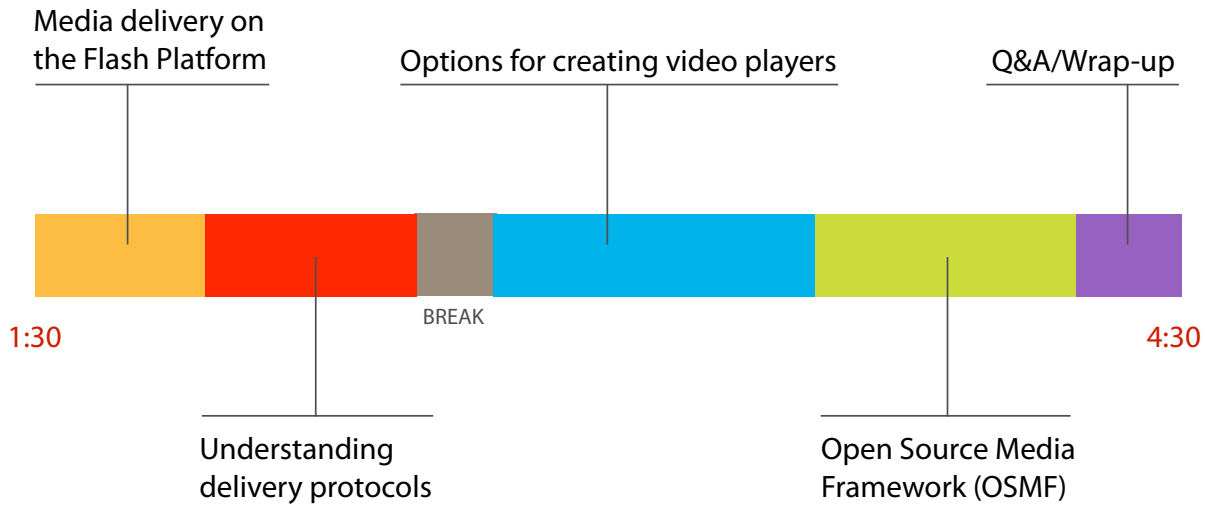
Flexible Protocols and Custom Player Development

Lisa Larson-Kelley | FlashConnections.com | Flash Platform Consultant

Flexible Protocols and Custom Player Development



First… why we are here!

- Out-of-the-video-box experience
  http://www.vimeo.com/9194146
- Streamlined playback application
  http://www.adobe.tv
- Expressive, immersive and interactive
  http://www.youtube.com/wariolandshakeit2008
- Customizable and dynamic
  http://beck.cnnbcvideo.com/

Flash Platform/Flash Media Server Family

Create/Capture	Stream/Protect	Receive
Fl Fx		
Flash CS4 Professional	Flash Media Interactive Server 3.5	Flash Player
Adobe Flash Builder	Flash Media Streaming Server 3.5	Adobe AIR
Flex framework	Flash Access 2.0	Adobe Flash Lite Player
Open Source Media Framework		
Adobe Flash Media Live Encoder 3		
Adobe Media Encoder CS4		

Flash Player 10.1

Receive

- Dynamic Streaming
- Hardware acceleration
- Speex audio codec
- RTMFP
- Full-screen with keyboard support
- Video4Linux webcam API
- 3D effects
- PixelBender
- Text Layout Framework
- Dynamic sound generation
- Expanded file APIs
- Color correction
- Flash Access support (desktop only)

UPDATE: Flash Player "Gala"

Hardware acceleration on Mac OSX 10.6.3 or later (HD H.264 only)

- MacBooks shipped after January 21st, 2009
- Mac Minis shipped after March 3rd, 2009
- MacBook Pros shipped after October 14th, 2008
- iMacs that shipped after the first quarter of 2009

Flash Player 9 and later

Receive

- High-quality codecs
- 8-bit video channel
- Encrypted RTMP
- Full-screen playback
- SWF search

Adobe AIR 2.0

Receive

- **Desktop applications**
- HTML/Javascript (AJAX), SWF content
- Cross-platform
- Repurpose existing content for online/offline delivery
- Play downloaded content protected with Flash Access (desktop only)

Flash Lite 3

Receive

- Over 800 million devices shipped
- 400+ device models enabled
- The number of Flash Lite shipped devices reached 1 billion in 2009 and more than 2.5 billion by the end of 2010*

*According to Strategy Analytics

Flash Player 10.1

Receive

- Coming soon
  to a phone near you*

**except one…*

Adobe

Flash Player 10.1

Receive

- Flash Player 10.1 allows your content to reach your customers wherever they are:
  - Desktops
  - Smartphones
  - Netbooks
  - Other Internet-connected devices
- Consistent and broadly adopted runtime
- Reuse code while adapting to individual device capabilities
  - GPU acceleration for video decoding and animation
  - Multi-touch gesture support
  - Accelerometer support
- Robust content protection powered by Flash Access 2.0
- HTTP Dynamic Streaming support

Adobe

Open Screen Project

Receive

- Built on the Flash Platform
  - Widest reach across operating systems and devices
  - A community of more than one million developers
  - Powerful, rich authoring tools
- Consistent runtime for standalone applications and web browsing
  - Optimized for high performance on mobile screens
  - Leverages native device capabilities (contextual applications)
  - Availability expected in the first half of 2010
- Support for major device platforms:
  - Android
  - BlackBerry® platform
  - Symbian® OS
  - Palm® webOS
  - Windows Mobile®
- Close to 70 ecosystem partners
- New partners include:

open screen project

Singular experience, multiple devices

SYMBIAN · Disney Interactive Media Group · PBS · NBC UNIVERSAL · nick

Flash Platform Authoring Tools

Create/Capture

Fl Flash Professional CS5

FC Flash Catalyst CS5

FB Flash Builder 4

CS5

Flash Professional CS5

Create/Capture

- **Create SWFs**
- Authoring environment for art, animation, and ActionScript code
- Code on timeline or import custom classes
- Pre-built components for user interface and video

Flash Builder 4 *(previously Flex Builder)*

Create/Capture

- **Create SWFs**
- **Leverage the Flex framework**
- Powerful coding tools (AS and MXML)
- Rich visual layout
- Interactive data visualization
- Skinning and styling
- Code refactoring
- Powerful testing tools
- Advanced data services

Open Source Media Framework (OSMF)

Create/Capture

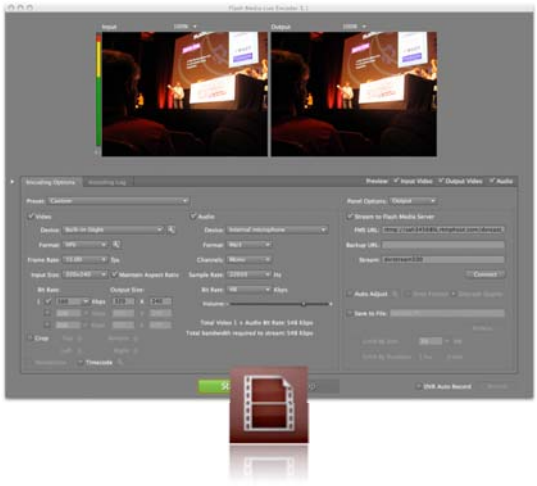
- Simplifies the development of media players
- Pluggable component architecture
- High quality, rich playback experiences
- Solves common problems
- API integration
- Quality of Service
- Reporting and analytics
- Lowers development costs, facilitates faster turnaround
- Open framework facilitates collaborative development
- Benefits publishers, Adobe tool users, and ecosystem partners
- FREE

open source media framework

Flash Media Live Encoder 3

Create/Capture

- Free from Adobe.com www.adobe.com/go/fmle
- Broadcast-level capture
- Any input source
- High quality Encoding
- Video: H264 | VP6 (up to HD)
- Audio: AAC | MP3
- Auto-adjust quality, buffer management
- Command-line control
- Image pre-processing
- Multiple destinations
- Metadata
- Local Archive
- VITC Timecode support
- Multiple bitrates
- DVR Support



Flash Media Server 3.5

Stream/Protect

- **Flash Media Streaming Server**
  - Efficient and affordable streaming
  - $995 full / $249 upgrade
- **Flash Media Interactive Server**
  - Interactive, multi-way communication applications with real-time video, audio and data sharing
  - $4,500 full / $349 upgrade
- **Flash Media Development Server**
  - All of the features of FMIS
  - Limited to 10 simultaneous connections
  - Free

Flash Access 2.0

Stream/Protect

- A content protection and monetization solution
- For streaming and download
- Over any protocol (i.e. progressive download, RTMP streaming, HTTP Dynamic Streaming, or file download)
- Using flexible usage rules (e.g. time-based, output protection)
- Support for a variety of business models (e.g. rental, subscription, electronic sell-through)
- Cross-platform: Windows, Mac, Linux
- Playback in Flash Player 10.1 and Adobe AIR 2.0 (desktop only in current version)
- Approved by studios as part of DECE (Digital Entertainment Content Ecosystem)

(formerly Flash Media Rights Management Server)

Useful Video Tools

- Adobe FMS Tools http://tinyurl.com/fmstools
  - FLVCheck
  - FMSCheck
  - F4V Post Processor
  - Dynamic Streaming Class
  - DVRCast application
  - FLVPlayback 2.5 for Flash CS4 and Flex
- Robert Reinhardt's tools
  - Video Validator for Flash http://labs.influxis.com/?p=54
  - Bitrate Calculator http://tinyurl.com/bitratecalculator
  - Bitrate Starter http://www.flashsupport.com/resources/

A word (or two) about HTML5.

Beyond the FUD

There is a time and a place for every technology.

