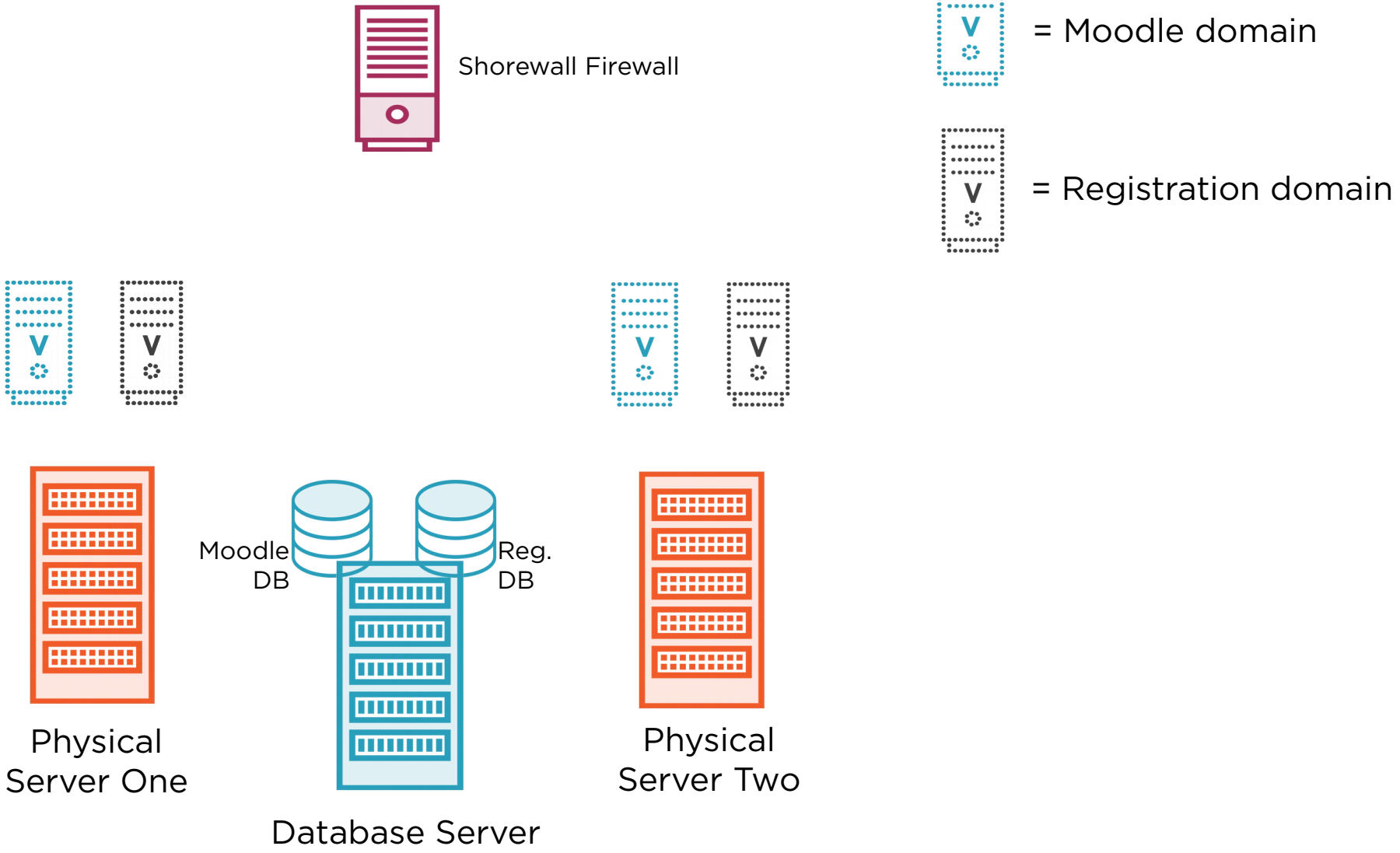


= Registration domain

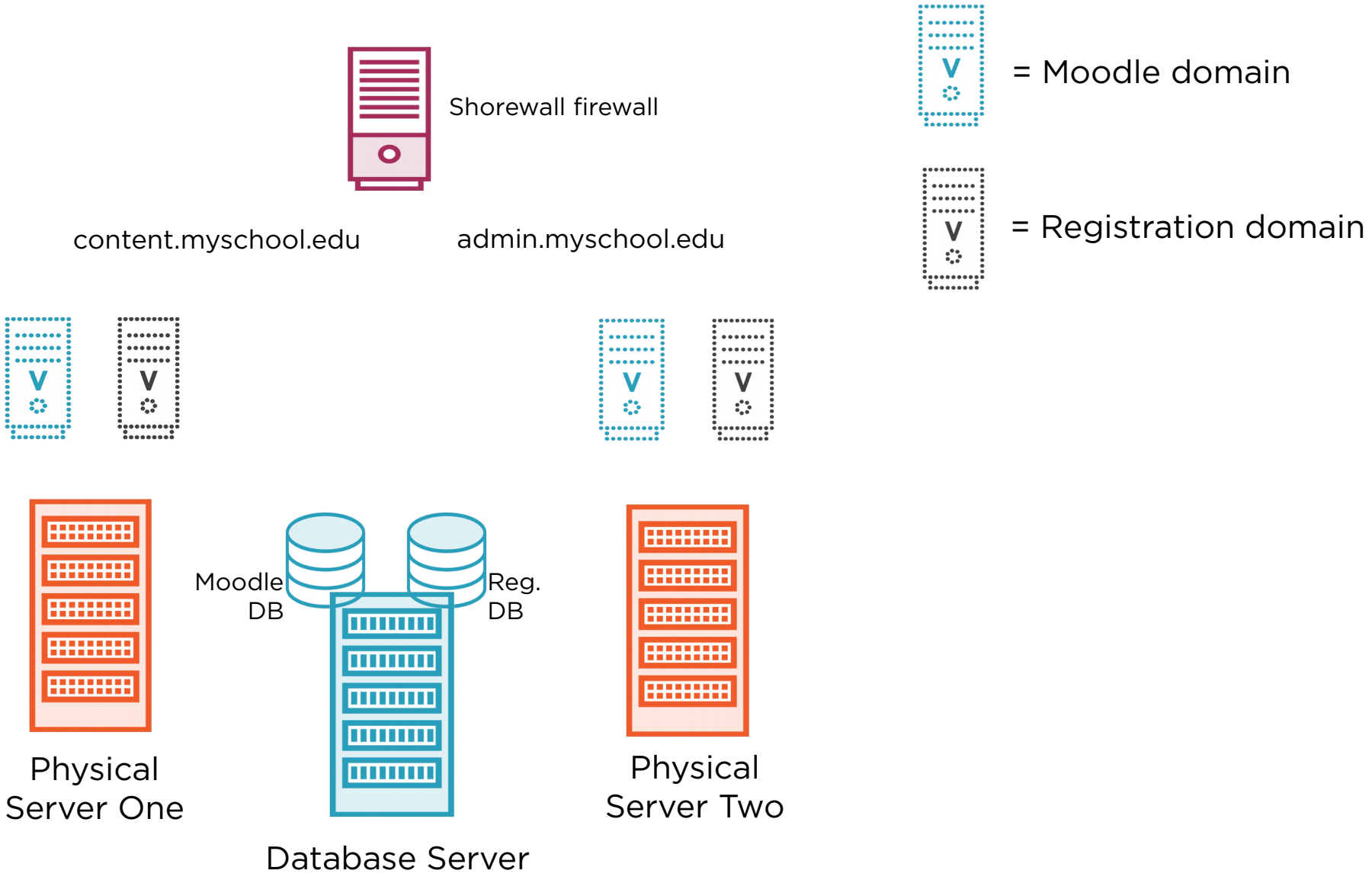
Databases


= Moodle domain
= Registration domain
Moodle DB
Reg. DB
Physical Server One
Database Server
Physical Server Two

