



**Microsoft**

# Live Broadcasting with Silverlight and Windows Media



Chris Knowlton  
Senior Program Manager  
IIS Media Team

# CBS - March Madness - WMS

Live Games Presented by: **State Farm** Bracket / Games On Demand What's Hot Historical Highlights

 (1) **Kansas** vs.  (13) **Davidson** BOSS BUTTON

[Full Game](#) [Full Game - Audio Only](#) [Highlights](#) [Buzzer Beater](#) [Game Stats and Team Info](#)  



00:00:00 Video Quality Better   Watch in Standard Definition

# FTV - French Open - IIS

This image shows a live broadcast of a tennis match on a clay court. The scene is viewed from an elevated position behind the baseline. A player in a white shirt is in a ready stance on the right side of the court, while another player is visible on the left. A yellow tennis ball is in mid-air near the center of the court. The background features a green wall with 'BNP PARIBAS' branding and several Lacoste brand markers. A scoreboard in the top right corner displays '2 3 4' and 'web'. A technical overlay in the top left corner shows 'Max', 'Frame Rate (25) fps', and 'Now Downloading Bitrate 3000 kbps Limit Max Bitrate 3000 kbps'. A vertical scale on the left side of the court indicates distances from 35K to 3M. The bottom of the image shows a video player interface with 'Replay' and 'Direct' buttons, a play button, and a volume icon. The text 'Diffusion en Silverlight Smooth Streaming - Version Beta' is visible at the bottom center.

Max

Frame Rate (25) fps

Now Downloading Bitrate 3000 kbps Limit Max Bitrate 3000 kbps

BNP PARIBAS

BNP PARIBAS

BNP PARIBAS

2 3 4

web

3M

2M

1.4M

1000K

700K

500K

350K

Merrier

IBM

Replay

Direct

17:05

Diffusion en Silverlight Smooth Streaming - Version Beta



# Agenda

- Technology Overview
- Enterprise Broadcasting
- Internet Broadcasting





**Microsoft**

# Technology Overview



# In This Section...

- Common Media Delivery Methods
  - Unicast
  - Multicast
  - Server Types
- Microsoft Media Ecosystem
  - Clients
  - Servers
  - Encoders
  - Content Management



# Media Delivery Methods



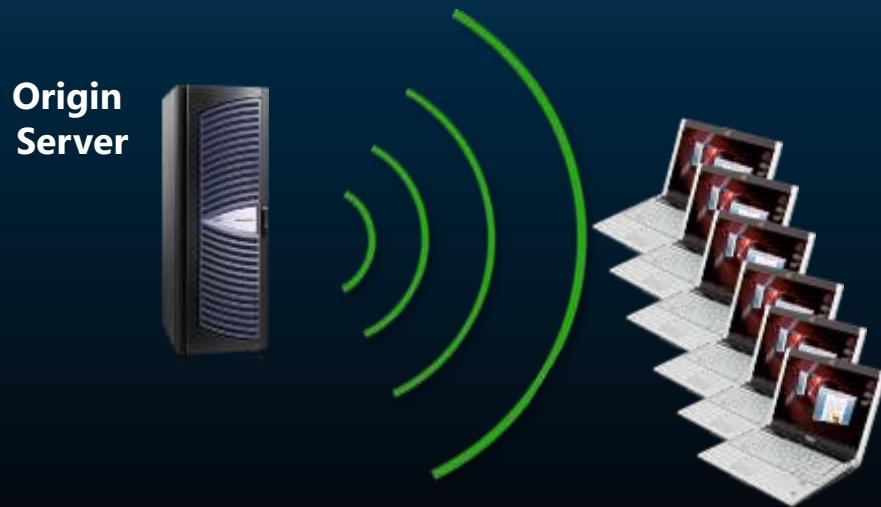
# Media Delivery Methods

- Multicast
- Unicast
  - Traditional Streaming
  - Progressive Download
  - Adaptive Streaming





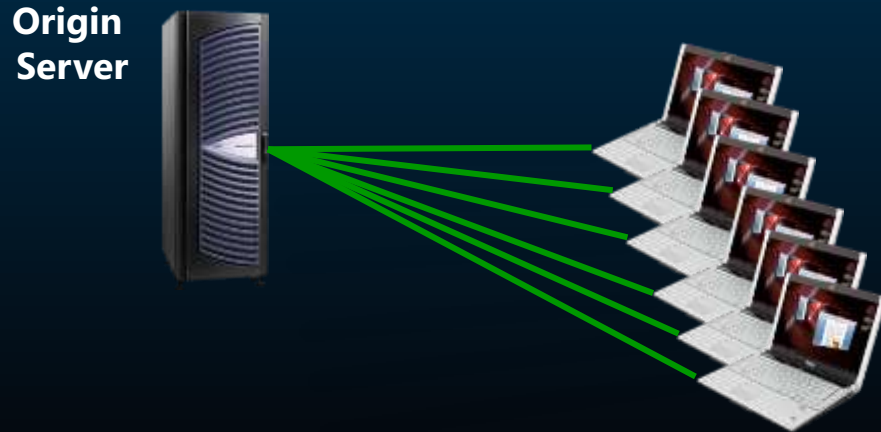
# Multicast



- A single one-to-many stream
- Uses bandwidth of only one stream
- Requires multicast-enabled networks
- Typically requires fewer servers than unicast



# Unicast



- Many one-to-one streams
- For private and public networks
- ...vs. multicast:
  - Requires more bandwidth per user
  - May require more servers



# Unicast Methods

Movie

3

3

3

3

3

Adaptive Streaming



Video @ 01:04?  
Video @ 01:06?  
Video @ 01:08?

Traditional Streaming



Progressive Download

Movie

3

3

3

3

3

# Streaming or Progressive Download?

- Traditional streaming
  - Designed for media delivery (stateful, UDP)
  - Works best in controlled networks – great for enterprise environments and IPTV
  - Scales poorly when demand tops provisioning
  - Easily disrupted by variable network conditions
- Progressive download
  - Scales efficiently over HTTP networks
  - Easily cacheable
  - Unthrottled download speeds waste bandwidth



# Adaptive Streaming

- Hybrid media delivery method
  - Acts like streaming but is in fact a series of short HTTP downloads
  - Video and audio delivered as series of small files
- Built for the Web
  - Leverages existing HTTP caches
  - Scales exceptionally well to meet high demand
- Adapts bit rate to local conditions
- Applicable to both on-demand and live delivery



# Server Types

- Origin Server
  - Provides the source content (file or encoder stream) to downstream servers and end users
- Middle Tier and Edge Server
  - Distribution Server
    - Typically requires per-event configuration
    - Receives streams from upstream origin servers and redistributes them to other servers and end users
  - Cache/Proxy Server
    - Typically requires one-time configuration
    - Proxies initial downstream requests to upstream servers
    - Splits live streams for downstream servers and end users



# Microsoft Media Ecosystem



# Microsoft Media Platform



Source



Encode



Server



Client



Microsoft  
Expression Encoder 3



Windows  
Media Encoder  
9 Series  
x64 Edition

Third-Party  
Carrier-Class  
Encoders



Windows Server<sup>™</sup>  
Internet Information Services 7.0



Windows  
Media Services 2008



Microsoft<sup>™</sup>  
Office SharePoint  
Server 2007



Microsoft<sup>™</sup>  
Silverlight



Windows  
Media Player 11

Microsoft<sup>™</sup>



# Encoders



# Microsoft Encoders

## ● Windows Media Encoder 9 Series

- Still very popular
- Only Windows Media codecs & format
- Scriptable via SDK
- Robust unicast and multicast streaming
- Live DRM support
- **Near end of product lifecycle**

## ● Expression Encoder 3

- Newer codecs
  - VC-1, H.264, AAC
- More import formats
- Windows Media for live and on-demand
- SDK for .NET devs
- Direct publishing to IIS
- Silverlight templates
- Smooth Streaming for on-demand video...



# More on Expression Encoder

- Expression Encoder 3 & Smooth Streaming
  - Easily encode Smooth Streaming presentations
  - Choose a Smooth Streaming Silverlight player from built-in templates – single file, or gallery player
  - Minimize files on disk for manageability
    - One file per video bit rate (or)
    - A single file for all audio and video content
  - Publish directly to an IIS server
  - Publish directly to a CDN using plug-in model
  - [Tune client heuristics](#)



# 3rd Party Encoding Solutions

- Announced support for Smooth Streaming:
  - Anystream (Grab Networks)
  - Digital Rapids
  - Envivio
  - Inlet Technologies
  - Rhozet
  - Telestream
  - Twofour Digital
  - VBrick
  - Viewcast
  - Winnov



# Servers





Windows Server® 2008

# Windows Server 2008 R2



# Delivering Media with Windows Server



- Traditional Streaming
  - Unicast
  - WMS RTSP
  - WMS HTTP
- Multicast Streaming
  - WMS Multicast

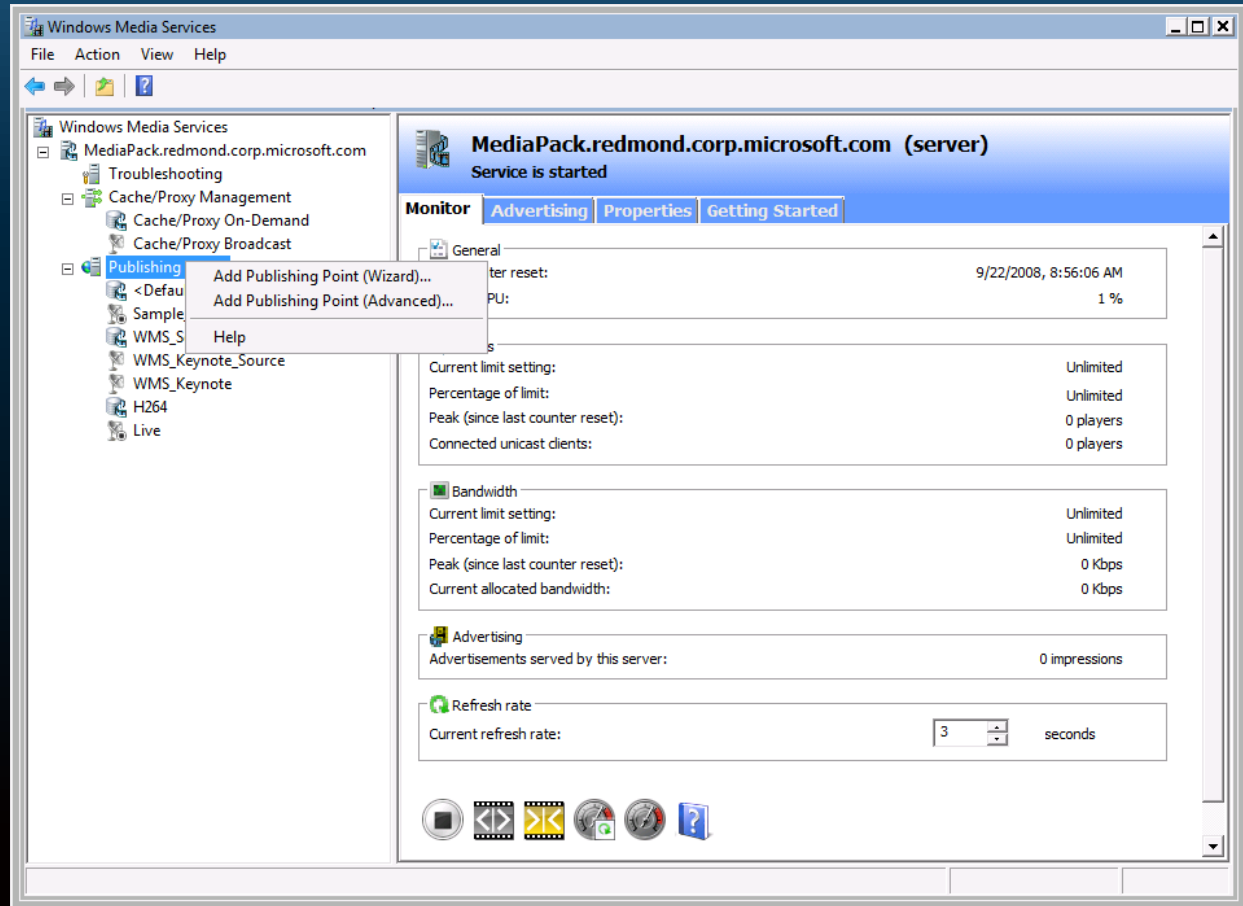


- Progressive Download
  - Bit Rate Throttling
  - Web Playlists
- Adaptive Streaming
  - Smooth Streaming
  - Live Smooth Streaming