Beyond the FUD

Advantages of HTML5

- Easy to set up with a couple of tags
- Runs on devices
- JQuery makes it even easier
- No compiling needed; can be written in Notepad
- Default player controls built in

Beyond the FUD

Disadvantages of HTML5

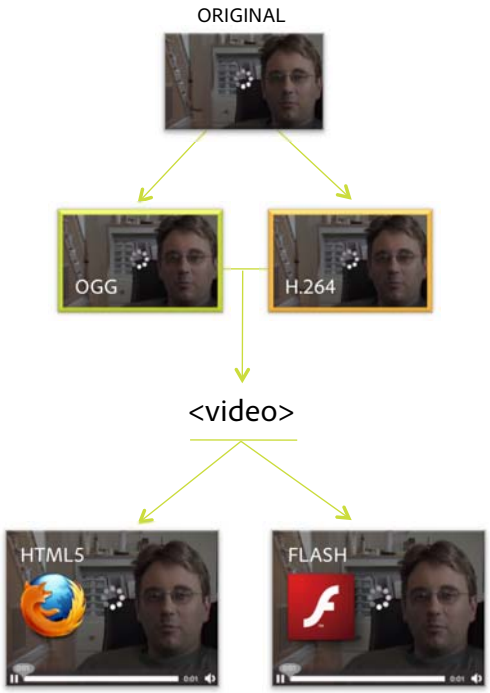
- Not supported in all browsers
  - Often fallback to Flash anyway
  - Must support lowest common denominator
  - Enable MIME types on server
- Complex development for full-featured player interactions
- Doesn't stream
- No live support
- Not protected
- Not multibitrate
- Lacking in measurement, advertising, QoS

HTML5 Video Playback

Here's what your video workflow looks like:

1. Make one version that uses Theora video and Vorbis audio in an Ogg container.
2. Make another version that uses H.264 baseline video and AAC "low complexity" audio in an MP4 container.
3. Link to both video files from a single <video> element.
4. If you detect a lack of HTML5 video support, replace the <video> element with a Flash-based video player.

* Source: http://diveintohtml5.org/



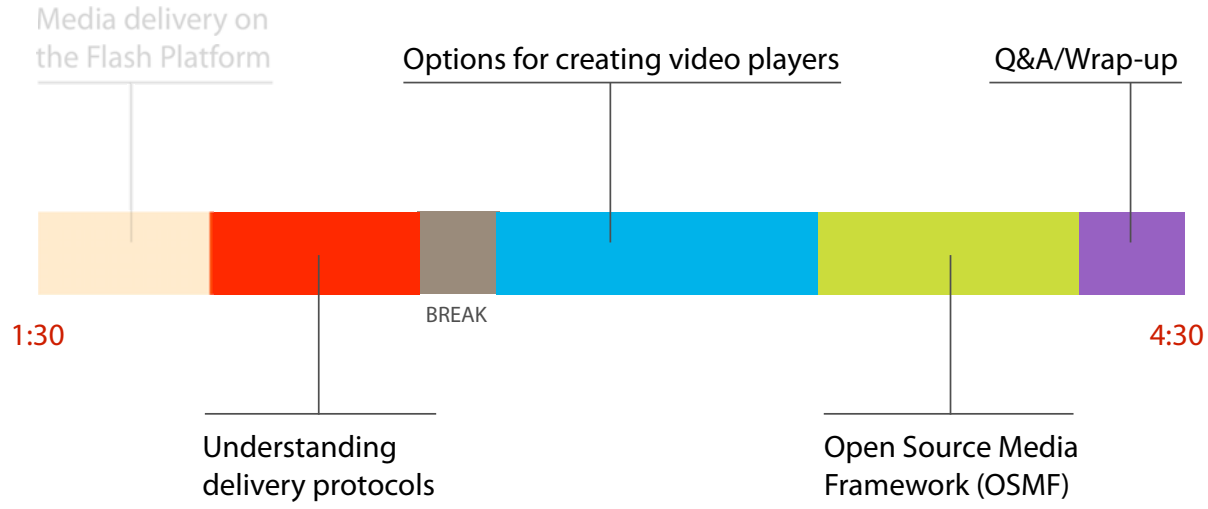
Questions?

NEXT: Protocols

2: Understanding Flash Media Delivery Protocols

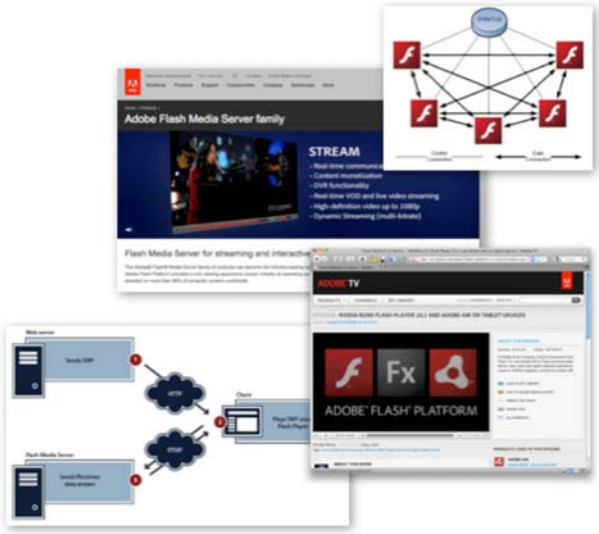
Lisa Larson-Kelley | FlashConnections.com | Flash Platform Consultant

Flexible Protocols and Custom Player Development



Flash Media Delivery

1. Progressive delivery
2. Streaming via RTMP
3. HTTP Dynamic Streaming
4. Peer-to-Peer via RTMFP



Progressive delivery

- Standard web server
- Easy deployment
- Downloaded (cached) on local hard drive
- Seek only to downloaded keyframes*
- YouTube, Facebook, etc.

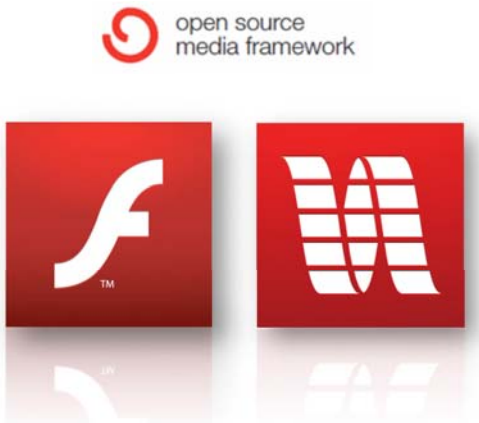
*without additional server-side code to access video file data and repackage

Streaming delivery (RTMP)

- Flash Media Server required
- Content never downloaded
- Various file protection options
- Seek to any frame
- Transfers only content watched
- Very low latency
- Live streams
- Integrated real-time, multi-way communication
- Deep, customizable logging
- Server-side playlists
- Stream recording
- Dynamic Streaming
- DVR functionality

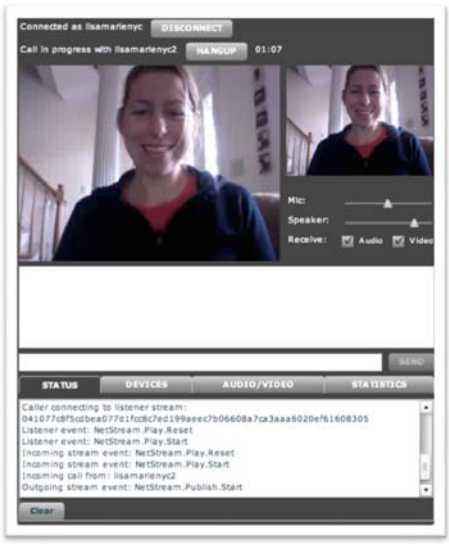
HTTP Dynamic Streaming *(Project Zeri)*

- New delivery method, requires Flash Player 10.1 or later
- Leverages standard HTTP caching infrastructures
- Open format, standards-based technology
- Protected HTTP Dynamic Streaming powered by Flash Access 2.0 enables file encryption and SWF verification
- Supports all Flash codecs (H.264, VP6, H.263, HE-AAC, MP3) and metadata
- Adaptive bitrate switching
- Live and on demand
- DVR support
- Dynamic ad insertion
- OSMF implementation

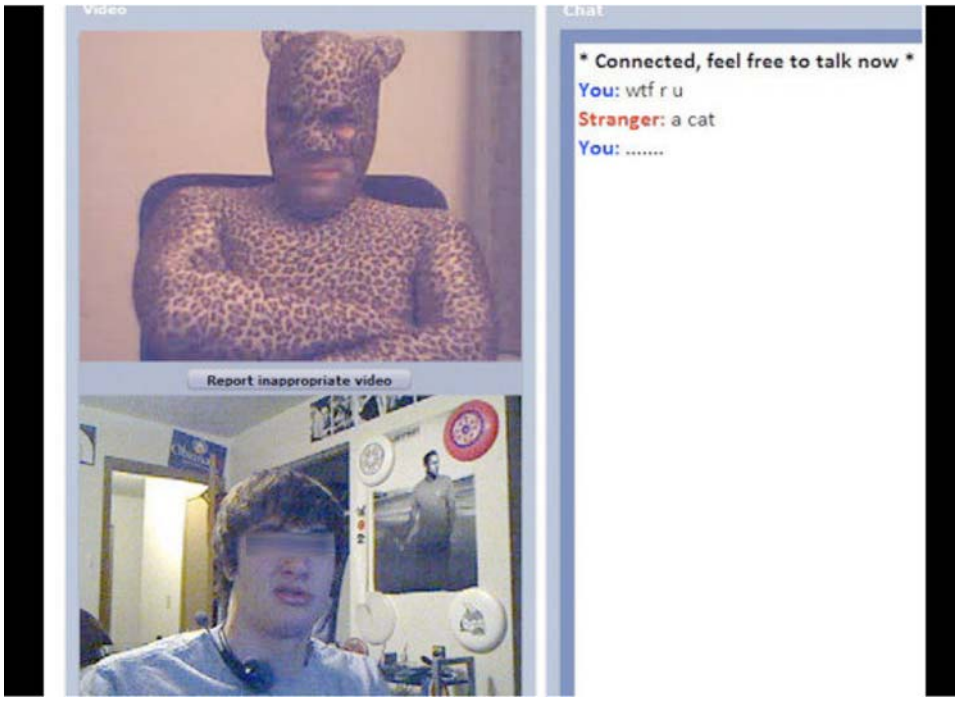


RTMFP (Peer-to-Peer)

- New delivery method, requires Flash Player 10.1, AIR 2.0 or later
- Very low latency
- Encrypted connection
- Will support:
  - Point-to-point
  - Swarming (large file download)
  - Distributed data storage
  - Live application-level multicast (Native IP Multicast/App level Multicast = "Fusion")
- Stratus service is the only way to use RTMFP at this time