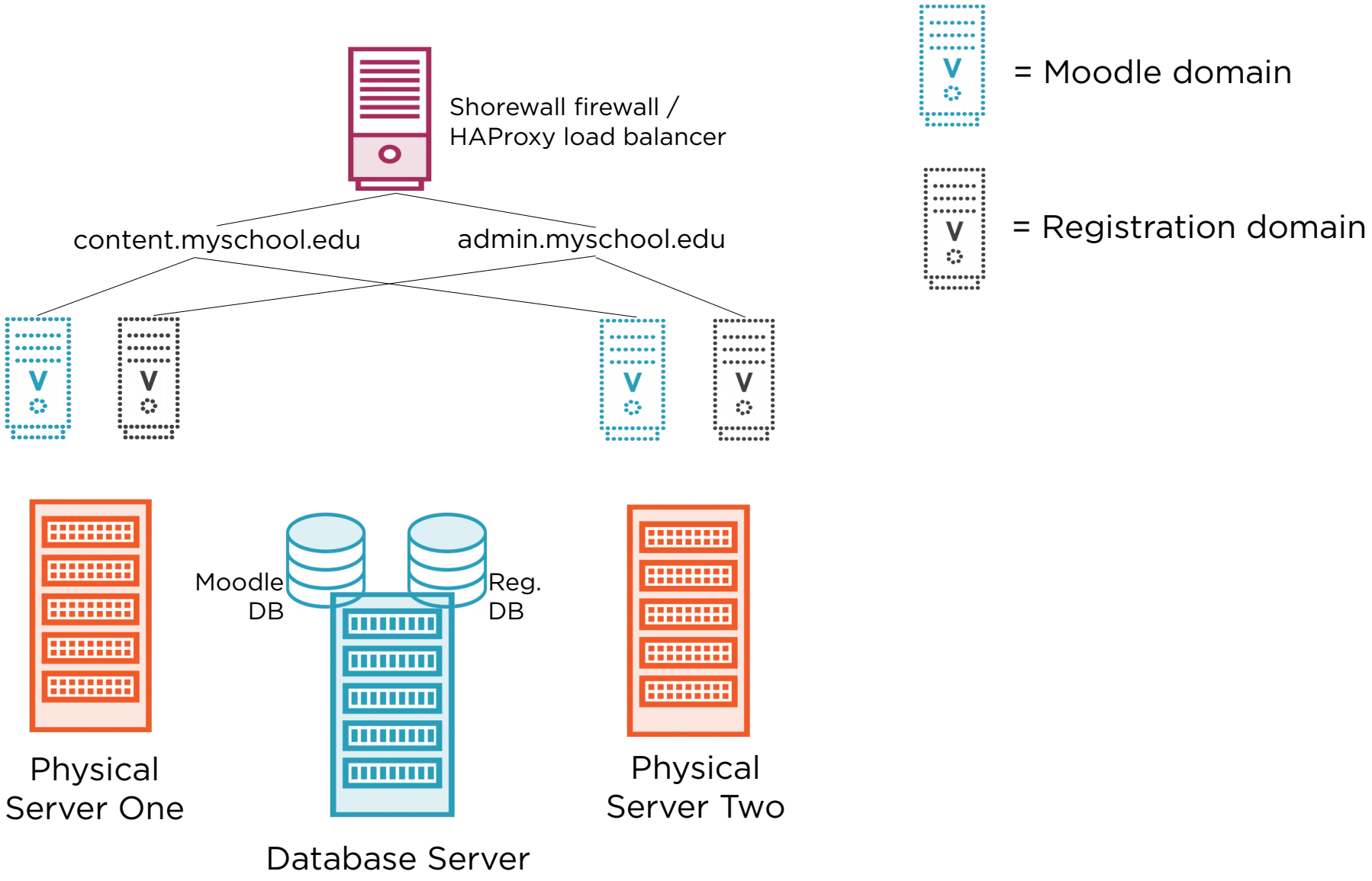
Firewall


Shorewall Firewall
= Moodle domain
= Registration domain
Moodle DB
Reg. DB
Physical Server One
Database Server
Physical Server Two

