# LiveStreamCast Application for Adobe Flash Media Interactive Server

LiveStreamCast Application for Adobe® Flash® Media Interactive Server

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

# Architecture of LiveStreamCast application for Flash Media Interactive Server

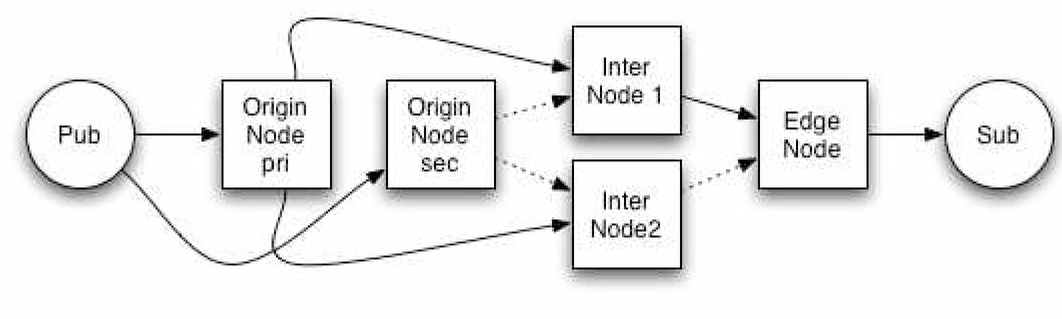
## Introduction

One of the functions of Flash Media Server (FMS) is to allow users to broadcast video, audio and data from one client to the other clients. However, one FMS server can only support limited number of live streams. This document shows how to set up a scalable structure to support more live streams through multiple layers of FMS servers. The entire setup involves different components such as the broadcasting client (typically Flash Media Live Encoder), server-side applications, and subscribing client SWFs.

The server-side architecture uses multiple  layers of server-side applications organized into Server Nodes. The layers include Origin Nodes, Intermediate Nodes and Edge Nodes. The applications are written in FMS ActionScript format (.asc).

You can set up any number of Intermediate Nodes and Edge Nodes and can change the number of layers from 2 to *n*. While a 2-tier setup (Origin Node and Edge Node) is functionally possible, in general, a 3-tier setup should be able to handle most situations.

To use the applications, create 3 different application folders in FMS (plus a simple monitor app on the Intermediate and Edge Nodes) and place the .asc files in each application folder. In a real scenario, the three layers of applications should be placed and running in FMSs at three different physical locations. For testing purpose, all applications can be running from a single FMS. A data flow diagram is shown below.



# System Requirements

- Flash Media Interactive Server 3.5 configured in Origin-only mode
  **Note**: Edge/Origin server configuration is not supported.
- Windows or Linux distributions.
- Flash Media Live Encoder 3.0 for live stream capture.
- Private VIP network between Nodes recommended.

# Installation

This section contains the following discussions:

- Architecture
- Installation
- Testing

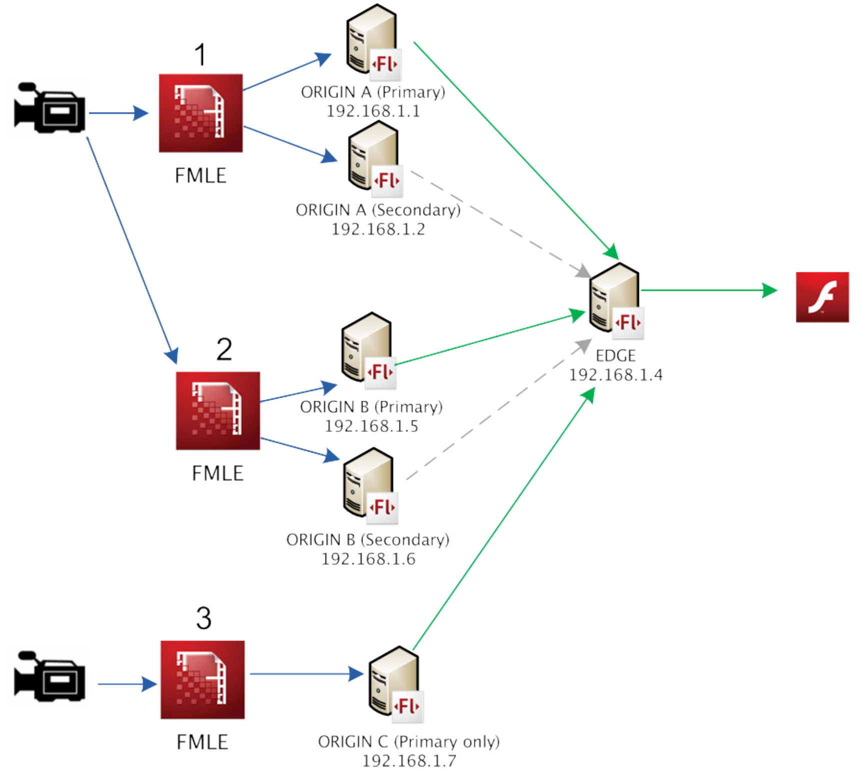
## Architecture

Several topologies are possible and are defined by the host.ini files on each Edge and Intermediate Node. The host.ini file contains values for Server Groups, the Primary and Secondary IPs of the upstream servers, and the application/instance name of the upstream application.

The host.ini format as follows (server group name is simply a label):

| server group name | Primary IP | Secondary IP | Application Name/Instance |
|---|---|---|---|
| Server Group 1 | aaa.aaa.aaa.aaa | bbb.bbb.bbb.bbb | livestreamcastapp |
| Server Group 2 | ccc.ccc.ccc.ccc | ddd.ddd.ddd.ddd | livestreamcastapp |
| Server Group N | yyy.yyy.yyy.yyy | zzz.zzz.zzz.zzz | livestreamcastapp |

## LiveStreamCast with multiple origin groups, multiple encoders, and no intermediate nodes



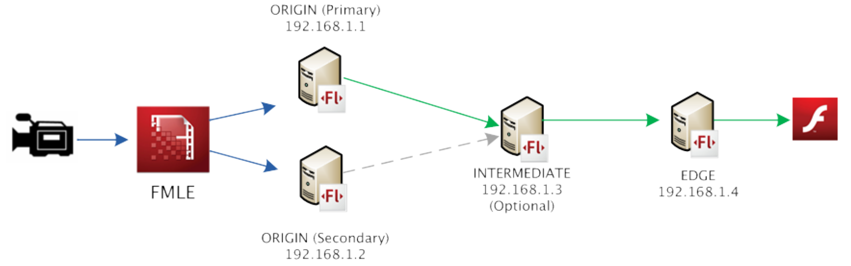
**host.ini (Edge Node)**

```
#Server Group 1: Primary and Secondary origins available from Encoder 1
server1 192.168.1.1;192.168.1.2
livestreamcast_origin

#Server Group 2:  Primary and Secondary origins available from same Encoder 2
server2 192.168.1.5;192.168.1.6
livestreamcast_origin

#Server Group 3: Primary only (no secondary)in the 3rd group from Encoder 3
server3 192.168.1.7;192.168.1.7
livestreamcast_origin
```

## LiveStreamCast with optional intermediate node



**In this scenario, an intermediate server is installed with the Edge Server node application.**

**host.iniINI (Intermediate Node)**

```
server1 192.168.1.1;192.168.1.2
livestreamcast_origin
```

**host.ini (Edge Node)**

```
server1 192.1.168.3;192.1.168.3
livestreamcast_intermediate
```

*Note that the Edge Node references the primary and secondary server as the same server.*

# Installation

To install LiveStreamCast in Adobe Flash Media Interactive Server 3.5, follow these node-specific instructions.

## Origin Node Installation

The Origin Node is the ingest server that the live encoder connects to. In a topology, you can configure any number of Origin Nodes, though two are usually sufficient to provide failover.