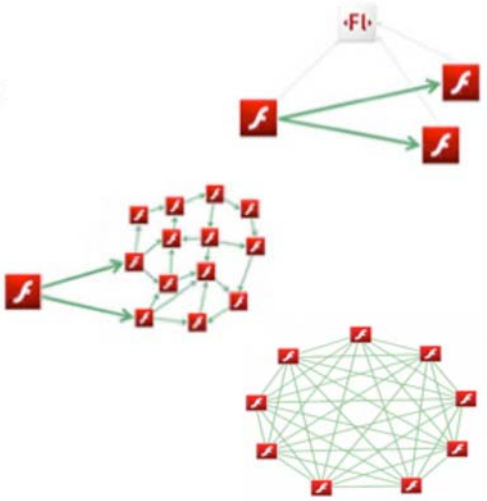


RTMFP (Peer-to-Peer)



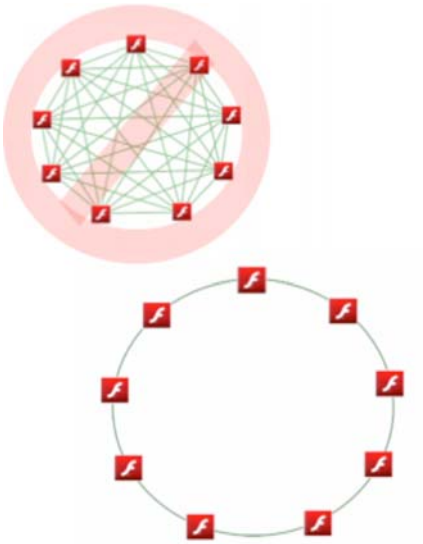
RTMFP (Peer-to-Peer)

- **1 to 1 (or 1 to few)**
  - Communicate via Server
    - RTMP (RTMPT, RTMPS, RTMPE…)
    - RTMFP
- **Communicate Directly**
  - RTMFP
- **1 to Many +++**
  - Communicate via Server
  - Communicate via **RTMFP Groups**



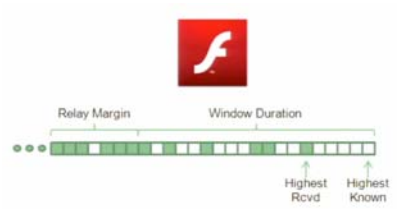
RTMFP Groups

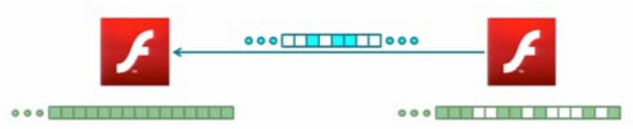
- Player needs to connect to millions of peers
- Solution: Overlay network
  - Not a full mesh
  - Optimized for multiple uses
  - RTMFP used as transport
    - NAT/Firewall traversal
    - Encryption
    - IP address mobility
    - Congestion Control
    - Partial reliability, etc.
  - Topology is distributed to peers rather than on a central server
  - Basically, can work like BitTorrent
- New API for developing apps that use RTMFP
- Dialog box will be displayed to client for Groups



RTMFP Multicast "Fusion"

- Uses NetStream API
- Designed for very low latency
- Works without ActionScript involvement
- Quality of Service parameters available
- A push-pull system
  - Starts out with pull (neighbor pulls data blocks from neighbor) <-Latency
  - Push reduces latency (send every 4th and 8th block to me without asking)





More RTMFP Information

More details:

- http://tv.adobe.com/watch/max-2009-develop/p2p-on-the-flash-platform-with-rtmfp
- http://labs.adobe.com/wiki/index.php/Stratus

Matthew Kaufman,
Adobe Sr. Computer Scientist,
Project Lead,
resident genius.

Content Protection for Flash Media

PROTECTION	PROGRESSIVE	STREAMING	HTTP DYNAMIC STREAMING
RTMP ▪ No local cache		●	
RTMPS ▪ SSL stream encryption ▪ Requires SSL certificate		●	
RTMPE ▪ Stream encryption ▪ No certificate required ▪ Built-in and turned on by default in FMS ▪ Use with SWF Verification		●	
Flash Access 2.0 ▪ File-based encryption ▪ Integrates with a policy server ▪ Downloadable or streaming protection ▪ Playlist protection	●	●	●

Additional Content Protection: Streaming only

PROTECTION	DESCRIPTION
SWF Verification	▪ Compare bytecode to approved SWF files
Access Plug-in	▪ Trap access requests and validate (database, xml list, etc.)
Authorization Plug-in	▪ Authenticate after connection is made ▪ Authorize playing or publishing of a stream ▪ Call server-side methods ▪ Limit time and duration of access ▪ Deliver specific content to specific authorization levels, etc.
Domain Access Control	▪ Whitelist (or blacklist) of domains/IPs (Adaptor.xml or vHost.xml)
User authentication with Server-side ActionScript	▪ User credentials (login and password) NetConnection.connect("rtmp...", "username", "password"); ▪ Encrypted token (MD5 Hash) `NetConnection.connect("rtmp...", 6aef79f07bc8f23c38e8979f3630f436);` ▪ Unique key NetConnection.connect("rtmp...", 349jh3k4324h9.234234098); ▪ Integration with SOAP, Flash Remoting, XML, HTTP Post (loadVars), or simple file access to validate the client

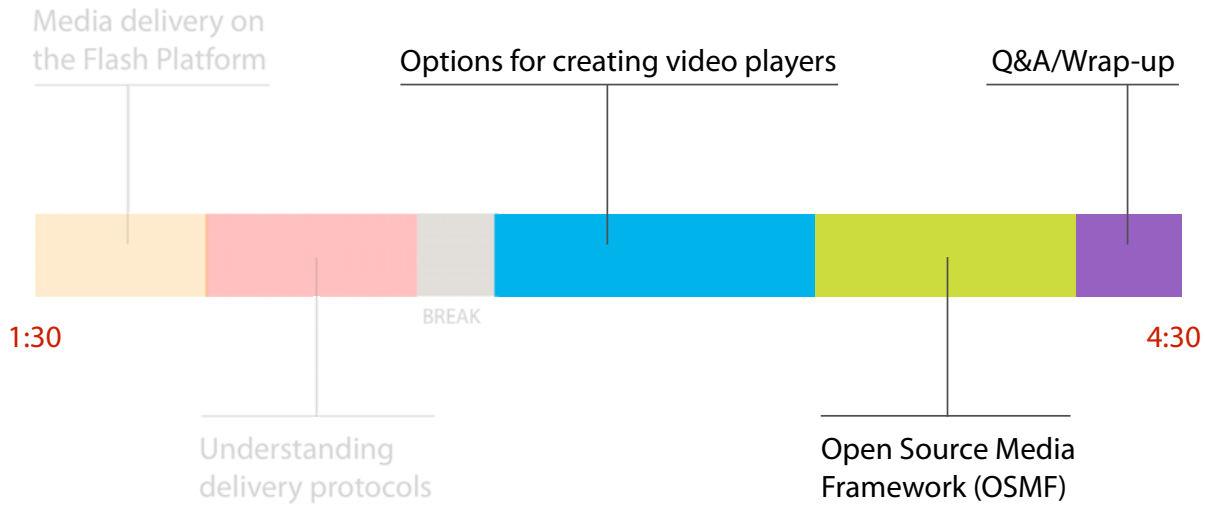
Questions?

NEXT: Options for Creating Video Players

3: Options for Creating Video Players

Lisa Larson-Kelley | FlashConnections.com | Flash Platform Consultant

Flexible Protocols and Custom Player Development



The Basic Media Player

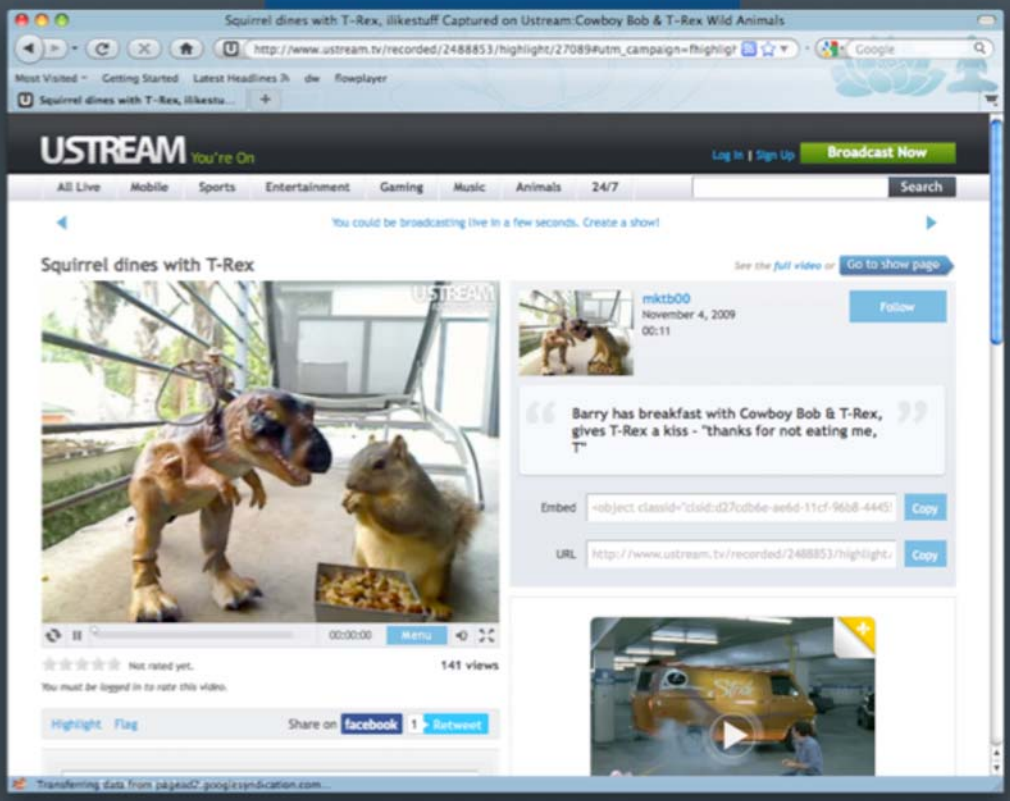
- Proprietary
- Limited functionality
- Little or no customization

Media Players Today

- Basic playback controls
- Full-screen mode
- Automatic bitrate shifting
- Rating systems
- Video sharing:
  Embed / Deep link/ Email
- Feed subscription / syndication
- Tracking and logging
- Monetization
- Advertising
- Search and discovery
- Mashup/create media
- Advanced: DVR, PiP, etc.