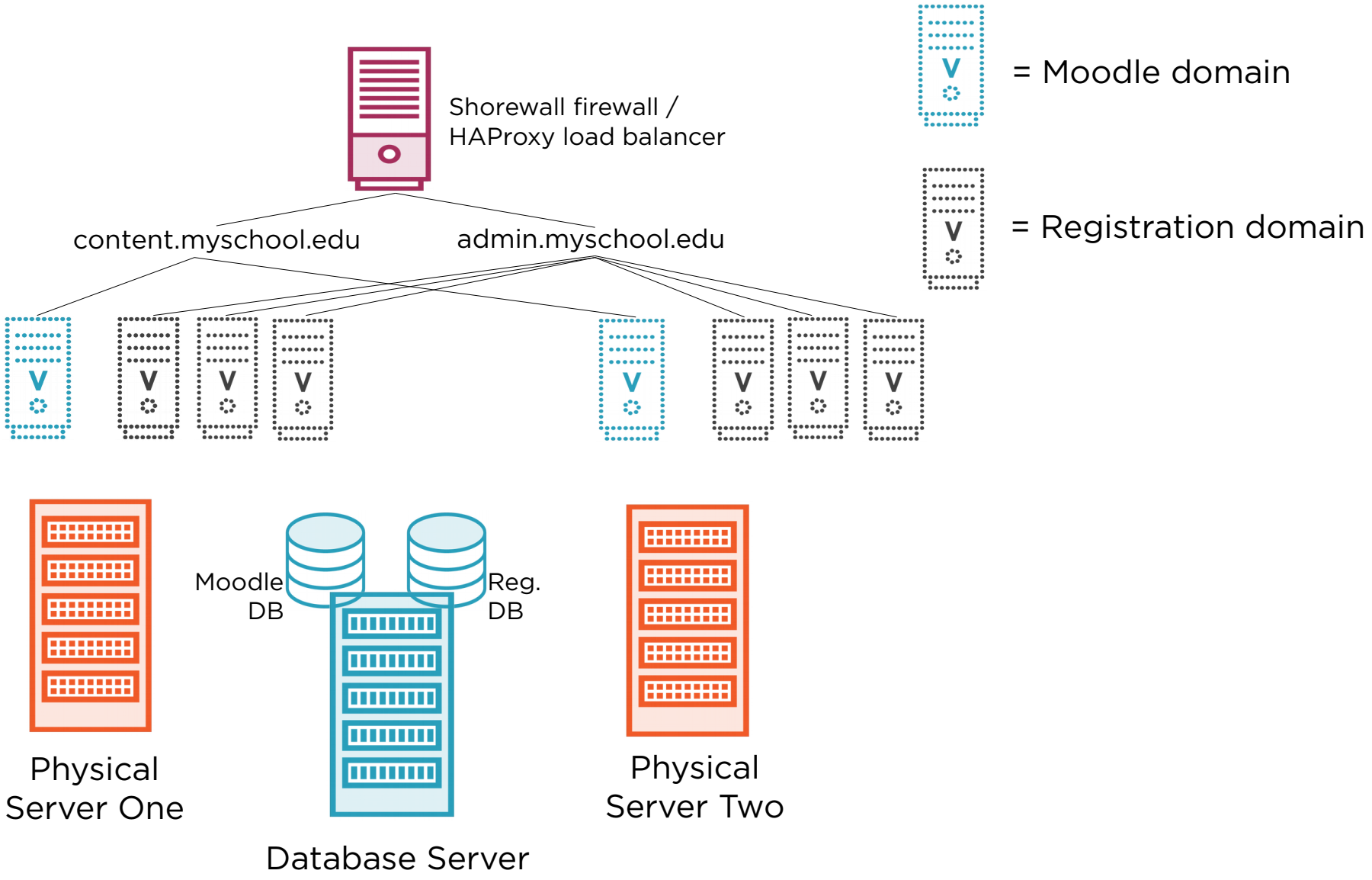
Load Balancer



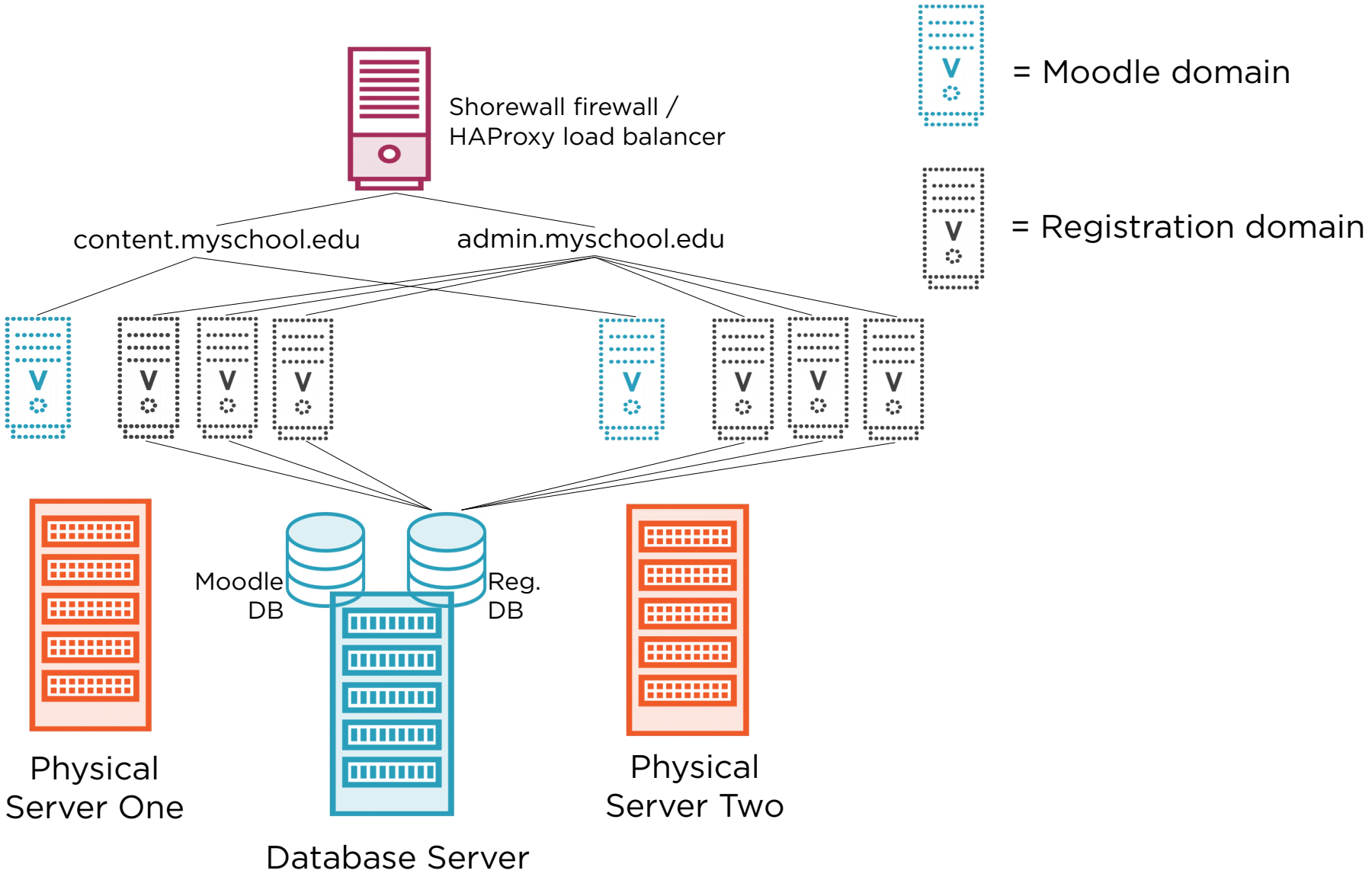
Load Balancer



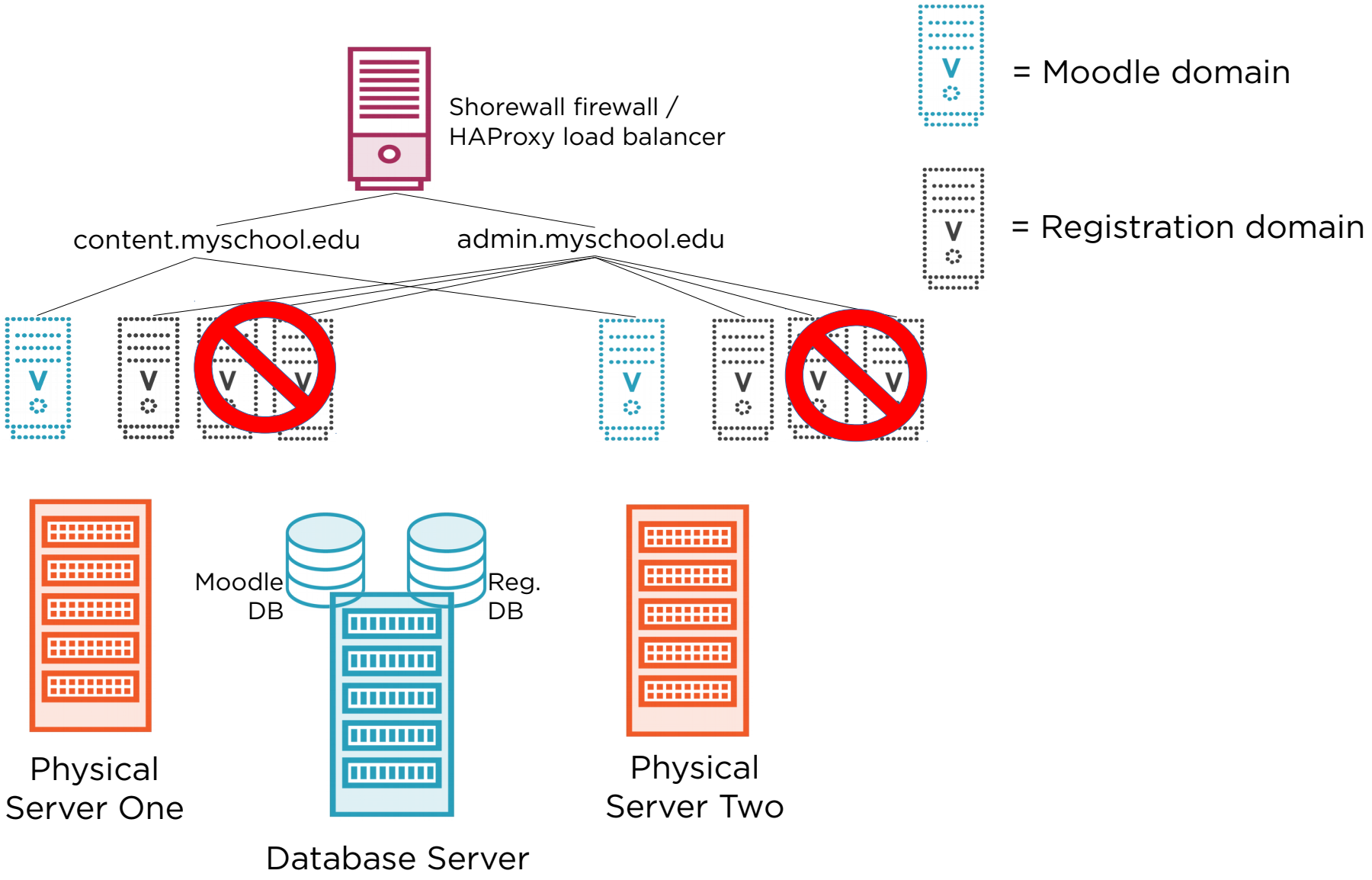