# Windows Media Services

- Reliable
- Scalable
- Affordable





# Core WMS Scenarios

- Enterprise
  - On-demand training
  - Live executive broadcast
  - Programmed broadcast
- Internet
  - News & entertainment (with ads)
  - Music & movie services (subscription, PPV, wireless)
  - Internet-based Radio/TV stations (with ads)
  - Radio & television rebroadcasts
  - Live broadcasts



# Key WMS Features

- Live & on-demand streaming
- Unicast (HTTP, RTSP) & multicast delivery
- Access control via authentication / authorization
- Server-side playlists
- Rich logging & advertising support
- Multiple-bit-rate (MBR) & Intelligent Streaming
- Fast Streaming & Advanced FF/RW
- Archiving & Play While Archiving
- Extensible plug-in model



# Key Newer WMS Features

- Server Core installation...
- Built-in Cache/Proxy plug-in...
- >2x scalability increase...
- 88% lower price for most Internet scenarios
- Remote Server Administration Tool
- MOM 2005 Management Pack



# Server Core Option

- Design
  - Minimal-footprint headless [installation option](#)
  - For running fixed-function server roles
  - Good option for [Windows Embedded](#) appliances
  - Supports WMS native code plug-ins
- Benefits
  - Eliminates GUI and client features
  - Reduces hardware requirements
  - Reduces overall attack surface
  - Reduces servicing costs



# WMS Cache/Proxy Plug-In

- Usage Options
  - Proxy – allows broadcast stream splitting
  - Caching
    - Opportunistic caching
    - Pre-stuffing (e.g., using DFSR)
    - Adheres to Expiry Date on content
  - Reverse Proxy
    - Provides a gateway server to users
    - Redirects content requests to an origin server



# WMS Cache/Proxy Plug-In

## ● Implementation

- Basic functionality for most scenarios
- Ships in the box, disabled by default
- Written in C++ (can be used on Server Core)

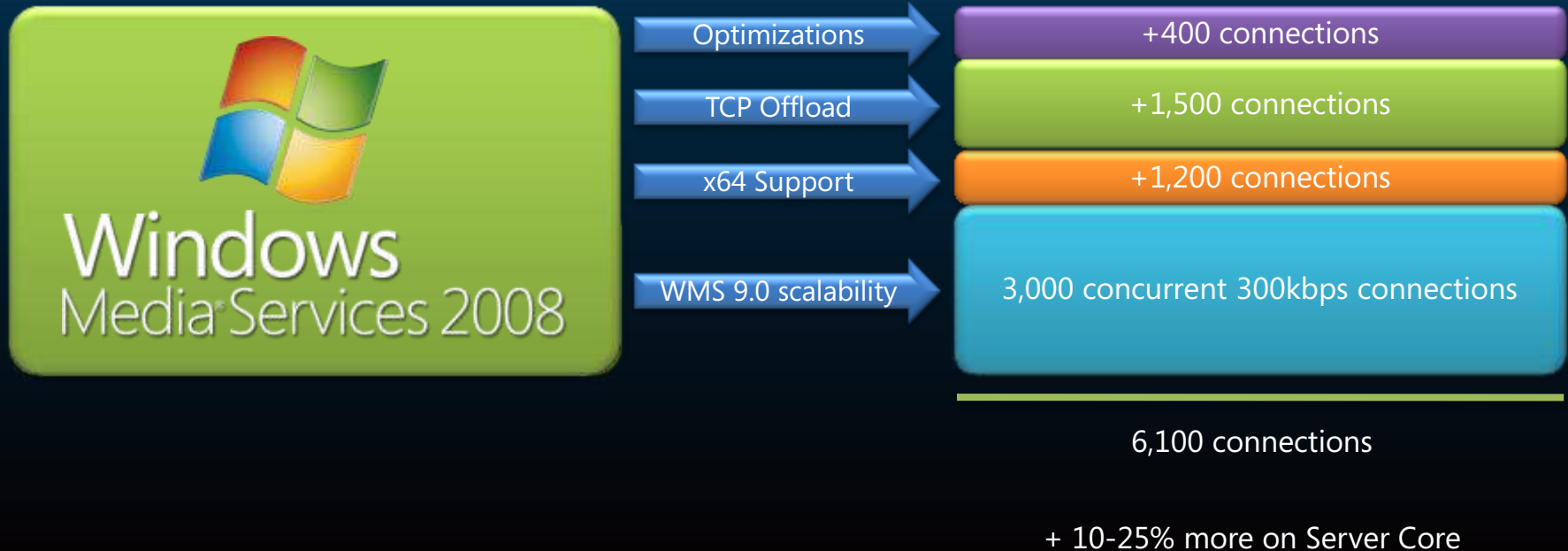
## ● Benefits

- Improved end user experience
- Reduced load on the origin server
- Reduced load on the network



# WMS Scalability

- Example on standard rack-mount server



# Affordable Options

- WMS 2008 is available for the following editions of Windows Server 2008:

<b>Edition</b>	<b>Cost</b>	<b>Features</b>
Web Server 2008	\$ 469 / server	99%, Web-facing
Standard	\$ 999 / server	99% of features
Enterprise	\$3,999 / server	All features
Datacenter	\$2,999 / proc.	All features

- Multicast streaming requires Enterprise or Datacenter edition





# WMS 2008 Summary

- For streaming of Windows Media content
- Free download for all editions of Windows Server 2008
  - Reduced footprint – Server Core install
- Includes WMS Cache/Proxy Plug-In
  - Reduces load on origin server and network
- See <http://www.iis.net/wms> to learn more



# IIS Media Services

Integrated Web/Media Platform on  
Windows Server



# IIS Media Services Goals

- Extend Media Engagement
  - Leverage scale of caching infrastructures
  - Deliver True HD (720p+) live & OD video
- Measure and Monetize Media
  - Real-time server- and client-side logging
  - Deliver the right ads at the right time
- Create One Web Platform
  - Consolidate multiple media formats
  - Manage Web and media content together

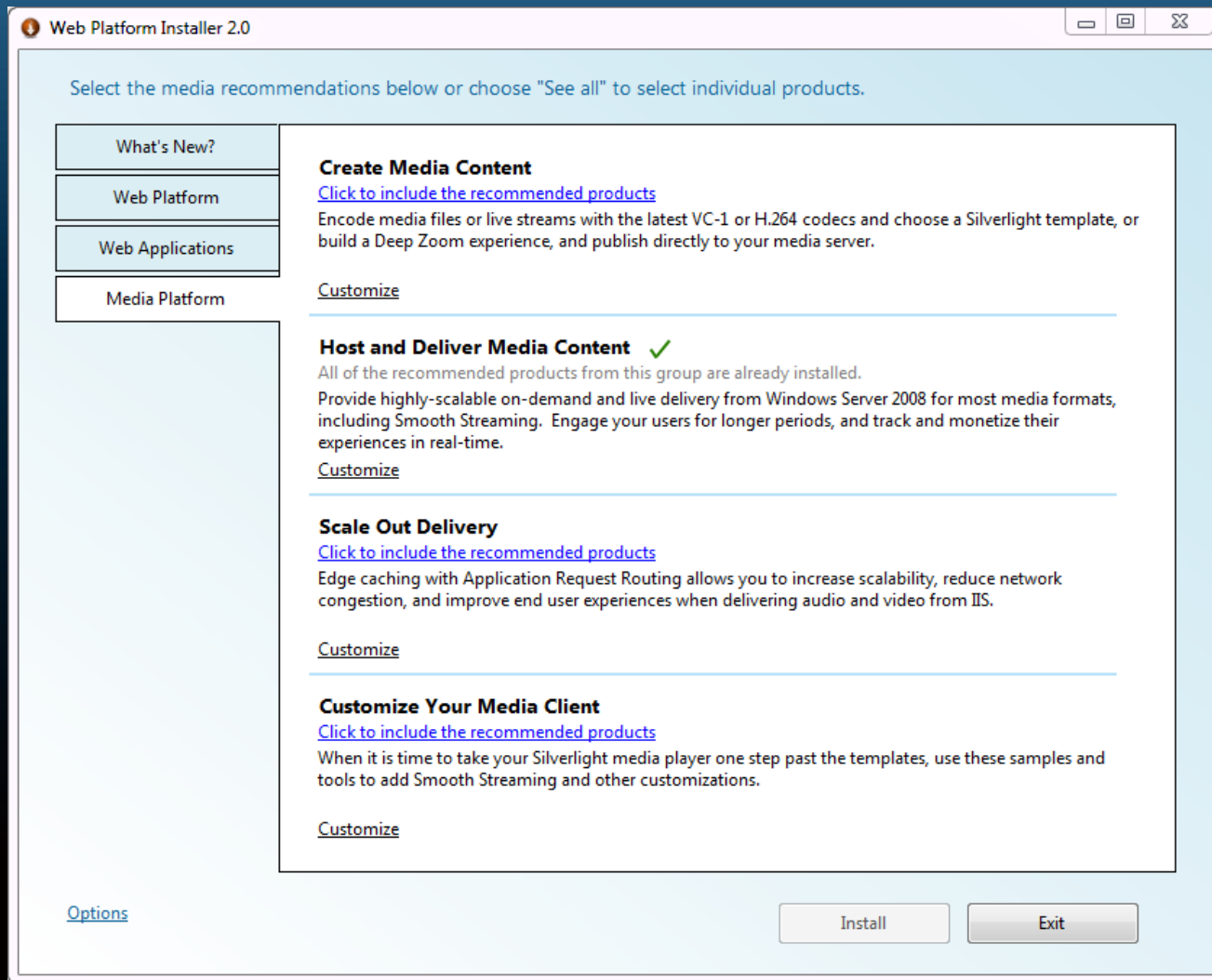


# IIS Media Services 3.0

- Enhanced delivery of media over HTTP
- Free IIS7 extension for Windows
- Download from <http://iis.net/media> via WebPI...
- Features...
  - Bit Rate Throttling
  - Web Playlists
  - Live and on-demand Smooth Streaming
  - Advanced Logging
  - Application Request Routing
  - Smooth Streaming Player Development Kit



