

Microsoft



Microsoft®
Silverlight™



Delivering Live and On-Demand Smooth Streaming

Video delivered over the Internet has become an increasingly important element of the online presence of many companies. Organizations depend on Internet video for everything from corporate training and product launch webcasts, to more complex streamed content such as live televised sports events. No matter what the scale, to be successful an online video program requires a carefully-planned workflow that employs the right technology and tools, network architecture, and event management.

This deployment guide was developed by capturing real world experiences from iStreamPlanet® and its use of the Microsoft® platform technologies, including Microsoft® Silverlight™ and Internet Information Services (IIS) Smooth Streaming, that have powered several successful online video events. This document aims to deliver an introductory view of the considerations for acquiring, creating, delivering, and managing live and on-demand SD and HD video. This guide will touch on a wide range of new tools and technologies which are emerging and essential components to deployed solutions, including applications and tools for live and on-demand encoding, delivery, content management (CMS) and Digital Asset Management (DAM) editing, data synchronization, and advertisement management.



A Step-by-Step Development Process

This paper was developed to be a hands-on, practical document to provide guidance, best practices, and resources throughout every step of the video production and delivery workflow. When delivering a live event over the Internet, content owners need to address six key areas, shown in Figure 1:

1. Production
2. Content Acquisition
3. Encoding
4. Content Delivery
5. Playback Experience
6. Management, Coordination, and Monitoring

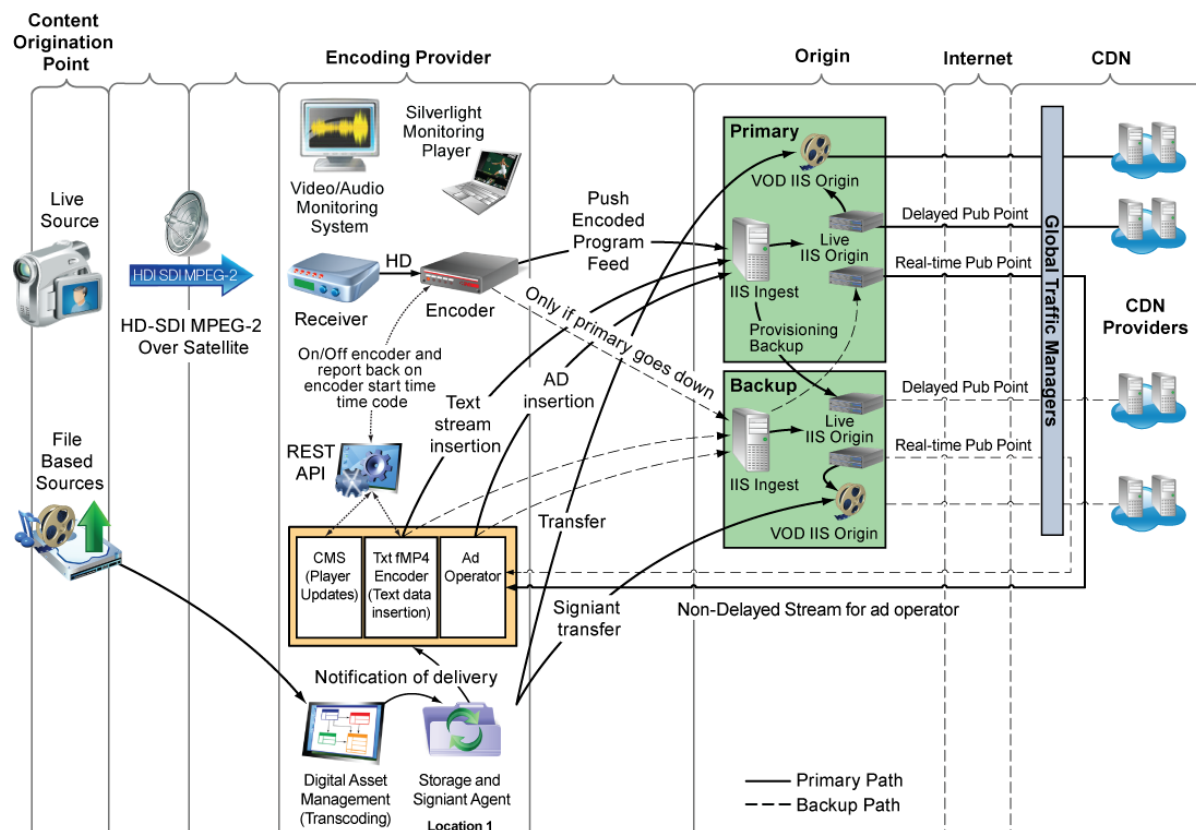


Figure 1. Sample Complete End-to-End Video Workflow Diagram

What is Smooth Streaming?

Smooth Streaming technology enables adaptive streaming of media to Microsoft Silverlight and other Smooth Streaming clients over HTTP. Smooth Streaming provides a high-quality viewing experience that scales massively on content distribution networks, making full HD media experiences (1080p+) a reality. The technology relies on Windows 2008 Server and Internet Information Services (IIS) Media Services technology.

Smooth Streaming dynamically detects local bandwidth and video rendering conditions and seamlessly switches, in near real time, the video quality of a media file that a player receives. Consumers with high-bandwidth connections can experience high definition (HD) quality streaming, while others with lower bandwidth speeds receive the appropriate stream for their connectivity, allowing consumers across the board to enjoy a compelling, uninterrupted streaming experience, and alleviating the need for media companies to cater to the lowest common denominator quality level within their audience base.

IIS Smooth Streaming uses the MPEG-4 Part 14 (ISO/IEC 14496-12) file format as its disk (storage) and wire (transport) format. Specifically, the Smooth Streaming specification defines each chunk/GOP as an MPEG-4 Movie Fragment and stores it within a contiguous MPEG-4 file (.MP4) for easy random access. One MPEG-4 file is expected for each bitrate. When a client requests a specific source time segment from the IIS Web server, the server dynamically finds the appropriate MPEG-4 Movie Fragment box within the contiguous file and sends it over the wire as a standalone file, thus ensuring full cacheability downstream (See Figure 2, below).

In other words, with Smooth Streaming, file chunks are created virtually upon client request, but the actual video is stored on disk as a single full-length file per encoded bitrate. This offers tremendous file-management benefits.

Smooth Streaming content is played using the Microsoft Silverlight 2 (or higher) Internet browser plug-in. All of the functionality, such as the parsing of the MPEG-4 file

format, HTTP downloading, bit rate switching heuristics, and more is provided in the Smooth Streaming client, which is written in Microsoft .NET code and runs as a managed plug-in to Silverlight. This provides an easy way to modify and fine-tune the client adaptive streaming code as needed, instead of waiting for the next release of the Silverlight runtime.

The most important part of Smooth Streaming client development is the heuristics module that determines when and how to switch bitrates. Simple stream switching functionality requires the ability to swiftly adapt to changing network conditions while not falling too far behind, but that's often not enough to deliver a great experience. The IIS Smooth Streaming Media Element (SSME) was created as a platform layer for Silverlight 3 that includes tunable heuristics, allowing publishers to more simply build experiences without requiring them to understand underlying details. This platform layer provides developers with a rich interface that is similar to ones that Silverlight developers are already familiar with. In addition, this layer also allows developers and third parties to more easily integrate additional rich functionality with Smooth Streaming, including analytics, advertising, DVR (such as Pause, Seek, Rewind, Fast Forward), rough cut editing, multiple camera angles, multiple languages, and DRM.

Even in its most basic usage, SSME addresses the following cases:

- Where a user has low or variable bandwidth
- Where a user has enough bandwidth but doesn't have enough resources (such as CPU cycles or RAM) to render the higher resolution video
- When the video is paused or hidden in the background (minimized browser window)
- When the resolution of the best available video stream is actually larger than the screen resolution, thus potentially wasting bandwidth
- Seamless rollover to new media assets, such as advertisements

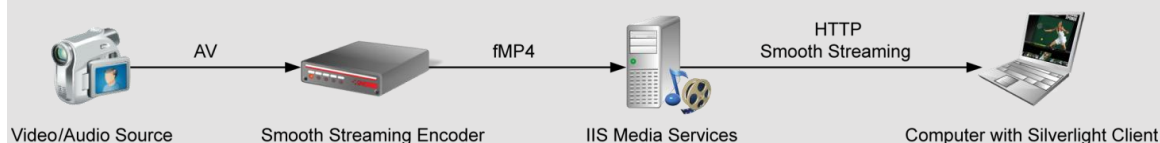


Figure 2: Simplified Smooth Streaming Workflow

For more information on Smooth Streaming, visit <http://go.microsoft.com/fwlink/?LinkId=167889>

Production: Creating the Video and Audio Content

The starting point of every event is production—the actual creation of the video and audio content that will be streamed live, or captured for later on-demand viewing. Video production can be simple, such as a single camera shoot with one audio source. More complex productions, such as professional sports events, might employ multiple cameras and audio sources mixed to create a single program feed that is streamed live. A full service

managed webcasting vendor can provide valuable guidance throughout every phase of production.