Meet Increased Admin Demand



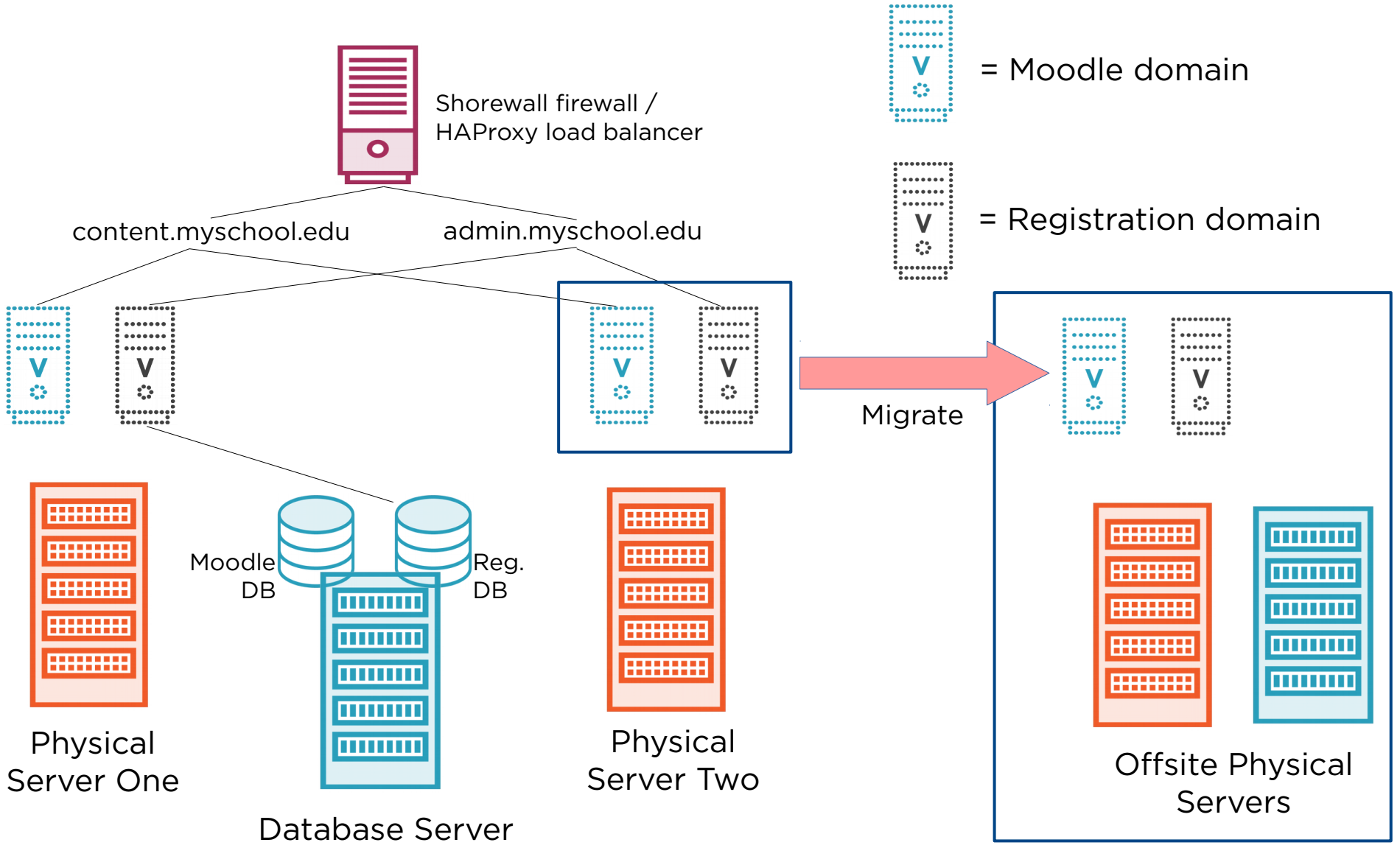
Meet Increased Admin Demand



Meet Increased Admin Demand



Offsite Replication



Linux Server Virtualization: Review

