



Content Readiness in Cable Workflows

Technical Brief

Table of Contents

Content Readiness and Your Cable Plant Workflow	3	Automating File-Based QC	13
Containers, Mezzanines, and Codecs	3	Semi-Automated QC Workflows	13
When Data Becomes Metadata	4	Asset Management and the Fully Automated Workflow.....	14
The Challenge of Quality Control.....	5	QC Reporting.....	15
QC Methodology: Human vs. Automated Monitoring	5	How is a Workflow Like a Web Browser?	15
Types of Errors: What Could Possibly Go Wrong?	5	When Systems Communicate.....	16
QC Tests, Checks, and Results.....	6	Conclusion.....	17
Examples of Content Readiness Testing in Cable Workflows	7		
Video on Demand	7		
The QC “Pre-Flight” Check.....	8		
Ad Insertion.....	9		
Ad QC	10		
Adaptive Bit Rate	11		
“Segmenting” and QC Issues	12		

Content Readiness and Your Cable Plant Workflow

File-based workflows are supplanting earlier, less integrated digital workflow architectures in cable plants everywhere. File-based environments streamline content production and delivery, a benefit that is becoming ever more urgent amid proliferating end-user platforms and the diverse formats that support them. The cable industry needs a workflow that can store and process content and then send it over many delivery paths simultaneously. The file-based workflow brings together all these functions and enables efficient management of both business and technical issues within the cable enterprise.

With this great power comes great responsibility: the requirement to ensure content is always ready for immediate distribution. This is “Content Readiness” and it means having all your material available in a form and condition suitable for distribution. It means content that’s not only free from objectionable distortion but also equipped with the right markers and metadata and resolution to deliver the best possible service to subscribers, regardless of the receiving platform.

This level of content readiness calls for a rigorous quality control (QC) program that inspects every individual file—from a 15-second ad to an HD movie—as it enters the cable plant and when it is stored for eventual playout. The sheer volume of content dictates some level automation in the QC process.

This document will briefly introduce the file-based system architecture and then will elaborate on potential QC challenges for three workflows of interest to cable providers: video on demand (VOD), ad insertion, and adaptive bitrate services. A final section will discuss automated quality control issues in a file-based cable workflow.

Containers, Mezzanines, and Codecs

The file-based workflow begins with an ingest server whose job is to process received content into files that facilitate management, storage, and distribution. Content arrives in diverse formats and it is necessary to process everything into a uniform format for internal use.

Codec	Container	Bit Rate
DV	MXF Op-Atom	100 Mb/s
MPEG-2 IMX	MXF OP1a	30–50 Mb/s
Apple ProRes	QuickTime	220 Mb/s
VC-3	MXF or QuickTime	220 Mb/s

Table 1. Commonly-used mezzanine file formats and codecs.

The output of this process is digital files organized into “containers,” also known as wrappers. A container usually embodies a single file, though some containers may be made up of multiple files linked together. In any case containers encompass all the vital information about the files they contain: compressed video, audio and also importantly, metadata.

There is no single universal container format. In fact there is a variety of container formats including:

- MPEG Program Stream
- MPEG-2 Transport Stream
- MP4
- 3GP
- QuickTime File Format
- Material Exchange Format (MXF)
- General Exchange Format (GXF)
- Advanced Systems Format (ASF)

...and more. Some formats are optimized for playout, others for editing or capture or other points in the workflow. The common MXF container format alone has multiple variants. Containers are a cornerstone of file-based workflows.

In the cable domain, MPEG Transport Stream files are of course ideal for playout, ready to be packetized and sent out over the network. The .TS container format can be used for offline processing and QC operations as well. In addition the Quicktime and MXF formats, among others, are well suited to editing and transcoding (converting) by means of codecs such as those in Table 1.