1. On each Origin Node, browse to the folder, `fms-install/applications`, or the custom applications folder identified in the `FMS.ini file`
2. In the ZIP package, browse to the originNode folder.
3. Copy the livestreamcast_originfolder into the applications folder.
4. (optional) If you have a secondary Origin Node, repeat steps 1-3
5. The ingest server for your live stream from Adobe Flash Media Live Encoder will be the Origin Node(s). Since there is no upstream FMS server, there is no host.ini file to configure.

## Edge Node Installation

The Edge Node is the actual server that the client connects to. It acts as a proxy for the Origin server and manages the upstream server connections automatically.

1. On the Edge Node, browse to the folder, `fms-install/applications`, or the custom applications folder identified in the `FMS.ini file`
2. In the ZIP package, browse to the edgeNode folder.
3. Copy the livestreamcast_edge & monitor folders into the applications folder
4. Configure the host.ini file by performing the following steps:
   a. Locate the host.ini file inside the applications/livestreamcast_edge folder.
   b. Replace the sample server configuration with your actual server addresses:
   ```
   <serverGroup1 label> <ADDRESS OF PRIMARY>;<ADDRESS OF SECONDARY>
   <ORIGIN APPLICATION NAME>
   server 1 192.168.1.1;192.168.1.2 livestreamcast_origin
   ```
   c. (optional) If you have multiple Origin Server Groups, add additional lines for each group in your host.ini file:
   ```
   # secondary origin available from Encoder 1
   server1 192.168.1.1;192.168.1.2 livestreamcast_origin
   # secondary origin available from same Encoder 2
   server2 192.168.1.5;192.168.1.6 livestreamcast_origin
   # no secondary in the group from Encoder 3
   server3 192.168.1.7;192.168.1.7 livestreamcast_origin
   ```

**Note**: The above configuration pertains to a 2-tiered topology of Edge and Origin Nodes. If you are configuring a 3-tiered topology, the Primary and Secondary IPs, and application name (livesteamcast_intermediate) would consist of the values for the Intermediate Node

## (Optional) Intermediate Node Installation

Intermediate Nodes are Edge server nodes configured to proxy the origin. You can have any number of intermediate nodes.

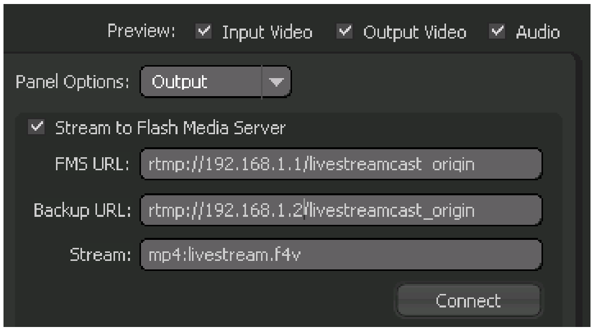
1. On the intermediate server node, browse to the folder, *`fms-install`*`/applications`, or the custom applications folder identified in the FMS.ini file
2. In the ZIP package, locate the intermediateNode folder.
3. Copy the livestreamcast_intermediate & monitor folders into the applications folder
4. Configure the host.ini file by performing the following steps:
    a. Locate the host.ini file inside the folder applications/livestreamcast_intermediate.
    b. Replace the sample server configuration with your actual server addresses
    ```
    <serverGroup1 label>  <ADDRESS OF PRIMARY>;<ADDRESS OF SECONDARY>
    <ORIGIN APPLICATION NAME>
    server 1 192.168.1.1;192.168.1.2 livestreamcast_origin
    ```
    c. Note – Edge Nodes will point to the Intermediate Node and not the Origin Node. You will need to configure your Edge Node host.ini file as follows:
    ```
    <serverGroup1 label>  <ADDRESS OF PRIMARY>;<ADDRESS OF SECONDARY>
    <INTERMEDIATE APPLICATION NAME>
    server 1 192.168.1.3;192.168.1.4 livestreamcast_intermediate
    ```
    d. (optional) If you have multiple Origin Server Groups, add additional lines for each group in your host.ini:

    ```
    # secondary origin available from Encoder 1
    server1 192.168.1.1;192.168.1.2 livestreamcast_origin
    # secondary origin available from same Encoder 2
    server2 192.168.1.5;192.168.1.6 livestreamcast_origin
    # no secondary in the group from Encoder 3
    server3 192.168.1.7;192.168.1.7 livestreamcast_origin
    ```

# Testing your installation

1. Ensure all server nodes are running (monitor the *FMS-install*/logs/access.00.log files on each FMS server to ensure all connections are made to the appropriate servers)
2. On a separate computer that can connect to the origin server node, install and launch Adobe Flash Media Live Encoder 3

3. Configure the Primary and Secondary server in Flash Media Live Encoder 3. The FMS URL setting specifies the primary origin server, and the Backup URL setting specifies the secondary origin server, as shown below:



Panel Options: Output

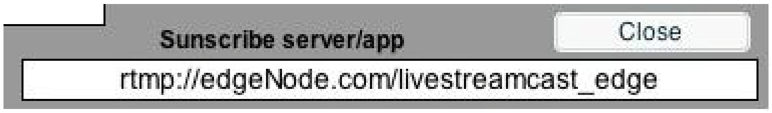
☑ Stream to Flash Media Server

FMS URL: rtmp://192.168.1.1/livestreamcast origin

Backup URL: rtmp://192.168.1.2/livestreamcast_origin

Stream: mp4:livestream.f4v

4. Open the LiveStreamCast Demo Publisher and Subscriber Flash client app that is included in this framework. This client app supports the necessary FCPublish and FCSubscribe events used by the LiveStreamCast framework. Refer to the livestreamcastDemoApp readme for details about this app. ***Note***: The standard Netstream play functionality in the FSM Sample Video Player does not support FCSubscribe and is not sufficient for use with LiveStreamCast.
5. Locate the Subscribers section of the Publish and Subscribe demo app. We will not use the Publisher section for this example.
6. Enter the **Server URI** of your Edge Node including the **name** of the stream (set in the encoder, defaults to livestream1).



Sunscribe server/app: rtmp://edgeNode.com/livestreamcast_edge

```
rtmp://192.168.1.4/livestreamcast_edge/mp4:livestream.f4v
```

7. From the Subscriber section, [Connect] to the EdgeNode. Upon a successful NetConnection the [Subscribe] button will enable.
*(Note, if you are streaming On2/Mp3 the URI will not require the prefix or the extension on the stream name)*

```
rtmp://192.168.1.4/livestreamcast_edge/livestream
```

8. Click [Subscribe] and the live stream video should play.

# Server Roles Description

## Origin Node

This FMS application receives the video stream from the publisher and distributes the data to multiple downstream Intermediate and Edge Nodes.

### Features

- Ability to accept connections from publishers and maintain a list of all publishers.
- Accepts connections from intermediate nodes and notifies intermediate nodes when a publisher is added or removed.
- Accepts play command from the intermediate nodes and distributes data to the intermediate nodes.

## Intermediate Node

This FMS application receives video streams from the Origin Node and passes the data to downstream Edge Nodes. It is the middle layer of the distribution network.

### Features

- Ability to read Origin IPs from a configuration table in the host.ini file.
- Connects to all Origin nodes in the configuration table and reconnects if a connection drops.
- Receives notifications from Origin nodes and maintains a list of active publishers for each Origin node.
- Sends play command to and receives data from Origin nodes.
- Reports connection/stream errors to Edge nodes.

## Edge Node

The Edge application receives connection requests from clients and delivers the data from upstream Intermediate Nodes to the clients.

### Features

- Ability to read Intermediate Node IPs from a configuration table in the host.ini file.
- Connects to all Intermediate Nodes in the configuration table and reconnects if a connection drops.
- Receives notifications from Intermediate Nodes and maintains a list of active publishers for each Intermediate node.
- Sends play command to and receives data from Intermediate Nodes.
- Reports connection/stream errors to Edge Nodes.