# Web Platform Installer



# Deliver Existing Content Better

## ● Bit Rate Throttling

- Save bandwidth – send only what is watched
- Use Dynamic Throttling to apportion bandwidth
- Configure throttling for many content types

## ● Web Playlists

- Control content playback, enforcing order
- Monetize content in ad-funded scenarios
- Configure playlists for many content types

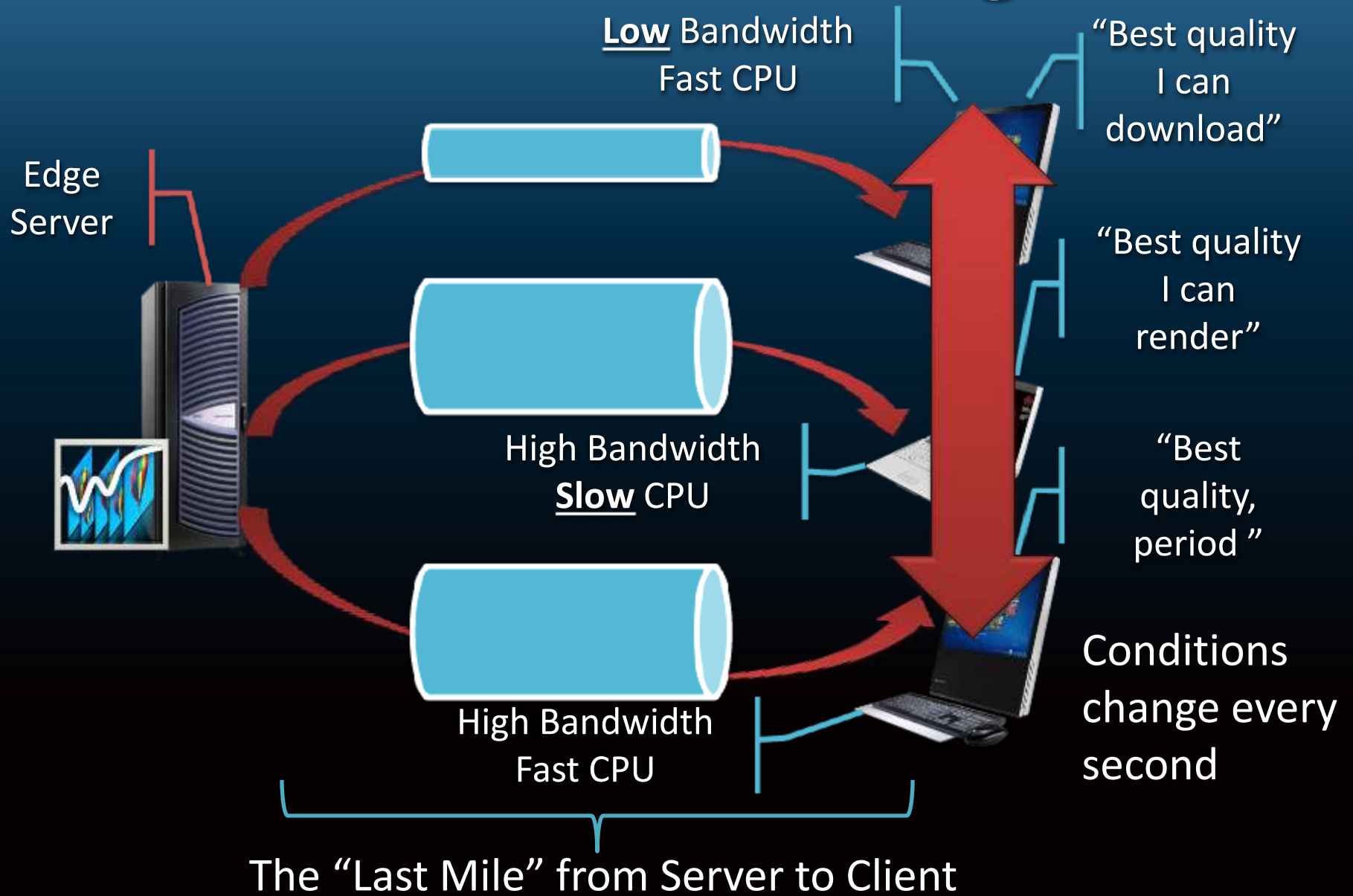


# Provide Smooth Playback

- Smooth Streaming
  - Microsoft implementation of HTTP-based adaptive streaming
  - Best of both worlds
    - Responsive and efficient like streaming
    - Cheap and scalable like progressive download
    - Superior user experience
- Most future Silverlight video events will be delivered with Smooth Streaming!



# The "Last Mile" Challenge

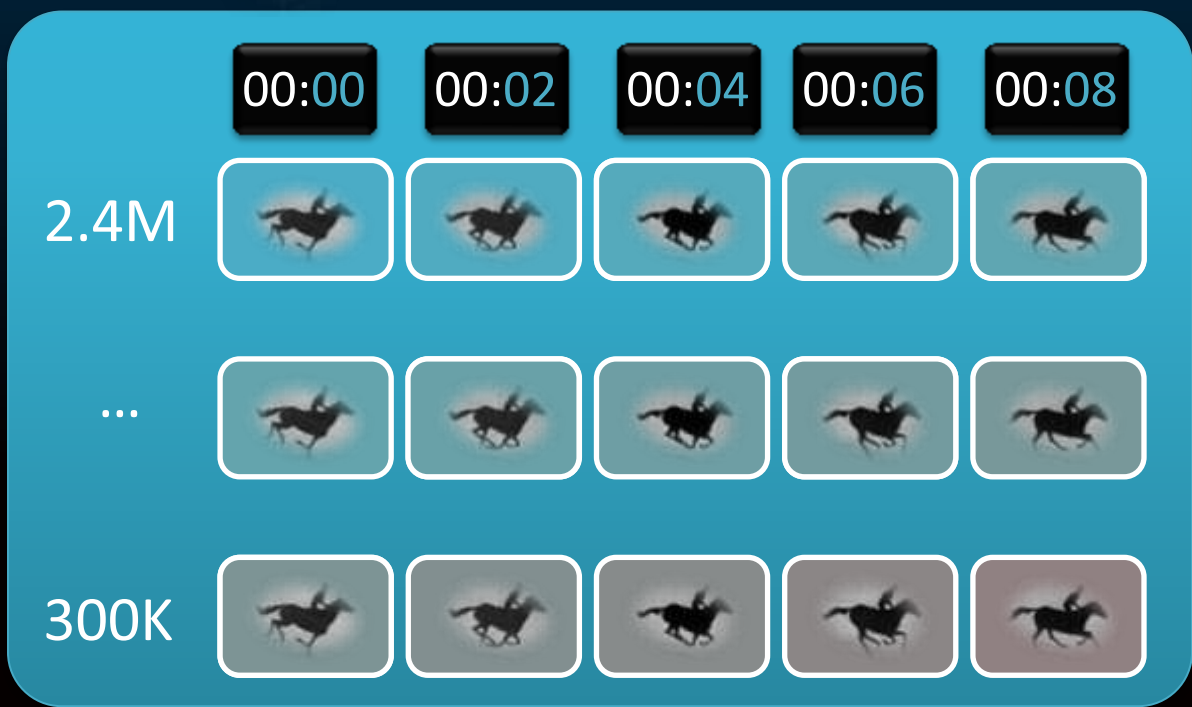




# Adapting Bit Rate in Real-Time



300K @ 00:00?  
700K @ 00:02?  
2.4M @ 00:04?  
1.5M @ 00:06?  
2.4M @ 00:08?



300K (start quickly)  
700K (good network)  
2.4M (great network)  
1.5M (glitch)  
2.4M (play on...)

Bit Rate  
Heuristics

# Solve the "Last Mile" Challenge

- Smooth Streaming – adapt bit rate in real-time
  - Best experience possible for conditions
  - Right content at the right time
- Give your users the best experience
  - Video playback does not stop
  - Best quality for client conditions at all times
  - Instant start-up, instant seek
- Leverage HTTP cacheability
  - Better QoS with the 10-20x reach of HTTP caching
  - Smart client heuristics self-correct video delivery



*demo*

# Smooth Streaming In Action

[www.iis.net/media/showcase](http://www.iis.net/media/showcase)

# Smooth Streaming Benefits

- Eliminates the “re-buffering” experience
  - Increases content stickiness 50 to 200%
- Dramatic increase in Quality of Experience (QoE)
- One click on “Watch Now”
  - No more high, medium, low
- Transparent user experience



# Top Smooth Streaming Requests

- Content protection
- Scaling to a growing audience
- Reporting, real-time monitoring, analytics
- Support for Live events with DVR
- Expanded encoder ecosystem support

