SMPTE Meeting Presentation

Cloud Transition Patterns for Media Enterprises

Shailendra Mathur, V.P. Architecture

Avid Technology

Gerald Tiu, Sr. Program Manager Azure Storage

Microsoft

Written for presentation at the

SMPTE 2017 Annual Technical Conference & Exhibition

Abstract. *Media enterprises are increasingly looking at how the cloud can be harnessed to support operational and business agility. With these transformations they are evaluating the implications to their business models, workflows and technology for content production. Multiple hosting choices are available to move equipment investment from machine rooms to centralized data centers and cloud environments. In moving the process running on bare metal in the machine to shared resource infrastructure in data centers or cloud, multiple choices of virtualization technologies are also present. On one hand, fast transformations are to be achieved by lifting and shifting known applications and appliances using virtual machines. On the other hand, use of containerized micro-services and cloud native architectures offer promise of agility and cost efficiency. The usefulness of each option will be presented by understanding current media production infrastructure needs and its mapping to virtualization as hosting technologies become available. The post production industry's deployment preferences leads to a notion of a common software platform to host media processes across the various cloud infrastructure choices available. In hybrid deployment models, the message*

bus based communication framework provides tremendous value and flexibility in order to ease the technology transformations.

Keywords. Cloud native, microservices, lift and shift, containers, VMs, Platform, Message Bus, Azure, MediaCentral, Avid, Microsoft,

Introduction

As the media industry undergoes a transformation from traditional on-premises infrastructure to cloud technologies, multiple infrastructure hosting choices and porting methods are available. This paper analyzes the different transition patterns in the media industry and describes solutions as well as a platform approach to ease this transition.

Cloud adoption and data center based centralization is well underway in the media industry. Different segments of the market have different product, workflows that corresponds to differing infrastructure needs. For some market segments such as computer graphics rendering, or VFX, cloud adoptions for associated compute-intensive processes is already very high. For other segments such as film post production, broadcast sports or news, certain stages of the media creation chain may see higher cloud adoption than others. For example, cloud technologies are amenable for encoding and streaming for film dailies processing at the beginning of the media creation chain. Similarly encoding and streaming for Over The Top (OTT) distribution sees heavy cloud storage and compute usage as well. Certain processes such as broadcast contribution and in-studio post production pose some special challenges in terms of infrastructure requirements.

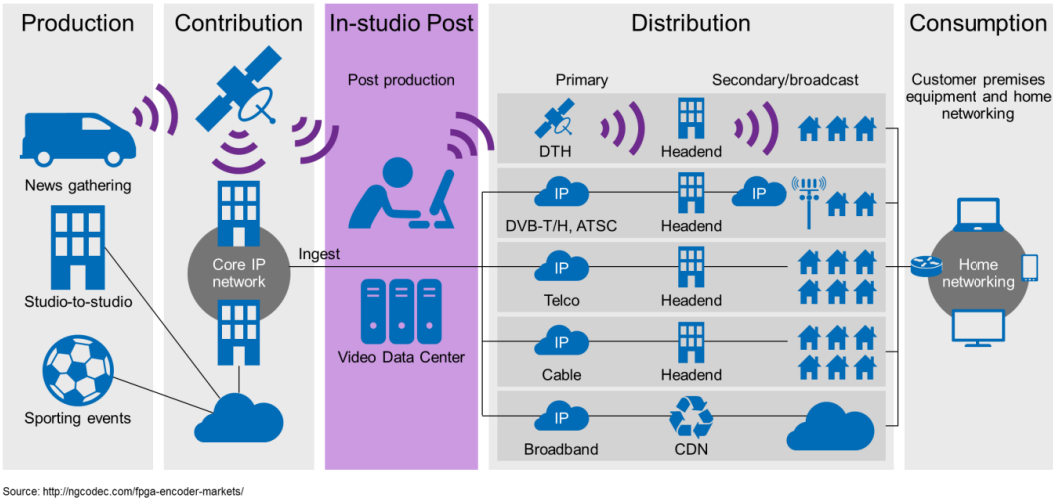

Figure 1 In-studio Post Production in the Media Production Chain

This paper discusses the infrastructure transition patterns using the requirements of in-studio post production environment in a broadcast chain as illustrated in Figure 1. As applied to news and sports productions, this environment brings in several different facets of media infrastructure and functionality such as real-time graphics, low latency editing and fast turn-around workflows that are good challenges to overcome in the transition from on premises to cloud infrastructure.

Along with infrastructure hosting choices, there is an associated debate around how existing media processes and applications should move to shared-resource and elastic model that centralized data centers and cloud provide. One approach ports existing software services and applications in their original form on virtual machines that mimic bare metal servers or

workstations. The other approach involves refactoring existing products into more atomic and dynamically composable services known as microservices. The former is known as the lift and fhift model, while the latter is known as a microservices-based approach. The lift and shift approach preserves person-years' worth of technology and intellectual property invested into the products that are being used within on-premises infrastructure today. On the other hand, the cloud-native implementation of small microservices provides scalability, cost-efficiency and resiliency that is ideal for the elastic and shared resource model that cloud infrastructures provide.

The question addressed in this paper is whether there is a method to ease this transition as cloud technologies catch up to the infrastructure needs of media productions. The answer lies in picking the right infrastructure hosting model, the right virtualization method and the use of a software platform that eases the transition.

To understand the transition path, this paper delves into the definition of the various hosting models as observed from plans from multiple media enterprises. It describes how the term Cloud is indeed used to describe a variety of hosting patterns. A discussion of different virtualization concepts that underlie the lift and shift versus microservices-based approach follows. The following sections identify the different types of products that are used in a post-production environment and the associated infrastructure needs. Finally, a software platform is discussed which allows for hybrid deployment models to be connected. The platform preserves the investments in traditional server and client applications while bringing the efficiency of new cloud-native microservices.

Cloud definitions and deployment patterns in the media industry

The goal of this section is to understand the different infrastructure as a service (IAAS) hosting patterns being applied by various Media Enterprises as part of their cloud transformations. The capabilities of each IAAS hosting model will impact the choice of products and workflows to be ported; the goal then is to understand the differences and provide nomenclature to each.