## Failover

When the LiveStreamCast app starts up, servers read their host.ini file to parse the list of upstream Server Groups and make Netconnections to them. LiveStreamCast app then builds an array of upstream servers with successful Netconnections.

A Netstream Play request goes out to all upstream servers and LiveStreamCast builds a second list of available servers in the order they respond. Streaming starts from the first Server to respond.

LiveStreamCast maintains a stream activity idle interval of 1 second. When stream inactivity exceeds that idle interval, LiveStreamCast switches to the Primary IP of the next available Server Group in the list. In the background, LiveStreamCast attempts to re-connect to the Primary IP of that first Server Group and after 3 attempts, attempts to connect to the Secondary IP of that Server Group. If a Server Group's Primary and Secondary IP point to the same machine, LiveStreamCast will try to re-connect to that same machine.

When a disconnected machine comes back on-line, that machine is added to the bottom of the list of available servers. LiveStreamCast will stay connected to any machine until another failover occurs, then LiveStreamCast switches to the next machine in its list.

Each node's application.xml overrides the global FMS application.xml and provides an automatic start mechanism to provide faster failovers.

For fast failovers, use multiple Server Groups. Use the same machine IP for both Primary and Secondary values, to configure a single machine for each Server Group.

# API Reference Guide

The design of this multi-layer structure requires servers to keep track of all live streams beings published to  Origin Nodes and pass the publishing information down to the subscriber through the other servers. Because of the complexity, both publishing and subscribing cannot be done directly through the use of NetStream.publish() and NetStream.play(). Instead, client-server and server-server communications are done through the use of command calls in NetConnection (that is, nc.call() ). A set of APIs is defined to publish to and subscribe from a server application. Notably, a publisher calls FCPublish and a subscriber calls FCSubscribe to send notification throughout the topology. Adobe Flash Media Encoder automatically sends FCPublish, any custom publishing app needs to emulate that behavior. Likewise, the subscribing application needs to send FCSubscribe (see the included subscriber.fla for a reference). See the API reference below for details.

# Origin Node

The top layer of server application, which allows connections from the client publishers and other servers. It contains 2 server script files - ExLiveStream.asc and main.asc.

**API defined in Origin Node:**

Methods coming in through NetConnection calls:

- **FCPublish** - Used by the client to request for publishing. The result will be returned through a callback function onFCPublish().
- **FCUnpublish** - Used by the client to stop publishing. The result will be returned through a callback function onFCUnpublish().
- **FMSubscribe** - Used by the client to subscribe. The result will be returned through a callback function onFMSubscribe().

- **FCUnsubscribe** - Used by the client to unsubscribe. The result will be returned through a callback function on FCUnsubscribe().
- **FCAddChild** (server-to-server) - When a child server (Intermediate Node) connects to the server application, the child server sends this command to the parent in order to register as one of the child servers. When an event happens, for example, a new publisher is added to the Origin Node, a control command (addStream or removeStream) is passed from the parent to the child server.

Methods going out:

- **onFCPublish** - The callback returned by the server application to the client. The client receives an info object that describes the result. The info object can be one of the following:
    - The publish request is accepted, info.code = "NetStream.Publish.Start", info.description = stream name.
    - The publish request is denied, info.code = "NetStream.Publish.BadName", info.description = stream name.
- **onFCUnpublish** - The callback returned by the server application to the client when server received an FCUnpublish call. The info object will be info.code = "NetStream.Unpublish.Success", info.description = stream name.
- **onFMSubscribe** - The callback returned by the server application to the client. Client will receive an info object that describes the result. The info object can be one of the following:
    - The stream is found, info.code = "NetStream.Play.Start", info.description = stream name.
    - The stream doesn't exist, info.code = "NetStream.Play.StreamNotFound", info.description = stream name.
    - The publish request was unpublished after play started, info.code = "NetStream.Play.UnpublishNotify", info.description = stream name.
- **onFCUnsubscribe** - The callback returned by the server to the client when server received a FCUnsubscribe call. The info object will be info.code = "NetStream.Play.Stop", info.description = stream name.
- **addStream** - Sent by the server to all the child servers when a new stream is published.
- **removeStream** - Sent by the server to all the child servers when a steam is unpublished.