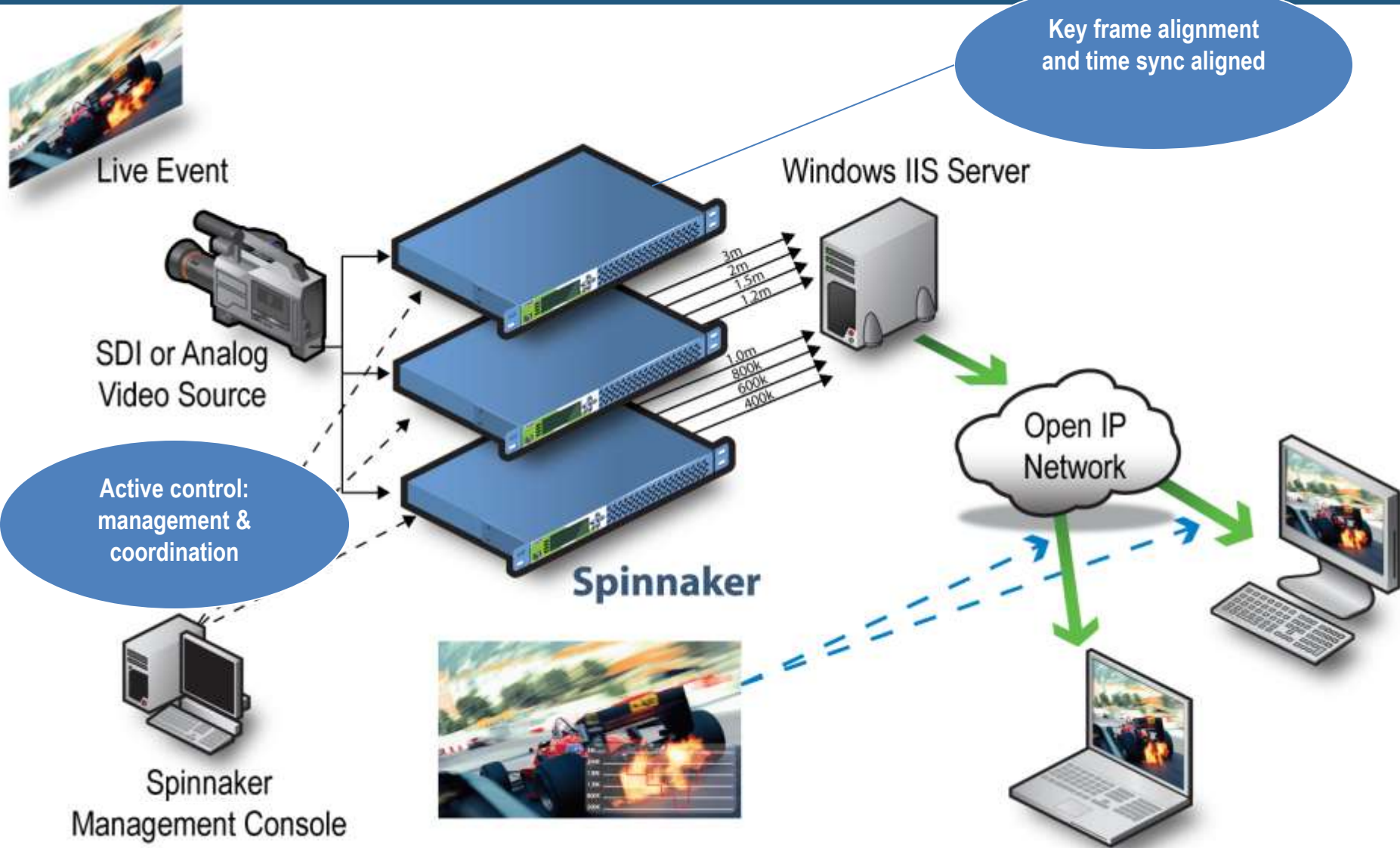


# Live Smooth Streaming

- Builds on Smooth Streaming
  - Cacheable HTTP delivery for Live events
  - Network Digital Video Recorder (DVR)
    - Pause, Instant Replay, Go to Start, Go to Live...
- Synchronized in-stream text & metadata
  - Captioning and subtitling
  - Sparse data (e.g., chapter markers)
  - Control events (e.g., ad insertion points)
- Optimizations for FF, Rewind, Slow, Scrub



# Live Smooth Streaming



# Scaling Out

- IIS Origin, Distribution, Edge servers
  - Each tier adds value to light up the network
  - Use for Content Ingest, Distribution, Failover, Load Balancing, Archiving, Ad Insertion, etc.
  - Application Request Routing (ARR)
    - A new IIS extension for HTTP caching & proxying
    - Provides Smooth Streaming request consolidation
    - Authenticates with upstream servers





# Scaling Out

- Origin, Distribution, Edge servers built on IIS
  - Each layer adds unique value to light up the network
    - Archiving, Load Balancing, Failover, Ad Insertion, etc.
  - Application Request Routing (ARR) extension
    - HTTP cache/proxy for IIS
    - Request consolidation
    - Delivery chain authentication
- Advanced Logging
  - Real-time integration for logging data analysis
  - Rich user engagement data to improve ROI
  - Centralized client logging for large networks



# Core IISMS Scenarios

IIS Media Services version

1.0

2.0

3.0

- Enterprise

– On-demand training	✓	✓	✓
– Live executive broadcast			✓
– Programmed broadcast			✓

- Internet

– News & entertainment	✓	✓	✓
– Music & movie services	✓	✓	✓
– Internet-based Radio/TV stations		✓	✓
– Radio & television rebroadcasts	✓	✓	✓
– Live broadcasts			✓



# Live Streaming Comparison

Live Streaming Features	WMS 2008	IISMS 3.0
Windows Media traditional streaming	✓	
Client-side logging	✓	✓
Broadcast stream-splitting	✓	✓
Broadcast streaming from files	✓	✓
High Availability content sourcing (encoder failover)	✓	✓
Archiving of live streams	✓	✓
Multiple-bit-rate streaming (Intelligent vs. Smooth)	✓	✓
HTTP Streaming	✓	✓
RTSP Streaming	✓	
UDP streaming	✓	
Multicast streaming	✓	
Play While Archiving (late joiner feature)	✓	✓
Edge caching and proxying	✓	✓
Full Network PVR		✓
Maximum scalability on existing HTTP caching networks		✓



# Windows Server Summary

- Windows Media Services
  - Proven traditional streaming
  - Unicast: HTTP, RTSP (TCP & UDP)
  - Multicast and other advanced features
- IIS Media Services
  - New hybrid of traditional unicast streaming and progressive download functionality
  - Smooth Streaming delivers better user experiences and the highest scalability



Microsoft

10-minute Break



# Clients



# Silverlight vs. Windows Media Player

	Windows Media Player 11	Silverlight 3
Pros	<ul style="list-style-type: none"><li>• &gt;10 years of product development</li><li>• Full Windows Media feature support</li><li>• Can be embedded in Windows apps</li><li>• WMS RTSP support</li><li>• Library management</li><li>• Portable device integration</li></ul>	<ul style="list-style-type: none"><li>• Highly customizable, rich UI</li><li>• Interactive video display</li><li>• Windows, Mac, Linux</li><li>• .NET support (C#, VB, JScript)</li><li>• Better dev &amp; design tools</li><li>• Smooth Streaming support</li></ul>
Cons	<ul style="list-style-type: none"><li>• Windows only</li><li>• Difficult to customize UI, controls</li><li>• Static video rectangle</li><li>• Only JScript, VBscript for web devs</li><li>• Inconsistent playback experience (OS version, codecs, hardware, drivers)</li></ul>	<ul style="list-style-type: none"><li>• More secure DRM (PlayReady)</li><li>• Consistent playback experience</li><li>• Self-sufficient browser plug-in</li><li>• No default player experience</li></ul>



# Smooth Streaming - Client

- New Smooth Streaming Player Development Kit
  - Smooth Streaming Player SDK
  - Push Encoder for simulating live streams
  - Sample Smooth Streaming Player
- Built on Silverlight for cross-platform reach
  - Runs on Mac, Windows and Linux
- Makes use of existing .NET skills
- Seamless integration between:
  - Expression Studio design tools
  - Visual Studio developer tools





# Smooth Streaming – Client

- Basic Playback controls - APIs, Events, Properties
- Advanced Playback – PVR, slow motion, scrubbing
- Monetization
  - Ad Playback integration – scheduling capabilities, tracking Ad progress
  - Live Ad Insertion with Live Smooth Streaming
  - Send analytics data to IIS Advanced Logging
- Content Protection – PlayReady integration
- Multi-track selection (camera angles, languages, etc.)
- Support for progressive download Ads/content



# Content Management



# Content Management

## ● File Shares

- Easy
- Simple to use
- Can apply ACLs

## ● SharePoint (MOSS)

- Asset libraries
- Built-in workflows
- Discoverable content
- Templates, e.g...
  - Podcasting Kit for SharePoint (PKS)...

## ● End-to-end Systems

- Video capture
- Slide synchronization
- Media delivery
- Authentication
- Real-time chat
- Content management
- Searchability
- Integrated
- Comprehensive
- Cost-effective



*demo*

# Podcasting Kit for SharePoint (PKS)

[http://getsharp.3sharp.com/Pages/  
pkshomepage.aspx](http://getsharp.3sharp.com/Pages/pkshomepage.aspx)



**Microsoft**

# Live Broadcasting in the Enterprise



# Enterprise broadcasting

- The Corpnet challenge
- Deployment options, with pros & cons
- Overview of how to do it with MS products
- WMS unicast or multicast workflow
  - to WMP
  - to Silverlight

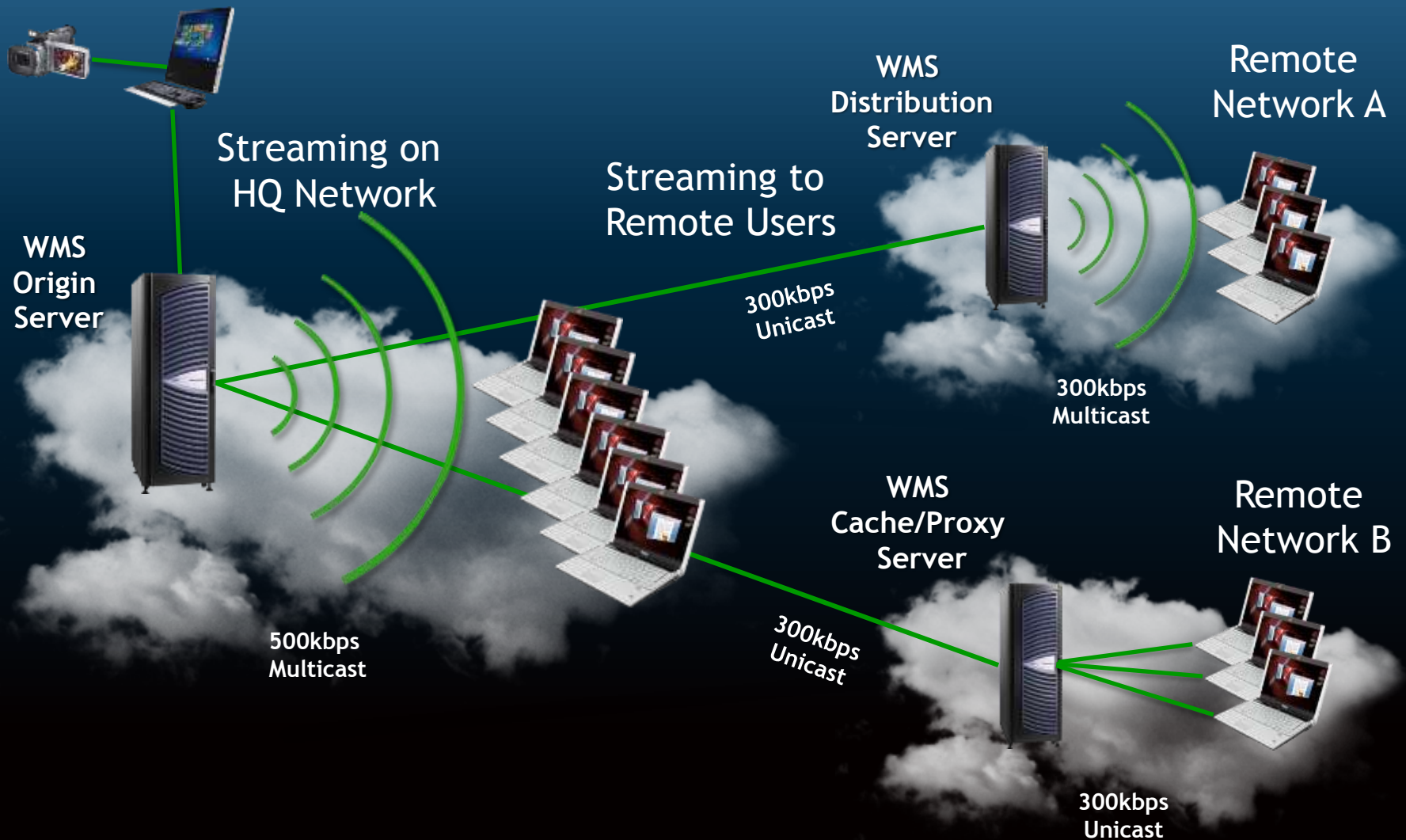


# Corpnet Streaming Challenge

- Corporate networks may need upgrades to meet video bandwidth needs
- Enabling a unicast-only network for video can be a significant project:
  - Add unicast cache/proxy servers, and/or...
  - Add multicast support to network equipment
- Besides some up-front CapEx, video delivery usually requires dedicated resources and on-going OpEx