To start with, the U.S. National Institute of Standards and Technology provides the following definition of cloud computing:

> "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

The NIST website calls out five essential characteristics of cloud computing

- on-demand self-service
- broad network access
- resource pooling
- rapid elasticity or expansion
- measured service.

In addition there are four deployment methods identified.

- private
- community
- public
- hybrid

This section will call out additonal deployment methods that are variations of the NIST definition. Thoough they may be variations, their characteristics are important enough to solve some of the challneges for the media enterprises. It should be noted that there will be overlaps between the different models as well.

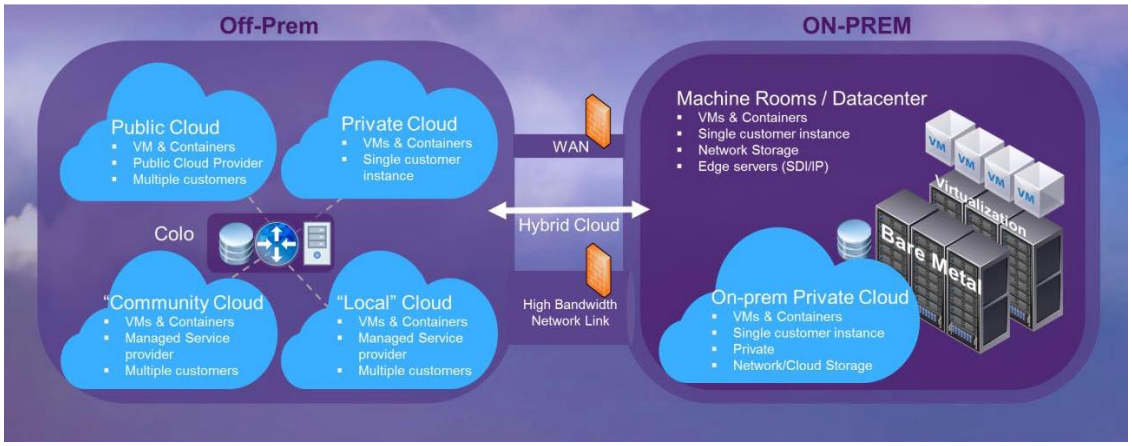

Figure 2 Cloud definitions as observed from media industry deployments

Machine room

Workstations and servers running within the studio space or dedicated suites are the first frame of reference. Machine rooms are the next reference that offers shared space, cooling and network for the infrastructure. These rooms typically belong to different departments. Bare metal servers and workstations and shared storage are typically installed there. Specialized SDI, Internet Protocol (IP) video, dedicated audio cabling is the norm here. Routers and switches for dedicated cabling as well as IP network switches define the media backbone in this traditional hosting model.

Machine rooms can host the full gamut of ingest, egress, asset management and client workstations, servers and storage. Due to short distances, the controllers such as mixers, edit controllers, color correction surfaces can be installed. KVM switches allow access to multiple workstations from the same keyboard, video monitor and mouse controls.

Data center

The first step towards a truly shared infrastructure on premises starts when different departments start sharing infrastructure running in common data centers. Data center based operations have been very common in the IT industry, but are becoming common place for shared media processing usage as well. The data center can be operated by the media

enterprise itself or by another hosted service provider. Regardless it is used only by a single media enterprise.

While bare metal servers running multi-tenant services are common in this environment, to achieve isolation or to provide on demand elasticity, virtual machines (VMs) can be used with commonly available hypervisors.

With bare metal and virtualized servers, comes the implication of moving away from dedicated cabling for ingest and egress processes. IP based interconnects play a bigger role in this model. Certain IP protocols that are LAN only, can work in this environment.

Certain SDI based servers can be co-located in the data center, but are typically connected to the IP backbone using SDI to IP converters.

Various tiers of shared networked storage and asset management systems are a natural fit here since they are typically centralized resource anyway.

If workstations or server-based desktops are hosted here for creative tool applications, HID devices and dedicated controllers will work only if the protocols used for the remote operation are IP enabled. IP based KVM or VDI remoting is useful in this situation as well as application models that rely on streaming media displays and web UIs.

Workgroup separation that was naturally enforced due to redundant infrastructure deployment for different productions can still be enforced through resource isolation. The advantage of the centralization is that based on the need of the production, the resources can actually grow or shrink elastically across the pool of resources available to serve multiple departments.

On-prem private cloud

A natural transition from the data center would traditionally be called a private cloud, but to reflect the fact that this version is hosted in customer owned datacenters, it will be referred to as an “On-premises private cloud” in this paper. The infrastructure model is similar to the data center model, but now the infrastructure management software changes to a cloud-management software stack. It can be maintained by the customer or third party provider. Dynamic resource allocation becomes much easier in this case since resource allocation and sharing is handled by the cloud vendor’s software. Azure Stack and VMware Cloud Foundation are examples of cloud software that manage on-premises data centers.

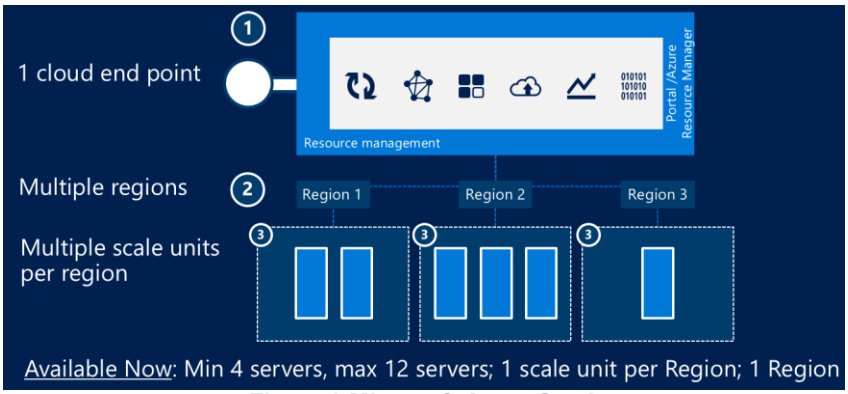

Figure 3 Microsoft Azure Stack

Utilizing Azure Stack allows for a common cloud software implementation shared between on-prem and cloud based deployments. Both VM appliances and cloud-native microservices can run in this environment. For products developed against a cloud-native framework, the development time and complexity of dealing with different PAAS frameworks gets eliminated.