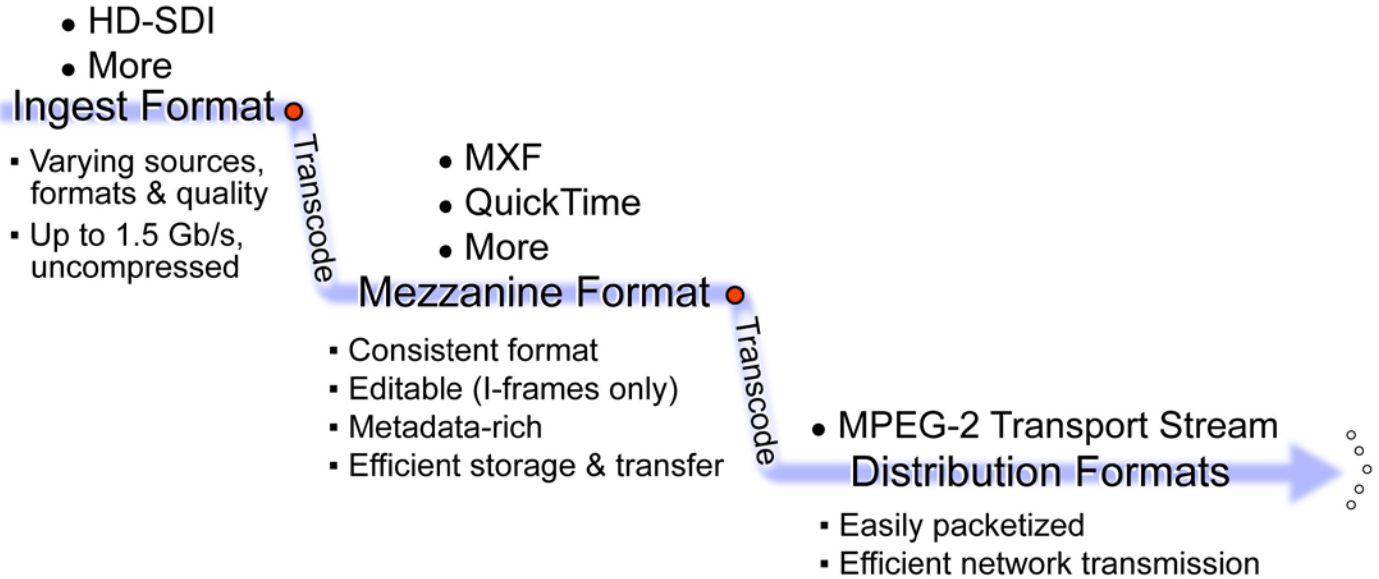


Figure 1. Mezzanine formats allow efficient handling and storage of files within the cable plant.

MXF and QuickTime formats are commonly used as “mezzanine files.” The term comes from the world of architecture, where it denotes an intermediate floor (often a lobby or a balcony) between two main floors. Similarly, mezzanine files are an intermediate step. They are working copies that are more expedient to use in the workflow than the original source material. Figure 1 is a symbolic illustration showing where the mezzanine’s transitional format fits in—midway between the high bit rates of the raw ingest and the lower distribution bit rates.

Though substantially compressed, the mezzanines suffer little noticeable loss in picture quality. They have sufficient resolution to minimize generation loss when transcoding. And being smaller files than the original source files (thanks to data rates of about 200 Mbits/sec compared to HD-SDI rates of 1485 Mbits/sec), they require less storage space and transfer time. Importantly, mezzanines are made up of I-frames only, which means that successive frames are complete and editable with no need for interpolation of I, P, and B-sequences, making editing and post-production tasks more efficient. For more details on the IPB hierarchy see the “Adaptive Bit Rate” section in this document.

When Data Becomes Metadata

Metadata is often stored in the file with the video and audio data, or it can be located in an auxiliary file in the same package. Metadata is overarching “data about the data”—a set of descriptors that can include the episode title, scene numbers, languages, ratings, and more. There may be information about the usage rights attached to the file. This specifies the number of layouts or the length of time in the licensing for the content. All these values are human-readable. In addition metadata may express attributes like frame size, frame rate, or aspect ratio; information that’s essential for correct playback. These entities are machine-readable.

Workflow operation is more efficient when metadata is written into the files. The MXF file format is being widely accepted in the broadcast industry specifically because it is metadata-rich. MXF is equally suitable for some cable applications.

The Challenge of Quality Control

QC Methodology: Human vs. Automated Monitoring

Almost like a law of physics, the challenge of quality control seems to grow with the square of content quantity. Perhaps there was a time when visual inspection of incoming programs was sufficient, but that era ended with the explosion of content required for services like Video On Demand (VOD). Human monitors are prone to overlooking subtle impairments, and of course can't detect metadata or embedded digital errors. Moreover, the task of scrutinizing thousands of old movies at two hours per selection can add up to decades' worth of man-hours. And with all the transcoded versions for diverse output platforms, the job just gets that much bigger.

