

Flash Video Across the Screens

realeyes.com

The Power of the Screens, & the Reach of Flash Video

The Benefits of Flash Video

- Ubiquity
- The Screens
- Capability
 - Hardware Acceleration
 - StageVideo
 - Live
 - VOD
 - P2P
 - Multicasting
- User Experience

Challenges

- UI
- Encoding
- File Types
- Apple (as usual)

Overview/Foundation

The Media Player Core

- Custom
 - NetConnection & NetStream
 - Pros: Can be light weight and low level control
 - Cons: Can be hard to build or support over time
- OSMF
 - Full media framework
 - Pros: Extensible, full featured, & supported by Adobe+
 - Cons: Can be complex and a bit heavy if simple is needed

Multi-Protocol Capabilities

- RTMP Based
 - TCP Based
 - Tunneling over HTTP via RTMPT
 - Encryption with RTMPE or RTMPS
- RTMFP
 - UDP Based
 - Supports Unicast & Multicast
 - Encrypted
- HTTP
 - True streaming over HTTP
 - VOD & Live
 - Requires content packaging

Security

- RTMPE+SWF verification
- User authentication
 - Client & Server
 - Token
- DRM: Flash Access 2.0
 - Industry Approved Standard
 - Coming to Mobile & TV Soon
 - Not cheap
 - Can be implemented as a service via BuyDRM or EZDRM

The Screens - Personal Computers

Desktop

- Often where the requirement begins
- Less limitations
- Creativity unleashed with video integration as a part
- Deployment Options:
 - Browser
 - Desktop

Development Considerations

- Handling different Flash Player versions
 - Different functionality supported
 - Conditional code compiling via compiler arguments
 - e.g. HTTP streaming & Streaming DRM support

Hardware Acceleration Basics

- StageVideo
 - Can offload video decoding and/or rendering
 - Result can be lower CPU & memory, w/increased pixel fidelity
 - Requires Flash Player 10.2+
 - Requires compiler argument: `-swf-version=11`
 - Always capable if system supported in FullScreen
 - HTML wmode should be "direct" if not in FullScreen
 - May work in Safari 4+ or IE 9 without "direct"

Hardware Acceleration Basics

- StageVideo (cont.)
 - **Rendered under display list - so nothing can be opaque**
 - Rendered into a flash.media.StageVideo instead of Video
 - H.264 is recommended and certain dimension may not work
 - Mac OS hardware decode of H.264 only on 10.6.3 +

<http://www.adobe.com/devnet/flashplayer/stagevideo.html>

Hardware Acceleration Basics

- StageVideo - API Pointers
 - **StageVideoAvailabilityEvent.STAGE_VIDEO_AVAILABILITY**
 - **StageVideoEvent.RENDER_STATE**
 - **VideoStatus.ACCELERATED**: The video is being decoded and composited through the GPU.
 - **VideoStatus.SOFTWARE**: The video is being decoded through software and composited by the GPU, if dispatched by StageVideo, or through software, if dispatched by Video.
 - **VideoStatus.UNAVAILABLE**: The video hardware has stopped decoding and compositing the video.

http://www.adobe.com/devnet/flashplayer/articles/stage_video.html

Hardware Acceleration Basics

- OSMF & StageVideo
 - Automatic w/OSMF 1.6 (sprint 2) or later
 - Can be disabled by:
OSMFSetting.enableStageVideo = false;
 - Can check if capable by:
OSMFSetting.supportsStageVideo;

Encoding Considerations

- Encoding for desktop
 - H.264 is best choice for StageVideo
 - Base, **Main** or **High** Profile
 - Dimensions divisible by 16
 - <http://www.eventsadobe.com/cookbook/>
 - http://www.adobe.com/devnet/flashmediaserver/articles/h264_encoding.html
 - <http://videorx.com/> [Products > Bitrate Starter]

- DEMO: The basic player w/OSMF+REOPS UI plugin
 - multi-protocol
 - dynamic streaming

Debugging & Testing

- OSMF debugging compiler arguments
 - - **CONFIG::LOGGING** :should be false for production
- Server Side ActionScript (if FMS)
- Using Flash Pro or Flash Builder for debugging

The Screens - Smart Phones (Android/iPhone/RIM)

Flash On Phones

- Many limitations but capabilities growing fast
- Offers expanded reach
- Deployment Options
 - Android
 - Web
 - Native installer - AIR (.apk)
 - iOS
 - Native installer (.ipa)
 - RIM
 - Web
 - Native installer - AIR (.bar)

General Considerations

- Limited hardware
- Fluctuating bandwidth and networks
 - 3G/4G/WiFi
- Consider slimming down the deployment
- Not all devices or platforms are created equal
- Can detect if mobile (phone/tablet) by:
 - Capabilities.cpuArchitecture
 - Capabilities.os

DEMO: Sliming OSMF

- Reducing the file size and bloat
- Only take what you need
- Sounds simple..not always
- Always check that only the classes you need are in the SWF

Android Considerations

- Flash 10.2 Support
- Many devices - nothing is standard
- Screen sizes & densities vary
- Supports all Flash Video playback methods
 - RTMP, RTMPE, RTMFP, etc.
- FlashAccess 2.0 DRM support soon

RIM Considerations

- Currently not available on phones but should be soonish?

iOS Considerations

- Cross Compiled
- Performance is more limited
 - No more than 400Kbps for video
 - Keep dimensions smaller
- In application video limited to VP6 & Sorenson Spark
- H.264 is available if offloaded to OS Native App
- RTMP & RTMFP supported, RTMPE is NOT supported
- iOS packaged AIR application can not load SWF's w/code

