

TechNote – AltitudeCDN™ Multicast+ and OmniCache Integration with Microsoft Stream

Version 1.0

AltitudeCDN™ Multicast+ and AltitudeCDN OmniCache™ support Microsoft Stream, bringing secure video multicasting and caching to enterprise environments that use Microsoft Stream for live events or Video on demand (VOD) streaming.

This guide describes how to use Multicast+, OmniCache, or both, to support Microsoft Stream video.

Contents

Contents	1
Introduction	3
Requirements	4
Using Multicast+ with Microsoft Stream	5
Component Overview	5
Functional Overview	6
Event Creation	6
Pre-Event Preparation	6
User Connection to the Event	6
Using OmniCache with Microsoft Stream	7
Component Overview	7
Functional Overview	7
Ramp Plugin Operation	7
Configuring the Multicast+ and OmniCache Integration with Microsoft Stream	8
Configuration Summary	8
Using Multicast+ with Microsoft Stream	8
Using OmniCache with Microsoft Stream	8
Activating the Program Guide	9
Configuring Microsoft Stream Signature Key Retrieval	9
Configuring the Ramp Plugin for Azure Media Player	10
Creating the Microsoft Stream Event	12
Creating the Multicast+ Event	12
Operational Notes	13

Introduction

With live events and VOD streaming, large numbers of viewers attempt to connect separately to external internet content delivery servers. This surge creates a high demand on network resources that can lead to poor video quality, or even denial of service for viewers and other network clients.

To avoid these issues, customers can use AltitudeCDN Multicast+ and AltitudeCDN OmniCache so resource-intensive network video traffic is processed locally, rather than traversing the enterprise WAN:

- **AltitudeCDN Multicast+** – A patented (US Pat. 9,516,390) solution that brings multicast support to any live video deployment that uses HTTP Live Streaming (HLS) or Dynamic Adaptive Streaming over HTTP (DASH).

By using Multicast+ Senders and Receivers, a video event is presented as a single multicast stream that is shared simultaneously by multiple viewers, allowing an event deployment to scale to large audiences, while conserving network bandwidth.

- **AltitudeCDN OmniCache** – A robust enterprise video caching-proxy that enables the efficient delivery of HLS and DASH streams behind the firewall.

By using OmniCache, retrieval of video assets from external CDN servers is greatly reduced, and shared content is internally distributed to enterprise WAN clients. The flexibility of this approach allows you to properly engineer network bandwidth usage to avoid service degradation, and maintain a high-level quality of service (QoS).

- **AltitudeCDN Multicast+ and OmniCache Microsoft Stream Integration** – Introduces a solution that incorporates secure video multicasting and caching with Microsoft Stream. The solution requires minimal additional configuration to your AltitudeCDN environment, and requires no changes to the stream player

The Multicast+ and OmniCache integration with Microsoft Stream extends the functionality of AltitudeCDN to provide support for Microsoft Stream video events. To use either Multicast+, OmniCache, or both, with Microsoft Stream, you first need to have a functional AltitudeCDN and Microsoft Stream environment, then perform the additional configurations described in this guide.

Requirements

To use AltitudeCDN with Microsoft Stream, you need the following:

Solution	Item	Recommendation
Multicast+ with Microsoft Stream	Altimeter Management Server	<p>Altimeter Management Server 2.2 or later:</p> <ul style="list-style-type: none"> The Management Servers must be installed and running on on-premise Windows or Linux platforms, positioned within your network and reachable by Multicast+ Senders and Receivers. For more information, see the <i>AltitudeCDN Altimeter Installation Guide</i>. The Management Servers require an external internet connection to retrieve Microsoft Stream signature keys for JSON Web Token validation. <p>Note: The retrieval of Microsoft Stream signature keys may require configuring an output HTTP proxy for the Altimeter Management Server. For more information, see Operational Notes.</p>
	Multicast+ Sender	<p>Multicast+ Sender v2.2 or later. The Senders must be installed and running on on-premise Windows or Linux platforms, positioned within your network so multicast traffic can reach the audience.</p> <p>For more information, see the <i>AltitudeCDN Multicast+ Deployment Guide</i>.</p>
	Multicast+ Receivers	<p>Any of the following:</p> <ul style="list-style-type: none"> Multicast+ Windows Receiver 2.2 or later. Multicast+ Mac Receiver 2.2 or later. Multicast+ Java Receiver 2.2 or later. <p>The Receivers must be installed on each client viewing device prior to the event. For more information, see the <i>AltitudeCDN Multicast+ Deployment Guide</i>.</p>
	Microsoft Stream Video ID	<p>The Microsoft Stream Video ID (VID) for a Stream event is required by Multicast+ to access and process the stream source.</p>
OmniCache with Microsoft Stream	OmniCache nodes	<p>OmniCache v2.0 or later – The OmniCache must be installed and running on an on-premises Windows or CentOS platform, positioned within the customer network so traffic can reach both the audience and external video source origin. For more information, see the <i>AltitudeCDN OmniCache Deployment Guide</i>.</p>