A better idea is an automated QC strategy that operates 24/7 and pinpoints errors so human operators can spend their time fixing rather than finding errors. Automated QC is more thorough and consistent, whether it's evaluating one short commercial or an entire archive of old TV westerns. Equally important, capable automated QC systems can detect errors that are invisible to the human eye: metadata that doesn't match the measured file attributes, for instance, or syntax errors that might pass through the industrial-strength systems in the cable plant but could crash an ordinary set-top box.

Types of Errors: What Could Possibly Go Wrong?

File-based video is a technical blessing but like all other media, it is susceptible to flawed source material. Some errors originate in the baseband product. Such flaws range from improper camera or microphone levels to amateurish, out-of-gamut "homemade" graphics in a tire commercial. These errors are part of the file and must be rectified eventually.

Incorrectly encoded files are another problem area. These can arise from a faulty encoder that produces syntax errors due to buffer overflows or similar technical issues. Or the encoder may be misconfigured, as when a standard-definition profile is applied to an HD source.

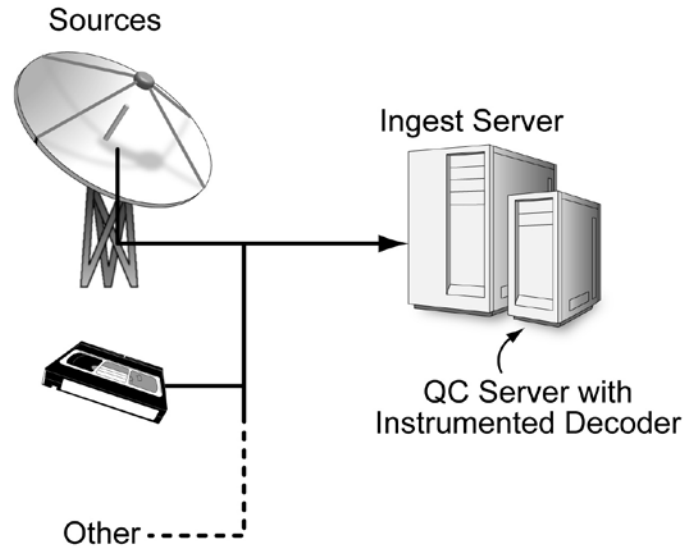


Figure 2. Ingesting source material into a file-based workflow.

Transfer errors are yet another hazard. Problems can attach themselves during any transfer from system to system, including the ingest transfer itself. Even content that is sent correctly can be received with errors due to interruptions or faulty equipment.

An "instrumented decoder" (Figure 2) is the preferred tool to ensure detection of all these error types. Whereas a conventional decoder attempts to recover gracefully from errors, an instrumented unit reports them. In addition the latter tool finds errors at two different levels: in the encoded bit stream, and in the decoded baseband image raster and audio samples. A file can be syntactically correct but still contain block frames, muted audio, and so forth. Only an instrumented decoder can find these faults reliably.



Figure 3. "Super white" gamut violation.



Figure 4. Blockiness due to over-compression.

QC Tests, Checks, and Results

Quality checks fall into three major categories:

■ Structural Checks

These QC checks focus on compliance with applicable industry standards and just as importantly, acceptance criteria for entities such as iTunes, Netflix and PBS. In addition CableLabs specifications for VOD content may apply.

"Compliance" typically encompasses format-related issues: codec and container type, the MPEG profile and level, GOP structure, bit rates, frame size, and frame rates. For example, there may be an incorrect number of streams, implying a missing audio portion. Or there may be a mismatch between the signalled bit rate and the actual bit rate.

■ Baseband Quality Checks

Errors can occur in the decoded image, the image raster or the decoded audio samples. These are the target of baseband checks. These measurements, when performed in the file-based domain, are very similar to observing a real-time live signal with a waveform monitor.

But file-based checks run faster than real playback time. The QC tool looks for video errors such as dropouts, frozen frames and unexpected letter- or pillar-boxes. Gamut problems such as super black or super white (Figure 3) also are revealed. In the audio domain, loudness violations such as clipping and CALM Act non-compliance can be not only detected (using the same ITU-BS.1770 algorithm used in a real-time loudness monitors) but also corrected.