To simplify the resource allocation and accounting by the cloud management software, the data center is composed of *Scale Units* of the same hardware infrastructure and services. Hence horizontal scalability is achieved by increasing the scale units. As per the figure above, while Azure Stack manages the scale units in the local region, it is also capable of elasticity beyond the current region. It's capable of binding together resources in different regions to effectively provide a single cloud identity.

With the compute and network hardware defined by the cloud vendor within the scale unit, custom hardware cannot be managed by the cloud software. However there are opportunities for equipment such as shared storage and edge servers to be hooked up to the network provided by the on-prem private cloud. The close proximity to the client applications makes many of the low latency, high bandwidth operations possible.

Private cloud

Next up is the increasingly familiar territory of private cloud. A private cloud can either refer to an on-prem or an off-prem cloud infrastructure. However the earlier section discusses the value of calling out the on-prem version separately from the off-prem version. Rgeardless of location, exclusive use of dedicated resources is guaranteed for a media enterprise. Isolation is provided either virtually through software or, in some cases, physically.

Due to the different infrastructure assumtions, the capability to host post production processes will follow either the on-prem private cloud model discussed previously, or the public cloud model. In the latter case, unlike the on-prem private cloud the distance from the facility may not allow for low latency workflows with on-prem equipment.

As the infrastructure assumption move to an off-prem model, one option available is to take advantage of colocation centers (Colo). These *colo* centers are facilities that provide custom infrastructure hosting facilities and virtual private high bandwidth connectivity and proximity to the cloud data centers. Due to the use of colo based custom infrastructure, workflows that require minimal latencies can be accommodated. Edge server or dedicated storages are examples of equipment that can be located here. For example editing operations that require low latencies and fast turn-around, can access media being ingested through edge servers on shared storage located in the Colo facility.

Community Cloud

A community cloud is a variant that shares infrastructure between several organizations that have similar concerns (security, compliance, jurisdiction, etc.). The shared needs can also be around similar infrastructure such as particular types of secure shared storage, ingest servers; Specific products such as remote editing and asset management; specific security compliant workflows qualified by audit agencies.

Local cloud

Local cloud is not a standard NIST definition, but can be seen as a variant of the community cloud with the caveat that it can also be a variant of a public or private cloud. The primary requirement here is of network and geographical proximity to the media enterprise facility. This proximity facilitates provisioning for high bandwidth and low latency connections to the client's facilities. As seen from previous sections, the low latencies and high bandwidth requirement are particularly useful for dealing with edge servers and for shared storage access that may be sitting on-prem instead of a colo facility. Cloud vendors suzh as Microsoft Azure provide technologies such as ExpressRoute that provide a private connection of on-premises equipment to Azure.

Public cloud

The public cloud is the standard multi-tenant shared infrastructure. From a media customer point of view, one big advantage over other hosting methods is the presence of infrastructure in multiple regions.

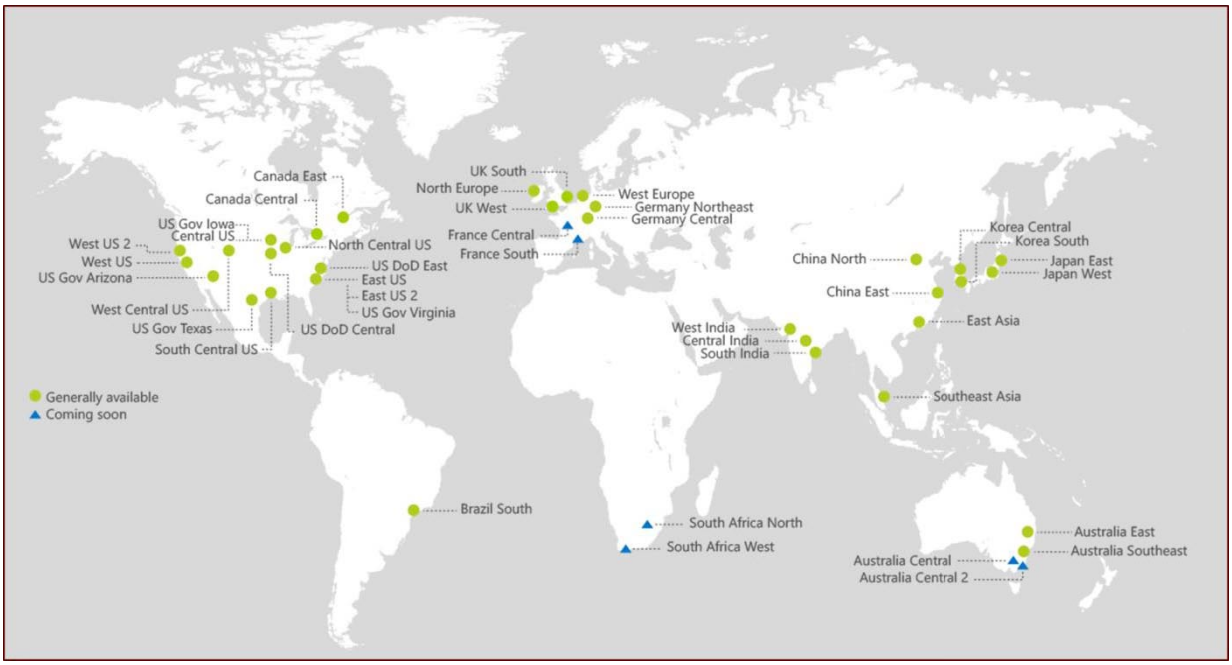

Figure 4 Microsoft Azure Regions

This distributing of data centers allows geographically distributed enterprises to work in a consistent fashion. The same geographical distribution allows for collaboration use cases. One example of this is the use by Avid of Azure public cloud for Pro Tools artists to collaborate across various geographies.