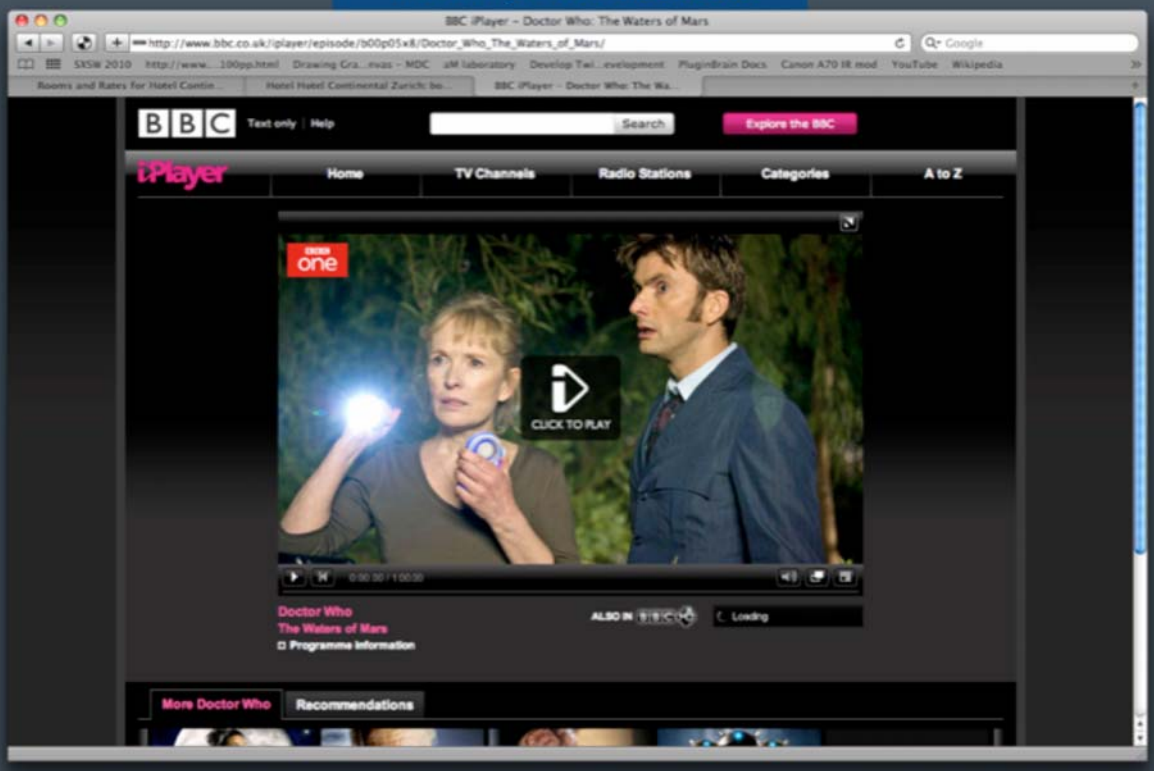
HULU www.hulu.com

uStream www.ustream.tv

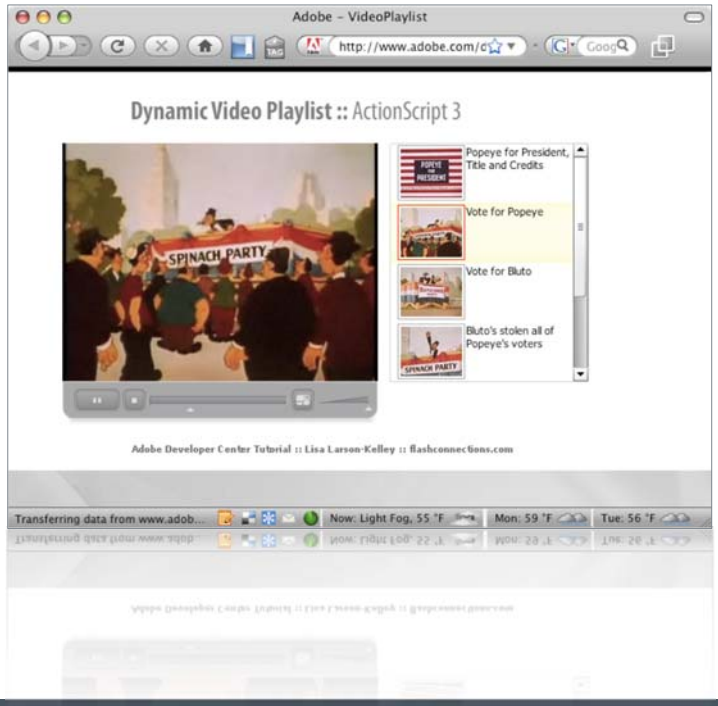


MLB www.mlb.tv

BBC iPlayer www.bbc.co.uk

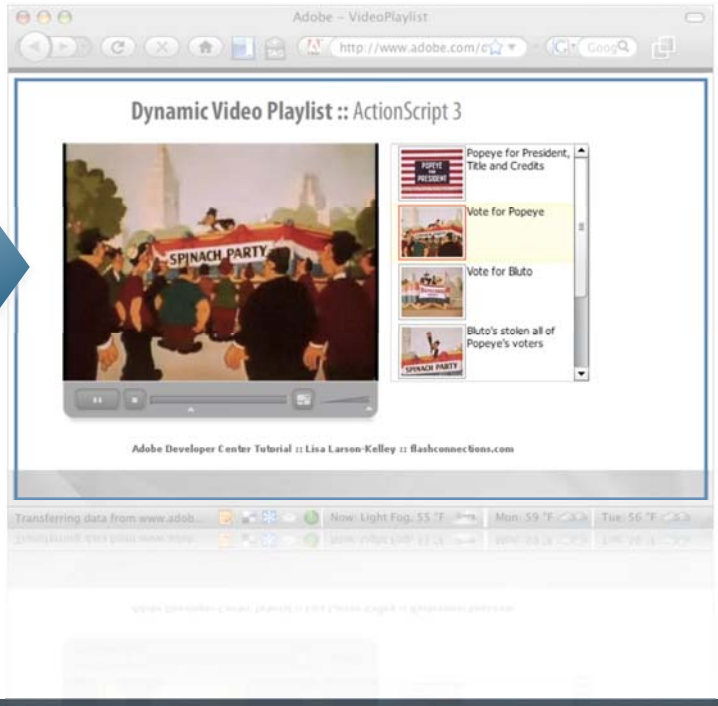


Typical Video Player with Dynamic Playlist

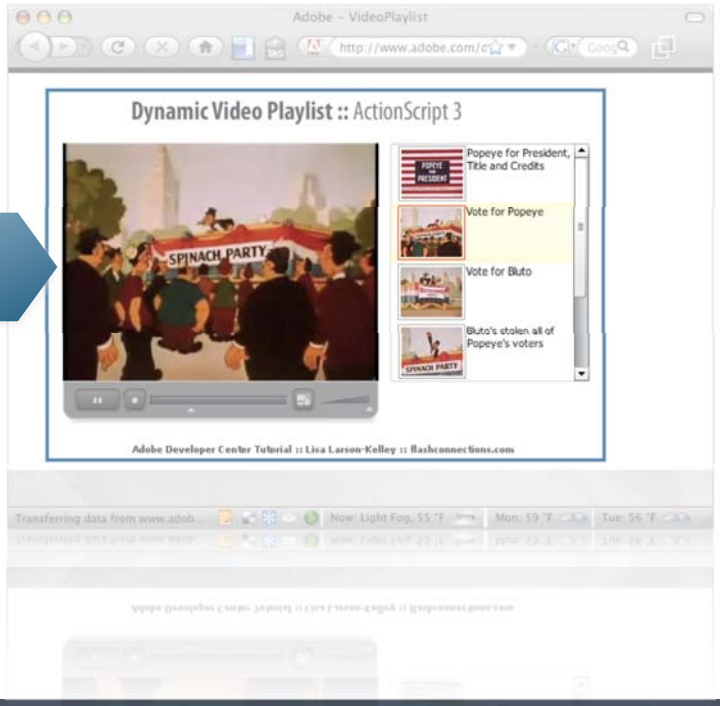


Typical Video Player with Dynamic Playlist

HTML w/embed script

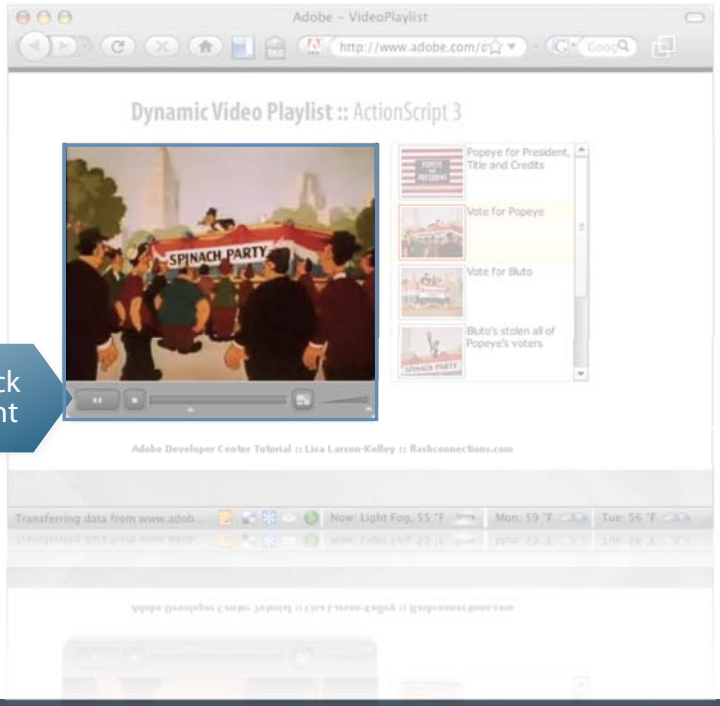


Typical Video Player with Dynamic Playlist



SWF

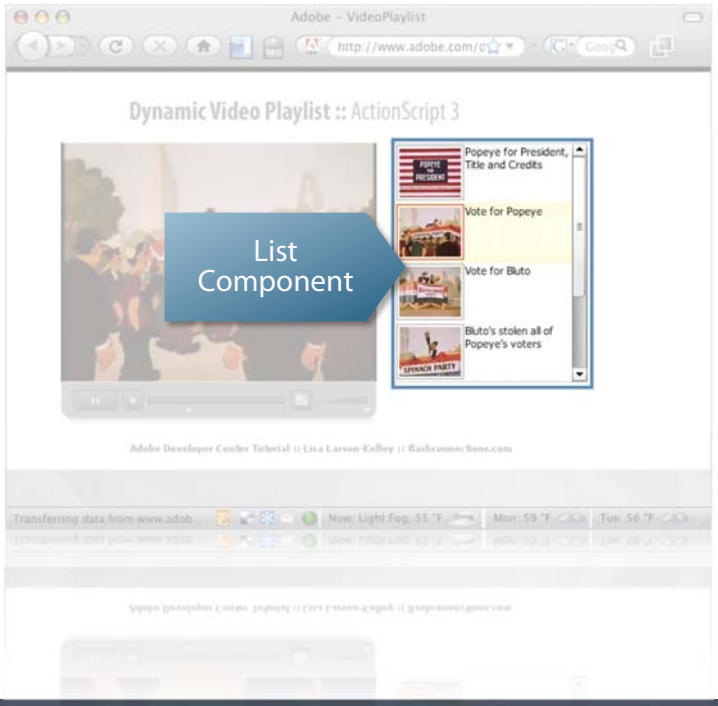
Typical Video Player with Dynamic Playlist