■ Encoded Content Checks

These tests watch for the low bit rates and over-compression that cause blockiness artifacts (Figure 4) that can be measured in the decoded image raster and reported. Field order problems in interlaced video are common as well, especially in Ad Insertion. These errors show up as motion artifacts. Similarly, MPEG errors such as incorrect slice order can cause large block distortions.

Files can get damaged during transfers. An interruption, for example, may cause a file to get truncated even though there are normally safeguards to ensure recovery from interruptions. An incomplete file like this would lack an End of Sequence marker.

All of these problems can be quickly exposed with a thorough a syntax check and all lend themselves to automated QC process that includes screening and reporting on every file.

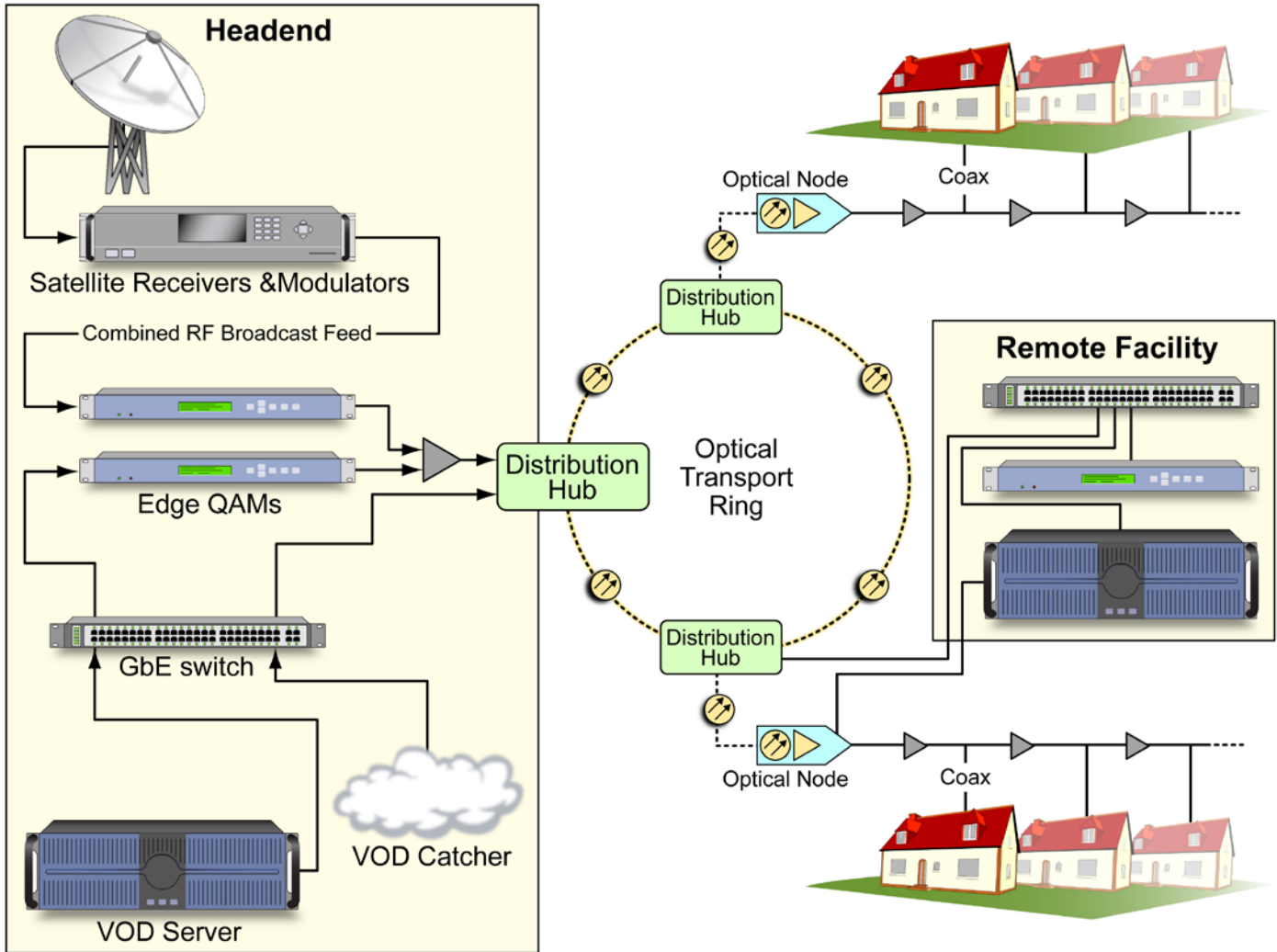


Figure 5. The Video On Demand (VOD) workflow.

