

MPlatform SDK

Release notes

2.12.0.14909 VERSION
August 2026

This document lists MPlatform SDK changes. Each release-note item includes a concise description of the affected workflow, the observable symptom, and the result.

25 ADDED	89 FIXED	3 UPDATED	0 CHANGED
--------------------	--------------------	---------------------	---------------------

DOCUMENT STRUCTURE

What is inside

Common Changes - 35 release-note items
File Recording and Network Streaming - 25 release-note items
Input/Output Devices - 11 release-note items
Decoding and Seeking - 21 release-note items
Plugins - 22 release-note items
Samples and Documentation - 3 release-note items

RELEASE OVERVIEW

MPlatform SDK - Version 2.12.0.14909

Added capabilities are listed first in each section. Select any item to jump to its detailed explanation.

Common Changes

ADDED	New GPU shader transitions based on gl-transitions.
ADDED	LTC reading from a selected embedded audio channel.
ADDED	xMedia Converter statistics.
ADDED	Effect-specific parameters for GPU mixer transitions.
ADDED	Runtime enumeration of GPU transitions and their default parameters.
ADDED	Configurable aspect ratio for Unreal Engine xMedia output.
ADDED	Independent in/out phases for GPU color-fade playlist transitions.
FIXED	Crash when 24-bit audio conversion is used together with audio channel remapping.
FIXED	MMixer background transitions did not fade audio.
FIXED	Dropped frames after CG commands during playlist playback.
FIXED	Duplicated frames in rotate + maintain_ar + gpu_pipeline scenarios.
FIXED	object::vfilter and object::afilter processing in gpu_pipeline mode.
FIXED	MDelay custom metadata reading from SBE files.
FIXED	Text outline rendering on small ARGB frames.
FIXED	Intermittent mixer audio when sync_mode=false.
FIXED	Audio distortion during normalization for specific source files.
FIXED	Frame cloning changing nRowBytes in gpu_pipeline mode.
FIXED	Mixer output pixel format being reset from HDYC to ARGB by background frames.
FIXED	Mixer solid-color background scaling after output format changes.
FIXED	MDelay properties set from code were not applied.
FIXED	external_process behavior consistency between code and registry configuration.
FIXED	pixelize video filter processing for ARGB formats.

FIXED	Audio-normalization properties passed through PlaylistAdd parameters.
FIXED	Audio noise during MFile normalization.
FIXED	TrackLoudnessGet results after applying audio normalization to MPlaylist.
FIXED	Repeated EOF events from finite PNG mixer sources after a GPU transition.
FIXED	Reusing writer format configuration after AttributesRemoveAll.
FIXED	FrameConvert color conversion for NV12, RGB8, YV12, and YV16 frames.
FIXED	MWriter state after ObjectClose following an encoder error.
FIXED	Excessive GPU load during mixer transitions.
FIXED	Video flicker after transitions in nested mixer workflows.
FIXED	Flicker during GPU mixer-stream transitions.
FIXED	Fast transcoding with rate_control=false in GPU and DX11 preview paths.
FIXED	MMixer crash while reading mask element attributes.
FIXED	Unreal Engine xMedia video-callback pacing.

File Recording and Network Streaming

ADDED	DVB-compatible MPEG-TS streaming over RTP.
ADDED	WebVTT subtitle support for HLS streams.
ADDED	DVB subtitle generation and insertion into DVB streams.
ADDED	Teletext support for EBU Teletext / ETSI EN 300 706 workflows.
ADDED	Selective seamless output switching for multi-output writers.
ADDED	Subtitle encoder and rasterizer formatting options with automatic canvas sizing.
ADDED	Multilingual DVB Teletext encoding, including Cyrillic character sets.
ADDED	SCTE-35 detection from HLS playlist tags.
FIXED	DVB subtitle packet passthrough after seeking during transwrapping, so subtitle PIDs remain available on receiver outputs.
FIXED	DVB subtitle side-data loss through MPLink chains.
FIXED	MWriter MXF timecode gaps when splitting output by max_duration.
FIXED	MWriter indexed audio PID settings for additional audio streams.

FIXED	Duplicated AC-3 descriptor in DVB PMT output.
FIXED	MXF pre-charge metadata loss during transwrapping.
FIXED	SRT streams with subtitle::codec=packets opening in VLC.
FIXED	Audio packet passthrough when source stream indices contain gaps.
FIXED	Timecode metadata in MP4 recordings produced by the NVIDIA encoder.
FIXED	setsar/setdar filter metadata in MOV and ProRes output.
FIXED	MXF finalization when MWriter records MMixer output.
FIXED	DVB subtitle packet timing after skipping initial source frames.
FIXED	DVB output when one of several external subtitle files is missing.
FIXED	DVB video continuity when MPlaylist switches to an item with SRT subtitles.
FIXED	DVB subtitle and Teletext receiver freeze when switching playlist items.
FIXED	DVB muxing stopping during long MPlaylist output with external subtitles.
FIXED	messageSize in single-operation SCTE-104 ANC packets.

Input/Output Devices

ADDED	ASIO device support as an LTC synchronization source.
ADDED	AJA LTC timecode input source option.
FIXED	DeckLink key/fill processing at frame rates above 30 fps.
FIXED	DeckLink custom ANC packet reception.
FIXED	Bluefish Kronos K8 audio loss in key/fill output modes.
FIXED	Long shutdown time for Blackmagic/DeckLink renderers.
FIXED	SDI auto-detect reinitialization when a 59.94p signal appears after no-signal startup.
FIXED	Audio stutter caused by missing BMD no-signal frame timing in live/render pipelines.
FIXED	NDI color metadata propagation when the source changes.
FIXED	HDMI DeckLink Mini Recorder 4K initialization crash in gpu_pipeline mode.
FIXED	Renderer frame-rate drops while seeking converted MPlaylist output.

Decoding and Seeking

ADDED	video.auto_rotation property for automatic rotation-flag handling in MFile.
ADDED	MXF SMPTE RP 188 and SMPTE 385M timecode support.
ADDED	Looping for MFile external-audio tracks with continuous timestamps.
FIXED	Delay when adding short play-while-recording files to a playlist.
FIXED	MMixer could stop after repeated switching between MFile sources.
FIXED	MMixer preview did not refresh after seeking a background MFile.
FIXED	Delayed application of MPlaylist item scale_type settings.
FIXED	MPlaylist position increasing during accelerated reverse playback.
FIXED	Playback-rate changes for mp:// background direct_out sources in MMixer.
FIXED	start_timecode metadata reading for GXF files.
FIXED	pause_out behavior during accelerated playback.
FIXED	mp:// links when FileRateSet exceeds max_forward_rate.
FIXED	MMixer seeking for interlaced files with gpu_pipeline enabled.
FIXED	playlist.on_next=pause_out behavior for live playlist items.
FIXED	Audio stopping after the opening seconds for specific MP4 files in MFile and MPlaylist playback.
FIXED	Audio loss when an mp:// playlist item's out point exceeded its duration.
FIXED	Scan-type detection for 10-bit JPEG 2000 files.
FIXED	Rare MPlaylist deadlock while adding break items.
FIXED	Timed MPlaylist stop commands in GPU-pipeline mode.
FIXED	PlaylistPosGet position calculation during accelerated playback.
FIXED	PlaylistPosGet next-index consistency after command-item removal.

