Open Video Player Tech Note

For Developers Moving from *Akamai Flash Media Player Framework v1.8* to *Open Video Player v1.0 (Flash)*

November 26, 2008

Intro

This document lists the code changes from *version 1.8* of the *Akamai Flash Media Player Framework* to *version 1.0* of the *Open Video Player* code base for Flash ActionScript 3.0. In addition to this document please see the ASDoc-generated documentation for specific details on classes and their properties, methods, and events.

Code Changes

This section lists the details of the code modifications for *v1* of the *Open Video Player* code base.

ActionScript Package Structure

The ActionScript package structure is described below. Please see the ASDoc files for specific information on these classes and their hierarchies.

org.openvideoplayer.events

- OvpError.as
- OvpEvent.as

org.openvideoplayer.net

- INetConnection.as
- OvpConnection.as
- OvpNetStream.as
- AkamaiConnection.as*
- AkamaiNetStream.as**
- AkamaiEnhancedNetStream.as

*this is a new and very thin class derived from OvpConnection and contains functionality specific to the Akamai network, such as the hostname/appname requirement for calling the connect method. Please see the ASDocs for more info.

**this is a new and very thin class derived from OvpNetStream and contains functionality specific to playing a stream over the Akamai network, such as live stream authentication.

org.openvideoplayer.rss

This package contains classes that model the Media RSS standard as well as custom classes to parse RSS feeds specific to the Akamai network.

org.openvideoplayer.utilities

This package contains utility classes for date comparison, Flash Player client version detection, http bandwidth estimates, and string and time functions.

Separate NetConnection and NetStream Classes

The OvpConnection class implements INetConnection, which allows it to behave like a NetConnection object. Its primary job is to provide a robust connection, trying multiple ports and protocols. By implementing INetConnection, client code can add listeners just as one would with a flash.net.NetConnection object.

Our goal was to make this class behave as much like a flash.net.NetConnection class as possible. Therefore, a client will typically listen for NetStatusEvent events on their OvpConnection instance. The Flex and Flash samples demonstrate this.

The OvpNetStream class 'is a' flash.net.NetStream. Client code will instantiate this class and give it either an OvpConnection object (or an object derived from OvpConnection) or give it the OvpConnection.netConnection property. The OvpConnection.netConnection property is the flash.net.NetConnection object which was successfully connected after the call to OvpConnection.connect(). Again, the Flex and Flash samples demonstrate this behavior.

Simplified Events and Errors

The Open Video Player code base contains one event class OvpEvent. This class contains a data property, the contents of which vary depending on the event. Please see the ASDocs for OvpEvent.

The OvpError class contains error numbers and descriptions for all errors dispatched by the code base. Please see the ASDocs for OvpError.

IDENT Removal

The AkamaiConnection class now detects the Flash Player version and will execute the previous behavior (meaning *Akamai Flash Media Framework v.1.8*) only if the client is running Flash Player version 9.0.60 or earlier.

Live Stream Subscribe Process

The FCSubscribe call was not added to the AkamaiNetStream class because this functionality will be obsolete and unnecessary with an Akamai server-side release scheduled for January 2009.

Stream Name Prefixes

The AkamaiNetStream class will now add the proper stream name prefix if it is missing and the file extension contains mp3, mp4, m4v, f4v, 3gpp, or mov.

Named Application Instances

The OvpConnection class' connect method accepts the standard Flash Media Server "*hostname/appname/instanceName*" format.

The AkamaiConnection class' connect method requires "*hostname/appname"*. Note this is a change from *Akamai Flash Media Framework v.1.8* which used the *isLive* property to determine the *appname if it was not supplied.*

[End of Document]