Using Multicast+ with Microsoft Stream

The sections below describe the components used in the Multicast+ solution, and provide a functional overview how Multicast+ is used with Microsoft Stream.

Component Overview

The following components are used to process Microsoft Stream events:

Item	Description
Program Guide	The Program Guide is a Multicast+ component that lists entries for video events currently available in your environment. Program Guide entries are sent from Multicast+ Senders, and Multicast+ Receivers listen for entry updates. When a Multicast+ Receiver has a request for a video event that matches a Program Guide entry, the Receiver uses information from the entry to start receiving the video event.
Altimeter Management Server	The Altimeter Management Server is used to: <ul style="list-style-type: none">• Create a Multicast+ event that associates the Video ID (VID) of a Microsoft Stream event with a multicast address/port.• Retrieve Microsoft Stream signature keys for JWT token validation.
Multicast+ Sender	The Multicast+ Sender is used to: <ul style="list-style-type: none">• Update the Program Guide when a new video event is available.• Add new processing information to the video event's Program Guide entry as it becomes available.• Encapsulate the video event segments for processing by Multicast+ Receivers.
Multicast+ Receiver	The Multicast+ Receiver listens for Program Guide entry updates, and when a request matches a Program Guide entry, the Receiver: <ul style="list-style-type: none">• Starts receiving on a multicast group using information from the Program Guide entry.• Receives the encapsulated video event segments from the Sender.• Processes and buffers the segments so they are available for the video player.
Ramp Plugin for Azure Media Player	The Ramp Plugin for Azure Media Player (Ramp Plugin) resides on the Microsoft Server, and is loaded into the browser of the local machine. The Ramp Plugin handles Stream/AltitudeCDN communications, and manages the general processing behavior for live events and VOD streaming.

Functional Overview

The sections below describe how Multicast+ is used with Microsoft Stream.

Event Creation

To use Multicast+ with Microsoft Stream events, two steps are needed:

- A Microsoft Stream event must be created, which generates a Video ID (VID) that identifies the event. is used .
- A Multicast+ event must be created in the Altimeter Management Server to associate the VID with a multicast address/port.

Pre-Event Preparation

Prior to the time users are expected to join the event, an administrator does the following:

- Starts the Microsoft Stream event.
- Starts the Multicast+ event in the Altimeter Management Server, which prompts the Sender to update the Program Guide with the event's VID and multicast address/port.
- Using a machine with an installed Multicast+ Receiver, connects to the Microsoft Stream event from the standard user webpage. The following occurs:
 - Microsoft Stream passes a signed JWT token through the Ramp Plugin to the Multicast+ Receiver. The JWT token contains the VID and video manifest URI for the event.
 - The Receiver validates the authenticity of the JWT token and checks the Program Guide for a corresponding VID. Once a matching VID is found:
 - The Receiver sends the JWT token to the Altimeter Management Server listed in the Program Guide.
 - The Management Server updates the Sender with the video manifest URI for the VID.
 - The Sender updates the Program Guide with the video manifest URI, and starts retrieving segments for the event.
 - The Ramp Plugin:
 - Checks the Program Guide and sees that the event has been updated with the video manifest URI.
 - Instructs the Receiver to use the multicast address/port associated with the VID from the Program Guide to begin receiving video segments.
 - If Multicast+ is not available, the Ramp Plugin immediately proceeds to its next configured option, such as using OmniCache (if configured), or reverting to unicast.

User Connection to the Event

When users access the event, the following occurs:

- Microsoft Stream passes a JWT token to the Ramp Plugin.
- The Ramp Plugin checks the Program Guide and confirms that a matching VID and video manifest URI exist.