Plugins

ADDED	SMPTE 436M CC, V-Chip, and AFD metadata support.
ADDED	cc.display_order option to choose CEA-608 processing order from packet-level or decoded-frame data.
ADDED	V-Chip/XDS insertion and parsing support.

ADDED	HTML5 MLCG editor.
FIXED	CG overlay color rendering when overlay.thread=0.
FIXED	CG overlay rendering for interlaced formats when overlay.thread=0.
FIXED	Extra spacing between adjacent CG text runs in tickers.
FIXED	CG plugin forcing ARGB32 instead of the configured HDYC format in playlist pipelines.
FIXED	CEF opening an external browser when HTML overlay is used from two applications at the same time.
FIXED	CG RSS ticker text width and spacing mismatch.
FIXED	Display of closed captions written to MXF files.
FIXED	Generated CEA-608 closed captions in files for VLC playback.
FIXED	Blurred CG text when decoder.convert_planar=true.
FIXED	MCCDisplay CEA-608 character ordering for specific files.
FIXED	CEF HTML plugin and WebCapture page loading on first start after OS boot.
FIXED	CEF audio mute and preview-control handling.
FIXED	CEF popup handling that could create an extra preview/window for new tabs.
UPDATED	CEF to version 147.0.11.
FIXED	L-shape resizing with external-process CG and runtime GPU-pipeline activation.
FIXED	Smooth scheduled appearance for CG items shown with cg-props::show.
FIXED	CG hang when hiding a ticker containing a non-RSS web page.
FIXED	Visible MPlaylist preview freeze while an HTML5 overlay loads.

Samples and Documentation

ADDED	MLCG REST control sample for MPlatform workflows.
UPDATED	MPlatform MLShader samples with the new shader-library project and shader catalog.
UPDATED	Subtitle workflows split into dedicated Conversion and Generation samples.

TECHNICAL DETAILS

MPlatform SDK release details

Descriptions below summarize the affected SDK workflow and result. Internal tracker references and raw issue text are intentionally omitted.

Common Changes

ADDED**New GPU shader transitions based on gl-transitions.**

APPLIES TO MPlatform

Applications using GPU transitions had a limited built-in transition set and could not directly use the newer gl-transitions style effects from the SDK package.

RESULT

The SDK includes the new GPU shader transition set.

ADDED**LTC reading from a selected embedded audio channel.**

APPLIES TO MPlatform

Applications receiving LTC embedded in a multi-channel audio signal needed to choose the exact audio channel used as the LTC source. Without that selection, synchronization could read the wrong channel or fail to lock.

RESULT

LTC can be read from a selected embedded audio channel.

EXAMPLE - Select an embedded audio channel for LTC

```
live.PropsSet("ltc.enabled", "true");
live.PropsSet("ltc_audio_device", "<input device>");
live.PropsSet("ltc_audio_channel", "1");
live.ObjectStart(true);
// Read timecode after the LTC source locks.
```

ADDED**xMedia Converter statistics.**

APPLIES TO MPlatform

Applications using xMedia Converter could not read the conversion statistics needed for monitoring, diagnostics, or UI status.

RESULT

xMedia Converter now exposes statistics through the SDK workflow.

ADDED**Effect-specific parameters for GPU mixer transitions.**

APPLIES TO MPlatform

GPU-pipeline mixer transitions ignored effect-specific options, so applications could not configure details such as the number or direction of Blinds bands.

RESULT

GPU mixer transitions now accept their documented effect-specific parameters.

ADDED**Runtime enumeration of GPU transitions and their default parameters.**

APPLIES TO MPlatform

Applications had to hardcode transition names and parameters to build transition pickers, which became error-prone as the GPU transition library expanded.

RESULT The SDK factory now exposes TransitionGetCount and TransitionGetByIndex.

EXAMPLE - Enumerate MPlatform GPU transitions

```
MFactoryClass factory = new MFactoryClass();
factory.TransitionGetCount(out int count);
for (int index = 0; index < count; index++)
{
    factory.TransitionGetByIndex(index, out string name, out string defaultParams);
    AddTransitionToUi(name, defaultParams);
}
```

ADDED

Configurable aspect ratio for Unreal Engine xMedia output.

APPLIES TO MPlatform

xMedia output created by the Unreal Engine integration always advertised a 4:3 aspect ratio, even when the rendered content used a different display shape.

RESULT Unreal Engine xMedia output can now specify its aspect ratio.

ADDED

Independent in/out phases for GPU color-fade playlist transitions.

APPLIES TO MPlatform

A fade through color always performed both the outgoing and incoming phases, so a playlist could not fade only the old item out or only the new item in.

RESULT fade(mode='color') accepts colorPhaseMode='both', 'out', or 'in' while preserving the existing default behavior.

EXAMPLE - Configure an outgoing-only color fade

```
item.PropsSet("transition_out::type",
    "fade(mode='color' colorPhaseMode='out')");
item.PropsSet("transition_out::time", "2.0");
// colorPhaseMode can be 'both' (default), 'out', or 'in'.
```

FIXED

Crash when 24-bit audio conversion is used together with audio channel remapping.

APPLIES TO MPlatform

24-bit audio conversion combined with explicit audio channel remapping could access the audio buffer incorrectly and crash.

RESULT The 24-bit audio conversion path now handles channel mapping safely.

FIXED

MMixer background transitions did not fade audio.

APPLIES TO MPlatform

When an MMixer background source was changed with transition=fade, the picture faded correctly but audio from the new source appeared immediately at full level instead of crossfading.

RESULT MMixer background fade transitions now fade the corresponding audio.

FIXED

Dropped frames after CG commands during playlist playback.

APPLIES TO MPlatform

Executing a CG command during playlist playback could introduce a visible timestamp gap or dropped frame around the command or item boundary.

RESULT Playlist playback remains frame-continuous when CG commands are executed.