When preparing a production, the first important technical requirements to consider are the desired video and audio formats. The most widely used video formats are standard definition (SD) and high definition (HD), with mono or stereo audio.

In the U.S., SD video typically refers to 480 horizontal lines of resolution, presented in an aspect ratio of 4:3 or 16:9 (see Table 1).

Resolution (WxH)	Active Frame (WxH)	Canonical Name(s)	Pixels (Advertized Megapixels)	Display Aspect Ratio (X:Y)	Pixel Aspect Ratio – Standard “4:3” (X:Y)		Pixel Aspect Ratio – Widescreen “16:9” (X:Y)		Description
					ITU-R BT.601	MPEG-4	ITU-R BT.601	MPEG-4	
720x480	710.85x480	480i/p	345,600 (0.3)	4:3	4320:4739	10:11	5760:4739	40:33	Used for 525-line/(60*1000/1001) Hz video such as NTSC-M
720x576	702x576	576i/p	414,720 (0.4)	4:3	128:117	12:11	512:351	16:11	Used for 625-line/50 Hz video, such as PAL-I

Table 1. SD Video Display Resolutions

HD video is defined with three major parameters:

1. **Frame Size** This is stated in pixels and defines the number of horizontal pixels times (x) vertical pixels (such as 1280×720 or 1920×1080).
2. **Scanning System** This is identified as progressive scanning (p) or interlaced scanning (i).
3. **Frame Rate** This is identified as a number of frames per second (fps).

Video Format Supported	Native Resolution (WxH)	Pixels (Advertized Megapixels)	Aspect Ratio (X:Y)		Description
			Image	Pixel	
720p 1280x720	1024x768 XGA	786,432 (0.8)	16:9	4:3	Typically a PC resolution XGA; also a native resolution on many entry-level plasma displays with non-square pixels.
	1280x720	921,600 (0.9)	16:9	1:1	Typically one of the PC resolutions on WXGA, also used for 750-line video as defined in SMPTE 296M, ATSC A/53, ITU-R BT.1543, digital television DLP and LCOS projection HDTV displays
	1366x768 WXGA	1,049,088 (1.0)	683:384 (Approx 16:9)	1:1 (Approx)	Typically a TV resolution WXGA; also exists as a standardized HDTV displays as (HD Ready 720p, 1080i) TV that used on LCD HDTV displays.
1080i 1920x1080	1280x1080	1,382,400 (1.4)	32:27 (Approx 16:9)	3:2	Non-standardized “HD Ready” TV. Used on HDTVs with non-square pixels.
1080p 1920x1080	1920x1080	2,073,600 (2.1)	16:9	1:1	A standardized HDTV displays as (HD Ready 1080p) TV, used on high-end LCD and plasma HDTV displays. Used for 1125-line video, as defined in SMPTE 274M, ATSC A/53, ITU-R BT.709.

Table 2. HD Video Display Resolutions

A variety of video options are available and ultimately the content source will influence how encoding is done and how the playback experience is designed.

Content Acquisition: Delivering the Content to the Encoding Team

Once all the production elements are in place, the next step is to determine how the source content will be acquired for encoding into a format suitable for distribution over the Internet. Content acquisition can take place onsite or offsite.

Onsite Acquisition

If sufficient connectivity is available at the event location (also known as the content origination location), then content acquisition can be done onsite (see Figure 3). This involves delivering video and audio of the program feed to the encoding team (typically in MPEG-2 SD-SDI format for SD content or MPEG-2 HD-SDI for HD content). The connectivity required for content acquisition depends on the desired output quality level, as well as the total aggregated bandwidth, across multiple bitrates, used inside the encoding profile. For example, for a sample SD profile, a minimum of 12Mbps of upstream bandwidth is required for each program feed stream. If a sample HD profile is used, a minimum of 18Mbps of upstream bandwidth for each program feed stream is required. Ideally, a separate public IP address should be provided for each encoder that will be used for easier control and management of the encoders.

Offsite Acquisition

If there is not sufficient connectivity at the event site, the program feed will have to be sent to an offsite location with sufficient bandwidth and encoding capabilities. This is typically done over satellite, fiber, or IP multicast.

Satellite transport—For satellite video transport, a satellite truck is used to uplink content from the content origination location (see Figure 4). In some cases a portable satellite flyaway system can also be used, while the downlink is handled at the offsite location using a satellite downlink antenna (or antennas, if content needs to be downlinked from multiple satellites). Simulcast antenna systems can track multiple satellites at the same time, however, they are much more costly than standard fully steerable antennas. Content acquisition by satellite requires purchasing time on the satellite used to transport video and audio. A full-service managed webcast vendor can provide assistance in reserving satellite transport time. The type of satellite truck and amount of satellite time that must be reserved will depend on the video

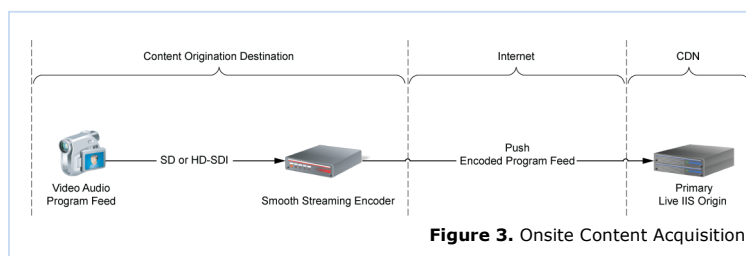


Figure 3. Onsite Content Acquisition

format (analog or digital; SD or HD). Satellites employ two different bands for video transport: C-band and Ku-band. Choosing the most appropriate band depends on several factors, such as the location from which the satellite uplink originates. C-band tends to be more reliable in harsh weather conditions, however, it is approximately twice as expensive as Ku-band. Satellite frequency is another important consideration. While a 9 MHz satellite slot is sufficient for an SD feed, a minimum of 18 MHz is required for an HD feed.

Fiber transport—Fiber is another alternative for transporting video content (see Figure 5). Vyvx® fiber is the most widely used method, and is available in two different service offerings: analog TV1, which can only be used for SD video; and digital 270MB ASI, which can transport both SD and HD video. While a mobile satellite truck can be sent to almost any location, fiber is not mobile—it is either present, or not, at the content origination location. Some locations, however, do offer temporary fiber services that can be used for the duration of an event.

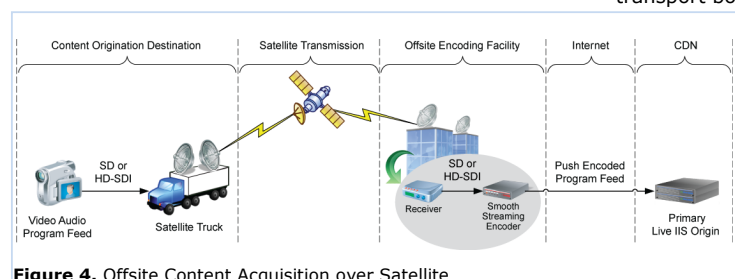


Figure 4. Offsite Content Acquisition over Satellite

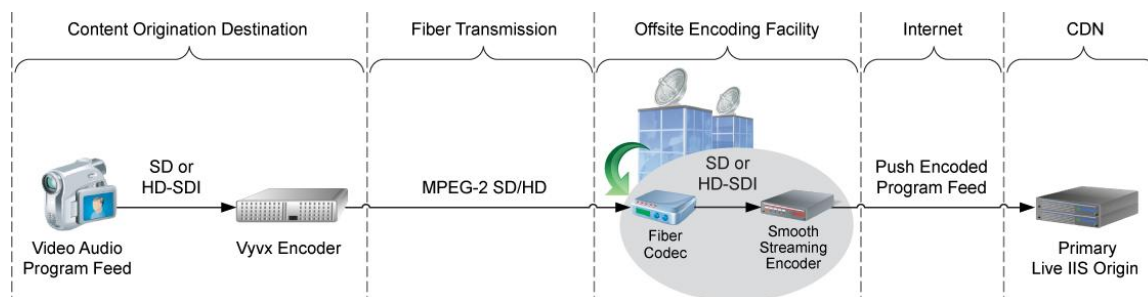


Figure 5. Offsite Content Acquisition over Vyvx® Fiber

IP multicast transport—Video content can also be transported over an IP connection using IP multicast technology (see Figure 6). For this method, raw video (such as HD-SDI) is encoded into a compressed format at the origination point using a codec such as H.264. The content is then multicast over IP using a dedicated point-to-point data connection such as DS-3, OC-3, or OC-12. At the destination, the content is decoded back to raw video and fed to the encoders. It is important to note that the video decoders must be capable of decoding the same codec used to encode the content. The type of data line

required will depend on the total aggregate bandwidth of all of the IP multicast streams being transported. For example, if two HD H.264 encoded video feeds at 17Mbps are transported, then a DS-3, which supports up to 45Mbps of data transport, is an appropriate choice. Video can be encoded using a variety of bitrates, depending on the compression codec used, the source content, and the desired output quality. For video encoded with the H.264 codec, 7Mbps for SD content and 17Mbps for HD will usually provide high-quality output.