- The Ramp Plugin instructs the Receiver to use the multicast address/port associated with the VID from the Program Guide to begin receiving video segments.

Using OmniCache with Microsoft Stream

The sections below describe the components used in the OmniCache solution, and provide a functional overview how OmniCache is used with Microsoft Stream.

Component Overview

The following components are used to process Microsoft Stream events:

Item	Description
OmniCache	OmniCache is used to minimize how often video assets are retrieved from external CDN servers, and distributes shared content to enterprise WAN clients.
Ramp Plugin for Azure Media Player	The Ramp Plugin for Azure Media Player (Ramp Plugin) resides on the Microsoft Server, and is loaded into the browser of the local machine. The Ramp Plugin handles Stream/AltitudeCDN communications, and manages the general processing behavior for live events and VOD streaming.

Functional Overview

The section below describes how OmniCache is used with Microsoft Stream.

Ramp Plugin Operation

Once configured for use with OmniCache, the Ramp Plugin:

- Upon startup, attempts to connect to OmniCache using a specified URL.
- If OmniCache is available, all subsequent requests are rewritten to pass them to OmniCache.
- Upon receiving requests, OmniCache checks to see if the requested resource exists in cache:
 - If the resource is available, OmniCache serves the request.
 - If not available, OmniCache makes an upstream retrieval request and caches the resource for subsequent requests.
- If OmniCache is not available, the Ramp Plugin immediately proceeds to its next configured option, such as making direct requests.

Configuring the Multicast+ and OmniCache Integration with Microsoft Stream

This section describes how to configure various AltitudeCDN components to work with Microsoft Stream.

Configuration Summary

To use AltitudeCDN with Microsoft Stream, perform the actions summarized below:

Using Multicast+ with Microsoft Stream

To use Multicast+ with Microsoft Stream, you need to do the following:

- [Activate the Program Guide.](#)
- [Configure Microsoft Stream key retrieval.](#)
- [Configure the Ramp Plugin for Azure Media Player.](#)
- [Create the Microsoft Stream event.](#)
- [Create the Multicast+ event.](#)

Using OmniCache with Microsoft Stream

To use OmniCache with Microsoft Stream, you need to do the following:

- [Configure the Ramp Plugin for Azure Media Player.](#)

Activating the Program Guide

(Multicast+ only) Multicast+ Senders and Receivers query DNS servers for Program Guide DNS entries that define the multicast group and port used by the Program Guide:

- Once the DNS entries are retrieved, Multicast+ Senders send video event updates to the Program Guide, which maintains a list of video events available in your environment. Multicast+ Receivers listen for Program Guide entry updates, and use entry information to begin receiving an event.
- If the DNS entries are not found, the Program Guide is not used. However, Multicast+ Senders and Receivers continue to perform DNS lookups even if the Program Guide is not used.

To create the Program Guide DNS entries, see *Activating the Program Guide* in the *AltitudeCDN Multicast+ Deployment Guide*.

Configuring Microsoft Stream Signature Key Retrieval

(Multicast+ only) To validate the JWT tokens used by Microsoft Stream, the Altimeter Management Server must be configured to retrieve Microsoft Stream signature keys by doing the following:

Note: The retrieval of Microsoft Stream signature keys may require configuring an output HTTP proxy for the Altimeter Management Server, using Java properties. For more information, see [Operational Notes](#).

1. Within the Altimeter Management Server, click Multicast+ > Settings to display the Settings tab.
2. For Third Party Integration > Microsoft Stream Key Retrieval URI(s), click the edit icon. The following appears:

Setting Close

Name: Microsoft Stream Key Retrieval URI(s)

Value: `https://api.microsoftstream.com/api/public/keys?api-version=1.0`
`https://refe.api.stream.azure-test.net/api/public/keys?api-version=1.0`

The URI for retrieving Microsoft Stream signer keys. A blank URI disables Stream automatic video id URI discovery, or multiple lines for multiple URIs.

The production URI is <https://api.microsoftstream.com/api/public/keys?api-version=1.0> [\[Click to set to this value\]](#)
If you have been provided a custom URI by Microsoft or Ramp, please use that instead.

Note that setting this field will cause Altimeter to access external network resources.