FIXED**Duplicated frames in rotate + maintain_ar + gpu_pipeline scenarios.**

APPLIES TO MPlatform

Combining rotation, maintain_ar, and gpu_pipeline could produce duplicated output frames in this processing path.

RESULT The rotate + maintain_ar GPU path now preserves correct frame delivery without duplicates.

FIXED**object::vfilter and object::afilter processing in gpu_pipeline mode.**

APPLIES TO MPlatform

Object-level video and audio filters could be skipped or processed incorrectly when gpu_pipeline mode was enabled.

RESULT object::vfilter and object::afilter are now applied correctly in gpu_pipeline mode.

FIXED**MDelay custom metadata reading from SBE files.**

APPLIES TO MPlatform

Custom frame metadata written before storing media into an MDelay SBE file was not available when that SBE file was opened later.

RESULT Custom frame metadata is preserved and can be read back from MDelay SBE files.

FIXED**Text outline rendering on small ARGB frames.**

APPLIES TO MPlatform

Text outlines could render incorrectly on small ARGB frames.

RESULT Text outline rendering is now correct on small ARGB frames.

FIXED**Intermittent mixer audio when sync_mode=false.**

APPLIES TO MPlatform

MMixer with sync_mode=false could produce intermittent or broken audio after a file source was added.

RESULT MMixer audio remains continuous when sync_mode=false is used in this scenario.

FIXED**Audio distortion during normalization for specific source files.**

APPLIES TO MPlatform

Audio gain normalization could create a loud distorted transient for some sources, especially after an initial silent part of the file.

RESULT Audio normalization handles this source pattern without producing the distorted transient.

FIXED **Frame cloning changing nRowBytes in gpu_pipeline mode.**

APPLIES TO MPlatform

Cloning a frame in gpu_pipeline mode could change nRowBytes and expose an unexpected row stride to downstream processing.

RESULT Frame cloning now preserves the expected nRowBytes value in gpu_pipeline mode.

FIXED **Mixer output pixel format being reset from HDYC to ARGB by background frames.**

APPLIES TO MPlatform

Mixer output configured for HDYC could be reset to ARGB by background frames, changing the pixel format seen by downstream processing or output devices.

RESULT Mixer output keeps the configured pixel format when background frames are processed.

FIXED **Mixer solid-color background scaling after output format changes.**

APPLIES TO MPlatform

Changing mixer background color and output format could leave the solid-color background scaled incorrectly when returning to a 4:3 format.

RESULT Mixer solid-color backgrounds scale correctly after output format changes.

FIXED **MDelay properties set from code were not applied.**

APPLIES TO MPlatform

MDelay properties set through PropsSet from code, such as video codec or default extension options, could be ignored even though the same settings worked from registry defaults.

RESULT MDelay applies the same properties when they are set from code.

FIXED **external_process behavior consistency between code and registry configuration.**

APPLIES TO MPlatform

external_process could behave differently depending on whether it was configured from code or through default/registry settings.

RESULT external_process behavior is now consistent across configuration paths.

FIXED **pixelize video filter processing for ARGB formats.**

APPLIES TO MPlatform

The pixelize video filter could produce incorrect output when processing ARGB frames.

RESULT The pixelize filter now processes ARGB formats correctly.

FIXED**Audio-normalization properties passed through PlaylistAdd parameters.**

APPLIES TO MPlatform

Audio-normalization properties passed in the MPlaylist.PlaylistAdd parameter string could fail to reach the created MFile, forcing applications to set those properties only after the item was added.

RESULT

Audio-normalization properties passed through PlaylistAdd are applied to the created source.

EXAMPLE - Passing normalization options while adding a playlist item

```
playlist.PlaylistAdd(  
    null,  
    sourcePath,  
    "object::audio.gain_limiter=true object::audio.gain_normalization=true",  
    ref itemIndex,  
    out itemFile);
```

FIXED**Audio noise during MFile normalization.**

APPLIES TO MPlatform

MFile audio normalization could generate audible noise for affected source files.

RESULT

MFile normalization handles these files without introducing the noise.

FIXED**TrackLoudnessGet results after applying audio normalization to MPlaylist.**

APPLIES TO MPlatform

MPlaylist audio normalization changed the output level, but TrackLoudnessGet continued to return the pre-normalization VU, peak, and RMS values.

RESULT

TrackLoudnessGet now reports the normalized playlist output levels.

FIXED**Repeated EOF events from finite PNG mixer sources after a GPU transition.**

APPLIES TO MPlatform

A finite-duration PNG used as a mixer source could emit a continuous series of EOF events after transitioning to a transparent MFile source with GPU decoding enabled.

RESULT

Finite PNG sources now stop their image worker at EOF and emit a single completion event.

FIXED**Reusing writer format configuration after AttributesRemoveAll.**

APPLIES TO MPlatform

After removing every attribute from a writer format configuration, selecting the same format again could fail until the writer object was recreated.

RESULT

AttributesRemoveAll restores the writer configuration defaults, allowing repeated ConfigSet calls without recreating the writer.

FIXED**FrameConvert color conversion for NV12, RGB8, YV12, and YV16 frames.**

APPLIES TO MPlatform

MPlatform FrameConvert could swap or misinterpret color channels when converting to NV12, RGB8, YV12, or YV16, producing visibly incorrect colors.

RESULT

FrameConvert now produces correctly ordered color data for these pixel formats.

FIXED**MWriter state after ObjectClose following an encoder error.**

APPLIES TO MPlatform

If MWriter entered eMS_Error, a successful ObjectClose could leave ObjectStateGet reporting eMS_Error instead of eMS_Closed.

RESULT

A successfully closed writer now reports eMS_Closed even when its previous state was an error.

FIXED**Excessive GPU load during mixer transitions.**

APPLIES TO MPlatform

Mixer transitions between SDI sources could cause a sharp GPU-load increase and visible stalls for specific interlaced input/output frame-rate combinations.

RESULT

GPU transition processing was optimized for the affected mixer format combinations.

FIXED**Video flicker after transitions in nested mixer workflows.**

APPLIES TO MPlatform

A mixer containing another mixer and playlist could begin flickering after a transition when source and output frame rates differed.

RESULT

Nested mixer transitions now maintain stable frame delivery across differing frame rates.

FIXED**Flicker during GPU mixer-stream transitions.**

APPLIES TO MPlatform

Starting transitions for multiple mixer sources with gpu_pipeline=true could inherit an old alpha-blend state and cause visible flicker.

RESULT

GPU transitions now reset blend state before rendering each transition.