Examples of Content Readiness Testing in Cable Workflows

Video on Demand

Video On Demand (VOD) is now widespread in the consumer market thanks to aggressive campaigns by content producers. VOD has become a substantial revenue source for cable providers, and continues to increase its market share at the expense of traditional consumer media. Figure 5 is an overview of the VOD flow from content capture to end user.

VOD implementation begins at the Headend with the acquisition of content. This may be via satellite receivers or by file transfers. Industry insiders use the baseball metaphor of “pitching” and “catching” to describe this process. A satellite antenna may act as the VOD catcher, though increasingly the task is simply one of transferring files from the Internet

cloud. Large (and successful) enterprises have been built up to provide tools that ensure the fastest possible transfer of massive data files such as multi-gigabyte movies. Increasingly, both the content provider and the cable operator maintain a cloud connection through a proprietary file transfer vendor to speed the exchange from pitcher to catcher. This method is likely to supplant the older FTP solution eventually, but conventional FTP is also in common use at this time.

Watermarking is added by the VOD Catcher. The purpose of watermarking is to build in an “invisible” undetectable means of tracing illegal copies. Conceivably a subscriber with the right tools could record a VOD movie and resell it, but a watermark identifies the material’s source and provides recourse against this kind of piracy. Metadata, added at ingest and stored in in an XML format compliant with the [CableLabs Asset Distribution Interface specification](#), plays an important role in automating VOD playback.

The QC “Pre-Flight” Check

Subscribers pay for the VOD product, often included as part of their subscription package, and they expect a level of quality that lives up to the vendor’s claims. Therefore the quality of the content stored in the VOD servers is crucial to the success of the whole process. The moment when the requested content exits the facility is clearly not the right time to evaluate its quality. Like a pre-flight check, it is important to confirm the quality of the product before it leaves the plant.

In Figure 5 the headend contains the broadcast feed which is modulated and inserted into the optical transport ring and onto the network. Of course this is also the distribution point for VOD. It is not necessary to run everything from a single centralized headend and in Figure 5 a remote facility (which may be one of several) shares the load. These remote sites have their own VOD servers which store separate copies of the VOD assets, increasing the efficiency of distribution in their locales. A high-speed network is required for streaming video to remote edge QAMs.

Incoming VOD assets can undergo QC checks when they are received at the Ingest cache, or while they are stored on the VOD servers. What kind of problems do we look for in the VOD workflow?

- File integrity problems: These can occur during the automated pitching-catching process at ingest. Files can become corrupted or truncated. Does the syntax check detect an EOS (End of Sequence) flag, and is the measured play time accurate?
- Format compliance issues: Files must be checked for compliance with CableLabs specification CEP 3.0 (Content Encoding Profile). Are the bit rates within the prescribed range; are the PID numbers correct (e.g. 481 for video or 482 for audio); are the GOP lengths correct?
- Does the VOD material comply with governmental regulations? Are the mandated CEA 608/708 captions present? Is the audio loudness within the limits set by the CALM Act? In every case these characteristics must be not only checked but also corrected when errors are found.
- Does the machine-readable metadata agree with measured values such as frame rate, displayed picture size, and play time?