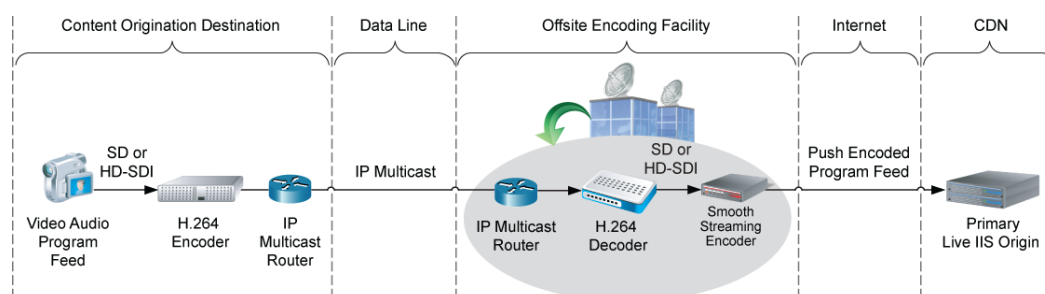


Figure 6. Offsite Content Acquisition using IP Multicast

Checklist

Consider the following technical requirements when planning for content acquisition:

- Ability to acquire live content and encode onsite (SD and HD sources).
- Ability to acquire live content offsite over satellite (both C- and Ku-band; SD and HD).
- Ability to acquire live content over video fiber (preferably Vyvx® Digital 270 ASI).
- Ability to acquire live content over IP multicast.

2009 Major Grand Slam Tennis Championship: A Global Production

The live webcast of the 2009 Major Grand Slam Tennis Championship posed several unique challenges. Work on the event spanned several time zones, and a seamless Video-on-Demand (VoD) workflow was essential, because several production elements from a variety of locations had to be brought together for distribution. Match replays, highlights, and interview files had to be transported from London and New York to Las Vegas. This content was then transcoded into the Smooth Streaming format for VoD distribution using automated workflow.

To meet these requirements, digital media service provider iStreamPlanet developed a hybrid solution to lower the costs while leveraging encoding resources available in regions where content was accessible. The HD feed was transported over fiber from London to New York, where it was post-produced, with additional commentary, commercials, and branding inserted into the program feed. The content was then placed on AMC1 satellite for US-based distribution. iStreamPlanet downlinked the HD feed, removed in-broadcast commercials, and encoded it live to the Smooth Streaming format. SD feeds from the event were not transported over fiber or any other transmission mechanism, and therefore unavailable in the U.S., so the encoding of the SD footage was performed onsite at the sports venue.

See [Appendix A](#) for sample hardware configuration information.

Professional Football: Pulling a Complex Program Together—Live

The 2009 professional football webcast series was a complex project as well. Developing an efficient workflow process was essential, because incoming content, such as advertisements and play-by-play data, had to be seamlessly inserted into the program in real time. Furthermore, four muxed SD feeds had to be smoothly synchronized with an HD feed (muxing is the process where multiple SD-SDI feeds are bundled together into a single ASI feed for ease of transport and lower costs).

To meet these challenges, iStreamPlanet muxed the SD feeds into a single ASI feed, which was uplinked to an AMC-6 satellite and downlinked in Las Vegas. The feeds were then demuxed into individual feeds and fed into Smooth Streaming encoders. The HD feed was uplinked to an AMC-1 satellite, then downlinked in Las Vegas and fed to the Smooth Streaming encoders.

Encoding: Converting the Content for Online Viewing

The next major step in a streaming media production is encoding. Encoding is a process that converts a raw video and audio source, such as SD-SDI or HD-SDI, to a desired online video format, or multiple formats. In this paper, we will focus on encoding content to the Smooth Streaming media format. Video content encoding can take place at an onsite or offsite facility.

Onsite Encoding

For onsite encoding at the content origination location, organizations can utilize a single Smooth Streaming encoder such as the Inlet Spinnaker® 7000 (see Figure 7), or a RoadEncode system which also includes an uninterrupted power supply, power cleaner, and audio mixer in addition to a single or dual Smooth Streaming

Encoder. The number of bitrates to be included in a profile will help determine whether a single or dual encoder is most appropriate. If your profile utilizes more than four bitrates, we recommend using two encoders.

Organizations should also consider whether backup encoders will be required for improved reliability and performance.

Smooth Streaming profiles can contain up to ten bitrates, depending on whether the source is standard or high-definition programming. To help ensure an optimal viewing experience, it is important to use relatively small increments when configuring bitrate settings for the profile. For example, instead of stepping viewers from 350Kbps to 950Kbps to 1950Kbps, step upward from 350Kbps to 600Kbps to 950Kbps to 1400Kbps to 1950Kbps. This technique helps make transitions from one bitrate to another less abrupt, for a much smoother, higher quality picture. Ideally, an SD stream should have



INLET

Spinnaker 7000

Live Streaming HD & SD Encoder
iStreamPlanet

Preset: ECN HD 6
Status: Running [\[Stop\]](#)

Summary

Presets

Input

VC1

H264

Scheduling

System

Help

Summary

[Alarms](#)

[Encoding Statistics](#)

A/V Status	Source	Format	Profile	Resolution	Bit Rate
VC-1 Video Stream 1	SDI	1920x1080	VC-1 Advanced	1280x720	2950 kbps
Stream 2			VC-1 Advanced	848x480	1950 kbps
Stream 3			VC-1 Advanced	736x416	1350 kbps
Stream 4			VC-1 Advanced	592x336	950 kbps
Stream 5			VC-1 Advanced	432x240	600 kbps
Stream 6			VC-1 Advanced	112x64	50 kbps
VC-1 Audio 1			WMAPro		48 kbps

Output Status	Network URL	Alternate URL
VC-1 Push Stream 1	http://origin102.sat10.msecn.net/live.isml/stream(2950a)	
Stream 2	http://origin102.sat10.msecn.net/live.isml/stream(1950a)	
Stream 3	http://origin102.sat10.msecn.net/live.isml/stream(1350a)	
Stream 4	http://origin102.sat10.msecn.net/live.isml/stream(950a)	
Stream 5	http://origin102.sat10.msecn.net/live.isml/stream(600a)	
Stream 6	http://origin102.sat10.msecn.net/live.isml/stream(50a)	

Pre-Processing Status	Inverse Telecine	De-interlacing
-----------------------	------------------	----------------

Figure 7. Inlet Spinnaker® 7000

a minimum of four bitrates, while an HD stream should have a minimum of six bitrates inside its profile.

We also recommend attaching audio to one of the lower bitrates. If connectivity issues arise, in which the encoder is unable to push larger bitrates out to the server, or delivery of larger bitrates is delayed, the audio will still make it to the server when attached to the lower bitrate, and will be present in the stream. For example, if using a profile that contains 350Kbps, 600Kbps, 950Kbps, 1500Kbps, 2250Kbps, and 3450Kbps, attach the audio to the 600Kbps stream instead of the 3450Kbps stream.

Connectivity between the encoder and the IIS origin server is another key consideration. Encoders push content to the IIS origin server or, in some cases, to an ingest server (a proxy server that receives encoder content, then forwards it to the origin server or servers). To help ensure proper delivery of all of the video content across all bitrates, adequate upstream bandwidth is vital. As a rule of thumb, employ two times the total aggregate stream bitrate. For example, two times 9.2Mbps would total 18Mbps.

Offsite Encoding

For offsite encoding, instead of receiving the program feed onsite, directly from the production sources, the video and audio is either downlinked from the satellite using a satellite downlink antenna(s), or taken from video fiber circuit or IP multicast. If reliability is a concern for a mission-critical project, a managed webcasting organization can configure the primary encoder to work in parallel with a redundant secondary encoder. Should the primary encoder fail, the encoding operation automatically rolls over to the backup encoder to minimize downtime. If multiple origin servers are available, the video encoder should be configured to push content to the primary origin, and to rollover to a backup origin server if connectivity drops, or if there is an issue with the primary origin. A managed webcasting organization may also choose to push content simultaneously to the primary and secondary servers. However, it is important to remember that a simultaneous push to primary and secondary servers will double bandwidth requirements. For example, when a single HD stream with a total aggregate bitrate of 9.2 Mbps is pushed to both primary and secondary origin servers, the required upstream bandwidth would be 36 Mbps. Alternatively, an organization could use the IIS Live Smooth Streaming Push Proxy feature, which auto-feeds a back-up stream from the primary ingest server to the backup ingest server, which can eliminate the need to

double encode upstream bandwidth. An organization could also set up a global traffic manager in front of the primary and secondary origin servers that allows for seamless transition from the primary to secondary origin.

