This article was published in the German magazine FKT, 08/09 2018

Editing in a cloud-based production environment

Introduction

For broadcasters, production houses – as well as post-production in general - multi-site workflows that include remote editing are challenging.

Traditionally, there have been two approaches to editing in regional locations or remote sites. Firstly, there is “proxy” or “low-res” editing. Editors work with specialized editing clients offering optimized functionality and use lower resolution, and therefore bandwidth, video in the editing interface. This is then linked back to the original media. After the edit, projects are either sent to “craft” edit clients, such as Adobe Premiere Pro, linking back to the high-resolution material, or a new clip is created or “fused”, normally by a server-side rendering engine. This approach still has many merits, especially in the case of journalistic or highlights editing which only require simple edits or voiceovers.

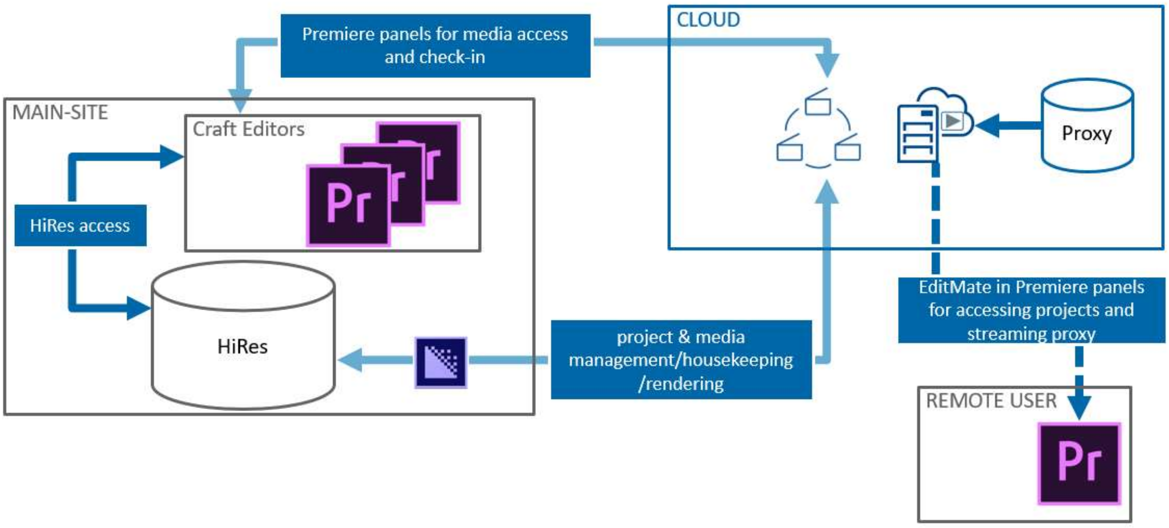
A second approach, as high-bandwidth connections have become more widely available, has been to connect directly to central storage with a craft edit client over a Virtual Private Network (VPN). Even with higher speed broadband connections, higher latencies mean this approach may not give users a “local-like” experience.

Cloud-based editing feels like it should be the answer, but cloud technology alone does not resolve the issues above.

in this article we consider the technical innovations required to make cloud-based editing a daily reality for editors. We will look at how a combination of streaming server and SMPTE RDD25-based HD Proxies and a hybrid Cloud architecture can be implemented.

Technical Overview

We look here at a hybrid craft editing installation, extended with remote editing. Video with high bitrate and storage consumption is located on premise and can be used by local craft editors. New files are rendered with a local Adobe Media Encoder which has access to the HiRes files. Proxy files are generated on growing files at the main site and are stored in the cloud. The streaming server and backend collaborative project management applications are also hosted in the cloud, enabling craft editors and journalists to exchange projects and to edit from remote locations.


Premiere panels for media access and check-in
CLOUD
MAIN-SITE
Craft Editors
Pr
HiRes access
HiRes
Proxy
project & media management/housekeeping /rendering
EditMate in Premiere panels for accessing projects and streaming proxy
REMOTE USER
Pr

Streaming Protocol

The choice of streaming protocol is key. MPEG DASH is useful for linear forward playback but is less capable around fast jumps and scrubbing. Although possible, scrubbing suffers due to the large segments that are transferred from server to client. Therefore, to improve performance, a protocol is implemented in which only the minimum essence data required is transmitted.

Proxy Format

The more information that a proxy can have, the nearer it is to the 'real thing'. RDD25 was originally conceived to describe low resolution proxies for use with low res editors. Extending the use of RDD25 with, for example, H.264 in MXF with 6-8Mbit bandwidth, 1920*1080, 16 audio tracks enhances the user experience significantly. Subjective tests with users show that good results can be achieved even with lower data rates.

Audio

Audio files are an important element of the editing workflow. In order to create waveforms, the whole file normally needs to be analyzed by the NLE client. Pre-created waveforms (or audio peak files), which are available immediately after import, speed up the workflow by reducing the analysis of files done by the client.

Fewer frames transferred when requesting chunks

Journalists – especially in sport –frequently need to scrub through large amounts of video to find the elements they need. Optimized search performance facilitates operations for both editors and journalists. A key improvement has been that only the frames needed for decoding are sent to the client: this reduces latency in streaming and leads to a better user experience.

Asynchronous / Half duplex – Full Duplex

Asynchronous frame requesting improves response times. In an asynchronous frame request scenario, the client can send multiple requests at the same time while in parallel receiving all return information.

Transmission protocol

To be able to transport single frames on demand and to ensure that the streaming server and client support transmission and reception (RX/TX) of requests and responses in parallel on the same TCP connection a proprietary protocol was introduced.

Conclusion

For a runner – Usain Bolt, for example – to improve his speed, there is no single element that he works on. He and his team work on improving multiple elements of his performance: if his stride can be a centimeter longer, his starting time a quarter of a second faster, then he will achieve a still-faster 100m. In the same way there is no single change that will improve and accelerate cloud-based editing workflows to the extent that cloud-based editing becomes the norm.

What is needed is teams of experts with in-depth understanding of the various elements to work on each area – server-storage connectivity, the use of TCP, reduction in the number of frames transferred, improved streaming protocols and better handling of audio files, as well as smart local caching.

Cloud editing is possible. With these improvements, it becomes a reality

About the author:

Karsten Schragmann is a seasoned product manager of professional audio and video products and solutions. At Arvato Systems he is responsible for driving the development and go-to-market strategy of the VPMS EditMate Production Asset Management solution.