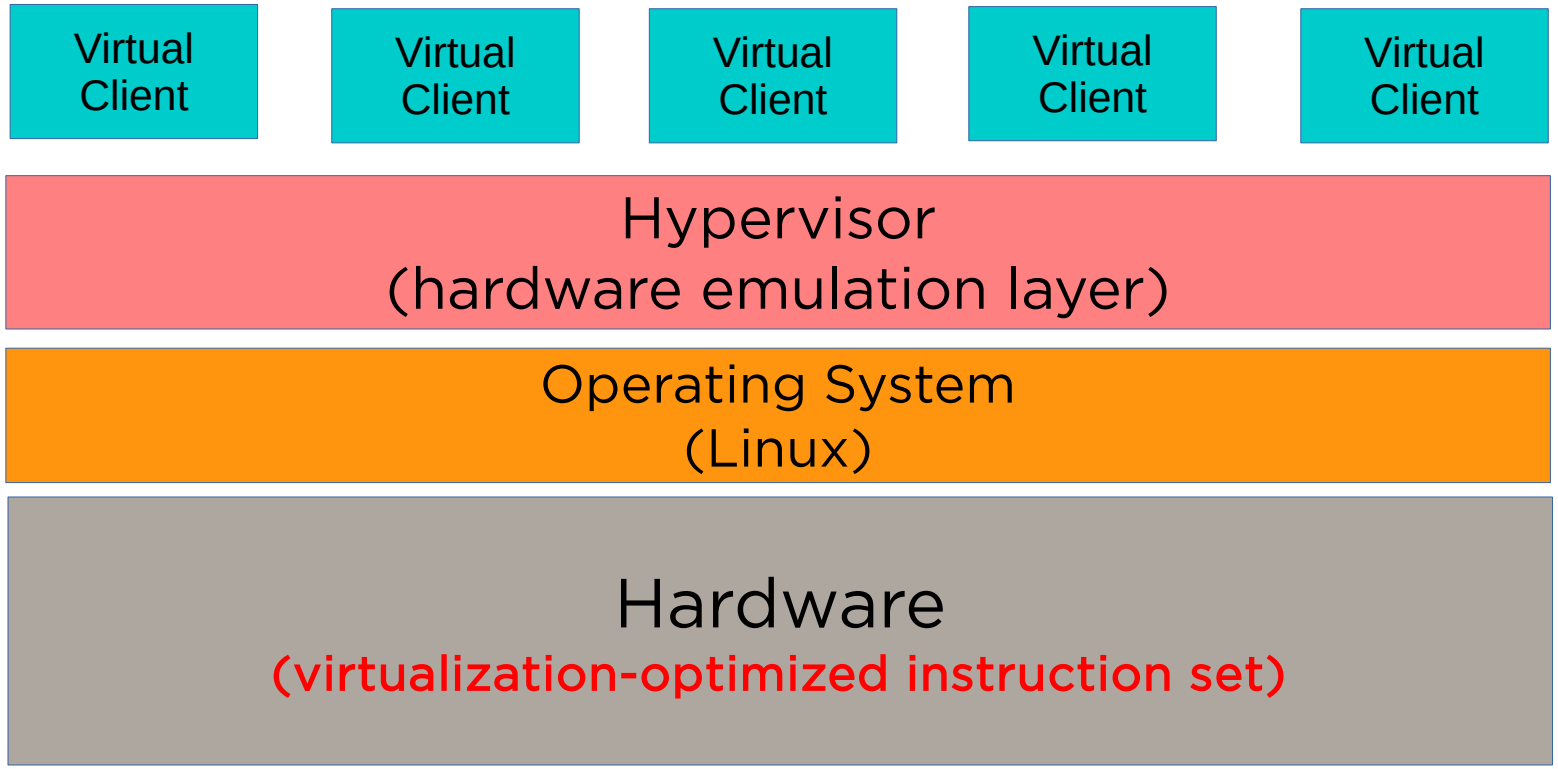
Design Considerations

- Fault isolation
- Performance isolation
- Consistency
- Server sprawl control

Hypervisor Categories:

- Type-1
  - "Bare-metal"
    - Xen
    - ESXi (vSphere)
- Type-2
  - System processes
    - VirtualBox
    - QEMU

Hardware Virtual Machines (HVM)