Save

3. Enter one or more URIs to be used for retrieving Microsoft Stream signature keys. For example:
`https://api.microsoftstream.com/api/public/keys?api-version=1.0`

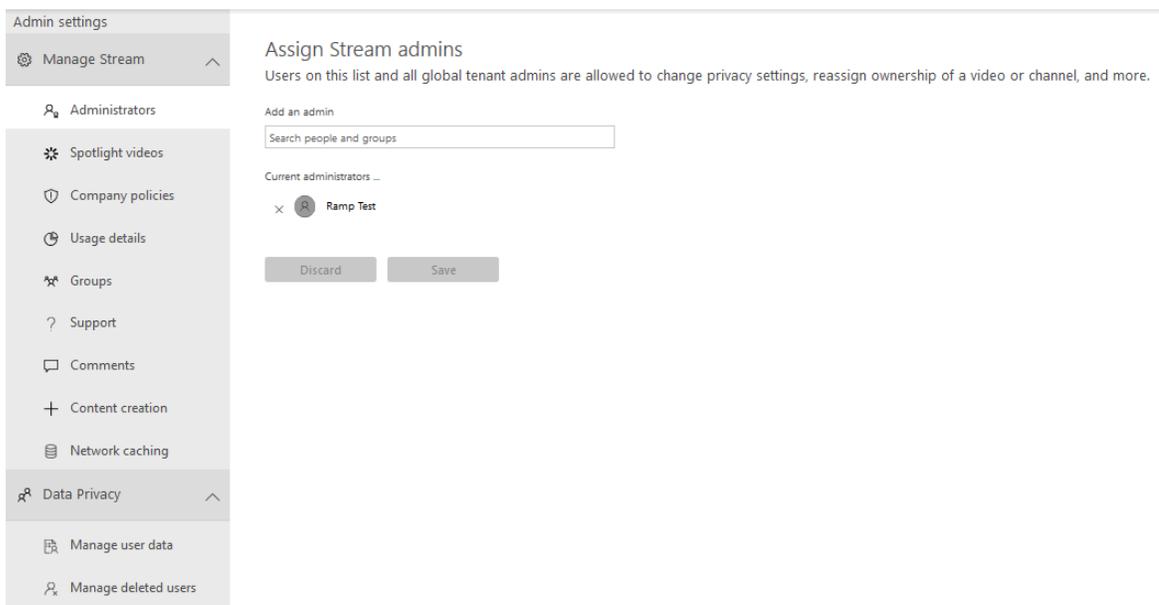
Note: A Microsoft stream signature key retrieval URI is required, otherwise, support for Microsoft Stream events is disabled.

4. Click Save, then click Close.

Configuring the Ramp Plugin for Azure Media Player

(Multicast+ and OmniCache) To work with AltitudeCDN, Microsoft Stream must be configured to use the Ramp Plugin for Azure Media Player by doing the following:

1. Log into Microsoft Stream using an account with administrative privileges.
2. From the Microsoft Stream welcome page, click the gear icon on the upper-right side of the page to display the Settings menu.
3. Select Admin settings. The following appears:



The screenshot shows the Microsoft Stream Admin settings interface. On the left is a navigation menu with categories: Admin settings (Manage Stream), Administrators, Spotlight videos, Company policies, Usage details, Groups, Support, Comments, Content creation, Network caching, and Data Privacy (Manage user data, Manage deleted users). The main content area is titled 'Assign Stream admins' and includes a search box for adding admins, a list of current administrators (Ramp Test), and 'Discard' and 'Save' buttons.

4. From the Admin settings menu, select Network caching. The following appears:

Enable a third-party network caching provider

Microsoft Stream works with third-party caching providers that offer software defined networks (SDNs) to help optimize video playback network traffic within your organization.