The hosting abilities of the post-production processes follow that of the private cloud; however, new multi-zone workflows become possible due to geographical replications.

From this section, the various definitions and capabilities the various cloud hosting models should help identify which environment is best for different types of workflows. It's not an either-or world. Either due to only partial satisfaction of infrastructure needs or the need for security for specific stages of production, hybrid deployment are preferred by many media enterprises. A hybrid environment also offers a solution to the burst-out needs of the enterprise during peak load. A lift and shift approach is very favorable to deal with the burst-out capacity issue since the workflows, products and tools remain consistent with what is being used on-premises.

Virtualization technologies for lift and shift and microservices porting

One of the goals of moving post-production processes to the cloud or centralized data centers is to benefit from resource pooling – whether to provision more processes on underutilized servers, or to aggregate resources to get even more powerful computing power. Another goal is of elasticity to deal with burst-out capacity under heavy load conditions. These conditions could arise due to a major news story or for a temporary setup of a production environment for a special project.

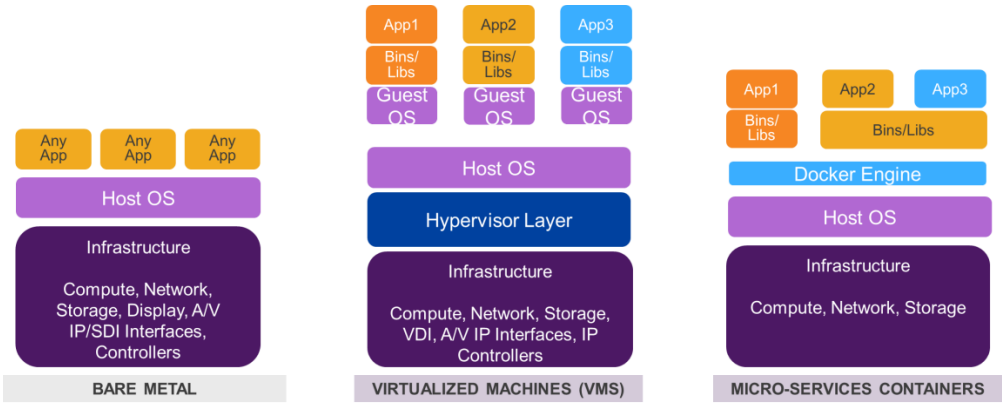

Figure 5 Bare metal, VMs and Container comparisons

A bare metal server typically runs a single operating system exposing a single set of hardware resources. For media processes while scalability can be achieved by running multiple processes under the host OS, it is up to the app developer to allow handshaking to share the same hardware resources such as dedicated SDI/IP ports, GPU resources and controller interfaces.

To allow for the same hardware resources to be shared across multiple media processes without the specific knowledge built into the apps, virtualization technology is utilized. Using resource partitioning, it also allows for shared access to the CPU, memory, network, storage and now virtual GPU access to simultaneously running services. For processes such as CPU based transcoding that requires standard compute, network and storage resources, the resource sharing can be very efficient. However processes such as baseband ingest that may need to work with non-shareable resources such as an SDI or IP port will not provide elasticity.

The virtualization technologies of interest that provide different capabilities for resource pooling and elasticity are Virtual Machines (VMs) and Containers.

VMs emulate a full server or workstation by running a full copy of the OS as well as virtualized drivers and emulations of the hardware. In comparison, containers take just enough of the operating system to support binaries and libraries required for applications or services running inside the container. Multiple containers can run in a single OS instance – providing resource isolation at a more granular level than VMs.

With VMs emulating bare metal OS and machine functions, applications and services can be run with almost no changes. Hence it is associated with the lift and shift approach. It should be noted that the biggest hurdle in the lift and shift of desktop applications are the ones currently based on macOS. VMs almost exclusively support Windows and Linux operating systems. Support for both Windows and Linux is now also present within containers. Hence application built for these OS's or applications based on web browser technologies are the only ones that can run in these environments.

An associated issue to watch for is the qualified support for custom drivers under hypervisors. While the application maybe portable under the lift and shift model, special attention needs to be placed for human interface devices, controllers, as well as I/O devices to serve in a virtual environment. Virtual desktop interface standards help solve some of these requirements. Certain App virtualization technologies can also pass-through or properly support certain custom drivers.

Containers can be used for large monolithic applications without refactoring, but they are most effective when associated with microservices. Container technologies are built for lightweight and rapidly deployable uses. As an example, containers built with Docker technology can be deployed rapidly across multiple infrastructure providers using container management, auto scaling and deployment software such as Kubernetes.

Decisions on which media-related processes to port to VMs and which ones to container are based on several factors

- time to refactor existing application and services into smaller microservices,
- ability of the original service to cleanly startup and shut down,
- availability of virtualized access to specialized hardware
- impact on performance
- uptime assumptions
- resource availability to allow for dynamically scalability

In general, unless dedicated hardware is involved, almost all post production processes will port easily to VMs. Existing products should be evaluated for refactoring using the criterion above as guidance. Of course, any new functionality should ideally be built as composable atomic microservices if a platform is available to execute them.

Post-production infrastructure transition

Figure 6 Post Production Workflow & Infrastructure

To understand the infrastructure needs of post-production products running in a cloud environment, it is necessary to understand the existing on-premises infrastructure used by current products in the studio within post-production workflows.

Workflow Transition

The processes for post-production can be grouped into four categories – ingest, asset management with associated shared storage, creative process, and finally packaging and egress of the prepared material for playout and delivery. In the transition to new hosting models, while the mechanics of achieving the workflows will change, these workflow processes themselves don't change. In a similar fashion, for each individual product participating in the workflow, the implementation may change but the behaviors that the creative talent works with should not change.

As an example of process-interactions being preserved, fast turn-around workflows require editing to start on media while it is being ingested on to shared storage. In a similar fashion, a playout through a video server can start while media is still being written to disk. Performing the same operation on cloud infrastructure requires the preservation of streaming writes to shared storage system, simultaneous read operations on the growing files from cloud based applications, and notification channels set up between applications to update the client applications. Any assumptions of full file transfers after ingest or renders are finished, or missing communication framework between the devices would not work.