Offsite Encoding Facility Requirements

For optimum availability, the offsite encoding facility should have primary and backup connectivity as well, preferably from two different ISPs, configured to automatically fail-over using BGP (Border Gateway Protocol) or HSRP (Hot Standby Routing Protocol). Conditioned power, including an uninterruptible power supply (UPS) and backup generator, is important as well. For additional flexibility, the encoding facility should also have the ability and capacity to accommodate content acquisition from a variety of sources, such as analog and digital satellite, C-band and Ku-band feeds, SD and HD, Vyvx® fiber, analog TV1 and digital 270 ASI circuits and IP multicast. The site should also be able to handle a wide range of tape formats, including Beta, Digi-Beta, DV and miniDV, HDCam, HD DVCPPro, and DVD, as well as any digital file source.

Some offsite encoding facilities can offer additional value-added services, such as recording to digital video recorder (DVR) systems for automatic rebroadcast at specific times or intervals, in a variety of formats. Some other desirable services include the ability to delay content, and support for Dual-tone Multi Frequency (DTMF) signaling, which is used by television broadcasters to indicate start and stop times of local commercial insertion points during station breaks. DTMF support lets organizations replace in-program commercials with their own advertisements or other customized content.

A video and audio router is required to manage all inbound video traffic and direct video and audio feeds to various encoders (Figure 8). All inbound video feeds should also be previewed on preview monitors.

In some situations, an organization will use multiple encoders to generate a single Smooth Streaming presentation (for example, encoding three bitrates on one encoder and three additional bitrates on a second encoder). To support this configuration, both encoders should be synchronized using vertical interval time code (VITC), a form of SMPTE time code embedded as a pair of black and white bars in a video signal. Encoders can also be synched using linear or longitudinal (LT) time code, which encodes SMPTE time code data as a Manchester-Biphase encoded audio signal. A time code generator will

provide optimal results for generating VITC or LTC time codes.

RESTful application programming interfaces (APIs) leverage RESTful Web services. A RESTful web service is a simple web service implemented using HTTP and the principles of REST. Such a web service can be thought about as a collection of resources. The definition of such a Web service can be thought of as comprising three aspects:

1. The base URL for the web service such as <http://example/resources>.
2. The MIME type supported by the web service. The most common types are XML, JSON or YAML, but the web service could also support any other valid MIME type.
3. The set of operations supported by the web service using HTTP methods (such as POST, GET, PUT or DELETE). Unlike SOAP-based web services, there is no official standard for RESTful web service because REST is an architecture (unlike SOAP, which is a protocol).

RESTful APIs are useful for synchronizing encoded content with external data feeds, such as play-by-play data associated with a sports event. These APIs retrieve a time code, which indicates when the video encoder has generated its first frame. Using this time code, the APIs then sync the external data with the program video feed. Additional API capabilities include the ability to start and stop encoders; forward publishing point and profile information to encoders; and periodically check the encoder status.

These sample settings are appropriate for high motion content, such as sporting events:

Video Bitrate (Kbps)	Width	Height	Frame Rate (fps)
3450	1280	720	29.97
2250	960	540	29.97
1500	720	404	29.97
950	544	304	29.97
600	400	224	29.97
350	288	160	29.97
50	112	64	29.97
Aggregate Bitrate		9200 Kbps	

Table 3. Sample Settings for Smooth Streaming HD Encoding Profile from NTSC source

These sample settings are appropriate for content with minimal motion, such as a corporate presentation:

Video Bitrate (Kbps)	Width	Height	Frame Rate (fps)
2950	1280	720	29.97
1950	960	544	29.97
1250	704	400	29.97
850	512	288	29.97
550	400	224	29.97
350	320	176	29.97
50	128	96	14.985
Aggregate Bitrate		8250 Kbps	

Table 4. Sample HD Smooth Streaming Encoding Profile Settings from NTSC Source

Video Bitrate (Kbps)	Width	Height	Frame Rate (fps)
1950	848	480	29.97
1500	720	404	29.97
950	544	304	29.97
600	400	224	29.97
350	288	160	29.97
50	112	64	29.97
Aggregate Bitrate		5450 Kbps	

Table 5. Sample Settings for Smooth SD Encoding Profile NTSC source

For both the HD and SD profile sets above, the video codec utilized is the VC-1 Advanced Profile. The audio codec is Windows Media Audio (WMA) 10 Professional – 48Kbps 44.1 kHz stereo. If encoding music content, we recommend increasing the audio bitrate to 96Kbps or 128Kbps, depending on the desired output quality and audio source.

Professional Football: Live Play-by-Play and Game Action

A recent Professional Football webcast required close coordination of the game's video and audio feeds with play-by-play statistics and data. To seamlessly integrate this content, iStreamPlanet utilized RESTful APIs to extract the video encoder start timecode, then synchronize the video feed with play-by-play data from the Game Stats and Information System (GSIS) feed generated by the Football League. It is important to note that time code can also be extracted directly from the manifest file. To do this, open the manifest file, take the first "c t=" line in the video chunks section, divide by 10,000,000, then divide by 3600. The result is the time displayed in a 24 hour clock format.

Example:

</QualityLevel>

<c t="346886540000" />

$(346886540000 / 10000000) / 3600 = 9.63573722222222222222222222222222$ 22 (approx 9:36am)

Checklist

The encoding process is a demanding step in streaming video delivery. Be sure to consider the following technical requirements as you prepare for video encoding:

- The encoding facility should have primary and back-up connectivity, with auto failover using BGP or HSRP. To further improve reliability, conditioned power is advisable, including an appropriate UPS and backup generator.
- Allocate two times the total aggregate stream bitrate of upstream bandwidth from the encoding location/facility.
- Use a video and audio router system capable of routing both HD and SD sources to deliver content to the video encoders.
- For additional quality control, video and audio preview and monitoring systems should be used to monitor content that is sent to the encoders.
- If encoding a single feed with multiple encoders, or encoding multiple video feeds that need to be in sync with multiple encoders, use a time code generator to sync video feeds that have been generated by the encoders.
- For SD content, utilize an encoding profile with minimum of four bitrates. A minimum of six bitrates are recommended for HD content.

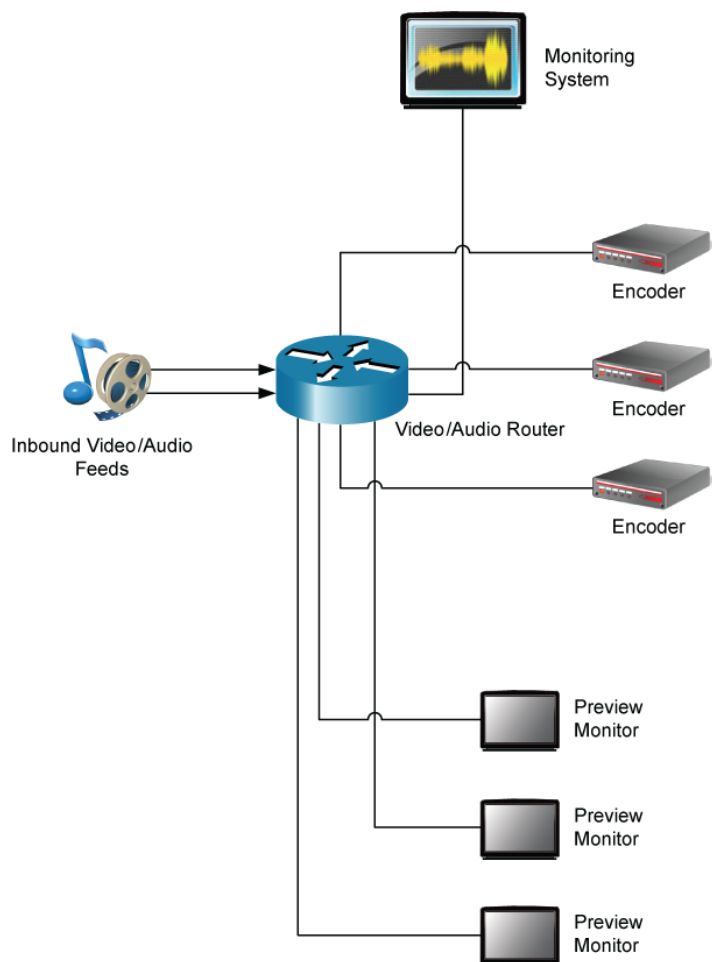


Figure 8. Audio/Video Signal Flow Diagram

Content Delivery: Delivering the Stream to Viewers

Once content is encoded and pushed to the IIS origin server, it can be delivered in two ways: directly from the origin server(s), or through a Content Delivery Network (CDN).

Direct Content Delivery

Programs can be delivered directly from one or more IIS servers that are load balanced to increase scalability. This delivery model is most appropriate for organizations serving content to a small audience (fewer than 1,000 end users) that is not geographically dispersed.

Delivery through CDN

To deliver content to an expected large audience that is geographically dispersed across different regions (such as North America, Europe and Asia), a CDN is the most effective model. This model utilizes a CDN HTTP infrastructure consisting of multiple caching devices that pull content from the Microsoft IIS origin server(s). The program content is distributed across a large number of HTTP cache devices that are geographically distributed, load-balanced, and capable of serving hundreds of thousands of viewers.