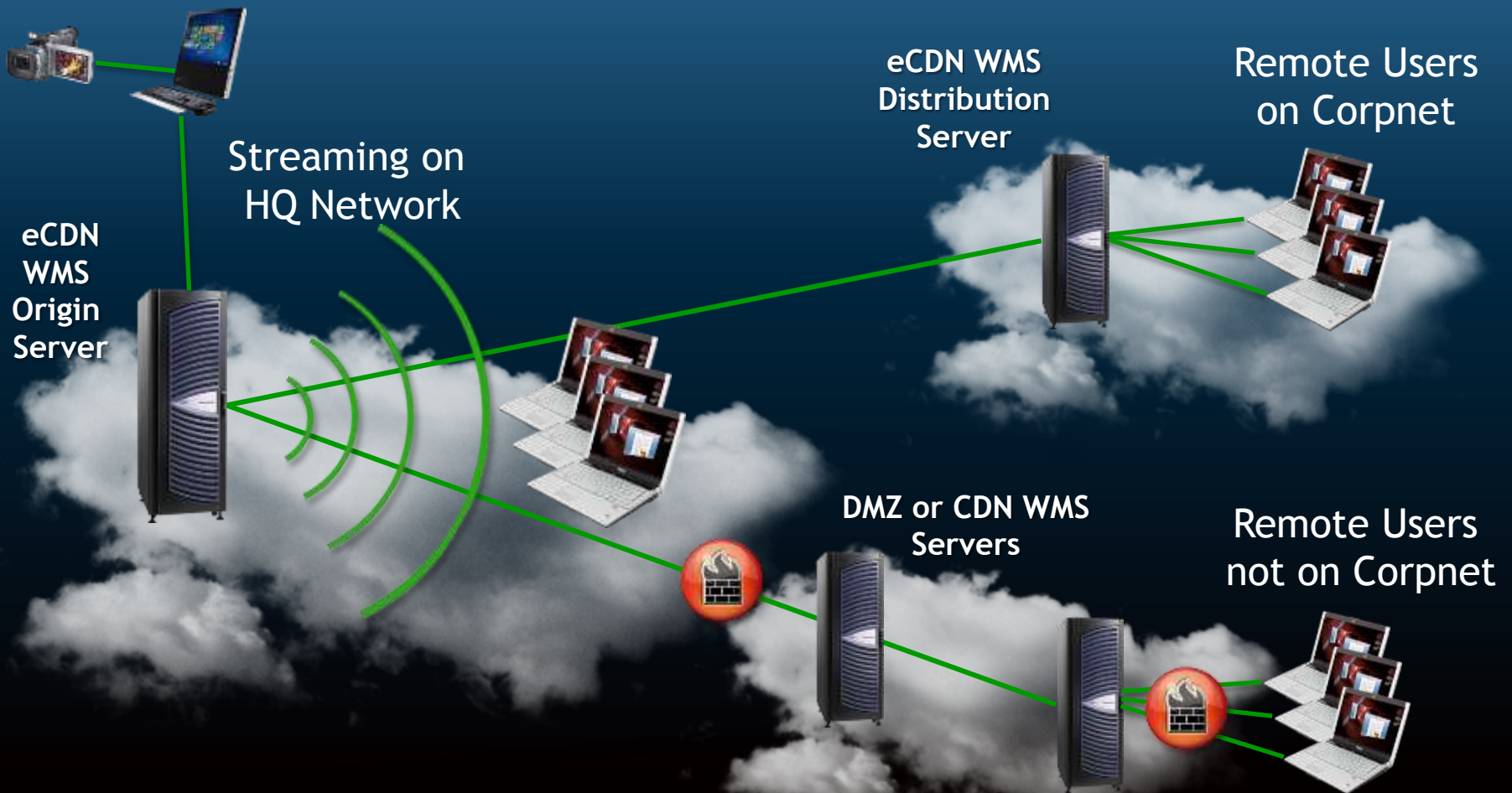


# Corpnet eCDN Deployment

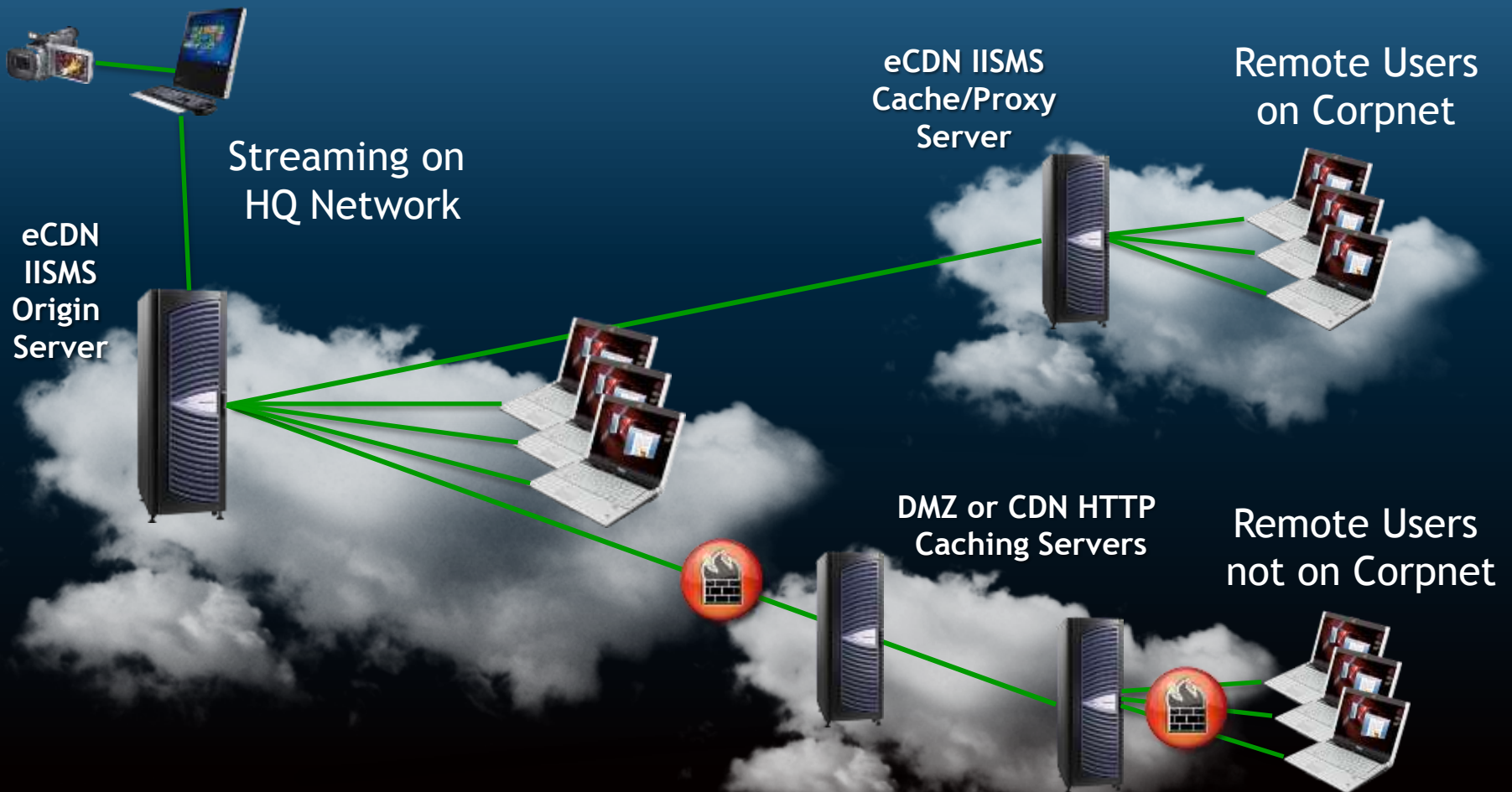




# Corpnet + Edge Deployment



# Smooth Streaming



# Deployment Comparison

Type	Pros	Cons
Corpnet eCDN Streaming	<ul style="list-style-type: none"><li>• All in-house – full control</li></ul>	<ul style="list-style-type: none"><li>• Requires streaming and networking ops resources</li><li>• Requires acceptable WAN links to each remote site</li><li>• Works best for many users when multicast is available</li></ul>
Corpnet eCDN + CDN Streaming	<ul style="list-style-type: none"><li>• Helps reach remote sites without good WAN links</li><li>• Reduces resource req'ts</li></ul>	<ul style="list-style-type: none"><li>• Adds CDN costs</li></ul>
Smooth Streaming	<ul style="list-style-type: none"><li>• If local bandwidth is sufficient for the number of users, UX is very good</li><li>• Good UX for remote users</li><li>• Makes use of HTTP caches</li></ul>	<ul style="list-style-type: none"><li>• May require adding HTTP caches or CDN costs</li><li>• May strain the network if there are many end users</li></ul>



# A Case for Corpnet Streaming

- Live enterprise video quickly brings your work community together without travel or event costs – smart, and eco-friendly!
- The benefits of live video delivery can be high, leading to improved morale and increased productivity



# Multicast for Silverlight

- Project Starlight for Silverlight 2 & 3
  - A WMS Multicast Plug-in for Silverlight
  - Leverages existing multicast infrastructures
  - Integrates transparently with Silverlight
  - Works wherever Silverlight works
  - Developed by [Qumu](#)
  - Available as open source on Codeplex:  
<http://projectstarlight.codeplex.com/>
- Multicast is built into Silverlight 4



# Live Multicast Workflow

- Simple workflow to test the Starlight multicast proxy for Silverlight
- Overview
  - Start a stream in Expression Encoder 3
  - Start a multicast publishing point in WMS
  - Upload the Starlight sample player to IIS
  - Install the Starlight multicast proxy on clients



# Live Multicast Workflow - EE3

- In Expression Encoder, click Live Encoding
- Select Live sources (e.g., cameras)
- Set up the Encoding options (e.g., Streaming 512k)
- Add Metadata (Title, Author, etc.)
- In the Output panel, set the streaming mode
  - Broadcast (pull mode)
    - Good for pulling content in from outside a firewall or intra-firewall streaming
    - Allows server to auto-reconnect if the network drops
  - Publishing Point (push mode)
    - Good for pushing a stream out from behind a firewall
- Press Start to begin



# Live Multicast Workflow - WMS

- In WMS, create a publishing point...
  - Broadcast publishing point
  - Multicast delivery
  - Source from `http://<encoder>:8080`
  - Create an .nsc file
  - Automatically retrieve stream formats
  - Save the .nsc file to an IIS Web site on the same server
  - Start the publishing point





# Live Multicast Workflow - IIS

- Download the server-side sample player from <http://projectstarlight.codeplex.com/>
- Unzip the Starlight files into a directory
- Choose or create a Site to use in IIS
- Create a virtual directory under your Web site pointing at the Starlight player files