# Intermediate Node

This is the middle layer of servers, which connects the Origin Nodes and the Edge Nodes. It contains three server script files - ExLiveStream.asc, FCRemoteConnection.asc and main.asc. It also contains an application.xml and a configuration file, host.ini. The application.xml file is used to load the application when the server starts, it contains the following section:

```
<Application name=" livestreamcast_intermediate">
```

```
<!-- Load the application when the server starts -->
<LoadOnStartup>true</LoadOnStartup>

<!-- Set the idle time for one year, which should be long enough -->
<MaxAppIdleTime>31536000</MaxAppIdleTime>

</Application>
```

The host.ini is used to configure the server app so that it will maintain a remote connection to the Origin Node. All the lines started with a "#" are comments and will be skipped. An example of host.ini:

```
#server group name primaryIP;secondaryIP appToConnect
server1 10.41.1.215;10.200.1.169 livestreamcast_origin
server2 10.41.1.216;10.200.1.170 livestreamcast_origin
```

**API defined in Intermediate Node:**

Methods coming in through NetConnection calls,

- **FMSubscribe** - Used by the clients to subscribe. The result will be returned through a callback function onFMSubscribe().
- **FCUnsubscribe** - Used by the client to unsubscribe. The result will be returned through a callback function on FCUnsubscribe().
- **FCAddChild** (server-to-server) - When a child server (Edge Node) connects to this application, the child server sends an FCAddChild command to the parent to register as a child server. When an event happens, a new publisher is added to the Origin Node and a control command (addStream or removeStream) is passed from the parent to the child server.

Methods going out,

- **onFMSubscribe** - The callback returned by the server app to the client. Client will receive an info object which describes the result. The info object can be one of the following:
    - The stream is found, info.code = "NetStream.Play.Start", info.description = stream name.
    - The stream doesn't exist, info.code = "NetStream.Play.StreamNotFound", info.description = stream name.
    - The publish unpublished after play started, info.code = "NetStream.Play.UnpublishNotify", info.description = stream name.
- **onFCUnsubscribe** - The callback returned by the server to the client when server received a FCUnsubscribe call. The info object will be info.code = "NetStream.Play.Stop", info.description = stream name.
- **addStream** - Sent by the server to all the child servers when a new stream is published.
- **removeStream** - Sent by the server to all the child servers when a steam is unpublished.

# Edge Node

This is the lowest layer of server structure that allows connections from the client subscriber. It is a repeating layer of the Intermediate Node and has the same server script files - ExLiveStream.asc, FCRemoteConnection.asc and main.asc. The application.xml and host.ini files are slightly different from an Intermediate Node and are shown below:

```
<Application name="livestreamcast_edge">

<!-- Load the application when the server starts -->
<LoadOnStartup>true</LoadOnStartup>

<!-- Set the idle time for one year, which should be long enough -->
<MaxAppIdleTime>31536000</MaxAppIdleTime>

</Application>
```

The host.ini file is used to configure the server app so that it maintains a remote connection to the Origin Node (via the Intermediate Node). All the lines started with a "#" are comments. An example of host.ini is shown below:

```
#server group name primaryIP;secondaryIP appToConnect
server1 10.21.1.215;10.100.1.169 livestreamcast_intermediate
server2 10.21.1.216;10.100.1.170 livestreamcast_intermediate
```

*Note: Both Edge and Intermediate Nodes also require the presence of the included monitor helper-application (an empty folder used by SSAS) to ensure fast failover functionality & backwards compatibility with earlier versions of FMS.*