Estimating Bandwidth

Provisioning the appropriate level of bandwidth is essential to successful content delivery. To estimate the egress bandwidth required to serve a particular audience size, use the following formula:

Total Egress Bandwidth = Average Bitrate Consumed by the Client x Maximum Number of Concurrent Clients

For example, if the average bitrate for a SD stream is 1.1Mbps, and the maximum number of concurrent clients is 1,000, the total egress bandwidth required would be 1.1Gbps. In other words, to serve 1,000 clients with an average connection of 1.1Mbps, the origin server would need a minimum capacity of 1.1Gbps or higher. When possible, planning and provisioning for additional egress bandwidth is desired in order to accommodate temporary spikes in traffic.

Determining the average bitrate utilized by viewers depends on several variables, including the encoding profiles used to encode the content, audiences' geographical location, and the default size of the Microsoft Silverlight media player embedded in a Web page. As of August, 2009, digital media service provider

iStreamPlanet has determined that an average bitrate for SD content was 1.1 Mbps, while the average bitrate for HD content was 1.5Mbps. These bitrates were based on the vendor's experience with several events with audiences in the U.S., from May through August, 2009, as well as the sample profiles presented in this document.

If the majority of your audience is located in specific geographical region, it is important that the CDN coverage of that region is large enough to accommodate all users, while providing additional capacity to accommodate flash crowds. Flash crowds are large spikes in a number of concurrent users requesting content. For example, an event presented to an average of 1,000 concurrent visitors might suddenly need to accommodate 50,000 additional requests for content. Use the total egress bandwidth to calculate how much capacity needs to be provisioned, while allowing for unpredicted surges in concurrent viewers.

Origin Server Hosting Options

When working with a CDN, the IIS origin server can be hosted by the content owner, while the CDN is used to deliver actual content using its large HTTP caching infrastructure.

Hosting the origin server can offer several advantages for the content owner, such as improved flexibility, customization, and administrative control. However, this approach can also result in additional costs for ongoing maintenance and updates. An excellent method to exercise control over IIS live and on-demand publishing points is to utilize IIS APIs and .NET to fully automate creation, removal, starting, stopping, and shutdown of publishing points from Web applications, such as content management systems.

The IIS ingest server should be used as a proxy if the IIS origin server has been configured to utilize a primary and backup server. IIS ingest servers are also recommended in situations where delayed delivery is needed, such as live advertisement insertion. In this case, a live publishing point is used to insert ads, while actual content is played from a delayed publication point.

For additional scalability, IIS servers can replicate content from one server to another. Taking advantage of these replication capabilities can dramatically improve reliability, as well as simply provide content owners with a backup version of their content.

Checklist

Consider the following technical tips when preparing for content delivery:

- Host your own origin and deliver content yourself only if delivering content to a small audience that is not geographically dispersed.
- Leverage a CDN HTTP infrastructure to deliver content to large audiences that are geographically dispersed.
- The path between the video encoder and IIS ingest server should be reliable, have sufficient throughput, and latency less than 75 ms.
- Use the Total Egress formula to calculate how much capacity is needed to serve your content.

Content Delivery at the 2009 Major Grand Slam Tennis Championship

For the 2009 Major Grand Slam Tennis Championship, digital media service provider iStreamPlanet hosted the IIS origin server, while the Akamai CDN supported HTTP content delivery.

Regardless of where the IIS origin is hosted, it is important that the path from the video encoder to the IIS origin ingest server is reliable, with sufficient throughput capacity. For example, if the encoder is pushing a single HD stream with an aggregate bitrate of 9.2Mbps, the connection to the ingest server requires 18Mbps throughput to account for any data bursts and optimal delivery of all video chunks to the server. Test and analyze the path between the encoder and server to avoid any potential degradation in quality and latency. A trace route between the two devices will ideally require fewer than ten hops, and latency between any of the routes should not exceed 75ms.

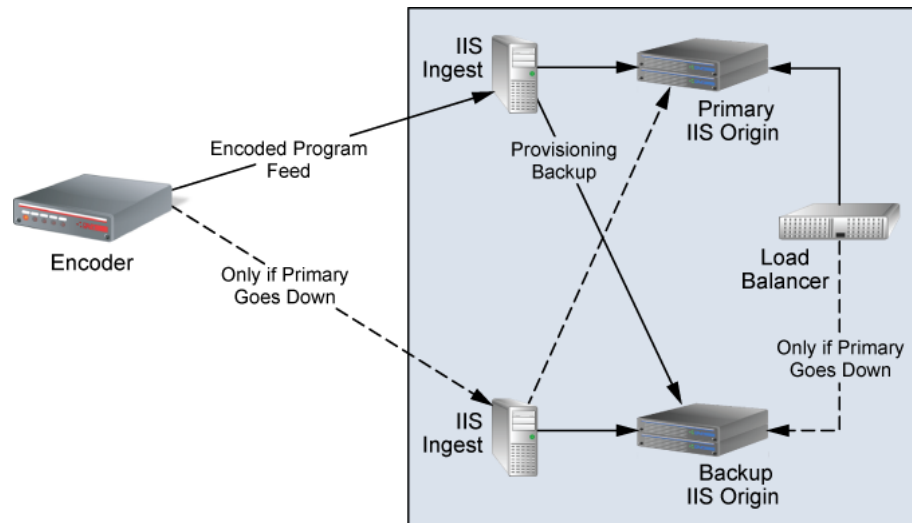


Figure 9. Sample Origin Implementation Diagram

About IIS Media Services 3.0

IIS Media Services is an integrated HTTP-based media delivery platform. By leveraging the massive scale of today's HTTP-based Web, IIS Media Services Smooth Streaming and Live Smooth Streaming deliver true HD (720p+) live and on-demand streaming for a compelling viewing experience. IIS Media Services 3.0 integrates five IIS extensions:

- Smooth Streaming dynamically detects and seamlessly switches, in real time, the video quality of a media file that a Silverlight player (or other client) receives, based on local bandwidth and CPU conditions. Consumers with high bandwidth connections can experience HD quality streaming while others with lower bandwidth speeds receive the appropriate stream for their connectivity, allowing consumers across the board to enjoy a compelling, uninterrupted streaming experience—for extended audience engagement and increased revenue opportunities.
- Live Smooth Streaming enables adaptive streaming of live events to Silverlight and other clients. Using HTTP to deliver live events takes advantage of the scale of existing HTTP networks, keeps content close to the end-user, and makes full HD (1080p+) a reality for live Web broadcasts, while boosting availability.
- Advanced Logging extension provides access to both real-time client and server side logs for media experiences delivered over HTTP. Advanced Logging also allows third-party analytics providers to easily access data in real-time, and extends the set of metrics that are captured from either the client or the server.

- Bitrate Throttling saves network bandwidth costs by controlling the rate at which media is delivered over HTTP. The throttle rate can be configured using server side rules or automatically adjust the download rate for detected media files based on format and encoded bitrate. When Bitrate Throttling is not used, content is downloaded at the maximum possible speed. If the customer drops off after watching 20 percent of a video clip, the content owner would have paid bandwidth for media that was never watched.
- Web Playlists let content owners deliver secure, server-controlled media playlists from a Web server. They enable organizations to control whether clients can Seek or Skip for each media asset. Web Playlists also obfuscate the location of media assets from the end user, and prevent client-side caching of those assets. Web Playlists offer content owners easy control over how media assets, including how advertisements are sequenced and experienced by their customers.

The Single Platform Advantage

Offering an integrated media delivery platform with IIS enables Web sites to manage and administer a unified infra-structure that delivers rich media and other Web content from a unified Web platform. Combining media delivery and Web content delivery on a common platform saves the cost of managing two separate server infrastructures to deliver a single Web site. Harmonizing server platforms also simplifies deployment and administration of new applications, servers and Web sites.

Playback Experience: Connecting with Your Audience

The rich media player is the area where the media is rendered. This area is the zone where the user is watching media content, and also interacts with it through a remote control layout. The remote control commands are subject to the type of media content that is being watched.

The Microsoft Silverlight client (see Figure 10) is an industry-leading platform to build rich media players, and we recommend using Microsoft Silverlight 2 or higher. Microsoft Silverlight 3 is preferable, because it offers better control over Smooth Streaming playback, using the Smooth Streaming Media Element, also known as SSME.

Microsoft Silverlight rich media Smooth Streaming players are designed and developed to play live and on-demand Smooth Streaming content. When live content is being delivered, users can pause or stop the video playback, leverage DVR to navigate to any part of live video from the start of the live broadcast to the current moment, and use fast forward and slow motion playback and instant replays. They can also adjust the volume or mute the sound, or enter a full screen viewing experience. Because of push technology, the program can automatically be converted to on-demand content. Depending on the level of customization, the player can also host an information panel that displays the channel being watched, the title of the program, and the date of delivery for the show and other metadata, as well as other applications that are seamlessly integrated into the rich media player, such as Twitter®, Facebook®, Q&A, chat, maps, e-commerce, overlay and instream ads, and many others.