FIXED**Fast transcoding with rate_control=false in GPU and DX11 preview paths.**

APPLIES TO MPlatform

Disabling rate control still limited a playlist/encoder pipeline to real time when gpu_pipeline or DX11 preview was enabled.

RESULT

The GPU/DX11 preview path now skips pacing waits when rate_control=false.

FIXED**MMixer crash while reading mask element attributes.**

APPLIES TO MPlatform

Reading the mask element through `ElementStringGet` could terminate the process because the mixer stream calculated an invalid row stride for non-RGB frames.

RESULT

Mixer mask-frame properties and row stride are now initialized safely for every supported format.

FIXED**Unreal Engine xMedia video-callback pacing.**

APPLIES TO MPlatform

Video callbacks delivered to Unreal Engine could arrive in bursts with long and short intervals even though the average frame rate remained correct, forcing applications to add extra buffering.

RESULT

xMedia callback delivery is paced more consistently for Unreal Engine sources.

File Recording and Network Streaming

ADDED**DVB-compatible MPEG-TS streaming over RTP.**

APPLIES TO MPlatform

Applications that needed DVB-compatible MPEG-TS delivery over RTP had no direct SDK-supported output path for that transport combination.

RESULT

DVB-compatible MPEG-TS streaming over RTP is now supported.

ADDED**WebVTT subtitle support for HLS streams.**

APPLIES TO MPlatform

Applications that generate HLS output can now include a WebVTT subtitle track. This is useful when an external subtitle file, such as ASS or SRT, must be passed through the reader and written as .vtt files next to the HLS playlist.

RESULT

HLS output with WebVTT subtitles is supported.

EXAMPLE - MPlatform HLS/WebVTT connected-object setup

```
string readerConfig =
    "external_process=false " +
    "subtitle_track=0 " +
    "experimental.out_subtitle_packets=2 " +
    $"external_subtitle='{subtitlePath}'";
string writerConfig =
    "format='hls' mux_buffers=0 hls_time=2 hls_list_size=0 " +
    "hls_playlist_type='vod' subtitle::codec='webvtt' " +
    "subtitle::metadata::language='eng' video::codec='libopenh264' " +
    "video::maxrate='5M' audio::codec='aac';
file.FileNameSet(sourcePath, readerConfig);
file.FilePlayStart();
writer.WriterNameSet(outputPlaylist, writerConfig);
writer.ObjectStart(file);
// After writing, read generated *.vtt files and check for expected cue text.
```

ADDED**DVB subtitle generation and insertion into DVB streams.**

APPLIES TO MPlatform

Developers implementing DVB subtitle insertion in MPlatform did not have a complete SDK sample showing the workflow.

RESULT A MPlatform DVB subtitle insertion sample was added.

EXAMPLE - External text subtitles to DVB output

```
string readerConfig =
    "external_process=false subtitle_track=0 " +
    "experimental.out_subtitle_packets=2 " +
    "$"external_subtitle='{subtitlePath}';
string writerConfig =
    "format='dwb' protocol='udp://' video::codec='n264' audio::codec='aac' " +
    "subtitle::codec='dvbtxt' subtitle::stream_id=400 " +
    "subtitle::metadata::language='eng' " +
    "subtitle.1::codec='dvbsub' subtitle.1::stream_id=401 " +
    "subtitle.1::metadata::language='eng';
```

ADDED

Teletext support for EBU Teletext / ETSI EN 300 706 workflows.

APPLIES TO MPlatform

Applications that need EBU Teletext / ETSI EN 300 706 subtitle or data workflows previously had no built-in SDK support for that Teletext output path.

RESULT Teletext support was added for EBU Teletext / ETSI EN 300 706 workflows.

ADDED

Selective seamless output switching for multi-output writers.

APPLIES TO MPlatform

Changing a writer destination to segment a local recording could also reconnect a parallel network output in a two-output writer configuration.

RESULT The seamless_switches option selects only the output indices that should move to the new destination, allowing other streams to continue uninterrupted.

EXAMPLE - Switch only the local MPlatform recording output

```
string writerConfig =
    "format='mp4' video::codec='libopenh264' audio::codec='aac' " +
    "$"stream.1::format='mpegts' stream.1::path='{udpUrl}';
writer.WriterNameSet(firstSegment, writerConfig);
writer.ObjectStart(source);
// Keep stream.1 running and switch only primary output index 0.
writer.WriterNameSet(secondSegment,
    "seamless_switch=true seamless_switches='0'");
```

ADDED

Subtitle encoder and rasterizer formatting options with automatic canvas sizing.

APPLIES TO MPlatform

Generated DVB subtitles and Teletext used a fixed internal option path, making it difficult to set fonts, colors, backgrounds, outlines, or a canvas matched to the output video.

RESULT Subtitle writer options now reach the encoder/rasterizer, and video width and height are supplied automatically as the subtitle canvas.

EXAMPLE - Configure generated DVB subtitle appearance

```
string writerConfig =
    "format='dwb' protocol='udp://' video::codec='libopenh264' audio::codec='aac' " +
    "subtitle::codec='dvbsub' subtitle::metadata::language='eng' " +
    "subtitle::font='Arial' subtitle::font_size=28 subtitle::font_color='FFFFFF' " +
    "subtitle::background=true subtitle::background_color='000000' " +
    "subtitle::background_alpha=128 subtitle::outline=true " +
    "subtitle::outline_color='000000' subtitle::outline_thickness=2";
```

ADDED**Multilingual DVB Teletext encoding, including Cyrillic character sets.**

APPLIES TO MPlatform

DVB Teletext output could replace or corrupt Cyrillic text and did not provide a way to select primary and secondary Teletext G0 character sets for mixed-language subtitle tracks.

RESULT DVB Teletext encoding supports language-aware and explicitly configured Latin/Cyrillic character-set combinations.

EXAMPLE - Select DVB Teletext character sets

```
string writerConfig =
    "format='mpegt' subtitle::codec='dvbtxt' " +
    "subtitle::metadata::language='mul' " +
    "subtitle::teletext_primary_charset='latin' " +
    "subtitle::teletext_secondary_charset='cyrillic_russian_bulgarian' " +
    "video::codec='libopenh264' audio::codec='aac';"
```

ADDED**SCTE-35 detection from HLS playlist tags.**

APPLIES TO MPlatform

HLS SCTE-35 markers carried in common playlist tags were not surfaced to the application, so it could not react to splice points described only by the m3u8 playlist.

RESULT The SDK detects CUE, SCTE35, SPLICEPOINT, OATCLS, and SCTE35 DATERANGE forms and attaches the marker to the first video frame of the related segment.