Cloud Compute Models

IaaS (Infrastructure as a Service)
- AWS EC2
- Azure Virtual Machines
- Google Compute Engine

PaaS (Platform as a Service)
- AWS Elastic Beanstalk
- Google App Engine
- Cloud Foundry
- Heroku

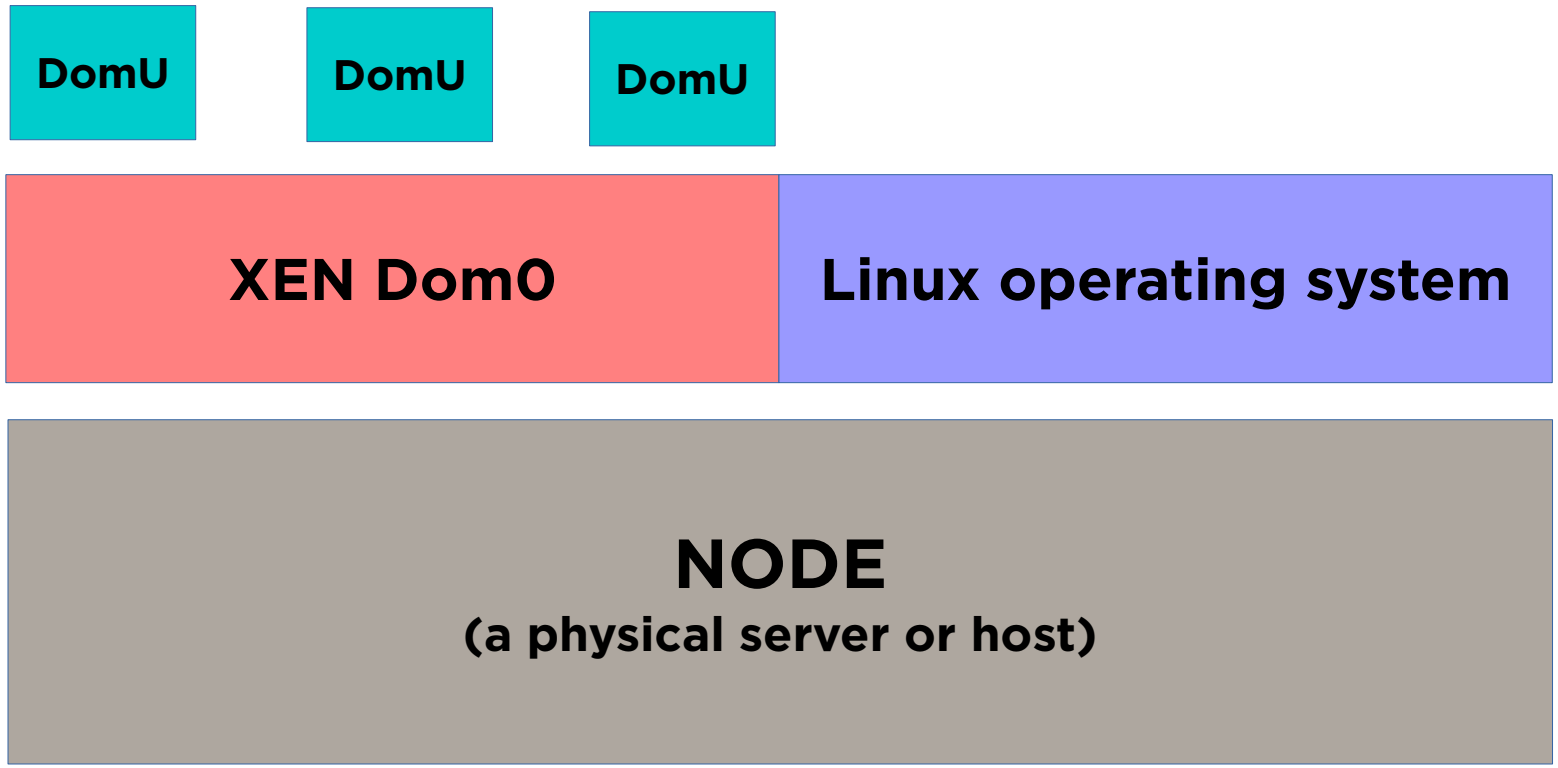
SaaS (Software as a Service)
- Google Apps
- Salesforce
- Dropbox
- WordPress.com

Sample XML Configuration File

Courtesy of:
http://libvirt.org/drvqemu.html

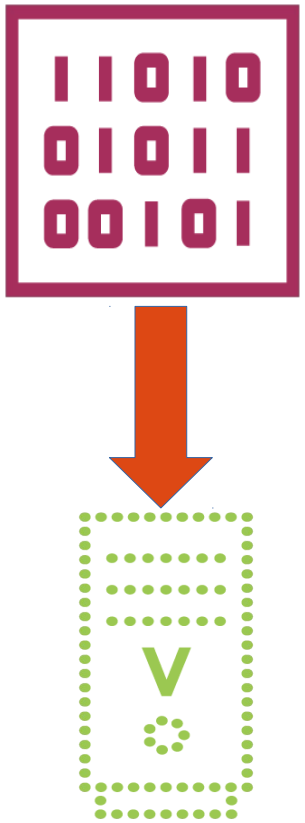
```
<domain type='qemu'>
  <name>QEmu-fedora-i686</name>
  <uuid>c7a5fdbd-cdaf-9455-926a-d65c16db1809</uuid>
  <memory>219200</memory>
  <currentMemory>219200</currentMemory>
  <vcpu>2</vcpu>
  <os>
    <type arch='i686' machine='pc'>hvm</type>
    <boot dev='cdrom'/>
  </os>
  <devices>
    <emulator>/usr/bin/qemu-system-x86_64</emulator>
    <disk type='file' device='cdrom'>
      <source file='/home/user/boot.iso'/>
      <target dev='hdc'/>
      <readonly/>
    </disk>
    <disk type='file' device='disk'>
      <source file='/home/user/fedora.img'/>
      <target dev='hda'/>
    </disk>
    <interface type='network'>
      <source network='default'/>
    </interface>
    <graphics type='vnc' port='-1'/>
```