Figure 10. Microsoft Silverlight Rich Media Player Experience

Checklist

The playback experience is the step at which your content finally reaches the viewer. Use these tips to help plan an optimal rich media experience for your audience:

- Configure your client access policy properly to help ensure that the desired audience can experience your program.
- Build player stream acquisition retry logic to make sure viewers are able to acquire the stream on initial load in situations where the client doesn't connect to the stream on the initial connection attempt. Player stream acquisition retry logic also enables the client to automatically reconnect, if it is temporarily disconnected from the stream (such as due to issues with Internet connectivity on the viewer's machine). If after multiple retries the client is still unable to make a connection to the stream, we recommend displaying a message stating that the client is unable to connect to the stream.
- Develop a comprehensive cross-browser Silverlight install experience to maximize your audience and enable ease-of-use.
- Match video player height and width with the encoded video height and width, to take advantage of screen heuristics that optimize video playback experience.
- Use a Content Management System (CMS) to update content inside the player, to help ensure content is fresh and relevant.
- To ensure a professional, high-quality webcast, utilize separate staging and production server environments so that playback, features, and functionality can be tested prior to publication on the production environment.
- Use TTL (time to live) inside your media feed so content updates can happen automatically, without user interaction.
- Use the Microsoft Silverlight 3 based player to take advantage of the Smooth Streaming Media Element and GPU acceleration.

Management, Coordination, and Monitoring: Complete Control

Management and Coordination

Managing and coordinating a video webcast is a complex endeavor. The key steps involved in developing a webcast, from production to encoding to playback, may involve a large number of groups, organizations, and people all working toward a common goal (see Figure 11). However, these disparate groups may not always be working together as a team, or have access to needed information.

A successful webcast requires a strong management team to closely coordinate a variety of groups, assign tasks, and track progress. A webcast technical director can play a vital role in running an event, deciding what content will be played, as well as when content is published and pushed to the staging and production environments.

Detailed documentation is also essential to running a smooth, accountable production team. A webcast production book, or play-by-play book, specifies points of contact and levels of responsibility, and plays a vital role in conducting live events over the Internet.

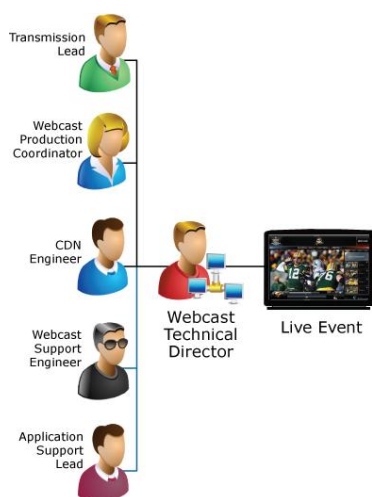


Figure 11. Live Event Management and Coordination Diagram

Keeping the Team Connected

During live events, we recommend establishing a conference bridge where all involved parties can communicate every aspect of the live event. This bridge enables production teams to quickly disseminate information and troubleshoot issues that might come up

during the live event. A conference bridge should include the transmission lead, webcast technical director, webcast production coordinator, CDN engineer, webcast support engineer, and an application support lead.

Utilizing fully managed live and on-demand streaming media services is one option that can help organizations simplify all aspects of online delivery, from production, content acquisition, encoding, content delivery, to playback.



Figure 12. Event Control Room During Live Event

Checklist

A effective management team can mean the difference between success and failure for a streaming webcast. The following tips can ease event planning and management:

- Make sure an experienced, qualified event management staff is in place as you plan the event. Although event management might seem like an additional expense of limited value, it is an absolute must when producing medium and large-scale events.
- Utilize a webcast technical director to run live events to help ensure seamless execution and workflow.
- Determine objectives, timelines, technical details and execution process for your live event, and document them in detail in a webcast production book. Distribute this play-by-play book to all involved parties.
- Monitor all inbound video and audio sources, encoders, receivers, decoders, and networking equipment.

Monitoring

Monitoring plays a vital role during live events, because any anomalies in the video or audio source, encoded feeds, or equipment performance must be quickly identified to minimize possible downtime or degradation in quality. Three key areas should be monitored during a live event:

1. **Inbound source video and audio feeds**—When monitoring inbound video and audio sources, it is important to have the ability to display all inbound sources so that they can be monitored in real time for any degradation in quality or interruptions.
2. **Outbound encoded video and audio feeds**—Outbound encoded feeds should be monitored directly from the origin, so that any anomaly in video and audio quality can be easily identified, and the source of the issue tracked to a specific device or process (see Figure 13). Smooth Streaming video is composed of multiple bitrates, so having ability to select and monitor a specific bitrate is critical. Two key statistics should be tracked: bitrate and corresponding video frames per second.
3. **Equipment monitoring**—For optimal performance, it is important to monitor all equipment used for content acquisition, encoding, and distribution to the IIS origin (see Figure 14 on the next page). This equipment monitoring should include receivers and decoders, encoders, and network devices.