EXAMPLE - Receive HLS SCTE-35 events from MFile

```
file.OnEventSafe += (channelId, eventName, eventParam, eventObject) =>
{
    if (eventName == "SCTE-35")
        HandleScte35(eventParam);
};
file.FileNameSet(hlsUrl, "");
file.FilePlayStart();
```

FIXED**DVB subtitle packet passthrough after seeking during transwrapping, so subtitle PIDs remain available on receiver outputs.**

APPLIES TO MPlatform

Applications that transwrap DVB subtitle packets through an MPlaylist and seek the source before starting receiver outputs could lose subtitle descriptors or subtitle PES packets on one output while another output still displayed subtitles.

RESULT DVB subtitle descriptors, PIDs, PES packets, and display segments are preserved after seeking during packet passthrough transwrapping.

EXAMPLE - MPlaylist DVB subtitle passthrough after seek

```
source.FileNameSet(sourcePath,
    "subtitle_track=0,1 experimental.out_subtitle_packets=2 external_process=false");
playlist.PlaylistAdd(source, "", "", -1, out item);
playlist.FilePlayStart();
playlist.FilePosSet(900.0, 0.0);
writer.PropsSet("encoder::subtitle.init_without_data", "true");
writer.WriterNameSet(outputUrl,
    "format='dvb' protocol='udp://' muxrate='8M' video::codec='n264' " +
    "audio::codec='aac' subtitle::codec='packets' subtitle.1::codec='packets'");
writer.ObjectStart(playlist);
```

FIXED**DVB subtitle side-data loss through MPLink chains.**

APPLIES TO MPlatform

DVB subtitle packet side data could be lost when a stream passed through MPLink chains, especially when playlist or external-process boundaries were involved.

RESULT DVB subtitle side data is preserved through MPLink chains.

EXAMPLE - DVB subtitle packets through MPLink

```
sourcePlaylist.ObjectNameGet(out string sourceName);
linkedPlaylist.PlaylistAdd(
    null,
    "mp://" + sourceName,
    "experimental.out_subtitle_packets=2 subtitle_track=0",
    -1,
    out item);
writer.WriterNameSet(outputFile,
    "format='mpegs' video::codec='n264' audio::codec='aac' " +
    "subtitle::codec='packets'");
writer.ObjectStart(linkedPlaylist);
```

FIXED

MWriter MXF timecode gaps when splitting output by max_duration.

APPLIES TO MPlatform

Splitting MXF output with MWriter by max_duration could create gaps in timecode between generated segments.

RESULT MXF segments created with max_duration now preserve continuous timecode.

FIXED

MWriter indexed audio PID settings for additional audio streams.

APPLIES TO MPlatform

MPEG-TS outputs with split audio channels could ignore audio.1::pid, audio.2::pid, and later indexed PID settings. Only audio::pid was used, while other audio streams received automatically assigned sequential PIDs.

RESULT Indexed audio PID settings are applied to additional audio streams.

EXAMPLE - Explicit audio PIDs for split audio streams

```
string writerConfig =
    "format='mpegs' protocol='srt://' split_channels='1' " +
    "audio::pid=4112 audio.1::pid=4124 " +
    "audio.2::pid=4128 audio.3::pid=4160";
```

FIXED

Duplicated AC-3 descriptor in DVB PMT output.

APPLIES TO MPlatform

DVB PMT generation for AC-3 audio could write two AC-3 descriptors for the same elementary stream. Some analyzers or receivers could then read duplicated or conflicting AC-3 metadata.

RESULT DVB PMT output writes a single correct AC-3 descriptor for the stream.

FIXED

MXF pre-charge metadata loss during transwrapping.

APPLIES TO MPlatform

MXF transwrapping could lose pre-charge metadata, so the output file did not preserve the timing/metadata structure of the source around the beginning of the essence.

RESULT MXF transwrapping preserves pre-charge metadata.

EXAMPLE - MPlatform MXF packet transwrap preserving metadata

```
string writerConfig =  
    "format='mxf' embed_cc='true' video::codec='packets' audio::codec='packets'";  
file.FileNameSet(sourceMxf, "");  
file.FilePlayStart();  
writer.WriterNameSet(outputMxf, writerConfig);  
writer.ObjectStart(file);
```

FIXED**SRT streams with subtitle::codec=packets opening in VLC.**

APPLIES TO MPlatform

Applications that pass subtitle streams as packets could start the writer before the first subtitle packet arrived. In that case the stream could miss subtitle initialization metadata and fail to open correctly in receivers such as VLC.

RESULT

Packet subtitle streams can be initialized before the first subtitle data packet arrives.

EXAMPLE - Packet subtitle stream initialization

```
writer.PropsSet("encoder::subtitle.init_without_data", "true");  
string writerConfig =  
    "format='mpegs' protocol='srt://' video::codec='n264' audio::codec='aac' " +  
    "subtitle::codec='packets' subtitle::metadata::language='eng'";
```

FIXED**Audio packet passthrough when source stream indices contain gaps.**

APPLIES TO MPlatform

Using audio::codec='packets' could omit later audio tracks when the source contained a missing stream index, changing the expected track order or channel layout in DVB output.

RESULT

Available packet audio streams are now written consecutively while preserving their source order and channel count.

FIXED**Timecode metadata in MP4 recordings produced by the NVIDIA encoder.**

APPLIES TO MPlatform

MP4 recordings created with the NVIDIA encoder and start_timecode could omit timecode information for GOP frames, making dropped-frame analysis unreliable.

RESULT

NVIDIA-encoded MP4 output now preserves timecode metadata across GOP frames.

FIXED**setsar/setdar filter metadata in MOV and ProRes output.**

APPLIES TO MPlatform

setsar/setdar filters changed the processed picture but MOV/ProRes files could still be written without the resulting sample/display aspect-ratio metadata.

RESULT

Calculated SAR is now propagated to the codec parameters used by the MOV muxer.

FIXED**MXF finalization when MWriter records MMixer output.**

APPLIES TO MPlatform

Stopping an XDCAM HD422 MXF recording sourced from MMixer could leave the file open/incomplete with no footer partition even though the essence data was present.

RESULT An empty audio end-of-stream no longer aborts encoder shutdown, so the MXF muxer writes its final header and footer.

FIXED**DVB subtitle packet timing after skipping initial source frames.**

APPLIES TO MPlatform

When a transwrap workflow skipped several seconds before starting its output, video began at the new position but DVB subtitle packets retained original timestamps and appeared late.

RESULT Attached subtitle packet timestamps now follow the adjusted frame timeline.