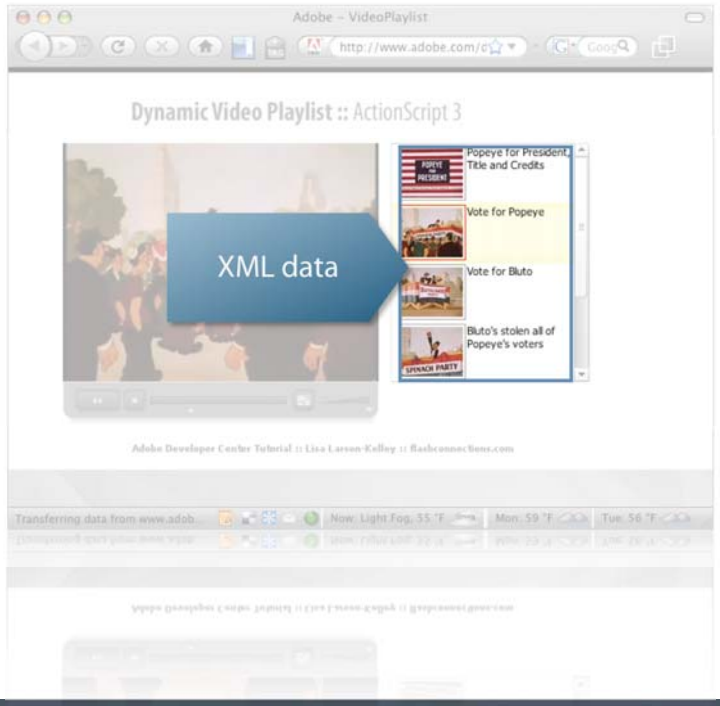
FLVPlayback Component

Typical Video Player with Dynamic Playlist

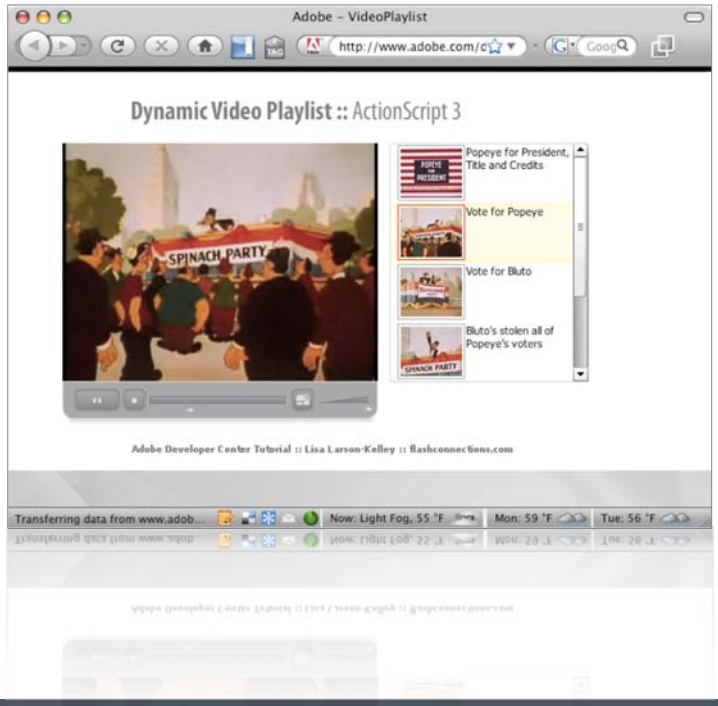
Typical Video Player with Dynamic Playlist



Typical Video Player with Dynamic Playlist



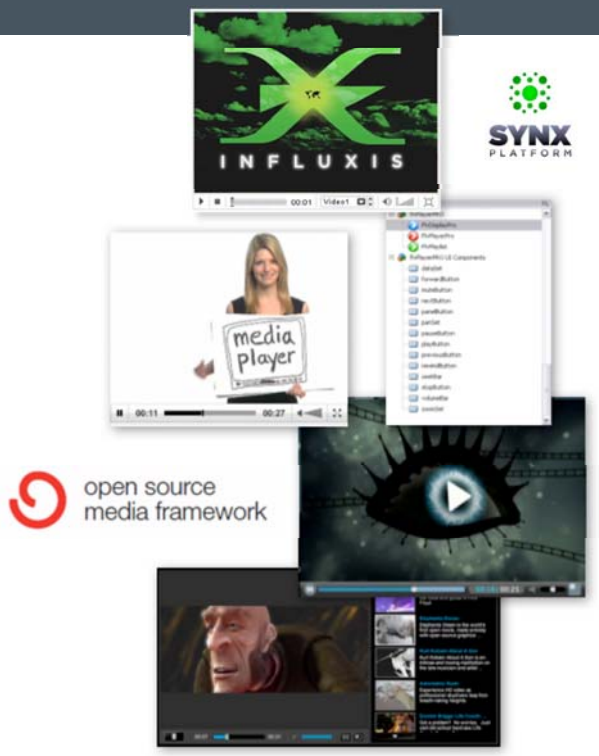
Typical Video Player with Dynamic Playlist



Lots of options... Too many?

- **No-code players**
  - Dreamweaver
  - FLVPlayback component
  - FLVPlayback UI components
  - Third-party
    - FLVPlayer PRO component
    - JW Player™ 5*
    - FlowPlayer*
    - Open Video Player (OVP) *
    - Influxis pre-built applications *
  - Open Source Media Framework*
- **Custom ActionScript players**
  (all of the above, except DW...)

* Open source



Media Player Programming Tasks

- **Basic Delivery (circa 2005)**
  - Connection management (connect/disconnect)
  - Stream management (play/pause/seek/volume)
  - User interface (Buttons/Frame/Fullscreen)
- **Beyond Basics**
  - Error handling
  - Authentication
  - Tracking
  - Cue points
  - Accessibility
- **Advanced**
  - Integrated advertising
  - Remote/live controls
  - Content protections
  - Dynamic playlists
  - DVR functionality
  - Picture in picture
  - Interactive features
  - Hotspot links
  - Data sharing
  - Multiple camera angles
  - Mobile delivery

Basic Connection Management (Progressive)

Step 1: Set up NetConnection

- ActionScript 3 class: **flash.net.NetConnection**
- URI: **netConnection.connect(null);**

Step 2: Set up the Stream controller

- ActionScript 3 class: **flash.net.NetStream**
- Stream name: **"video.flv"**

Basic Connection Management (RTMP Streaming)

Step 1: Connect to Flash Media Server

- ActionScript 3 class: **flash.net.NetConnection**
- URI: **rtmp://[server]/[application]**

Step 2: Wait for connection to be accepted + monitor connection

- ActionScript 3 event: **flash.events.NetStatusEvent**

Step 3: Set up the Stream controller

- ActionScript 3 class: **flash.net.NetStream**
- Stream Name: **"video.flv"**

Flash Media Connection Events (RTMP Streaming)

ActionScript Events let you monitor connection status changes

- **flash.events.NetStatusEvent**
  - "NetConnection.Connect.Closed"
  - "NetConnection.Connect.Failed"
  - "NetConnection.Connect.Success"
  - "NetConnection.Connect.Rejected"
  - "NetConnection.Connect.AppShutdown"
  - "NetConnection.Connect.InvalidApp"

Controlling the Stream

Step 1: Connect the Stream to the NetConnection

- ActionScript 3 class: flash.net.NetStream

Step 2: Set the buffer size

- NetStream.bufferTime

Step 3: Use the NetStream.play() command

- NetStream.play([streamname], [start], [length],[reset playlist])
  - eg: NetStream.play("myVideo",0,-1,true);

Step 4: Attach commands to the user interface

Flash Media Stream Events

ActionScript Events let you monitor stream status changes

- **flash.events.NetStatusEvent**

NetStream.Buffer.Empty;	NetStream.Buffer.Full;	NetStream.Buffer.Flush
NetStream.Play.Start	NetStream.Play.Stop	NetStream.Play.Failed
NetStream.Pause.Notify	NetStream.Unpause.Notify	NetStream.Seek.Notify
NetStream.Seek.Failed	NetStream.Seek.InvalidTime	
NetStream.Play.StreamNotFound	NetStream.Play.InsufficientBW	

- Additional events for H.264 containers
  - NetStream.Play.FileStructureInvalid
  - NetStream.Play.NoSupportedTrackFound

TIP: "Preload" an FLV:

Tell a NetStream to play() the designated external file, then immediately pause it. The browser continues the download. You can retrieve bytesloaded and bytestotal from the NetStream object.Play video, wait for NetStream.Buffer.Full status event then pause. (Metadata also loads in the background).