As an example of preserving the user experience of creative talents with the applications is the full support of pen and tablets for drawing and manipulating shapes. When an application is running in a remote cloud environment the tablet for input and the desktop display local to the user needs very low latencies for best hand eye coordination. VDI solutions are increasingly

geared to solve the low latency problems; however the network latency and bandwidth between client and host will effect the performance.

Usage of a cloud infrastructure can on also open up new workflows that did not exist with in-studio infrastructure. In the audio world, multiple users collaborating across geographies by exchanging tracks and sessions with each other would not be possible if not for the public cloud infrastructure. While the creative tools for that implementation may be on-premises, the full collaboration engine is built as a microservices architecture.

Infrastructure Transition

To make it easier to map to the cloud infrastructure world, the terminology used to describe cloud resource consumption is used to characterize the infrastructure used in post-production workflows. The categorization is for connectivity, compute and storage usage.

Connectivity

Several types of connectivity mechanisms are used in studios.

- Specialized connectors between host machines and peripheral devices
- Video and audio cabling such as SDI or AES/EBU
- Internet protocol (IP) networks between computing nodes and data stores.

Most of the connectivity to cameras, decks and audio devices is based off of analog and digital standards that have been around for years. Whether for ingest, egress or for monitoring in any of the stages of post-production, connectivity to most professional equipment has been through SDI, AES/EBU, XLR and other such interconnects and cabling. This cabling goes through routers and switchers based off these standards.

The introduction of IP based audio and video standards such as SMPTE 2022, SMPTE 2110, AVB and AES67 is paving the way for a transition to IP based networks from traditional network technologies. The move to cloud will rely on this IP transition because the special purpose networks previously in use are unlikely to be available in the cloud.

The infrastructure in-studio has always had an IP network backbone to work with network based shared storage and for regular control and data traffic between media systems. One benefit of the non-IP cabling and routing till now has been the effective segregation of IP network traffic from the network traffic created by ingest, egress and monitoring applications. These applications have strict latency requirements. For cost-effectiveness, it is highly desirable that a common IP network infrastructure be shared among all applications. In addition to increased capacity, quality of service, bandwidth shaping, and routing features of IP networks will need to ensure non-interference with regular operations of other systems on the same network.

In a studio environment reference signals have been a mainstay for multi-cam scenarios. Multi-channel synchronized mixing in downstream products. IP video and audio standards are allowing that synchronization to be applied in virtualized world as well. Since low latency operations are important, for certain activities such as interactive editing the latency of the network and number of hops needs to be taken into account in the move to the cloud.

File-based ingest and egress of various camera cards, shuttle drives and physical media such as XDCAM optical disks required the ingest operator to have access to client-side workstations where these devices can be mounted and used. These devices sometimes require special drivers to mount these devices. As part of any move to cloud or on-prem data centers, this local access to mounted devices needs to be factored in.

Within a studio or machine room environment, it's easy enough to connect a workstation through HDMI cables or other specialized video and audio interconnects to dedicated audio and video monitors. Similar cabling is of course not possible, however using a combination of IP based video and audio standards as well as VDI technology the same desktop and broadcast monitoring information needs to be provided to remote users.

The same requirement holds for specific Human Interface Devices (HID) such as tablets. Specialized controllers used for audio mixing, editing operations and color corrections have the same user-side connectivity. They all require very low latency feedback with the system hosting the application. Some of the standard HID devices that typically go through the OS software stack also get support through various remoting protocols such as PCOIP or Microsoft Remote Display Protocol (RDP). To use these protocols, one method is to run software remoting clients on local workstations. There are also hardware clients such as the PCOIP Zero Client that offers common HID device connectivity. Other custom protocols as used with many of the specialized controllers such as audio mixers are not as easily available today for remote control unless they are IP enabled.

Compute

As can be expected, almost all compute processes in a studio workstation and machine room environment are heavily based on CPU processing. This typically uses common multi-core CPU technology from Intel and AMD. Compute intensive processes use multi-threading as well as SIMD instructions sets, which are again, ubiquitous. Most of this same compute power is present in virtualized form from the IAAS providers, albeit choices may be dictated by a specific set of pre-packaged VM instance types.

Dedicated GPUs are required for most professional media workstations but not necessarily available in all servers. GPUs are heavily used in the creative tools for graphics and video effect acceleration. They are the primary methods of driving display monitors. GPUs are also present in broadcast graphics servers or render farms accelerators for video VFX processing.

Moving into a cloud and data center world, GPU availability is not necessarily a problem but the choice and configuration of servers is much more limited. The GPUs in servers are typically compute-only GPUs since display interconnects are not typically needed. This is a natural consequence of the servers being remote from the user. While VDI technology interacts and in some cases uses the GPU for desktop display streaming purposes, configurations for creative tools running in remote servers may still need attention to ensure low latency and minimal bandwidth for effective display and control.

There are several media processing tasks for which dedicated silicon is still required for satisfactory performance. DSP-based audio processing is a good example. FPGA-based encoding and decoding during ingest and an egress process is another. In both cases customized I/O or processing cards have been a requirement. Programmable FPGAs are

starting to become available in certain cloud environments, however the primary direction here has been to depend on CPU-based acceleration of these processes. If issues such as latency requirements are solved by other means such as ensuring proximity and low latency networks, the creation of microservices provides the best promise for the cost-for-performance challenge that custom hardware typically solves.

Storage

Media productions are one of the heaviest consumers of different types of storages compared to other data intensive industries. Within the media industry the storage use is differentiated based on the speed and capacity required for the workflows.

The fastest storage is in fact the memory available within workstations and servers. During the execution of any media process it's where all intermediate results and application processes are stored. Since memory is universally available, there are no issues between in-studio or cloud environments.

Within servers and workstations, SSD or spinning disk based storage is the next requirement for temporary storage of intermediate results or ingested or egressed media. Again SSDs and spinning disk availability pose no problem between the different infrastructure providers.