FIXED**DVB output when one of several external subtitle files is missing.**

APPLIES TO MPlatform

If a playlist item referenced multiple language subtitle files and exactly one file was unavailable, the stream could stop or discard subtitle tracks from files that did exist.

RESULT Unavailable external subtitle files are skipped independently while valid tracks and the media stream continue.

FIXED**DVB video continuity when MPlaylist switches to an item with SRT subtitles.**

APPLIES TO MPlatform

A DVB UDP writer could remain in Running state but stop sending video after MPlaylist advanced to the next item with external SRT subtitle tracks.

RESULT Subtitle timing is clipped correctly at playlist boundaries, keeping video and the new item's subtitle streams active.

FIXED**DVB subtitle and Teletext receiver freeze when switching playlist items.**

APPLIES TO MPlatform

A receiving decoder could stop progressing when a DVB stream with generated subtitles or Teletext crossed an automatic MPlaylist item boundary.

RESULT DVB output remains decodable and continues advancing across playlist item switches.

FIXED**DVB muxing stopping during long MPlaylist output with external subtitles.**

APPLIES TO MPlatform

A DVB UDP writer could stay in Running state while muxer PTS values and network output stopped after several minutes of playlist playback with external SRT-to-DVB subtitle encoding.

RESULT Subtitle errors and unavailable sender streams no longer stop muxer progression or the DVB output.

FIXED**messageSize in single-operation SCTE-104 ANC packets.**

APPLIES TO MPlatform

Single-operation SCTE-104 packets counted the payload descriptor byte in messageSize, producing a value one byte larger than required by SCTE-104.

RESULT messageSize now describes only the SCTE-104 message after the payload descriptor.

Input/Output Devices

ADDED

ASIO device support as an LTC synchronization source.

APPLIES TO MPlatform

Applications that synchronize by LTC could not use an ASIO audio device as the LTC source. This mattered when the timecode reference was available only through an audio interface instead of a video device input.

RESULT ASIO devices can now be selected as LTC synchronization sources.

ADDED

AJA LTC timecode input source option.

APPLIES TO MPlatform

Applications using AJA hardware could not select the board LTC input as the timecode source directly from the SDK workflow.

RESULT AJA LTC input can now be selected as a timecode source.

FIXED

DeckLink key/fill processing at frame rates above 30 fps.

APPLIES TO MPlatform

DeckLink key/fill output could fail or drop performance for frame rates above 30 fps on affected key/fill hardware workflows.

RESULT DeckLink key/fill processing supports the affected high-frame-rate modes.

FIXED

DeckLink custom ANC packet reception.

APPLIES TO MPlatform

Custom ANC packets could be missed or received incorrectly in DeckLink capture workflows.

RESULT DeckLink custom ANC packet reception is now handled correctly.

FIXED

Bluefish Kronos K8 audio loss in key/fill output modes.

APPLIES TO MPlatform

Bluefish key/fill output could lose audio while preparing fill/key video frames.

RESULT Audio is preserved during Bluefish key/fill output processing.

FIXED

Long shutdown time for Blackmagic/DeckLink renderers.

APPLIES TO MPlatform

Blackmagic/DeckLink renderers could take an excessive amount of time to shut down.

RESULT DeckLink renderer shutdown now completes without the previous long delay.

FIXED**SDI auto-detect reinitialization when a 59.94p signal appears after no-signal startup.**

APPLIES TO MPlatform

If an SDI input was initialized with no signal and a 59.94p signal appeared later, the preview could show split, stacked, or otherwise incorrectly reassembled video until the device was reinitialized.

RESULT SDI auto-detect reinitializes correctly when a 59.94p signal appears after no-signal startup.

FIXED**Audio stutter caused by missing BMD no-signal frame timing in live/render pipelines.**

APPLIES TO MPlatform

Live capture pipelines using Blackmagic/DeckLink inputs could miss proper frame timing while one input was in no-signal state. This could affect downstream live processing and renderer synchronization.

RESULT No-signal frames now carry timing suitable for live capture pipelines.

FIXED**NDI color metadata propagation when the source changes.**

APPLIES TO MPlatform

NDI workflows could lose or propagate incorrect color metadata when the source changed.

RESULT NDI color metadata is now propagated correctly through source changes.

FIXED**HDMI DeckLink Mini Recorder 4K initialization crash in gpu_pipeline mode.**

APPLIES TO MPlatform

Initializing an HDMI DeckLink Mini Recorder 4K input with gpu_pipeline enabled could crash the application.

RESULT DeckLink Mini Recorder 4K initialization is now stable in gpu_pipeline mode.

FIXED**Renderer frame-rate drops while seeking converted MPlaylist output.**

APPLIES TO MPlatform

Seeking through a converted multi-file playlist connected to SDI output could reduce renderer FPS substantially.

RESULT Active frame-rate stabilization is now honored for the converted playlist path used by the renderer.

Decoding and Seeking

ADDED**video.auto_rotation property for automatic rotation-flag handling in MFile.**

APPLIES TO MPlatform

MFile required application-side logic to apply source rotation flags automatically during playback.

RESULT

video.auto_rotation can now apply rotation flags automatically in MFile.

ADDED**MXF SMPTE RP 188 and SMPTE 385M timecode support.**

APPLIES TO MPlatform

MXF files carrying SMPTE RP 188 or SMPTE 385M timecode metadata could expose incomplete timecode information to SDK applications.

RESULT

The SDK reads SMPTE RP 188 and SMPTE 385M timecode from MXF files.

ADDED**Looping for MFile external-audio tracks with continuous timestamps.**

APPLIES TO MPlatform

External audio attached to MFile could not be looped when it reached end-of-stream, which made repeating video workflows lose their companion audio.

RESULT

file::external_audio_loop=true restarts external audio and advances timestamps continuously across loops.

EXAMPLE - Loop an external audio track in MFile

```
file.FileNameSet(videoPath, "");
file.PropsSet("file::audio_track", "-1");
file.PropsSet("file::external_audio", externalAudioPath);
file.PropsSet("file::external_audio_loop", "true");
file.FilePlayStart();
```

FIXED**Delay when adding short play-while-recording files to a playlist.**

APPLIES TO MPlatform

Adding a file to a playlist while it was still being recorded could pause for several seconds when the current recording was shorter than about six seconds.

RESULT

Short play-while-recording files can now be added without waiting for a longer live index to accumulate.

FIXED**MMixer could stop after repeated switching between MFile sources.**

APPLIES TO MPlatform

Applications that repeatedly switch an MMixer background between two MFile sources through mp:// links with background.direct_out could see the selected program source stop instead of continuing playback.

RESULT

Repeated MMixer background switching between prepared MFile sources remains in playback.