# Live Multicast Workflow - SL

- Download the Starlight multicast client plug-in from <http://projectstarlight.codeplex.com/>
- Install the Starlight plug-in on PCs and Macs that will receive multicast streams
- Point your browser to the SamplePlayer on your IIS site
- Type the .nsc URL in the player text box





**Microsoft**

# Live Broadcasting on the Internet



# Internet Streaming Challenge

- Traditional streaming Pros:
  - Was designed for efficient media delivery
  - Works well in small networks
- Traditional streaming Cons:
  - Doesn't scale to today's Internet audience
    - RTSP standardized in 1996-1998
    - Estimated number of Internet users in 1998:  
100 million (source: DNS.net)
    - Estimated number of Internet users in 2009:  
1500 million (source: Miniwatts Marketing Group)

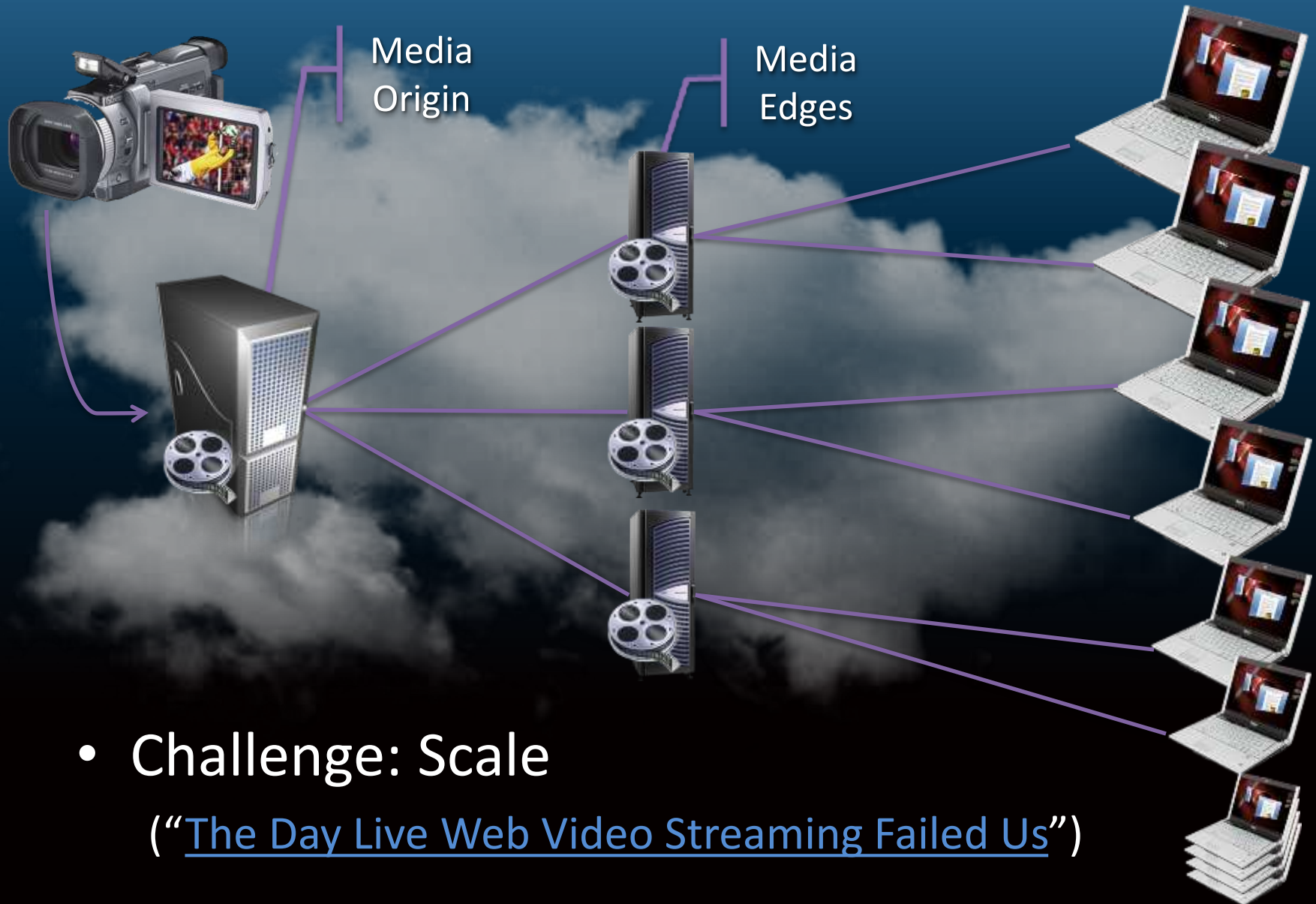


# Internet Streaming Challenge

- Tradition streaming Cons (cont'd):
  - Doesn't take into account today's Internet structure and organization
    - When RTSP was being standardized, none of the top CDNs even existed!
    - HTTP was and still is the #1 Internet protocol
  - Not a perfect solution for:
    - The wide range of end user bandwidths
    - High bandwidth volatility



# Using a Traditional Streaming Network



- Challenge: Scale

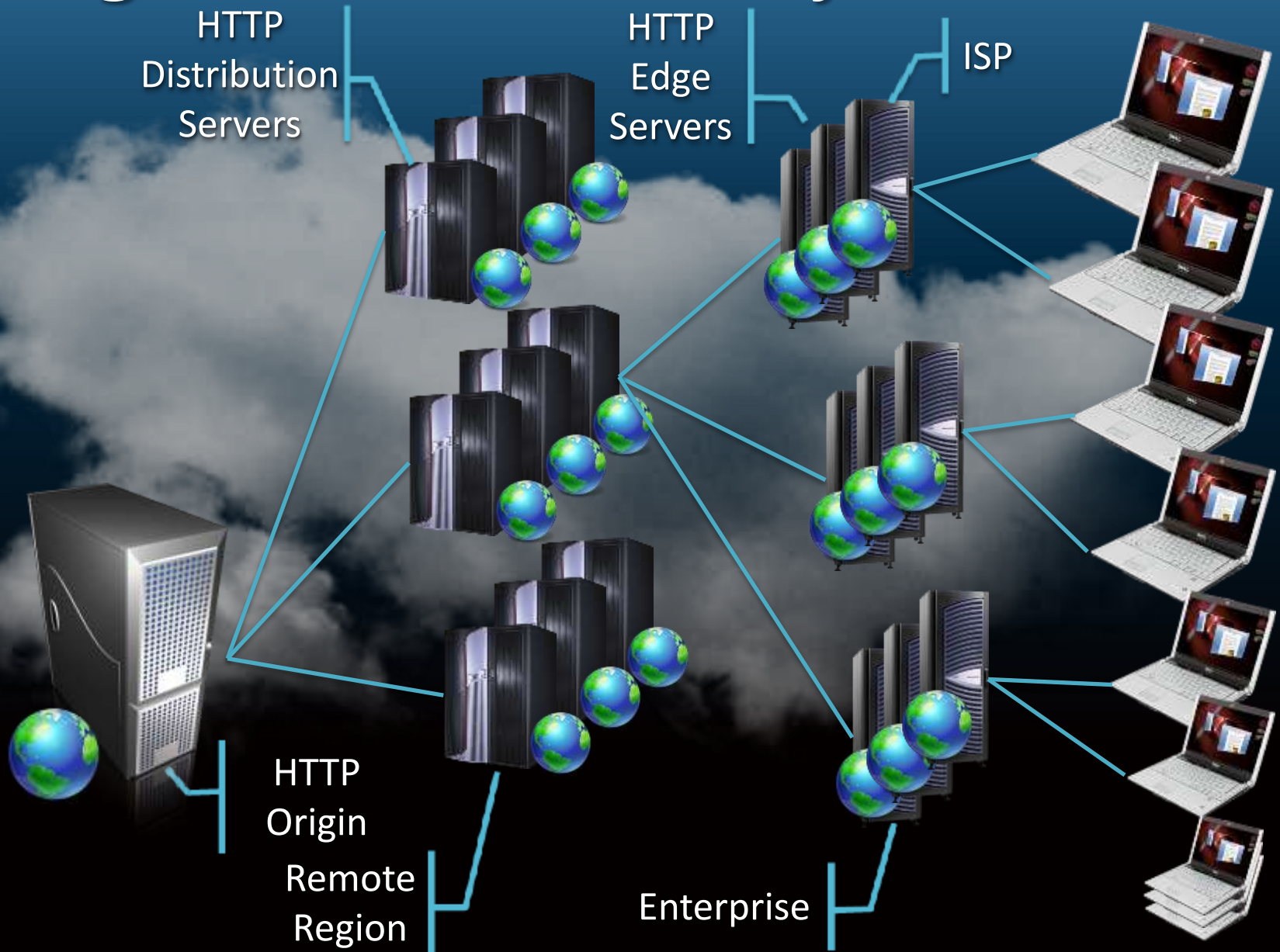
(“The Day Live Web Video Streaming Failed Us”)

# HTTP Is (Still) King

- Majority of Internet traffic is data (text, images) transferred via HTTP
- All CDNs had to first build out an advanced, reliable HTTP infrastructure
  - Non-HTTP servers rarely a priority
- Web caches are commoditized
  - Cheap to deploy in large quantities
  - Windows, Linux, hardware, etc.