Portable storage was already mentioned as an issue previously. This is purely due to the physical separation of the servers from the client-side device to which the user connects the portable media storage. There are solutions for those cases such as file-sharing software, USB forwarding through remote protocols and, client-side hardware clients such as the PCOIP Zero Client.

Custom databases require their own storage for storing metadata and asset information that is typically different from the tiered media storage. The storage tier is defined by the quality of service required for specific workflows. Parking or Archive storage can today use blob or block storage exposed through object storage APIs or as filesystems.

Collaborative real-time workflows in large post-production departments require guaranteed real time performance of multiple high bandwidth streams to multiple simultaneous clients reading and writing to the storage. This poses latency as well as bandwidth requirements on the infrastructure. The geographical separation makes a big difference in this context.

A platform based communication framework

In the previous section, the discussion led to conclusions of bare metal services, VMs and containers all having to co-exist as part of cloud transitions. Other sections provided different hosting models that may be suitable for specific media enterprise needs. Another observation was that many enterprises are choosing a hybrid deployment model with in-studio equipment connected to on-prem data centers, which in turn may be connected to processes running in the public cloud. These conclusions point to the need for a platform that provides a communication framework running within each hosting models as well as across them.

Most of the inter-service communication in media enterprises has been developed organically either by developers in these enterprises implementing home-grown frameworks, or using

specific SOA products offering workflows orchestrations, or with generic Platform-As-A-Service (PAAS) providers. Depending on the extent of use of product or platform frameworks, as well as the targeted hosting models, the complexity of transition will differ for each enterprise. The key here is that if the product or the platform is portable across the IAAS providers then the services built for them will be portable as well. However if the PAAS is tied to particular IAAS providers then the portability is hampered.

Within products and platforms that host services there are different communication patterns possible. Due to microservices being functionally atomic in nature, the need for a lightweight and fast messaging framework is much more necessary than it was with monolithic services that encapsulated multiple functions and could communicate within the process.

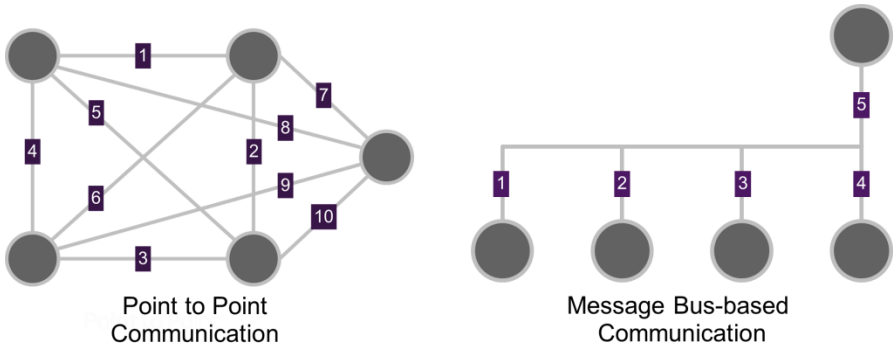

Figure 7 Inter-service communication patterns

Inter-service communication for mircoservices can follow two architectural patterns

- Point to point communication
- Event based execution

The point to point model shown in figure 7 requires each service to talk to the others through APIs by knowing their end points. When implemented as direct connections any resiliency has to be dealt by the services themselves. The direct API communication model proves problematic when the services need to run in different hosting domains. However there are implementation methods that provide brokering of API calls to offload some of the connection management.

The event-based execution model works by services executing in response to the reception of particular events to which they subscribe. This is the most decoupled method of execution since there is no need for services to know each other's end points. This communication model offers the advantage of new microservices being added dynamically without having other services knowing about them.

A message bus based communication framework

A message bus architecture provides a lightweight message brokering system that are part of a platforms running on the different hosting models.

The advantage of the message bus architecture is that it is an open standard application layer that other vendors can support. Another advantage is that it automatically takes care of queuing,

routing operations for messages as well as publish and subscribe operations for events. It handles resiliency and security.

The communication framework challenge of dealing with multiple hosting frameworks gets solved due to the bus federation capabilities it provides. The same bus can run in multiple hosting environments and pass messages between services that span those hosting boundaries. It can support both API based and event based execution models mentioned above. The message bus just acts as the conduit for messages that encapsulate the API calls and reponses, as well as event passing.

Bridging service communication across hosting and virtualization models

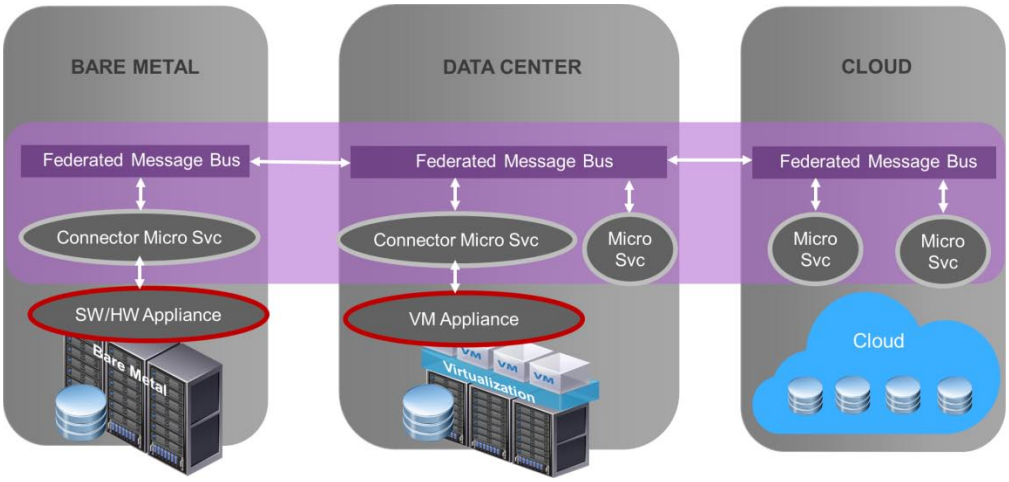

Figure 8 Communication across hosting and virtualization boundaries

As shown in the figure above, with the message bus running across multiple hosting environments, communication can be routed across hosting boundaries. The same bus can send messages to end points that act as proxies of other services running underneath. As shown in the figure above, hardware and software appliances can be abstracted out and can play nicely with other microservices through this connector service that acts as a proxy. The connector service can perform API based request and response actions on the bus or, it can participate in event choreographies on behalf of the appliance it communicates with.