FIXED**MMixer preview did not refresh after seeking a background MFile.**

APPLIES TO MPlatform

Cueing a paused MFile to a new position and then assigning it as an MMixer background through mp:// could leave the preview showing the old frame until playback started.

RESULT

MMixer preview updates to the cued background frame after FilePosSet.

FIXED**Delayed application of MPlaylist item scale_type settings.**

APPLIES TO MPlatform

MPlaylist item scale_type settings could be applied too late during item switching, so a few frames appeared with the wrong scale before the intended letterbox or crop mode took effect.

RESULT

MPlaylist applies item scale_type in time for the transition to the item.

FIXED**MPlaylist position increasing during accelerated reverse playback.**

APPLIES TO MPlatform

During reverse accelerated playback, MPlaylist position reporting could move forward instead of backward, so timing logic based on PlaylistPosGet observed the wrong direction.

RESULT

MPlaylist position now follows the actual reverse playback direction.

FIXED**Playback-rate changes for mp:// background direct_out sources in MMixer.**

APPLIES TO MPlatform

When an MFile was connected as an MMixer background through mp:// with background.direct_out=true, FileRateSet on the source MFile could be ignored and the background continued at normal speed.

RESULT

Playback-rate changes on the linked MFile are reflected by the MMixer background direct_out path.

FIXED**start_timecode metadata reading for GXF files.**

APPLIES TO MPlatform

GXF files could expose missing or incorrect start_timecode metadata to the SDK.

RESULT

start_timecode metadata is now read correctly from GXF files.

FIXED**pause_out behavior during accelerated playback.**

APPLIES TO MPlatform

pause_out could be skipped during accelerated playback. Instead of pausing on the final frame at the out point, playback could continue to the background.

RESULT

pause_out is respected during accelerated playback.

FIXED**mp:// links when FileRateSet exceeds max_forward_rate.**

APPLIES TO MPlatform

Linked-source workflows using mp:// could apply playback or rate control differently through the link than on the source object itself.

RESULT

Linked-source timing and playback-rate handling were corrected.

FIXED**MMixer seeking for interlaced files with gpu_pipeline enabled.**

APPLIES TO MPlatform

Seeking interlaced files through MMixer could return incorrect frames when gpu_pipeline mode was enabled.

RESULT

MMixer seeking now handles interlaced files correctly with gpu_pipeline enabled.

FIXED**playlist.on_next=pause_out behavior for live playlist items.**

APPLIES TO MPlatform

For live playlist items, playlist.on_next=pause_out could behave like pause_in and show the first frame of the next item instead of holding the last frame of the live item.

RESULT

playlist.on_next=pause_out now holds the live item's outgoing frame.

FIXED**Audio stopping after the opening seconds for specific MP4 files in MFile and MPlaylist playback.**

APPLIES TO MPlatform

A specific MP4 file produced only a short burst of audio or noise at the beginning in MFile/MPlaylist playback, after which decoded audio samples stopped even though video playback continued.

RESULT

Audio decoding now continues beyond the opening seconds for the affected file structure.

FIXED**Audio loss when an mp:// playlist item's out point exceeded its duration.**

APPLIES TO MPlatform

An mp:// playlist item could lose audio when its configured out point was later than its duration, even while video remained available.

RESULT

Playback boundaries now honor the later out point without prematurely ending audio.

FIXED**Scan-type detection for 10-bit JPEG 2000 files.**

APPLIES TO MPlatform

Some 10-bit JPEG 2000 files were reported with the wrong progressive/interlaced scan type when the decoder did not copy field order into the decoded frame.

RESULT

Scan type now falls back to stream field-order metadata for the affected JPEG 2000 files.

FIXED**Rare MPlaylist deadlock while adding break items.**

APPLIES TO MPlatform

Calling BreaksAdd while playlist frames were being delivered could rarely deadlock the UI and playback threads because playlist locks were acquired in opposite orders.

RESULT

Break insertion and frame delivery now use a consistent lock order.

FIXED**Timed MPlaylist stop commands in GPU-pipeline mode.**

APPLIES TO MPlatform

A playlist stop command with time set could pause forever under gpu_pipeline=true unless audio conversion happened to be enabled.

RESULT

GPU frames are now treated as valid media for timed-stop progression, so playback resumes after the requested interval.

FIXED**PlaylistPosGet position calculation during accelerated playback.**

APPLIES TO MPlatform

After FileRateSet increased playback speed, PlaylistPosGet could report a position behind the expected media time because source-rate metadata was treated as an absolute position.

RESULT

MPlaylist now uses source-rate metadata as an anchor and advances position correctly at non-default playback rates.

FIXED**PlaylistPosGet next-index consistency after command-item removal.**

APPLIES TO MPlatform

Two edit sequences that produced the same final MItem collection could return different next indices from PlaylistPosGet after a command item was removed and re-added.

RESULT

MPlaylist recalculates its cached next item after removal, so identical item collections return the same next index.

Plugins

ADDED**SMPTE 436M CC, V-Chip, and AFD metadata support.**

APPLIES TO MPlatform

SMPTE 436M workflows needed SDK support for closed-caption, V-Chip, and AFD metadata carried with the media.

RESULT

SMPTE 436M CC, V-Chip, and AFD metadata are now supported.

ADDED**cc.display_order option to choose CEA-608 processing order from packet-level or decoded-frame data.**

APPLIES TO MPlatform

CEA-608 workflows needed a way to choose whether caption processing order follows packet-level data or decoded-frame data.

RESULT cc.display_order now lets applications select the CEA-608 processing order.

ADDED V-Chip/XDS insertion and parsing support.

APPLIES TO MPlatform

Applications could not generate and parse V-Chip/XDS metadata directly through the SDK workflow.

RESULT V-Chip/XDS insertion and parsing are now supported.

ADDED HTML5 MLCG editor.

APPLIES TO MPlatform

Applications that expose CG/graphics editing workflows did not have the new HTML5-based MLCG editor available as part of the SDK package.

RESULT The HTML5 MLCG editor is now included.

FIXED CG overlay color rendering when overlay.thread=0.

APPLIES TO MPlatform

CG overlays rendered with overlay.thread=0 could produce incorrect colors compared with the same CG content rendered through the normal overlay path.

RESULT CG overlay colors are preserved when overlay.thread=0 is used.

FIXED CG overlay rendering for interlaced formats when overlay.thread=0.

APPLIES TO MPlatform

CG items could disappear when overlay.thread=0 was used together with interlaced video conversion, even though the same CG content was visible for progressive formats.

RESULT CG overlays render for interlaced formats when overlay.thread=0 is used.

