XaaS

An introduction for the broadcast industry

April 2017
Appliances

- Purpose-built dedicated
- Performs a specific task
- Little or no dependency on an underlying platform
Appliances in Broadcast

- Broadcast engineering is traditionally **appliance** based.
- The **roles** of these are typically well-defined.
- **Signal flow** dominates design.
Software Based Appliances

- Many are entirely software based
- Standard operating systems and COTS x86 hardware
Computers in Disguise

- Often avoid management by IT
- Failed to capitalize on benefits of data center technology
Why Appliances?

Developers

Broadcasters

Vendors
Why Appliances?

**Developers**
- Resource contention
- Always on
- Dependency control
- Simplified testing

**Broadcasters**
- Familiar designs
- Simple failure blocks
- Minimal system interdependency
- Simple redundancy
- Easy troubleshooting
- Single-vendor support

**Vendors**
- Guaranteed performance
- Simplified supportability
Why *not* Appliances?

- IT has the **same problems**, has achieving **gains** in efficiency, scalability, and reliability
- Recent major progress in real-time production **video over IP**
- IP “plumbing” between appliances doesn't realize the full **potential** of a broadcast data center
The Future

Appliances → Virtualization → IaaS → SaaS
Virtualization in IT

- IT industry has been virtualizing since 1998*
- x86 virtualization exceeds 75%* in most organizations

*Source: Babcock, 2016
Hypervisors

**Type 1**

- VM
  - Applications
  - Dependencies
  - Operating System
- Hypervisor
- Bare Metal

**Type 2**

- VM
  - Apps
  - Dependencies
  - Operating System
- Hypervisor
- Operating System
- Bare Metal

deployed by Telestream
Virtualization Performance

*General Compute*

96%*

*Developers*

96%


(Calculated using mean of ESXi, Hyper-V, Zen compute overhead for Linux and Windows VMs)

*Source: Telestream Tests*
Virtualization Management

Unmanaged Virtualization → Managed Virtualization → Private Cloud

Complexity of management system
Integration of infrastructure system
Steps to deploy new virtual instances
Virtualization Management

- Managed Virtualization
- Inf. Request
- Provision
- Store
- Virtualization System
- Apps
- Network
Virtualization Management

Private Cloud

Inf. Request → Provision

Cloud Manager

Hypervisor

Images

Store

Apps

Network
Virtualization Management

Private Cloud

Provision

Cloud Manager

Hypervisor

Images

Store

Apps

Network
Virtualization Management

Private Cloud

Provision - API

Cloud Manager

Images

Hypervisor

Apps

Network

Store
Virtualization Benefits

• Hardware utilization
  – Combine multiple VMs per blade
  – Less hardware $
  – Less power, cooling, labor, less real estate $$

• Standardization
  – Single hardware platform
  – Blades are simple generic modules
Managed Virtualization Benefits

• Portability
  – Between sites, between data centers, DR

• Ease of Deployment
  – Software task, not physical

• Elasticity
  – Systems can grow and shrink
**Heavyweight Virtualization Benefits**

- **Increased hardware utilization**
  - Software already saturates the bare metal
- **Standardization**
- **Portability**
- **Easy Deployment**
- **Elasticity**

Still Apply
XaaS
Software as a Service
Platform
Infrastructure
Other Stuff

telestream
XaaS

• Usually refers to commercially available services, wherein the underlying dependencies are abstracted away

• Sometimes refers to the intra-organization or even intra-system services abstracted internally

• Powered by virtualization, containerization and orchestration technologies
XaaS Offerings

**Infrastructure**
- Functionality
- Applications
- Dependencies
  - Operating System
  - Hypervisor
  - Bare Metal

**Platform**
- Functionality
- Applications
- Dependencies
  - Container Host
  - Operating System
  - Hypervisor
  - Bare Metal

**Software**
- Functionality
  - Application
  - Dependencies
  - Operating System
  - Hypervisor
  - Bare Metal
Public vs. Private

Public
- **external** provider
- Usually offered to **many** customers

Private
- **Internal** offerings
- Owned/managed by the **customer**
Public vs. Private

**Public**
- OPEX
- Bandwidth/sovereignty concerns
- Their security
- Scaling on demand
- Customization of abstracted

**Private**
- CAPEX (hardware)
- Local data
- Your security
- Scaling to your capacity
- Full control
Practical Considerations
Redundancy

**Hardware**
- Compute failure
- Disk failure
- Power failure

**Software**
- Software malfunctions
- Data corruption
- Maintenance errors
- Operating System instability
Redundancy

1+1 Appliance

- Application
- Operating System
- Bare Metal

Software

Hardware
Redundancy

1+1 Unmanaged Virtualization

VM

Application

Operating System

Hypervisor

Bare Metal

Software

Hardware

VM

Application

Operating System

Hypervisor

Bare Metal
Redundancy

1+1 Managed Virtualization + HA

VM
- Application
- Operating System

Cluster
- Hypervisor
- Bare Metal

Software

Hardware

telestream
Security

API Approach

Your Network

Public Internet

VPC

*
Security

VPN Approach

Your Network

VPN

VPC
Security

VPN Approach

Your Network

VPC

*telestream*
## Latency

<table>
<thead>
<tr>
<th>Transaction</th>
<th>Transactions</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP File Transfer</td>
<td>1</td>
</tr>
<tr>
<td>SMB File Transfer</td>
<td>19</td>
</tr>
</tbody>
</table>

Centralized Configuration

Fixed Instance(s)

Discrete Configuration
Centralized Configuration

Dynamic Instances

Discrete Configuration Manual Management
Centralized Configuration

Dynamic Instances

Centralized Configuration
Humans / Design

Core

API

Media

API

Studio

API

Control

API

Media

API

Infrastructure

API
Thank You