Bridging Client applications across hosting and virtualization models

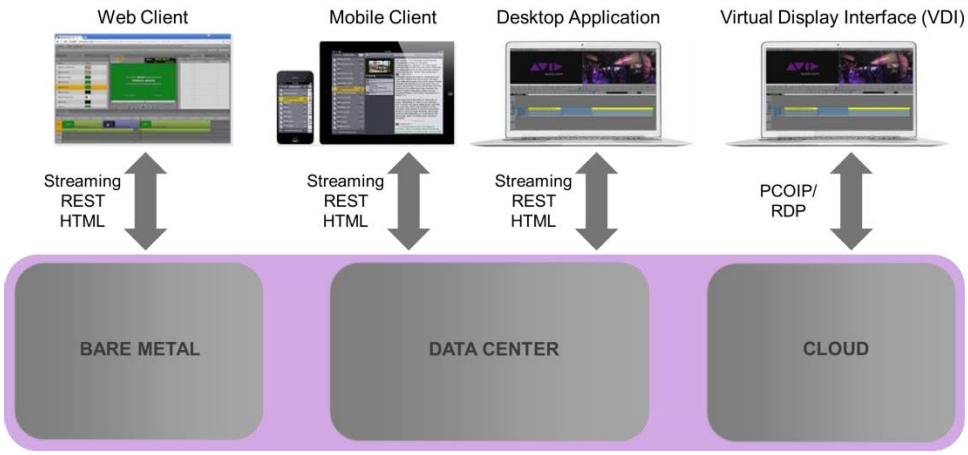

Figure 9 Client Application communication options

So far we've looked at the communication framework that allows connectivity between different hosting models, as well as to services running under the different virtualized and non virtualized deployment models.

The other critical part is the commnications framework with the client application frameworks typically used in post production. These applications can be control and monitoring applications for ingest and egress processes; They can be metadata logging, search and asset management applications; or they could be creative tool applications that need access to asset management or media services running in the hosting environments.

As was seen from the analysis of the post production processes, the applications can themselves run in to different environments – within mobile devices, web browsers, on desktops applications.

For browser or mobile hosting, HTML5 and Javascript based web apps provide the appropriate user presentation. For desktop apps, there is a choice of two modes - one running locally on a users desktop desktop and accessing media and metadata remotely, and the other with the application actually running on the back-end server, but providing a native client experience through Virtual Desktop Interface (VDI) protocols such as PCOIP or Microsoft's Remote Desktop Protocol (RDP). Both technologies have advanced rapidly recently for low latency and desktop interactions appropriate for media clients.

With VDI technology being the enabler, a major benefits of running the applications in a remote server is to ensure security and control. Security patches, user access and other security tools can be administrated centrally in the hosted environment compared to the multiple workstations that would otherwise need to be administered and maintained.

Storage media access to the web, mobile and desktop clients can be through the brokering of remote playback engines that use streaming protocols or on-demand sample access to the client applications. Storage media access for VDI-based options is through direct access to the media storage by the remote application running on the back-end sever.

The situation where local ingest or low latency connection to HID or specific controllers is needed, the option of running the desktop application on the remote client machine works best.

The situation where high-quality media access and fast performance on media stored in remote storage is required, the VDI based desktop option is best.

Any calls from the applications to the services needs to be through authenticated API calls in the communication framework before being passed to the same message bus framework that services themselves use to communicate with each other. With support for the communication framework across the hosting envrionment and the ability for the communication fabric to talk to any of the virtualization methods, effectively the applications can now be completely decoupled from service running under any of the hostnig models.

A media platform to bridge the hosting models

To implement the solutions prescribed for working with different virtualization methods, different hosted infrastructures, and communication methods, a common software platform is required. This platform is different from the one that a generic PAAS service provider would provide since as per previous sections, it is built for the special needs of media productions and creative processes.

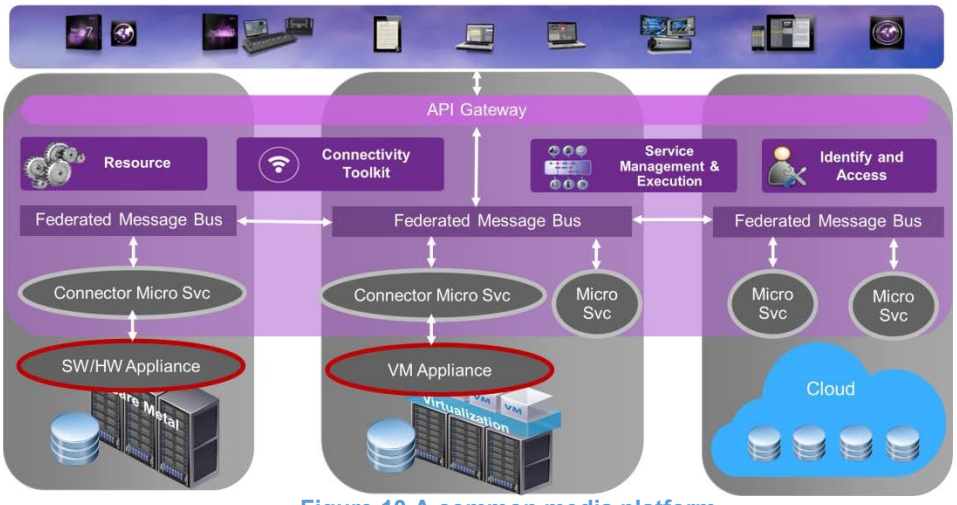

Figure 10 A common media platform

A base assumption of the platform is that it runs in each hosted environment, including bare metal, managing the resources available in those environments. It offers the common service management modules to register manage and execute the various microservices. This includes microservices and connector services built by various vendors through a connectivity toolkit. It offers security using an identity and access management system that works with other identity providers available on the hosting platform. Secure access to services is provided from external client applications through an API gateway.