FIXED Extra spacing between adjacent CG text runs in tickers.

APPLIES TO MPlatform

Adjacent CG text runs could get an extra inserted space. This broke punctuation and words where only part of the word used a different style.

RESULT Adjacent CG text runs keep their authored spacing.

FIXED CG plugin forcing ARGB32 instead of the configured HDYC format in playlist pipelines.

APPLIES TO MPlatform

The CG plugin could force ARGB32 output in playlist pipelines even when HDYC was explicitly configured.

RESULT CG processing now preserves the configured HDYC pixel format in this workflow.

FIXED**CEF opening an external browser when HTML overlay is used from two applications at the same time.**

APPLIES TO MPlatform

Running HTML overlay from two applications at the same time could open an external browser window instead of keeping the page inside the SDK-controlled embedded browser path.

RESULT HTML overlay stays in the embedded browser path when multiple applications use it.

FIXED**CG RSS ticker text width and spacing mismatch.**

APPLIES TO MPlatform

CG RSS ticker text could use different width or spacing metrics from a normal text ticker, so visually equivalent ticker content did not line up.

RESULT RSS ticker text metrics match normal CG ticker text more closely.

FIXED**Display of closed captions written to MXF files.**

APPLIES TO MPlatform

Closed captions embedded into MXF output could fail to display after the file was written, making the recording look as if the caption payload had been lost.

RESULT Closed captions written into MXF output are preserved for display.

EXAMPLE - MPlatform MXF output with embedded captions

```
string writerConfig =
    "format='mxf' embed_cc='true' anc_capture='false' " +
    "video::codec='mpeg2video' audio::codec='pcm_s16le' audio::ar='48000'";
writer.WriterNameSet(outputMxf, writerConfig);
writer.ObjectStart(file);
```

FIXED**Generated CEA-608 closed captions in files for VLC playback.**

APPLIES TO MPlatform

Files written with generated CEA-608 captions could contain caption data that VLC did not display, even though the application attempted to embed captions during encoding.

RESULT Generated CEA-608 captions written by the SDK are compatible with VLC playback for this workflow.

EXAMPLE - MPlatform embedding generated CEA-608 captions

```
ccDisplay.PropsSet("cc_type", "608");
string writerConfig =
    "format='mp4' embed_cc='true' scc_capture='false' " +
    "video::codec='libopenh264' video::b='5M' audio::codec='aac'";
writer.WriterNameSet(outputFile, writerConfig);
writer.ObjectStart(file);
```

FIXED**Blurred CG text when decoder.convert_planar=true.**

APPLIES TO MPlatform

CG text could become blurred for certain text colors when decoder.convert_planar=true was enabled, even without scaling artifacts.

RESULT CG text remains sharp with `decoder.convert_planar=true`.

FIXED**MCCDisplay CEA-608 character ordering for specific files.**

APPLIES TO MPlatform

CEA-608 captions decoded through MCCDisplay could show characters in the wrong order for specific files, while reference playback displayed the text correctly.

RESULT MCCDisplay preserves the expected CEA-608 character order for these files.

FIXED**CEF HTML plugin and WebCapture page loading on first start after OS boot.**

APPLIES TO MPlatform

After a fresh OS boot, HTML plugin or WebCapture pages could fail to load on the first SDK start and require another attempt.

RESULT HTML plugin and WebCapture pages load correctly on first start after boot.

FIXED**CEF audio mute and preview-control handling.**

APPLIES TO MPlatform

CEF-backed preview/control workflows could handle mute state or preview control inconsistently, causing embedded-page audio or control state to differ from the application expectation.

RESULT CEF mute and preview-control handling is consistent with SDK control state.

FIXED**CEF popup handling that could create an extra preview/window for new tabs.**

APPLIES TO MPlatform

CEF popup or new-tab requests could create an extra preview/window instead of being handled inside the intended SDK browser workflow.

RESULT CEF popup handling no longer creates the unwanted extra preview/window.

UPDATED**CEF to version 147.0.11.**

APPLIES TO MPlatform

HTML overlay, CG/HTML, and WebCapture workflows rely on the embedded Chromium/CEF runtime, so browser compatibility and rendering behavior depend on that bundled runtime.

RESULT The embedded CEF runtime now uses 147.0.11.

FIXED**L-shape resizing with external-process CG and runtime GPU-pipeline activation.**

APPLIES TO MPlatform

An L-shape layout did not resize correctly when the CG plugin ran in an external process and `gpu_pipeline` was enabled at runtime.

RESULT External-process CG output is created on the GPU path correctly, so L-shape resizing behaves consistently.

FIXED**Smooth scheduled appearance for CG items shown with `cg-props::show`.**

APPLIES TO MPlatform

Scheduling `cg-props::show=yes` displayed a hidden CG item immediately, ignoring the configured introduction time, while scheduled hiding faded correctly.

RESULT Scheduled CG show operations now use the configured intro fade duration.

FIXED**CG hang when hiding a ticker containing a non-RSS web page.**

APPLIES TO MPlatform

Adding a regular web-page URL as ticker text and then hiding the ticker could block CG and the host application.

RESULT Ticker URL handling now rejects non-RSS pages instead of adding an unsupported web item.

FIXED**Visible MPlaylist preview freeze while an HTML5 overlay loads.**

APPLIES TO MPlatform

Enabling or disabling an HTML5 overlay could freeze the actual playlist preview while the embedded browser initialized, making otherwise continuous file decoding appear stalled.

RESULT CEF creation and other long HTML-overlay operations now run outside playlist frame locks, so preview frames continue updating during page load.

Samples and Documentation

ADDED**MLCG REST control sample for MPlatform workflows.**

APPLIES TO MPlatform

Developers integrating MLCG graphics with MPlatform did not have a ready REST API sample showing connection, session selection, timeline commands, and variable updates.

RESULT A new MPlatform sample demonstrates MLCG REST control in an SDK workflow.

UPDATED**MPlatform MLShader samples with the new shader-library project and shader catalog.**

APPLIES TO MPlatform

The older MPlatform shader sample did not show the current shader-library workflow with a catalog, metadata, and reusable HLSL examples.

RESULT The MPlatform shader sample set was refreshed to the shader-library workflow.

UPDATED**Subtitle workflows split into dedicated Conversion and Generation samples.**

APPLIES TO MPlatform

The former Subtitle Insertion sample combined file conversion and live subtitle generation in one project, making the two integration paths difficult to follow and extend.

RESULT Each SDK now includes a Subtitle Conversion sample for external SRT/ASS files and a Subtitle Generation sample for live-source workflows.