Sizing Considerations

- Screen sizes can differ greatly
 - Resolution (Capabilities.screenResolutionX & Y)
 - Density (Capabilities.screenDPI)
 - Physical Dimensions (resolution/dpi)
- UI Elements (such as video control bar) need to be designed and sized for touch interaction

<http://www.paultrani.com/blog/index.php/2010/12/sizing-graphics-correctly-across-screens/>

Sizing Demo

[Review Size & Layout Code]

Hardware Acceleration: Phones

- Is needed - they have puny CPU's
- Phones generally have powerful video decoders
- Supports 1 StageVideo instance at a time
- Implemented the same as desktop

Encoding Considerations

- Main or Base Profile
- Dimensions divisible by 16 (8 if necessary)
- Encode to display size, don't scale in app

4:3	16:9
640x480	640x360
512x384	512x288
480x360	480x272

http://www.adobe.com/devnet/devices/articles/mobile_video_encoding.html

<http://www.adobe.com/devnet/devices/articles/encoding-guidelines-android.html>

- Demo:
 - Android
 - iOS packager

Testing & Debugging

- Android
 - ADL Device Simulation
 - Remote debugging w/Flash Builder
 - USB Connection
 - Over Network
- iOS
 - ADL Simulator
 - Remote debugging w/Flash Builder
 - Over Network

The Screens - Tablets (Android/RIM/iPad)

Overview

- The mid-size personal experience
- Higher-end user experiences, but still limited environment
- Enhanced UI for medium format touch experience
 - Enough real-estate for video + additional UI
- Front Facing cameras becoming standard
 - Video Chat is a growing demand
- Deployment Options - Same as phones

Development Considerations

- Motorola Xoom first mainstream on Android 3
 - Current Flash can't access front camera
 - StageVideo is problematic at the moment
 - Both should be fixed shortly - hopefully
- iPad 2
 - Front facing camera works
 - No StageVideo support at this time - possibly ever?
 - Same limitations as iOS phones but more CPU
- Enhanced RIM (Playbook) video capabilities & CODEC's
 - H.264, MPEG4 & WMV

Encoding Considerations

- Same as Mobile

4:3	16:9
-	1280x720
768x576	1024x576
640x480	640x360
512x384	512x288
480x360	480x272

- Demo Tablet Video:
 - Xoom (but don't have one here - sorry its @ NAB)
 - iPad 2

- Take it further
 - Sample: P2P chat tablet demo

Testing/emulating

- Android
 - ADL Device Simulation
 - Remote debugging w/Flash Builder
 - USB Connection
 - Over Network
- iOS
 - ADL Simulator
 - Remote debugging w/Flash Builder
 - Over Network

The Screens - TV's (Samsung TV & Google TV)

Overview

- Big screen, little CPU - what to do?
- Emerging market - plenty more to come
- High-resolution and high expectations of end-users
- Deployment Options:
 - Web (Google TV, TV Browser)
 - AIR (Samsung App Store - eventually)

Development Considerations

- Hardware acceleration is a must
 - If in browser use only wmode="direct"
- Home bandwidth + wireless = unreliable
 - Use Adaptive Bitrate streaming
 - Implement reconnection/resume playback (FMS 3.5.3+)
- Limited UI and user input controls
 - New keyboard APIs
- Flash Access 2.0 DRM (soon)
- Know your stage dimensions
 - Overscan may occur
 - Keep things title safe (7.5% inset on full dimensions)
- Fonts

DEMO: RECONNECT RESUME

- <http://aspexamples.adobe.com/flash/streamreconnect.html>

Hardware Acceleration Notes

- Any content even old Flash Video Object will be rendered as StageVideo
 - This is likely temporary to handle legacy content
- Any StageVideo unsupported functionality will be disabled
- Only supports playback of 1 video at a time
-
- Use StageVideo like normal but know even if you don't for now it will likely be hardware rendered if possible

Screen Sizes & dimensions

TV Resolution	Stage Dimensions	Title Safe Area
540p	960x540	x=72 y=40 w=816 h=460
720p	1280 x 720	x=96 y=54 w=1088 h=618
1080p	1920 x 1080	x=144 y=81 w=1632 h=918

Adobe Recommended TV Encoding Guidelines:

- Video codec: H.264, Main or High profile, progressive encoding
- Resolution: 720i, 720p, 1080i, or 1080p
- Frame rate: 24 or 30 frames per second
- Audio codec: AAC-LC or AC3, 44.1 kHz, stereo
- Combined bit rate: up to 2Mbps (or higher depending on available bandwidth)
- Audio bit rate: up to 192 kbps
- Pixel aspect ratio: 1 × 1

Links

- http://www.adobe.com/devnet/devices/flash_platform_tv.html
- <http://blogs.adobe.com/ktowes/2011/04/sneak-peak-future-adobe-technology-for-http-streaming-across-multiple-devices.html>
- http://www.adobe.com/devnet/devices/articles/video_content_tv.html
- <http://labs.adobe.com/technologies/flashplayer10/live-player/>
- <http://tv.flash.com/>
- <http://www.google.com/tv/>
- <http://code.google.com/tv/web/>
- <http://www.adobe.com/devnet/devices/fpmobile.html>
- **Reconnect/Resume:** http://help.adobe.com/en_US/flashmediaserver/devguide/WSae44d1d92c7021ff-1f5381712889cd7b56-8000.html
-
- <http://delicious.com/coderjun/video>

<http://realeyes.com/>

david@realeyes.com

jun@realeyes.com

john@realeyes.com