# Using an HTTP Delivery Network



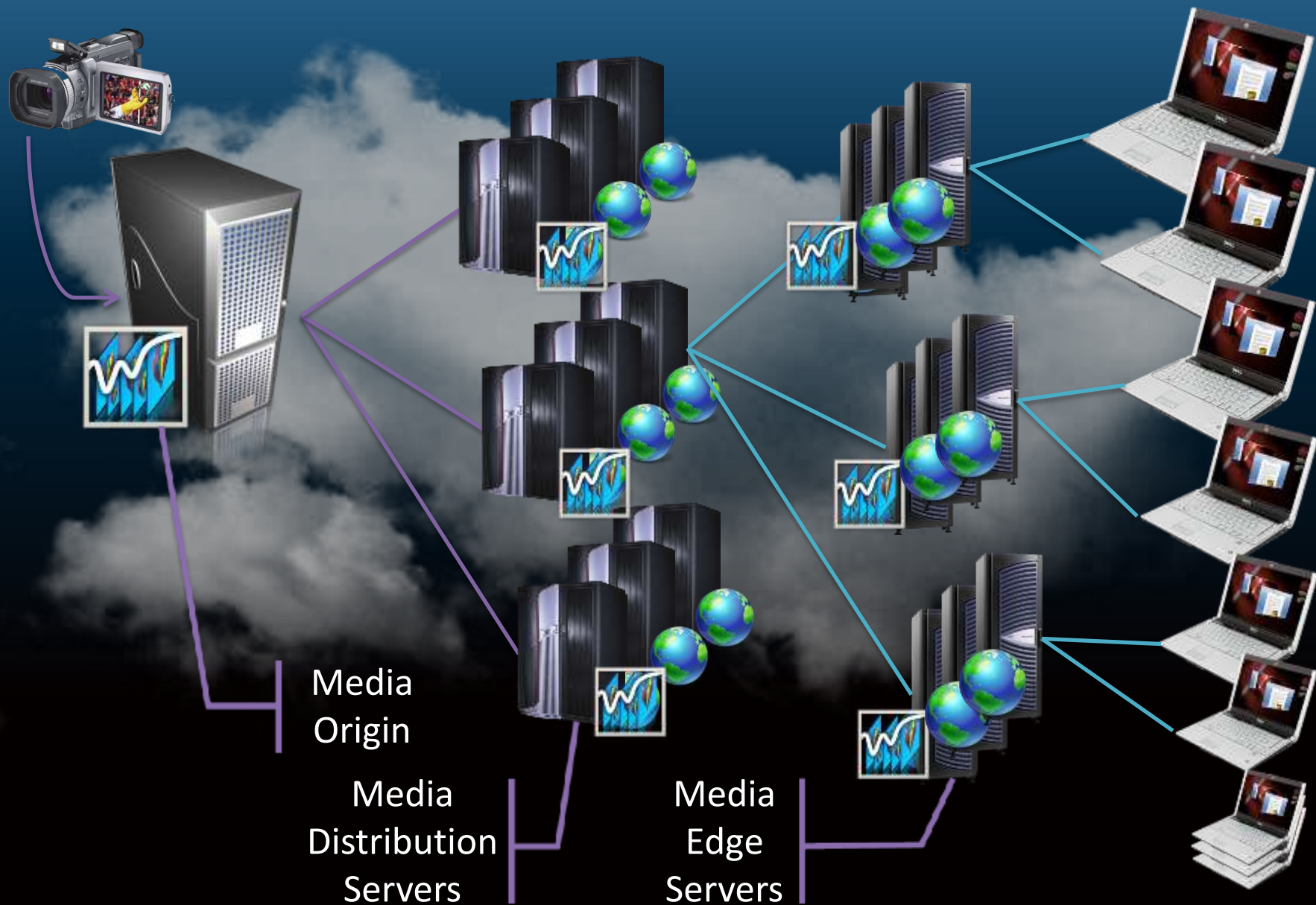


# A Case for Smooth Streaming

- Why force or invent new streaming protocols when the Internet is already built to handle high volume HTTP traffic?
- Instead of trying to adapt the entire Internet to streaming protocols - why not just adapt media delivery to the Internet?



# Smooth Streaming over the Internet



# Content Provider Benefits

- Cheaper to deploy
  - Can utilize any generic HTTP caches/proxies
  - Doesn't require specialized servers at every node
- Better scalability and reach
  - Reduces "last mile" issues because it can dynamically adapt to inferior network conditions
- Smooth Streaming clients...
  - Adapt to local conditions
  - Eliminate guessing by content providers about which bit rates are accessible to their end users



# End User Benefits

- Fast start-up and seek times
  - Start-up/seeking can be initiated on the lowest bit rate before moving up to a higher bit rate
- No buffering, no disconnects, no stutter
  - User must still meet a minimum bit rate requirement
- Seamless bit rate switching based on network conditions and video rendering capabilities
- A generally consistent, smooth playback experience



# Smooth Streaming Architecture



# Smooth Streaming Workflow

## IIS Smooth Streaming Media Workflow



**Expression Encoder 3** encodes video at different quality levels, typically with each in a contiguous MP4 file, then publishes the manifests, the Silverlight 3 player, and video files to IIS7

**IIS Smooth Streaming** creates small cacheable fragments from the contiguous MP4 files, then delivers the Web page, the client manifest, the Silverlight 3 player, and the cacheable MP4 fragments

**Silverlight 3** reads the manifest and plays MP4 fragments, adaptively switching between quality levels

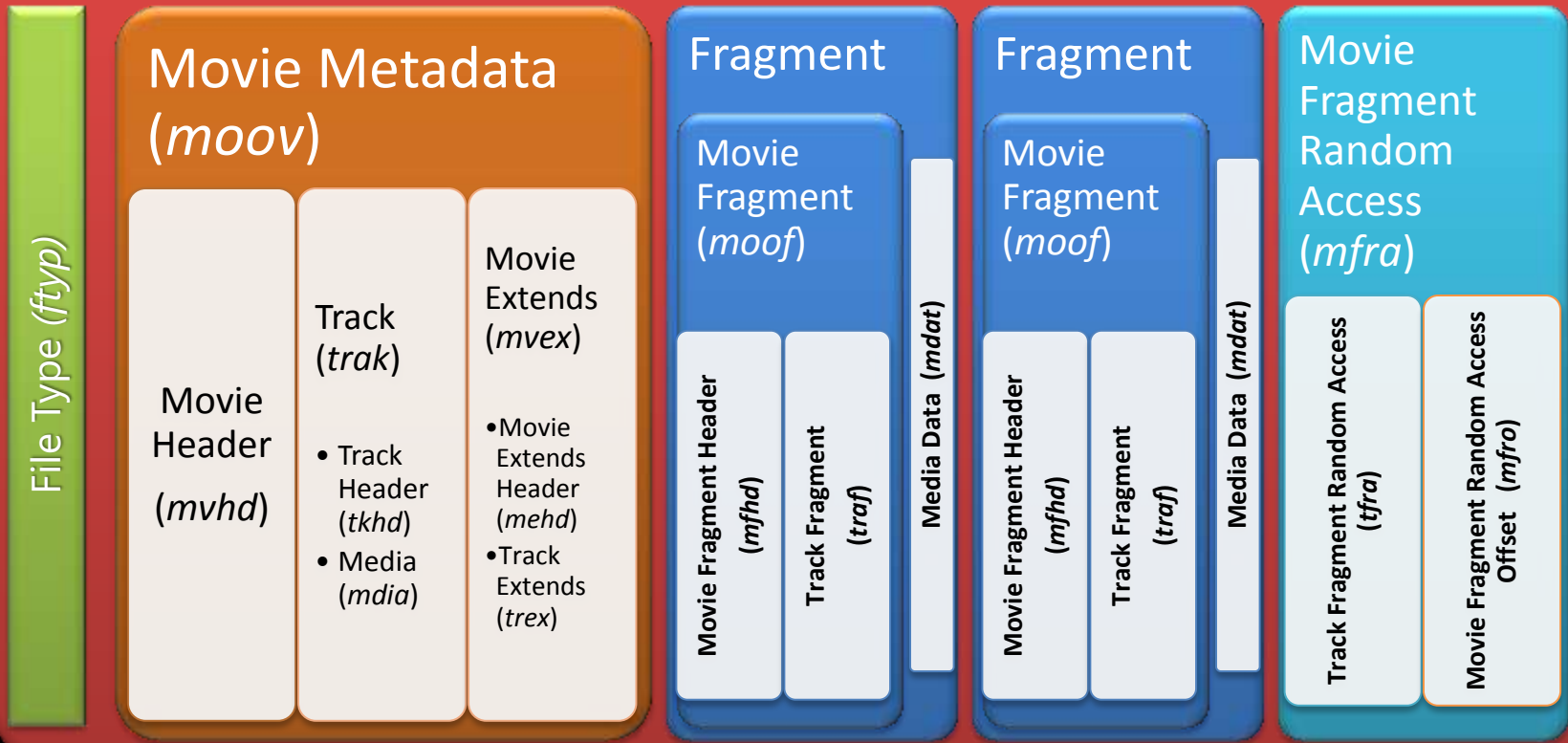


# Smooth Streaming Design

- Smooth Streaming File Format based on MP4 (ISO Base Media File Format)
- Video is encoded and stored on disk as one contiguous MP4 file
  - One file per bit rate (or) one file containing all rates
- Each video Group of Pictures (GOP) is stored in a Movie Fragment box
  - This allows accurate fragmentation at key frames
- Contiguous files are split into cacheable fragments when responding to a client request



# Smooth Streaming MP4 Disk Format





# On-Demand Presentations

- MP4 files containing video/audio fragments
  - New file extensions instead of \*.mp4
  - \*.ismv for files containing video & audio tracks, or video-only
  - \*.isma for files containing only audio tracks
- On-demand server manifest file
  - File extension: \*.ism
  - Defines relationships between media tracks, bit rates, and files
  - Uses SMIL 2.0 XML format
- Client manifest file
  - File extension: \*.ismc
  - Defines streams, codecs, bit rates, resolutions, markers
  - Uses XML format



# On-Demand Server Manifest

```
1  <?xml version="1.0" encoding="utf-16"?>
2  <!--Created with Expression Encoder version 2.1.1206.0-->
3  <smil xmlns="http://www.w3.org/2001/SMIL20/Language">
4  <head>
5    <meta
6      name="clientManifestRelativePath"
7      content="NBA.ismc" />
8  </head>
9  <body>
10 <switch>
11 <video
12   src="NBA_3000000.ismv"
13   systemBitrate="3000000">
14   <param
15     name="trackID"
16     value="2"
17     valuetype="data" />
18 </video>
19 <video
20   src="NBA_2400000.ismv"
21   systemBitrate="2400000">
22   <param
23     name="trackID"
24     value="2"
25     valuetype="data" />
26 </video>
51 <video
52   src="NBA_500000.ismv"
53   systemBitrate="500000">
54   <param
55     name="trackID"
56     value="2"
57     valuetype="data" />
58 </video>
59 <audio
60   src="NBA_3000000.ismv"
61   systemBitrate="64000">
62   <param
63     name="trackID"
64     value="1"
65     valuetype="data" />
66 </audio>
67 </switch>
68 </body>
69 </smil>
```



# Smooth Streaming Client Manifest

```
1 <?xml version="1.0" encoding="utf-16"?>
2 <!--Created with Expression Encoder version 2.1.1206.0-->
3 <SmoothStreamingMedia
4   MajorVersion="1"
5   MinorVersion="0"
6   Duration="4084405506">
7   <StreamIndex
8     Type="video"
9     Subtype="WVC1"
10    Chunks="208"
11    Url="QualityLevels({bitrate})/Fragments(video={start time})">
12    <QualityLevel
13      Bitrate="3000000"
14      FourCC="WVC1"
15      Width="1280"
16      Height="720"
17      CodecPrivateData="250000010FD3FE27F1678A27F859E80C9082DB8D44A9C00000010E5A67F840" />
18    <QualityLevel
19      Bitrate="2400000"
20      FourCC="WVC1"
21      Width="1056"
22      Height="592"
23      CodecPrivateData="250000010FD3FE20F1278A20F849E80C9082493DEDDCC00000010E5A67F840" />
24    <QualityLevel
25      Bitrate="1800000"
26      FourCC="WVC1"
27      Width="848"
28      Height="480"
29      CodecPrivateData="250000010FCBF81A70EF8A1A783BE80C908236EE5265400000010E5A67F840" />
```



# Client-Side Implementation

- Client first downloads the client manifest
- Client requests fragments in the form of RESTful URLs:
  - [http://video.foo.com/NBA.ism/QualityLevels\(400000\)/Fragments\(video=610275114\)](http://video.foo.com/NBA.ism/QualityLevels(400000)/Fragments(video=610275114))
  - [http://video.foo.com/NBA.ism/QualityLevels\(64000\)/Fragments\(audio=631931065\)](http://video.foo.com/NBA.ism/QualityLevels(64000)/Fragments(audio=631931065))
- IIS looks up the quality level and time offset in the server manifest (\*.ism) and determines the physical location of the requested fragment
- Because the wire format (fragment) is just a subset of the disk format (MP4), the extraction process is simple
  - No re-muxing necessary
- Dynamic stream switching logic is fully implemented in Silverlight application code – no server-side detection