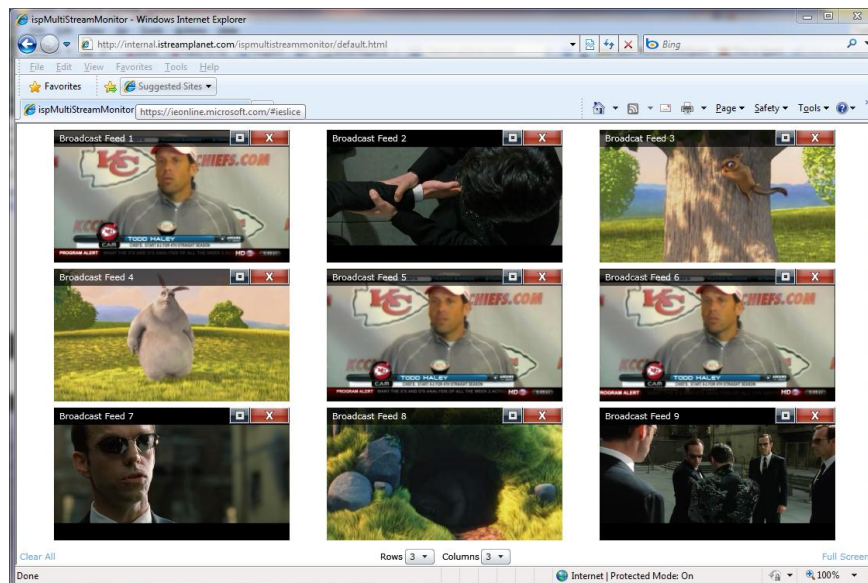


Figure 13. Outbound Encoding Feeds Multi-Screen Silverlight-based Monitor Player

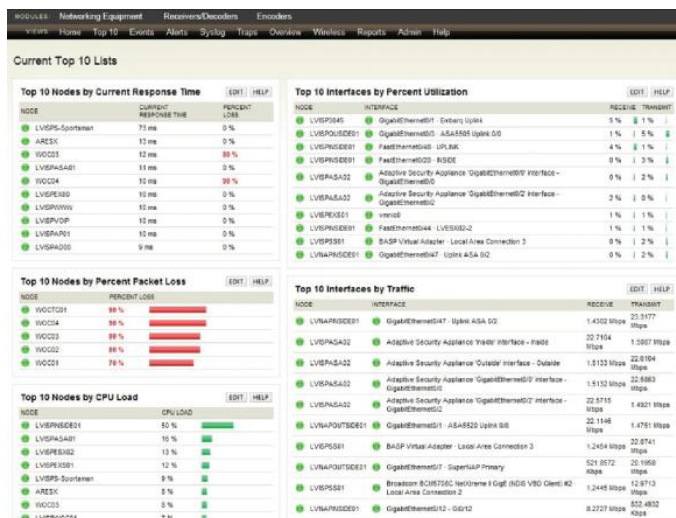


Figure 14. Equipment Monitoring System

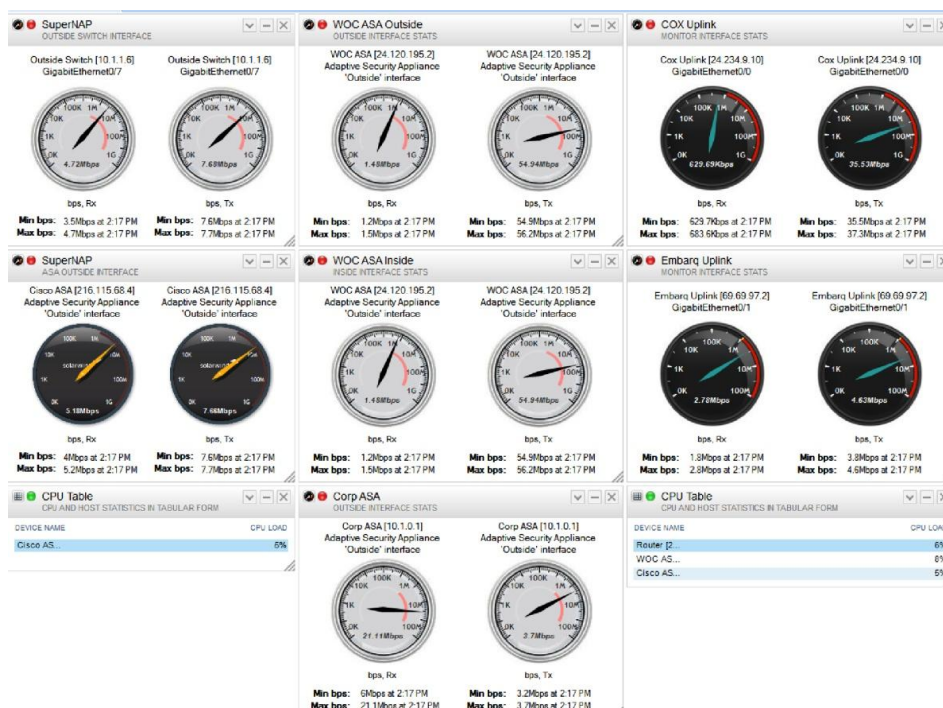


Figure 15. Network Monitoring System

On-Demand Video Workflow

On-demand video workflow (see Figure 16, below) can provide additional flexibility and playback options for content owners. This type of workflow can be used on its own for on-demand events, or in conjunction with live video workflow. For example, an organization might want to develop a short program of video highlights using captured live video content—while the event is still happening. On-demand video workflow can also be used to publish a live event for video on-demand playback, or to transcode a variety of disparate media into Smooth Streaming media format for playback in conjunction with a live content.

Key elements of the on-demand video workflow include:

- CMS (Content Management System)
- Digital Asset Management (DAM)
- Live Ads Insertion Application
- RCE (Rough Cut Editor)
- Text Stream Data Insertion Application

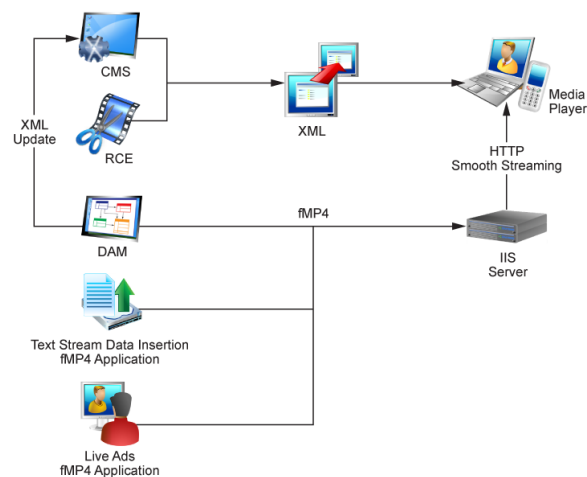


Figure 16. Video On Demand Workflow Diagram

Professional Football: Live and On-Demand Together

For a professional football webcast, highlights were created from a live feed, while the live event was happening (see Figure 17). At the same time, additional interviews and videos were produced and transcoded into Smooth Streaming format for video on-demand playback. Digital media services provider iStreamPlanet utilized a combination of on-demand and live video workflows, to enhance and enrich viewers' rich media experience.



Figure 17. A recent professional football webcast featured both live and on-demand content.

Content Management System

A Content Management System (CMS) is used to manage metadata for both live and on-demand content, such as the content title, description, duration, ratings, and other data. Organizations can also use a CMS to publish an XML feed that contains metadata and URLs for content thumbnails and media assets (see Figure 18). This XML feed is used by the video player to populate and play the content.

Ideally, a CMS should be able to publish content to both the staging and production environment. The staging environment is used to confirm that the program content and its associated metadata have been properly published, and that they are displaying and playing inside the media player as intended. After the program content and metadata have been validated, an XML file containing all metadata and links is then pushed into production.

Some organizations may wish to use a CDN to deliver the XML file, to take advantage of its caching capabilities, and improved scalability and reliability. However, it is important to keep in mind that changes may take time to take effect in a cached environment. Most CDNs support content purging to speed up this update process. However, when employing this technique, remember to purge only the CMS XML file—not the entire site that contains media assets and other content.

```
<?xml version="1.0" encoding="utf-8"?>

<Clips>

  <Clip ID="1"

    Duration="30"

    Type="VOD"

    ThumbnailUrl="http://www.example.com/clipthumb
    1.png"

    Source="http://www.example.com">

    <Title><![CDATA[Clip1]]></Title>

    <Description><![CDATA[This is the description for
    clip1]]></Description>

  </Clip>

  <Clip ID="2"

    Duration="332"

    Type="PB"

    ThumbnailUrl="http://www.example.com/clipthumb
    2.png"

    Source="http://www.example.com">

    <Title><![CDATA[Clip2]]></Title>

    <Description><![CDATA[This is the description for
    clip2]]></Description>

  </Clip>

</Clips>
```

Figure 18. Sample XML

Rough Cut Editor

The Rough Cut Editor, provided by Microsoft, is a standalone, Silverlight-based component proxy and metadata editing solution that lets end users edit audio-visual content in a cross-platform web browser experience.

It gives users a multi-step process to create non-destructive rough edits from proxy files (in VC1 or H.264 format), and save the edit decisions as metadata in a back end service for further workflow processing.

This Rough Cut Editor uses the media capabilities of Silverlight 3, such as live and on-demand true HD Smooth Streaming. Following are three key scenarios where Rough Cut Editor can be used (see Figure 19):

Trimming an archived version of a live feed for video on-demand playback—Encoding of a live event usually begins 15 to 30 minutes prior to its start time. For example, a slate may appear stating that the event will start shortly, to keep early visitors engaged or, in some cases, to manage access to the content in the event of a flash crowd. After the live event ends, a content owner may post a slate indicating when an archived version of the event will be available. Rough Cut Editor lets content owners delete this additional pre-show and post-event content, if desired, rather than forcing viewers watch it

Creating video clips from a live video feed while the live event is still running—In this situation, organizations can use Rough Cut Editor to create video clips from a live Smooth Streaming feed while the live event is still happening. For example, highlights from a live sporting event can be created using Rough Cut Editor, and posted for on-demand playback even while the event is taking place.

Creating thumbnails from video content—Rough Cut Editor can be used to create thumbnails from video files or live streams. In situations where Rough Cut Editor is connected to the CMS, these thumbnails can be managed using the CMS, and published for display inside the video player.

Rough Cut Editor also provides a variety of additional features, including support for creation of video clips with advertising pre- and post-rolls. The application can also be used to insert metadata into finished media.

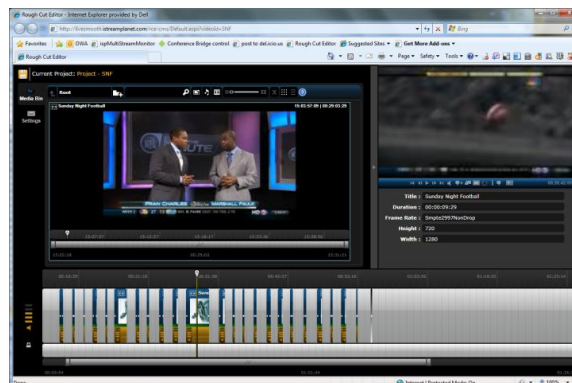


Figure 19. Rough Cut Editor enables content owners to refine and customize video content.

Text Stream Data Insertion Application

A program's manifest file can contain not only video and audio tracks, but a text track that includes a variety of data that is synchronized with the video and audio content (see Figure 20). For example, play-by-play data from a sporting event can be inserted into a manifest file as a fragmented MPEG-4, then displayed inside the media player window alongside the streaming video content. This capability gives organizations a unique opportunity to create engaging, immersive rich media experiences, with external data seamlessly synchronized with video and audio content.

```
<?xml version="1.0" ?>
<SmoothStreamingMedia MajorVersion="1"
MinorVersion="0" Duration="0" IsLive="TRUE"
LookAheadFragmentCount="2"
DVRWindowLength="0">
  <StreamIndex Type="video" SubType="WVC1"
Chunks="0"
Url="QualityLevels({bitrate})/Fragments(video={start time})">
  <StreamIndex Type="audio" SubType="WMAPRO"
Chunks="0"
Url="QualityLevels({bitrate},{CustomAttributes})/Fragments(audio={start time})">
  <StreamIndex Type="GSIS" SubType="CAPT"
Chunks="0" ParentStreamIndex="video"
ManifestOutput="TRUE"
Url="QualityLevels({bitrate})/Fragments(text={start time})">
</SmoothStreamingMedia>
```