> - flvs can be 'preloaded': Tell a netstream to play() the designated> external file, then immediately pause it. The browser continues the> download it. You can retrieve bytesloaded and bytestotal from the> netstream object. This one is quite handy - I play the video and then wait for the NetStream.Buffer.Full status event then pause, which ensures it's ready to go as soon

Flash Media URLs

- Streaming URLs are not the same as web or progressive download URLs
- Many publishing systems/CDNs use single URL strings that include
  - Server
  - Stream Name
  - Authentication
  - Other parameters
- URL parsing is done in ActionScript and broken into
  - Connection
  - Stream Name / Source
  - Authentication details

URL Structure for RTMP Streaming

- Basic URL
  - rtmp://172.16.4.128/vod/mystreamname
- H264-based or MP3-based URL
  - rtmp:// 172.16.4.128/vod/mp4:mystreamname
  - rtmp:// 172.16.4.128/vod/mp3:mysound
- FLVPlayback (Source)
  - rtmp:// 172.16.4.128/vod/mp4:mystreamname.f4v
  - http://webserver.com/myPlaylist.smil
- Application instances
  - rtmp:// 172.16.4.128/vod/_definst_/mp4:mystreamname
- Stream folders
  - rtmp:// 172.16.4.128/vod/_definst_/folder/mp4:mystreamname
- Query-based authentication
  - rtmp:// 172.16.4.128/vod/mp4:mystreamname?token=234h234jkh&account=llk

URL Structure for Progressive Delivery

- Basic URL
  - http://webserver.com/videos/myVideoName.flv
- H264-based or MP3-based URL
  - http://webserver.com/videos/mp4:myVideoName
  - http://webserver.com/videos/mp3:mySoundName
- FLVPlayback (Source)
  - http://webserver.com/videos/mp4:mystreamname.f4v
  - http://webserver.com/myPlaylist.smil

Hands on!

Third-party: NO CODE

- LongTail Player (JW Player)
- FlowPlayer

Hands on!

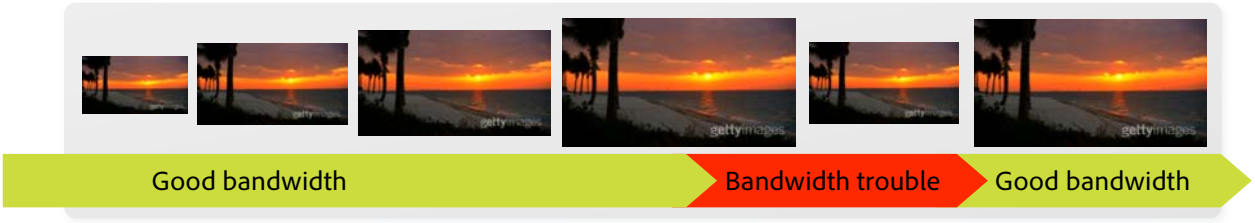
Basic Video Player

- FLVPlayback
- UI Components
- Video Object
- Video Object: RTMP Streaming

Dynamic Streaming

Deliver the best quality experience as network conditions change

- Flash Media Server feature
- Quick start video
- Highest quality for hardware
- Smooth switch to the best bitrate
- No disruption in audio
- H.264 and VP6 support
- Developer can control switching
- Advanced metrics available for QoS monitoring



Dynamic Streaming Best Practices

- Wide range of bitrates, but not too close in value
- Can switch between clips of different codecs, video bitrates, frame sizes
- Keep audio codec and bitrate the same across all clips (mono if possible)
- Constant keyframe interval may yield smoother transitions (5 seconds rec.)
- Recommended client-side buffer 6-10 sec.

VIDEO SIZE TYPES	VIDEO SIZE	4:3 ASPECT SIZE	16:9 ASPECT SIZE	TOTAL BIT RATE (KBPS)	VIDEO BIT RATE (KBPS)	AUDIO BIT RATE (KBPS)	% US BROADBAND CONSUMERS*
QCIF	176×144	144×108 192×144	192×108 256×144	**48** **96**	32 80	16 (mono) 16 (mono)	Modem and ISDN 2%
CIF	352×288	288×216 320×240	384×216 384×216	**300** **500**	268 372	32 (stereo) 128 (stereo)	Low-end DSL 4%
D1	720×486	640×480 640×480	852×480 852×480	**800** **1200**	672 1072	128 (stereo) 128 (stereo)	Faster DSL 25%
HD	1280×720	– –	1280×720 1280×720	**1800** **2400**	1672 2272	128 (stereo) 128 (stereo)	Cable modems 69%

SOURCE: *Dynamic streaming on demand with Flash Media Server 3.5,* Adobe Flash Media Developer Center.

- Details at http://www.adobe.com/devnet/flashmediaserver/articles/dynstream_on_demand.html

Dynamic Streaming ActionScript 3.0 (FP 10)

- **DynamicStream.as** and **DynamicStreamItem.as**
  - Flash CS4 Professional
  - Adobe Flex® 3
  - Flex SDK
- **FLVPlayback Component 2.5**
  - Flash CS4 Professional
  - Adobe Flex® 3
- **Open Source Media Framework 1.0**
  - Dynamic Streaming built into the framework
- **http://www.adobe.com/products/flashmediaserver/tool_downloads/**

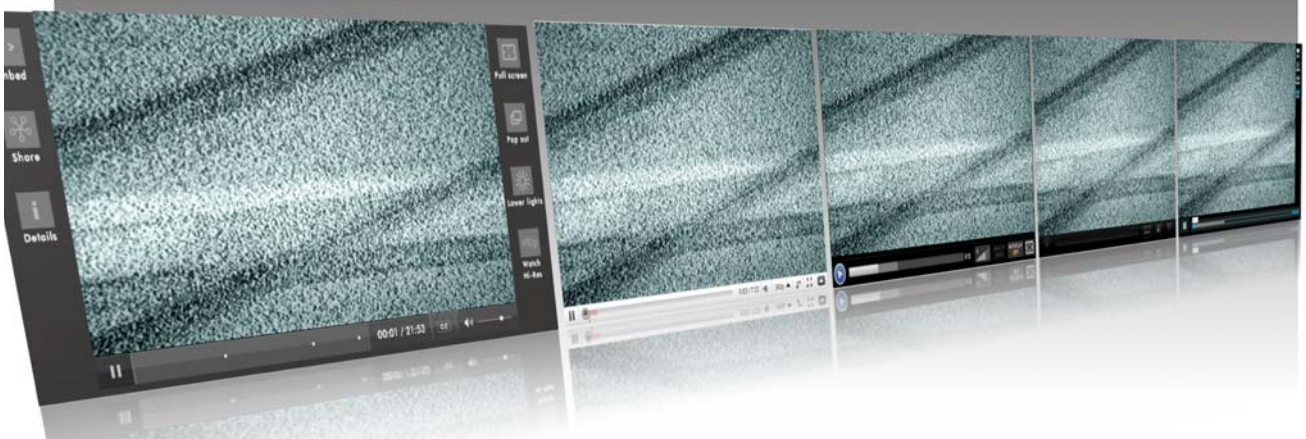
Dynamic Streaming ActionScript 3.0 (FP 10)

- **NetStream.play2(playParam: NetStreamPlayOptions);**
  - streamName
  - oldStreamName
  - start
  - len
  - transition
- **NetStatusEvent**
  - NetStream.Play.Transition
  - NetStream.Play.TransitionComplete
- **NetStreamInfo**
  - currentBytesPerSecond
  - maxBytesPerSecond
  - byteCount
  - droppedFrames
  - playbackBytesPerSecond
  - SRTT

So what does all this add up to for us?

Complexity.

So what's the problem today?

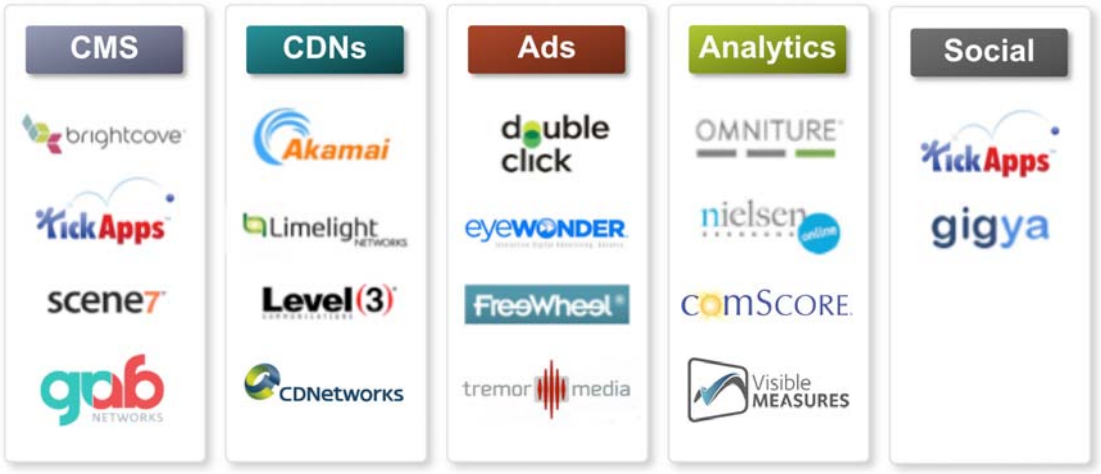


- Duplicated efforts to solve basic problems
- Deployment of new video features takes too long
- Increasing complexity + fragmentation
- Hindered integration with 3rd party services

Open Source Media Framework (OSMF)

- **Simplifies the development** of media players
- **Pluggable** component architecture
- High quality, rich playback experiences
- **Solves common problems**
- API integration
- Quality of Service
- Reporting and analytics
- Lowers development costs, facilitates
- faster turnaround
- **Open framework** facilitates collaborative
- development
- Benefits publishers, Adobe tool users,
- and ecosystem partners
- **FREE**

OSMF: Integration with Service Providers via Plug-ins



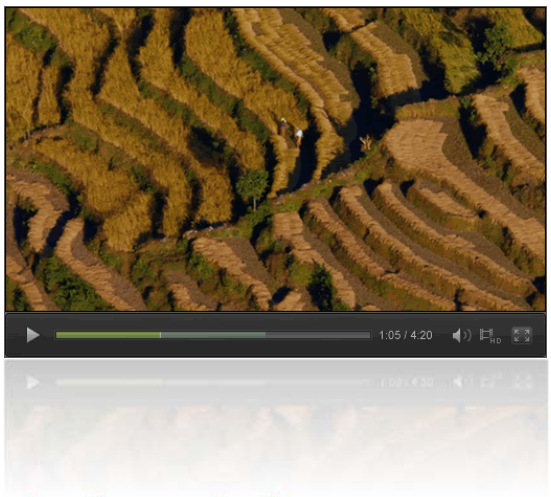
- *Assemble* functionality with plug-ins
- Extensible architecture
- Compile in or load dynamically
- Open API

Strobe.swf

OSMF is powerful and flexible, but…

- Requires programming expertise

What if you want the power without the programming?