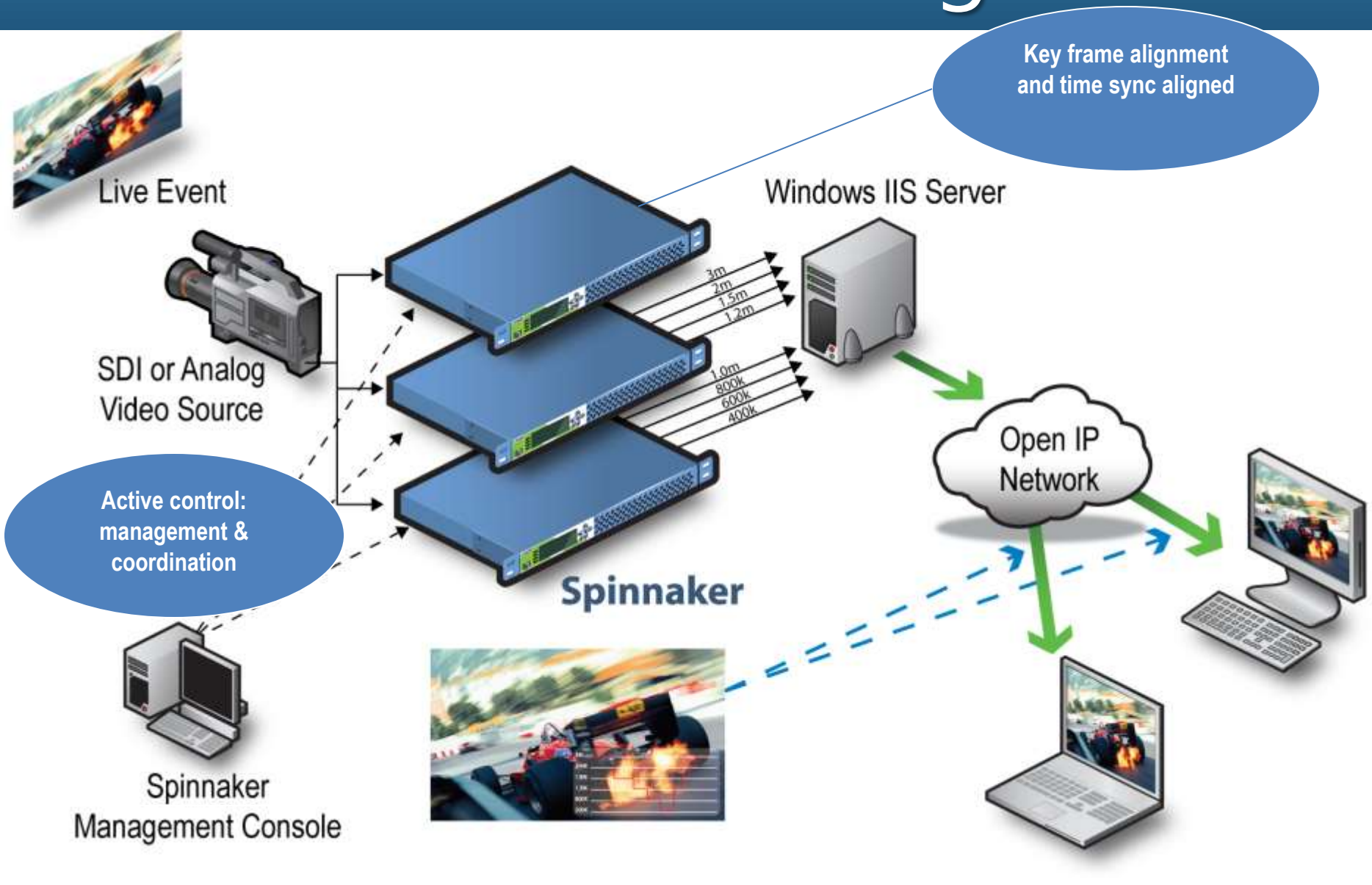


# Live Smooth Streaming

- Live publishing point identified by \*.ismv extension
- Very similar to on-demand from client perspective
- Server continually appends MP4 file on disk with new fragments from encoder
  - Allows DVR-like seeking
    - Back to start of event
    - Forward to the current live point



# Live Smooth Streaming



# Live SS Workflow - Server

- Create Live SS publishing point

The screenshot displays the Internet Information Services (IIS) Manager interface. The main window is titled 'Live Smooth Streaming Publishing Points' and contains a table with columns: File Name, Source, Archive, Fragment, and Stream. A dialog box titled 'Add Publishing Point' is open in the foreground, containing the following fields and options:

- File name: Demo
- Title: My Live Smooth Streaming Demo
- Estimated duration: 00:00:00
- Live source type: Push Encoder (selected in a dropdown menu)
- Archive media:
- Publish media fragments:
- Publish media streams:

Buttons for 'OK' and 'Cancel' are visible at the bottom of the dialog box. The background window shows the 'Connections' tree on the left with 'ZAMBELLI-X301 (REDMOND)' expanded to 'Sites' > 'Default Web Site'. The 'Actions' pane on the right shows 'Publishing Point' > 'Add...' and 'Help' > 'Online Help'.



# Live SS Workflow - Encoder

- Configure one or multiple encoders to push to publishing point
- Why multiple encoders?
  - Most professional VC-1/H.264 video encoders can't output more than 2 HD video streams per unit
  - Encoding video up to 720p HD at 6-8 bitrates would therefore require 2-3 encoder units
  - Video, audio and data can be encoded on separate units





# Live SS Workflow - Encoder

- Encoder pushes fragmented MP4 stream to server in body of a long-running HTTP POST request
  - [http://video.foo.com/Demo.isml/Streams\(720p\)](http://video.foo.com/Demo.isml/Streams(720p))
- Encoder inserts Live Server Manifest Box into start of MP4 stream
  - Contains a SMIL-formatted Live Server Manifest very similar to On-Demand Manifest
  - Describes all tracks in that encoder's stream



# Live SS Workflow - Encoder

- Encoder can also insert Stream Manifest Box into the MP4 stream
  - Identifies all streams from all encoders
  - If present, server will wait for all encoders
- Each sent fragment contains a box with absolute time and duration
  - Used to keep fragments from multiple streams in sync



# Live SS Workflow - Server

- Server parses the encoder manifests and starts collecting MP4 fragments
- Server builds a cumulative runtime index in memory for all incoming fragments
- Server archives MP4 fragments into local Smooth Streaming MP4 files
- When the broadcast is done, server generates the 'mfra' index box



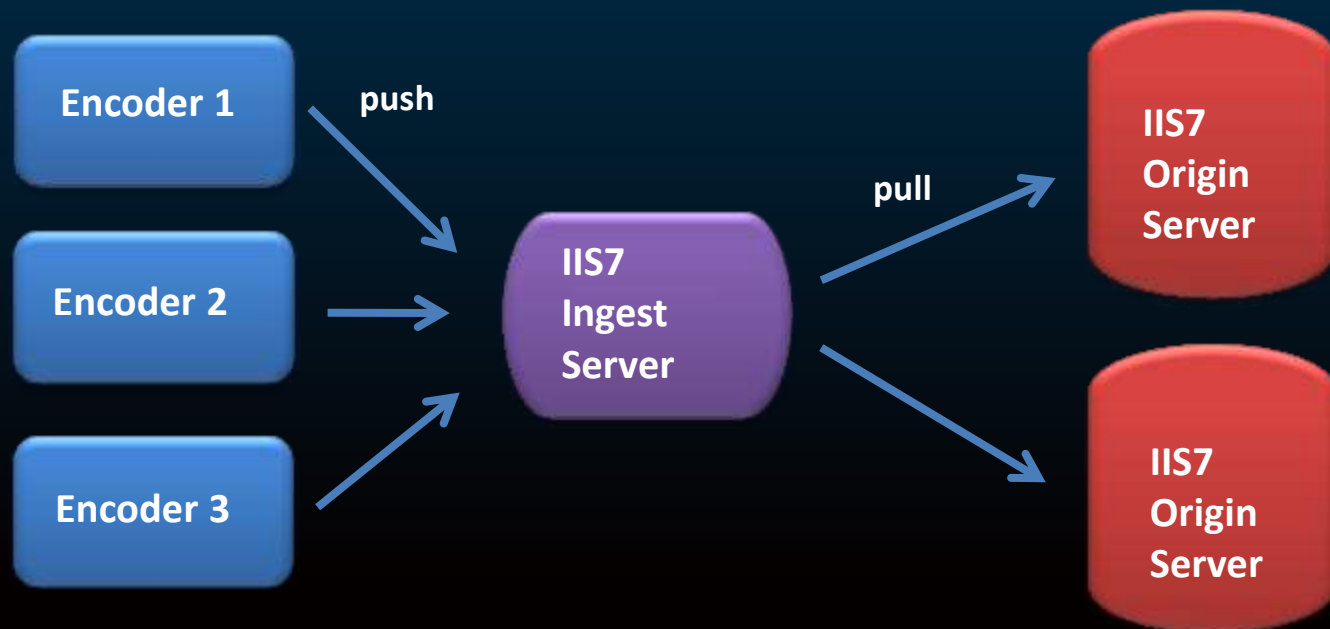
# Live SS Workflow - Client

- Client requests a manifest
  - <http://video.foo.com/Demo.isml/Manifest>
- Server adds up all the encoder manifests and the runtime fragment index and sends the current version to the client
- Client builds its own cumulative runtime index based on any additional fragments sent by the server



# Ingest Servers

- Live Smooth Streaming differentiates between Ingest and Origin servers



# Failover and Redundancy

- Several points of failover
  - Encoder
    - Encoder can push to multiple ingest servers
    - If an encoder fails, server will continue ingesting streams from other encoders
  - Ingest Server
    - Ingest server can also act as push proxy
    - Replicates data as it's ingested
  - Origin Server
    - If ingest server fails, can switch to backup ingest



# Failover and Redundancy

- Client resilient to network failures
  - If requested fragment is not available, client will retry several times before moving on to another bit rate fragment
- Spreading multi-bitrate encoding across multiple encoders creates redundancy
  - Worst case scenario: If an encoder fails and no backup is available, client will keep trying until it finds an available bit rate



# Live Smooth Streaming

- IIS Media Services 3.0 also features:
  - Temporary DVR archive
    - Delete DVR archive after live broadcast is done
  - Sliding window DVR archive
    - Only archive the most recent NN minutes
    - Ideal for 24/7 live broadcasts
  - Archive segmentation
    - Create a new archive file for every NN minutes





Q&A



# Getting Started

- Windows Media Services 2008
  - Free Download
  - <http://www.microsoft.com/windowsmedia/forpros/serve/prodinfo2008.aspx>
- IIS Media Services
  - Free Download
  - <http://www.iis.net/media>



# Participate

- Community

- Windows Media -

- <http://www.microsoft.com/windowsmedia/community.aspx>

- IIS - <http://forums.iis.net/1145.aspx>

- Blogs

- <http://blogs.iis.net/media>



# **Microsoft<sup>®</sup>**

*Your potential. Our passion.<sup>™</sup>*

© 2009 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.

MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.