To begin, license and setup your chosen SDN solution subject to the selected third-party provider's [terms of service and privacy policy](#), which will govern your use of the SDN provider's solution. Your use of the SDN provider's solution will not be subject to the Microsoft volume licensing terms or Online Services Terms. If you do not agree to the third-party provider's terms, then do not enable the SDN solution in Microsoft Stream. Once you deploy your chosen SDN solution, enable the following settings to start using it within Microsoft Stream. [Learn more](#)

Stream tenant ID: 453694b6-f851-4d2a-96ed-f0784091eb7f

Enable 3rd party network caching provider

On

Choose a caching provider

Ramp

Customer ID

ThisIsTheCustomerID

Analytics report URL

Configuration settings

```
{
  "params": {
    "maddr": "0",
    "receiverHost": "ramp-demo.multicast-receiver.altitudecdn.net",
    "omnicache": "cache.ramp.com:10010"
  }
}
```

Discard Save

Verify setup

5. Within the “Enable a third-party network caching provider” dialog, do the following:

Setting	Value
Set 3rd party network caching provider	Set the slider to “On”.
Choose a caching provider	Select “Ramp” from the drop-down list.
Customer ID	Not used.
Analytics report URL	Not used.
Configuration settings	<p>Specify the configuration settings to be used for the Ramp Plugin.</p> <p>Note: Additional configuration parameters can be included as directed by Ramp Support.</p> <p>Parameters for Multicast+ Support:</p> <p>At minimum, the following is required:</p> <pre>{ "params": { "maddr": "0", "receiverHost": "ramp-demo.multicast-receiver-altitudecdn.net", } }</pre> <p>Where:</p> <ul style="list-style-type: none"> maddr – Indicates the multicast address/port of the multicast channel. <p>The special setting “0” directs the Receiver to obtain the multicast address/port from the Program Guide.</p> <ul style="list-style-type: none"> receiverHost – Enter a valid certificate-backed secure host name for HTTPS communications. For example: <pre>ramp-demo.multicast-receiver-altitudecdn.net</pre> <p>Parameters for OmniCache Support:</p> <p>At minimum, the following is required:</p> <pre>{ "params": { "omnicache": "<omnicache host>[:<port>]", } }</pre> <p>Where:</p> <ul style="list-style-type: none"> omnicache – Specifies the URL used to connect to OmniCache. For example: <pre>cache.acme.com:10010</pre>

6. Click Save.

Creating the Microsoft Stream Event

(Multicast+ only) To use Multicast+ with a Microsoft Stream event, you need to obtain the Video ID (VID) that identifies the Stream event by doing the following:

Note: These instructions summarize how to create and start a Microsoft Stream event. For more information, see the Microsoft documentation.

1. Configure an encoder for live streaming with Microsoft Stream.
2. Within Microsoft Stream, create and publish a live event.

When the Microsoft Stream event is published, the URI changes at the top of the page, and includes the assigned VID for the event. For example:

```
https://acme.com/create?vid=94e3d457-afd1-4a40-8b2f-93e117e2aeb6
```

3. Copy the highlighted portion of the URI to obtain the VID. This value is used when the Multicast+ event is created in the Altimeter Management Server.
4. Start the Pre-live event as directed by the instructions displayed on the Microsoft Stream page.

Creating the Multicast+ Event

(Multicast+ only) To create the Multicast+ event, do the following:

1. Within the Altimeter Management Server, create a Multicast+ event. For more information, see the *AltitudeCDN Altimeter Management Server User Guide for Multicast+*.
2. Within the Details dialog for the Multicast+ event, do the following:
 - Paste the VID that was copied as part of [Creating the Microsoft Stream Event](#) into the Content Server URL field.
 - Add "mstream://" as a prefix to the VID. For example:
mstream://94e3d457-afd1-4a40-8b2f-93e117e2aeb6
 - Select the DASH as the Source Format from the drop-down list.
 - Select Live from the Source Type drop-down list.
 - Click Save.
3. Start the Multicast+ event.
4. Using a machine with an installed Multicast+ Receiver, connect to the Microsoft Stream event from the standard user webpage. This action retrieves the video manifest URI, updates the Program Guide, and allows subsequent users to quickly access the event.

Operational Notes

Once you have deployed the Multicast+ and OmniCache integration with Microsoft Stream, note the following:

- For convenience of future integration code, the Receiver triggered 404 message with body “Not found but receiver triggered: <maddr>” will now also be accompanied by an explicit HTTP response header “X-RampMulticastPlus-ReceiverTriggeredMulticastAddress” with the maddr string.
- The Multicast+ support for Microsoft Stream does not currently support HTTP proxy, but provides support for SOCKS proxy. For example:
 - (-DsocksProxyHost=\${proxyHost} -DsocksProxyPort=\${proxyPort})