Strobe.swf

- Built on OSMF
- A compiled .swf hosted on Adobe.com for anyone to embed anywhere on the web
- Free

*In development now – stay tuned at **www.osmf.org** for more details*

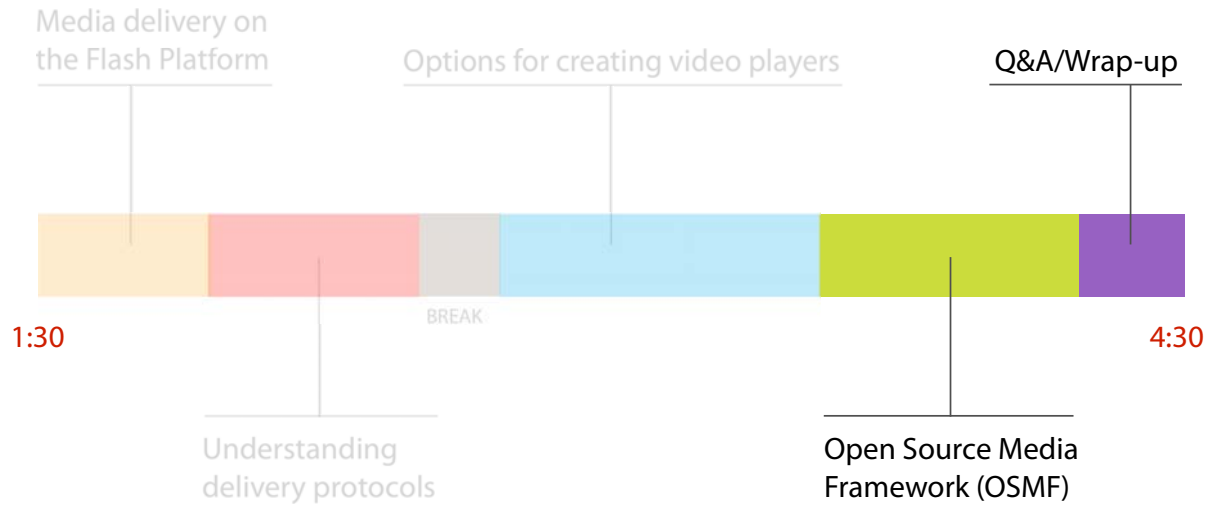
Questions?

NEXT: In depth – Open Source Media Framework

4: Open Source Media Framework (OSMF)

Lisa Larson-Kelley | FlashConnections.com | Flash Platform Consultant

Flexible Protocols and Custom Player Development



4: Open Source Media Framework Agenda

- Framework overview
- Building blocks of an OSMF player
- Layout API
- Installing OSMF
- How to build an OSMF Player
- Hands-on: HelloWorld(s)!
- Hands-on: HelloTrait
- Hands-on: HelloPlayButton

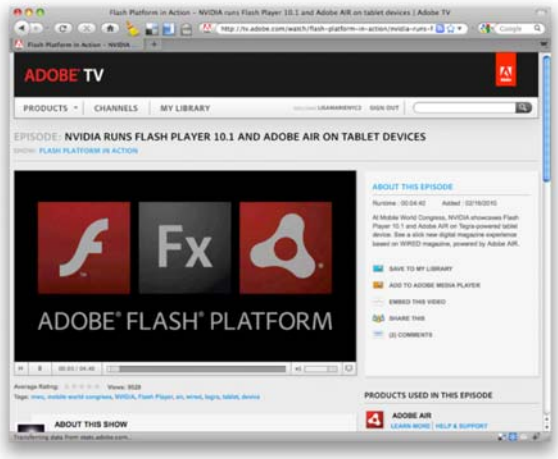
OSMF: Feature Complete 1 (version 0.95)

- API Lockdown complete
- New features in this sprint:
  - Over 130 bug fixes
  - Player size optimizations
  - Ad insertion and Recommendations examples
  - User authentication with NetConnection for FMS
  - Syndication Support (i.e. Atom, RSS 2.0, iTunes, Media RSS)
- Minimal: 35k compiled SWF
- Every feature: 128k compiled SWF
- **http://www.osmf.org**

OSMF 1.0 Features

- Video Quality of Service
  - Error handling
  - Dynamic Streaming
  - HTTP Streaming
- Rich interactive experiences
  - Multiple media types
  - Integrated advertising
  - Dynamic playlists
- Third-party plug-ins
  - CDNs
  - Advertising
  - Analytics

open source media framework



OSMF 1.0 Roadmap *(Q2 2010)*

- **High Quality Video**
  - On-demand and Live
  - Streaming (RTMP and HTTP)
  - Progressive Download
  - FMS 3.5: Dynamic Streaming and DVR
  - Flash Access 2.0
- **Rich Media Experiences**
  - Layout inside and outside player
  - Media: video, audio, images, SWF
  - Playlists: sequential and parallel compositions
  - Cue points
  - Metadata-driven experiences
- **Plug-ins for Services**
  - Ad server calls and ad rendering
  - Tracking and reporting events
  - CDN connections

What's Next

- Ready-to-deploy player SWF
- Flash Authoring visual component
- Default chrome
- External configuration of chrome, layout, plug-ins
- Monitor QoS metrics
- Mobile optimizations
- Flash Media Server 3.x
- Flash Access 2.x

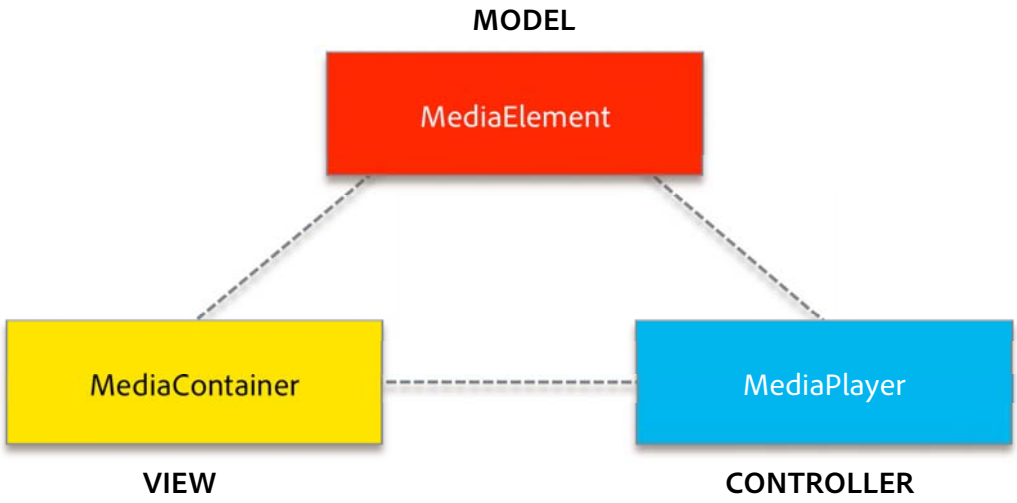
Rich media experiences and advertising

Integration with both standard and proprietary playlist formats

OSMF: Supported File Formats

- Streaming audio
  MP3, AAC, Speex, and Nellymoser
- Streaming video
  FLV, F4V; MPEG-4: MP4, M4V, F4V, 3GPP
- Progressive audio
  MP3
- Progressive video
  FLV, F4V, MP4, MP4V-ES, M4V, 3GPP, 3GPP2, MOV
- Images
  PNG, GIF, or JPG
- SWF files

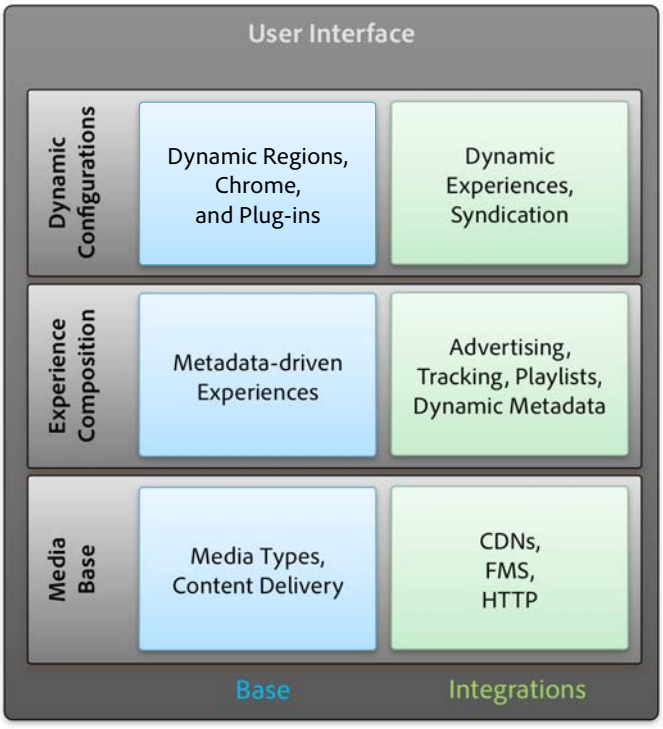
Framework Overview



Architectural Overview

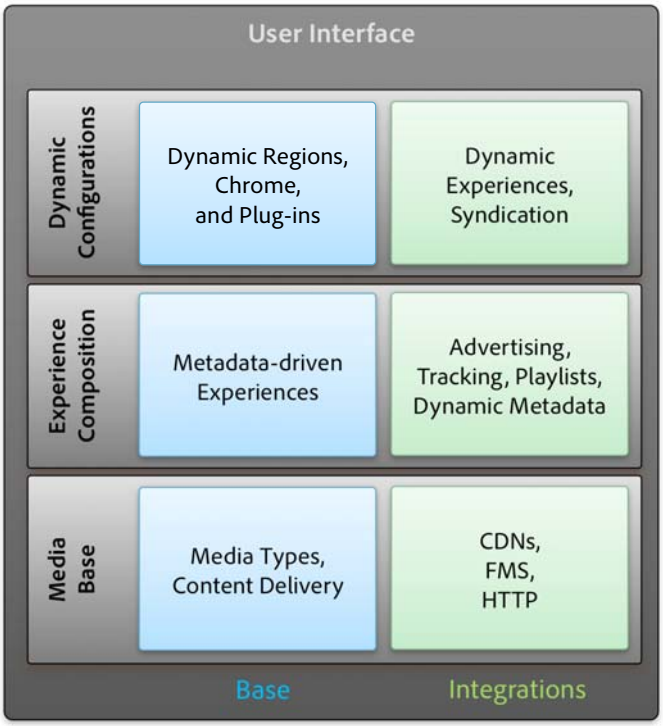
Layered, pluggable architecture

- Integrate at multiple points
- Support a range of use cases
- Take only what you need

