Designing Private Clouds by Leveraging Systems Thinking

Ratnadeep (Deep) Bhattacharjee  
Head of Product Management  
ZeroStack Inc.
Agenda

- IT Infrastructure 101
- Evolution of Software
- Private Clouds Today
- An Alternative Approach
- Q&A
About me

1999

Software Engineer

2007-2009

Product Manager

2016
system design & management

IT Infrastructure 101
Software for Routine Office Activities

Virtual Desktops
Software to Run Core Business Apps

Finance
Supply Chain
Healthcare
Retail
Chip Design
BioTech
IT Infrastructure = Hardware + Software
That Powers Your Apps

Hardware

Server + Storage + Network

Infrastructure Software

Management

Operating System

App runs on it
IT Infrastructure Delivery Models

Public Cloud

• Ease of use
• Low operational headaches
• Pay as you grow
• Feature Agility

Private Cloud

• Greater control
• Predictable performance
• Secure & Compliant

• Diminished control
• Inconsistent performance
• Compliance & security

• High software cost
• Operational complexity
• Requires specialists
IT Stakeholders

CIO
IT Managers
Deliver business value
Reduce Costs
Define corporate IT policies

IT Administrators
Provision IT (day 0)
Operate IT (day n)
Enforce corporate IT policies
Make IT users happy
Make CIO happy

IT User (App Teams)
Build/Test Apps
Deploy Apps
Get resources NOW
What corporate IT policy?
Before we get into the details of Private Clouds…
… A look at how software has evolved
Pets versus Cattle
**Scale Up Architecture (Pets)**

- System designed to eliminate failure. Specialized hardware may be needed.
- Manual steps necessary to handle greater load.
Scale Out Architecture (cattle)

- System grows with addition of new “resources”
- Designed to handle failure. Failed resources are discarded.
- Required less maintenance
- Resource addition can be automated, reducing operational cost
- Fits the pay-per-use public cloud model very well (hence, the name Cloud Native)
API, API, API

Programmatic Interface

- Well Defined Feature(s)
- Well Defined SLA
- Well Defined Cost

Software + Hardware

Service

Features Include:
- Detailed analytics and reporting
- Easy-to-use APIs and webhooks
- DKIM customization

<table>
<thead>
<tr>
<th>EMAILS PER MONTH</th>
<th>EMAILS PER MONTH</th>
</tr>
</thead>
<tbody>
<tr>
<td>40,000</td>
<td>100,000</td>
</tr>
<tr>
<td>$9.95/mo</td>
<td>$19.95/mo</td>
</tr>
</tbody>
</table>

Get Started

SendGrid

Leadership, Innovation, Systems Thinking
Public Clouds Have Established Dominant Design: Everything as a Service

<table>
<thead>
<tr>
<th>Abstract The Infrastructure</th>
<th>Offer Services (*aaS)</th>
<th>Users pay based on usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public Cloud</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Programmatic Interface (API)

- **Compute** as a Service
- **Storage** as a Service
- **Network Services**
- **Database** as a Service
- **Big Data** as a Service
- **Software** as a Service (SaaS)
Software Defined Everything

Then

Delivered As Hardware

Now

Delivered As Software

Firewall

Load Balancer

Router

Networks
Private clouds of Today
What constitutes a Private Cloud?

**Consumption**
- API
- App Store
- Self-Service
- Automation
- Cloud Services

**Management**
- Policy Definitions
- Automation
- Capacity Planning
- Chargeback
- Multi Site Awareness

**Infrastructure**
- Orchestration Engine
- Monitoring
- High Availability
- Cloud Svcs Management
- Virtualization Engine
- Hardware Manager

- Components often delivered by different vendors
- Single vendor may deliver a “suite” of different products as a single offering
How Private Clouds are Normally Built

1. Choose compatible hardware
2. Make sense of all Software pieces
3. Learn the deployment tools
4. Take Expert Help

1 to 6 months

Finally!

Your Private Cloud!

Longer deployment cycle means slower time to value
Different Hardware For Different Purposes

Server Hardware

Storage Hardware

Networking Hardware

Server Management Software

Storage Management Software

Network Management Software

Complex Hardware Software Integration
Each hardware Component has to be optimized
Asymmetric Design (management puppies)

- Management Hardware and Software needs more attention
- High Availability = Twice the number of Controller hardware = More $$$
- Static setup of Controller infrastructure
- As application workloads increase more controller nodes have to be added
- Much more operational expertise needed to manage this system
All Components Run On-Premises

- Patches and upgrades have to be handled by the customer
- New Features cannot be frequently delivered
- Specialized skills needed to deal with maintaining the on-premise software
An alternative approach
Converge

Converge Compute and Storage

Offer Network constructs in software on the same hardware

Run Management Software on same hardware

Small Building Blocks. Scale as needed
No complex hardware + software integration.
Redistribute Management Software Elements

### Cloud

- ZeroStack User Interface
- Map-Reduce Analytics
- Stream Processing
- Monitoring & Event Processing
- Big Data Layer
- Visualization

### Data & Control Plane

- ZeroStack Control Plane
- HW / SW Manager
- Software-Defined Network
- Software-Defined Storage
- OpenStack Services
- Linux + Hypervisor (KVM)

Secure workload and data. Agility in consumption and management.
On Premise Control Plane: Symmetric & Scale Out Design

• No separate controller nodes. Distributed control plane

• Each host has a controller VM

• Each host runs a subset of management software

• As more hosts are added new instances of services are spun up (scale out automatically)
Self Healing

Distributed Cloud Services

Control Plane

1. Leader election
2. Host failure
   - Leader detects node and service failure
   - Leader computes new service mapping
   - Services migrated to match new service map
Self Healing

1. Non-leader nodes monitor leader health
2. Initiate new leader election when leader becomes unavailable
3. New leader restores previous leader’s state from Distributed WAL
4. Re-balance services as necessary
SaaS Platform Delivers Management Features

Management Features

Cloud Consumption Portal

User Interface

- In-Memory KV Store Cluster
- Persistent Store
- Smart Analytics Batch Processing
- Data Processing Cluster
- Cloud Gateway

Cloud

- Cloud Building Block

On-Premises

Cloud Gateway
Benefits of Alternative Approach

Reduce time to value
- Up and running in minutes
- Model ensures users spend less time setting things up

Reduce Costs
- Dedicated on-premises h/w has better performance
- Self Healing nature reduces number of experts needed

Scale Flexibly
- System can grow based on customer needs
- System is multi site aware

Keep IT secure
- Main control plane is on premises
Summary: An New Model for Cloud

- **Public Cloud**
  - User Workflows
  - Operations
  - Management
  - Apps & Data

- **Private Cloud**
  - User Workflows
  - Operations
  - • Feature agility
  - • Real-time insight and preventative analytics

- **On-Premises**
  - • Maintain control and visibility
  - • Data locality
  - Management
  - Apps & Data
  - User Workflows
  - Operations
  - Management
  - Apps & Data
Summary – A New Model for Cloud

Public Cloud
- Ease of use
- Low operational headaches
- Pay as you grow
- Diminished control
- Inconsistent performance
- Compliance & security

Public Cloud Experience:
- Deployed in minutes
- No expertise necessary
- Scale on demand

Private Cloud Control:
- Maintain data ownership
- Service reliability
- Better user experience

ZeroStack

Private Cloud
- Greater control
- Predictable performance
- Secure & Compliant
- High software cost
- Operational complexity
- Requires specialists
system design & management

Q&A