Figure 20. Sample Condensed Manifest File with Text Stream

Live Ads Insertion Application

A program's manifest file can also support an advertising insert track, which contains information about live ads or ad pods—all tightly synchronized to the program's video and audio content. For example, an ad campaign created inside DoubleClick's DART® system could be inserted inside a manifest file at a specific time, for playback inside a media player. Using a live ad insertion application helps ensure that an advertisement will always be shown at the correct time within the program. At the same time, the application gives the ad operator greater control over the advertising experience.

```
<?xml version="1.0" ?>
<SmoothStreamingMedia MajorVersion="1"
MinorVersion="0" Duration="0" IsLive="TRUE"
LookAheadFragmentCount="2"
DVRWindowLength="0">
  <StreamIndex Type="video" SubType="WVC1"
Chunks="0"
Url="QualityLevels({bitrate})/Fragments(video={start time})">
  <StreamIndex Type="audio" SubType="WMAPRO"
Chunks="0"
Url="QualityLevels({bitrate},{CustomAttributes})/Fragments(audio={start time})">
  <StreamIndex Type="ad-insert" SubType="ADVT"
Chunks="0" ParentStreamIndex="video"
ManifestOutput="TRUE"
Url="QualityLevels({bitrate})/Fragments(ad-insert={start time})">
</SmoothStreamingMedia>
```

Figure 21. Sample Condensed Manifest File with Ad-Insert

Digital Asset Management

The last element of on-demand video workflow is Digital Asset Management (DAM). A DAM system is used to manage media assets (see Figure 22). In a live or on-demand environment, it can be used to transcode various media formats into the Smooth Streaming format. After transcoding is complete, DAM can transfer the Smooth Streaming media files to the designated IIS origin using FTP, or using other file transfer applications such as Signiant or Aspera.

The DAM workflow is built around six key steps:

1. High quality media files (such as mpeg-2 HD, mov HD, or avi HD) are transferred to a watch folder inside DAM. File transfer accelerator applications such as Signiant or Aspera can substantially reduce transfer times.
2. The watch folder is configured on DAM and tied to an automated workflow system.
3. Automated workflow (see Figure 23) is created with designated Smooth Streaming transcoding profiles. For best results, we recommend using SD profiles for SD content, and HD profiles for HD content.
4. Once files are transferred into the watch folder, the auto-mated workflow is invoked and files are automatically transcoded using specified Smooth Streaming profiles.
5. Once transcoding is complete, DAM automatically transfers the finished files to a designated IIS origin server, and e-mail notification is sent to the specified recipients.

Automated workflow also offers the option to create thumbnails and perform XML transforms if CMS is not available for these tasks. To provide a unified user interface look and feel for all users, as well as provide critical process functionality through a central location, Microsoft Office SharePoint Server 2007 can be used as the Presentation Layer, interfacing with the back end DAM system and providing process integration with other systems. This Web presentation front-end layer is designed to meet the needs of intranet and Internet users. Its out-of-the-box capabilities provide a rich collaborative environment that streamlines everyday business activities, while taking advantage of the latest development technologies such as Web parts, ASP.NET, and the Windows Workflow Foundation. Organizations can integrate Microsoft Office SharePoint Server 2007 with the back end DAM system using Content Management Interoperability Services (CMIS), a set of product, technology, and protocol-agnostic standards. CMIS provides specifications for the data model, as well as the services to be exposed to the application integrating with the AM or ECM system. Microsoft Office SharePoint Server 2007 can use CMIS to integrate with DAM systems either at the back end (for example, using Microsoft BizTalk Server), or at the front end (for example, using Silverlight).

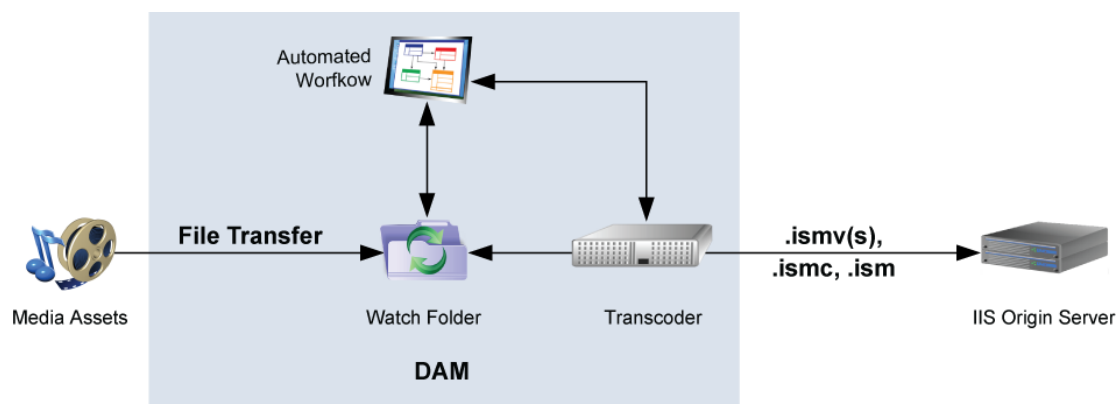


Figure 22. Digital Asset Management

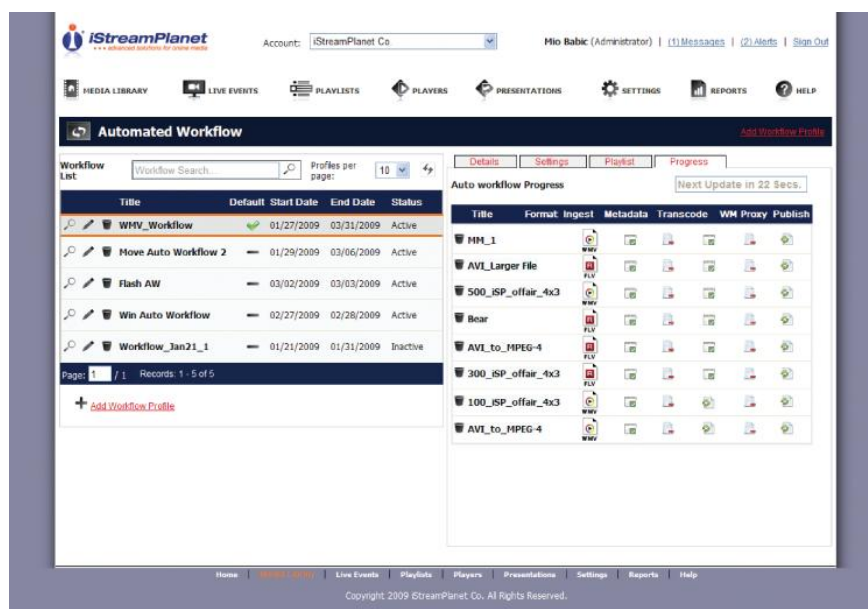


Figure 23. A DAM system enables organizations to manage media assets and an automated workflow.

Checklist

On-demand video workflow gives content owners a wide variety of options to produce, publish, and customize their rich media content. Consider these tips for making the most of on-demand video workflow:

- Use CMS to manage the process of publishing content to staging and production environments, and to manage content metadata.
- Leverage RCE to trim archives of live events; create clips from live streaming events that are still in progress; and to create thumbnails.
- If outside data sources are available to enhance users' viewing experience (such as statistics or play-by-play data for sporting events), use a text stream data insertion application to combine that data with audio and video for seamless, synchronized delivery of rich, integrated programming.
- To monetize your live events, use a live ads insertion application to insert and seamlessly deliver ads at a specific time and location within the video.
- Use DAM with automated workflows to ingest and automatically transcode your media assets into Smooth Streaming media.

Conclusion

Streaming video webcasts are an exciting way to create compelling content that will resonate with the people you are trying to reach. Delivering high-quality streaming media is among the most effective ways to connect with your audience and customers. As new tools and technologies are rapidly emerging, it has never been easier or more cost-effective to produce rich media content with professional results. By taking advantage of Microsoft Silverlight, IIS Media Services, and Smooth Streaming technology, even growing businesses can deliver customized, interactive webcasts to a wide range of audiences. A full service managed webcast provider can provide a complete, end-to-end solution to manage and coordinate the video production and delivery process every step of the way.

With careful planning and best practices, combined with the right technology and production team, today's organizations can take their first steps toward unlocking the benefits of live and on-demand streaming video.

For more information, please visit:

www.microsoft.com/media

<http://www.microsoft.com/silverlight>

<http://iis.net/media>

<http://www.istreamplanet.com>

Appendix A: Sample Hardware/ Software Configuration

For the 2009 Major Grand Slam Tennis Championship, only two origin servers (primary and back-up) were required to perform virtual data chunking. The hardware and software were configured as follows:

Origin Server: Dell PowerEdge 2950

CPU: Intel Xeon E5420 @ 2.5 GHz 2.49 GHz

Memory (RAM): 24GB

OS: Windows Server 2008 Standard 64-bit running IIS
Media Services 3.0



Microsoft
1 Microsoft Way
Redmond, WA 98052 425.882.8080
www.microsoft.com



iStreamPlanet
6630 West Arby Ave., Suite 112
Las Vegas, NV 89118 702.492.5900
www.istreamplanet.com