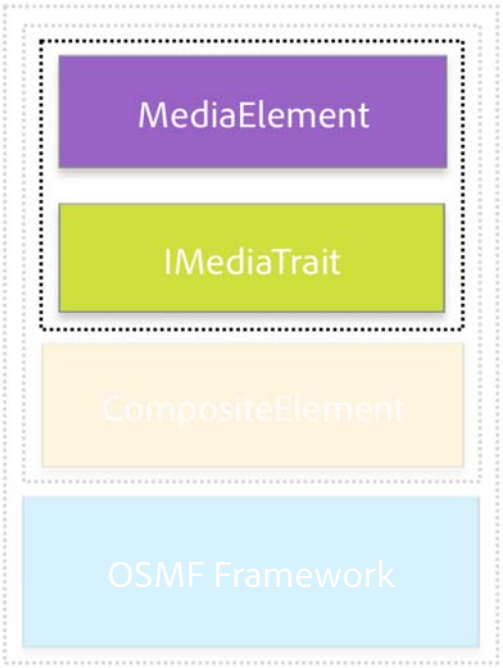


High-level steps for building an OSMF player

1. Start with OSMF as the foundation of your player
2. Add plug-ins for advertising, analytics, CDNs, social
3. ***Design a fabulous UI for your audience***
4. Deploy the player
5. Over time, upgrade to get new features



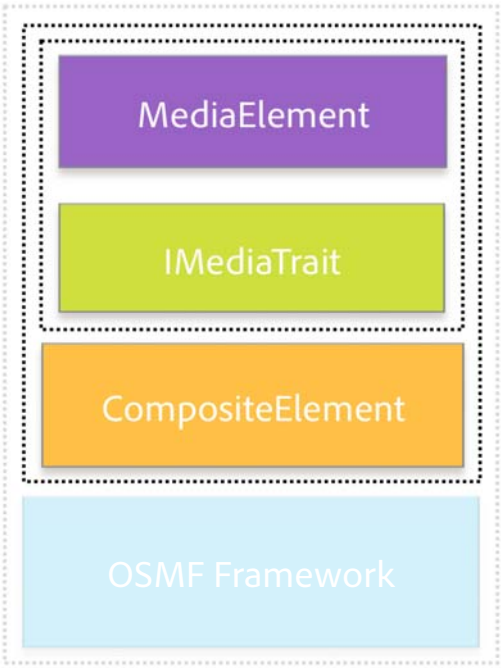
Framework Overview



LEVEL 1: Media Base

- MediaElement – A unified media experience
  - Video, Audio, Images, SWFs
- Traits – IMediaTrait – Can be dynamic
  - Audible
  - Bufferable
  - Loadable
  - Pausable
  - Playable
  - Seekable
  - Spatial
  - Switchable
  - Temporable
  - Viewable

Framework Overview

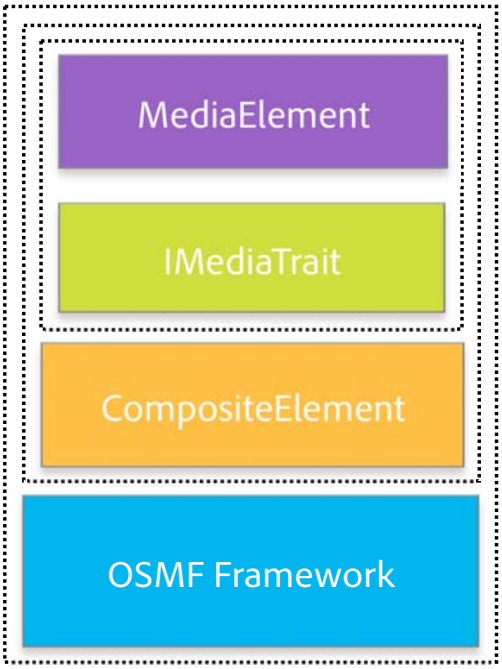


LEVEL 2: Media + Experience Composition

- CompositeElement – MediaElements with children
  - SerialElement
  - ParallelElement

Adobe

Framework Overview

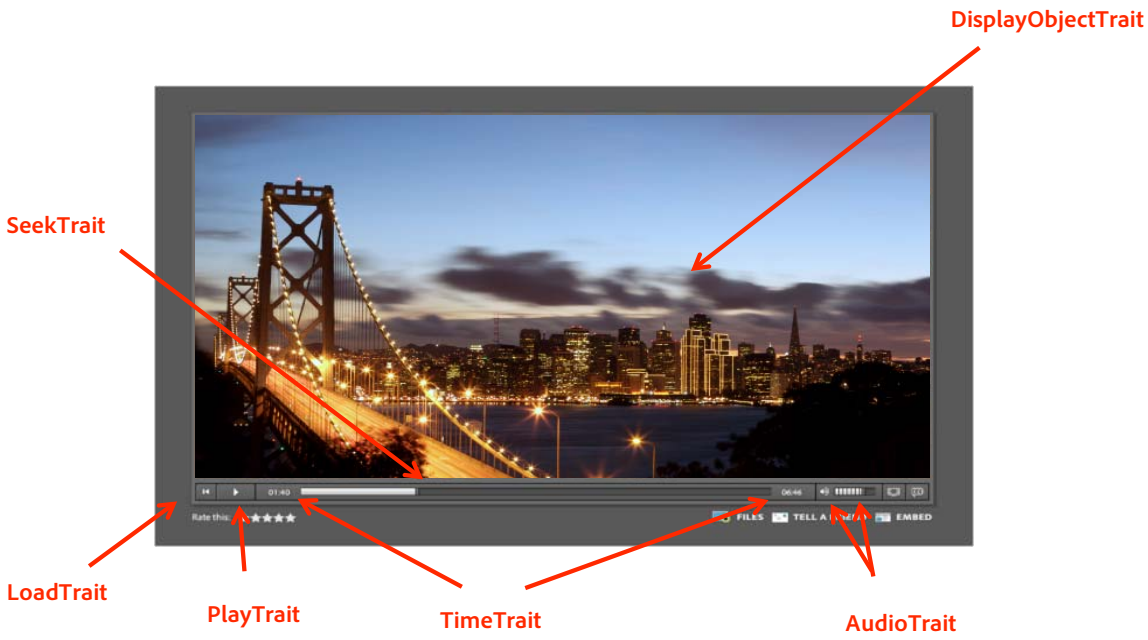


LEVEL 3: Full OSMF Framework

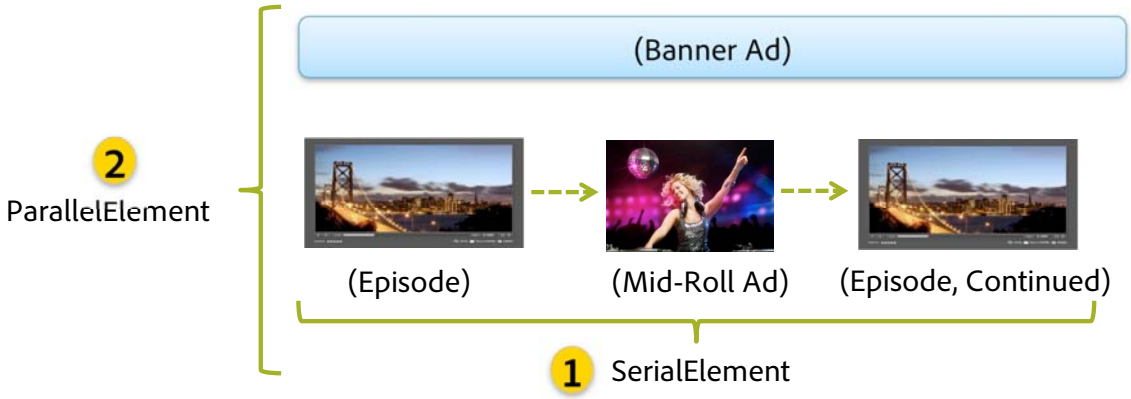
- Plug-ins (Compile time or run time)
- Layout API
- Javascript communication
- QoS, DS, CDNs
- Playlists, ads, tracking
- Dynamic chrome, tweakable default UI
- Syndication

Adobe

OSMF Traits



OSMF Composition



1
```
var serial:SerialElement = new SerialElement();
serial.addChild( /* MediaElement for Episode */ );
serial.addChild( /* MediaElement for Mid-Roll Ad */ );
serial.addChild( /* MediaElement for Episode, Continued */ );
```

2
```
var parallel:ParallelElement = new ParallelElement();
parallel.addChild( /* MediaElement for the Banner Ad */ );
parallel.addChild( serialElement );
```

Hands on!

- Most basic player
- Second most basic player
- Traits
- Intelligent playback controls

Next steps

- OSMF Blog
  http://blogs.adobe.com/osmf/
- OSMF User Group
  http://groups.adobe.com/groups/7af970e6e4/summary
- Adobe Developer Forums
  http://forums.adobe.com/community/opensource/osmf/developers
- Tutorial
  http://www.adobe.com/devnet/flash/articles/video_osmf.html
- OSMF Presentation from MAX
  http://max.adobe.com/online/session/332

Questions?

Source files and more information:

www.flashconnections.com

twitter.com/lisamarienyc

Flash 411: Video Questions, Answered!
Adobe.TV

Flash Video for Professionals
Expert Techniques for
Integrating Video on the Web
Wiley Publishing/Sybex