XEN Architecture



The KVM Process

STEP ONE:
Create or modify image with qemu-img



STEP TWO:
Use image to start installation with qemu-kvm

NOTE: qemu-kvm = qemu-system-x86_64 and the kvm wrapper

KVM Managers

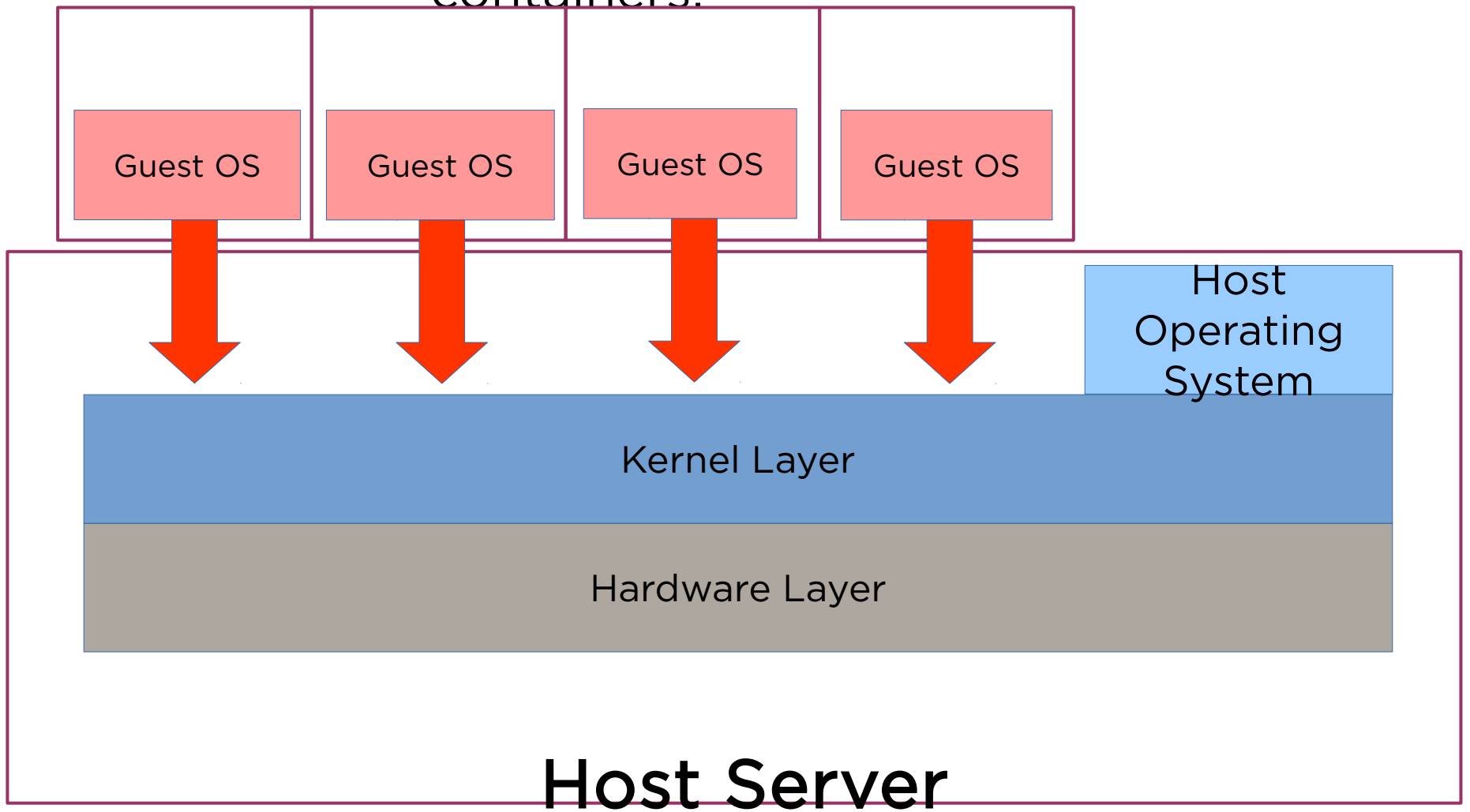
- libvirt (virsh)
- virt-manager
  - Non-GUI version; including virt-install
- vmbuilder
- KVM

Sample Pool Entry from KVM .XML Configuration

```
<pool type="netfs">
        <name>virtimages</name>
        <source>
          <host name="nfs.example.com"/>
          <dir path="/home/datauser/current-files"/>
          <format type='nfs'/>
        </source>
        <target>
          <path>/var/current-files</path>
        </target>
      </pool>
```

LXC Design

Fully isolated LXC containers:



The Packer/Vagrant Process