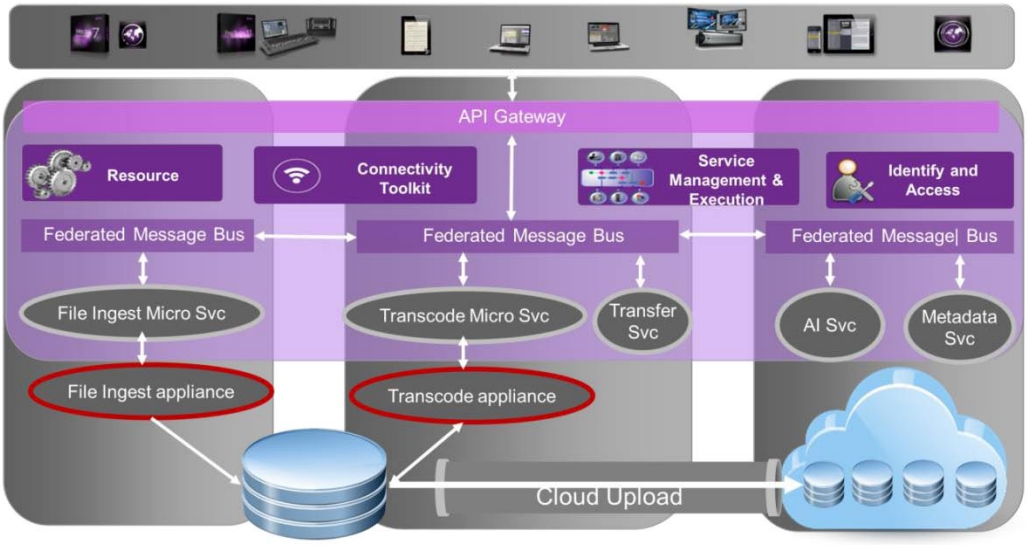

Figure 11 Media workflow example

The figure above shows an example of an event-based choreography possible with this platform. In response to a file ingest running in an on-prem bare-metal appliance, a virtualized transcode appliance in an on-prem datacenter can be triggered. The ingest and transcode processes use a network connected shared storage residing on-prem but connected to both locations. A completion of the transcode triggers a transfer from the on-prem storage to the cloud blob storage, where a microservice triggers an AI analysis service provided by the cloud vendor. The completion of that AI service triggers a service to store the metadata and to issue another event that is subscribed to by one of the application that consequently fetches the metadata through an API call to the API gateway for review.

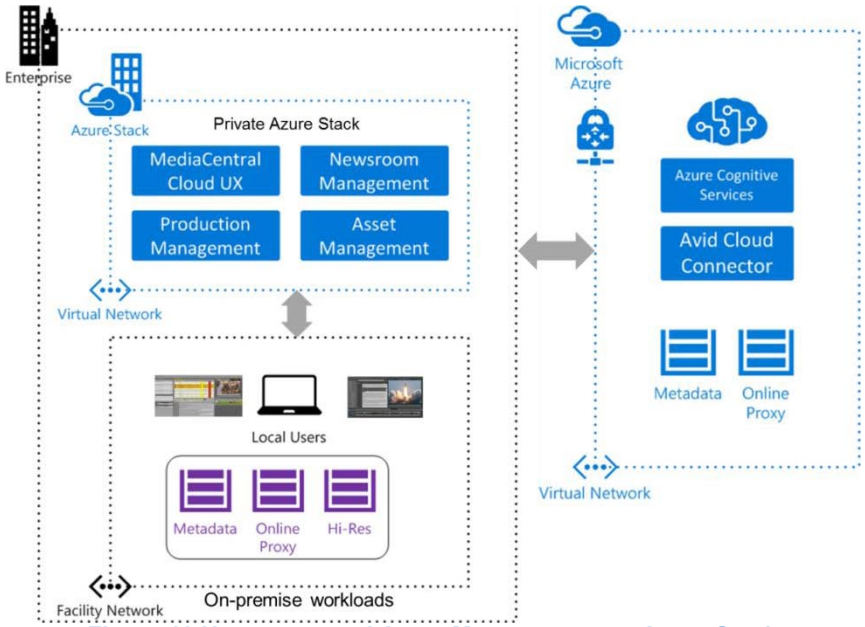

Figure 12 Newsroom and Asset Management on Azure Stack

Several aspects of the above-mentioned platform are already being used on-premises by multiple media enterprises. The public cloud hosting model has been shown during NAB 2017 and IBC 2017. Using an early version of Microsoft's Azure Stack, during IBC 2017, Avid and Microsoft showed a hybrid workflow between an on-prem private cloud and the Azure public cloud. Working with a large international news media organization, the workflow showed newsroom and media asset management modules accessed and edited through a rich web browser UI, as well as Azure public cloud based cognitive services running on assets stored on on-premises shared storage.

Conclusions

From this paper it can be concluded that the cloud definitions goes beyond the one from NIST to span multiple hosting patterns. The different hosting model described may be variations of established IAAS pattern, but are important to understand for media enterprises to make the right choices as they transition to the cloud.

The various opportunities and capabilities of the different hosting models, as well as the appropriate virtualization methods dictate which post-production processes should be be ported with a lift and shift approach, and which ones should be used in microservices framework.

To ease the transition from one hosting environment to another, and to bridge hybrid deployment models, the need for a platform to bridge the hosting and virtualization models was presented. This included an example of a real media workflow that was demonstrated at IBC 2017.

Web References

The NIST Definition of Cloud Computing, SP 800-145 (DOI), , 28th Sept, 2011

"Microservices - a definition of this new architectural term", James Lewis, Martin Fowler, 25th March 2014, https://martinfowler.com/articles/microservices.html

"Final Version of NIST Cloud Computing Definition Published", Oct 25 2011, https://www.nist.gov/news-events/news/2011/10/final-version-nist-cloud-computing-definition-published

ST 2022-6:2012 - Transport of High Bit Rate Media Signals over IP Networks (HBRMT)

What is SMPTE ST2110 and Why Does It Matter?, John Mailhot, 2017