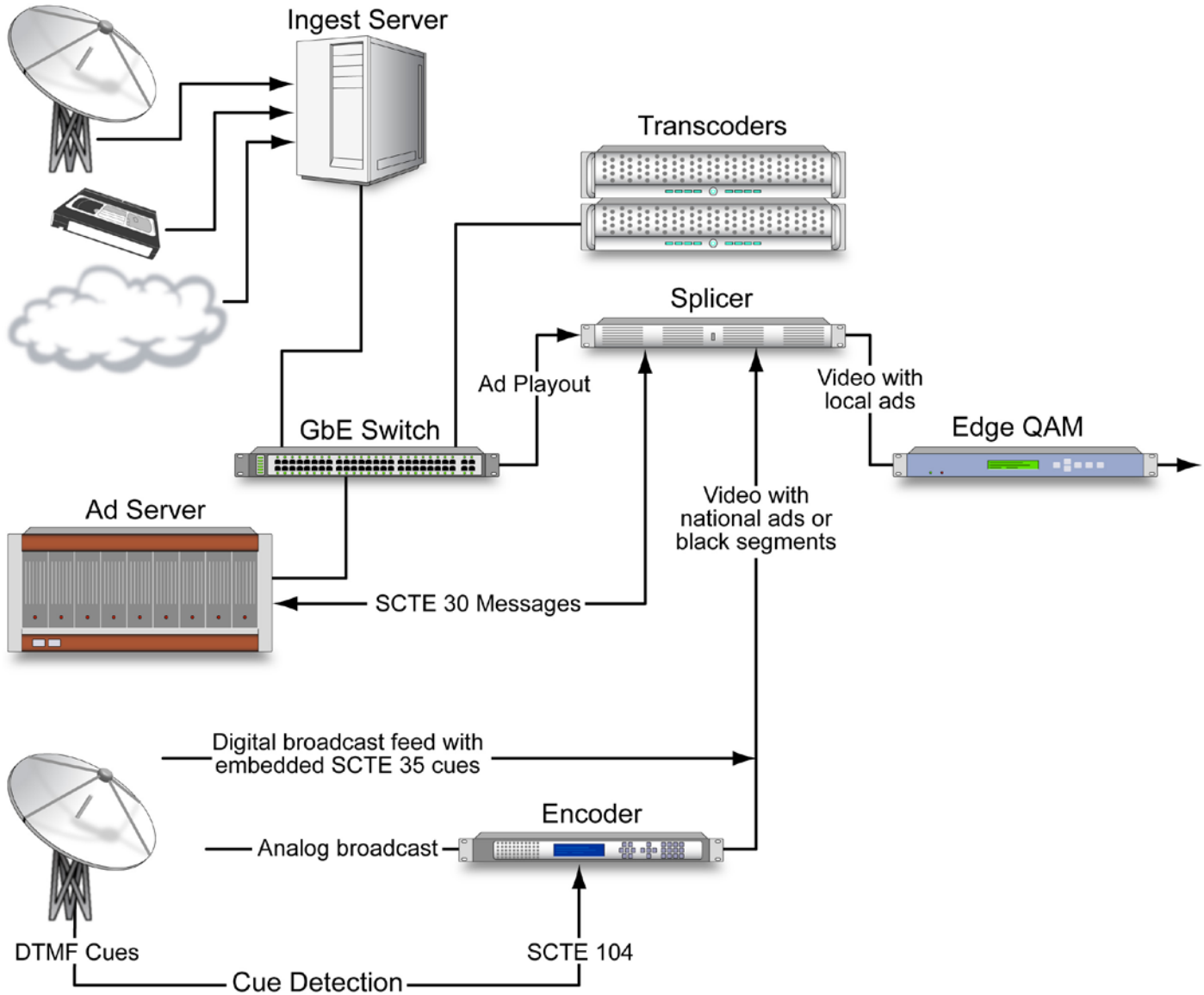


Figure 6. Ad insertion workflow.

Ad Insertion

Advertising is of course the lifeblood of scheduled cable and broadcast networks. A content stream isn't complete until advertising is embedded. In the ad insertion workflow (Figure 6), the challenge is one of managing myriad codec, container, and media formats. Local commercials may be delivered in the

form of tapes, DVDs, memory cards, or digital files. National ads typically arrive as files from a content delivery network. The ingest process must ensure that all of this is stored in a consistent file format. In every case, vigilant QC is required to ensure continuity of visible quality as programming switches from entertainment to ad content and back.

Rigorous incoming inspection is the ideal, and theoretically content that fails to meet acceptance standards will be rejected. But the reality is that cable operators often must groom local ads before they are suitable for distribution. The mattress outlet at the mall and the car dealership in town sometimes produce ads on a scant budget with too little attention to normal video standards such as gamut compliance and loudness.

In the early part of the ad insertion workflow, the QC focus is on content quality rather than absolute format correctness. Ads that don't meet minimum standards, for example having a very low bit rate and consequently poor picture quality, are candidates for rejection. Gross errors in play time (for example a 32-second spot for a 30-second timeslot) are another disqualifier. Other flaws such as loudness violations may be accepted with the understanding that the operator will correct them—sometimes with additional charges.

SCTE standards specify various types of ad cues and messaging encoded in diverse ways. Incoming content may include national spots that can be used “as is” or replaced by local spots. There may be black segments meant to be populated with local ads drawn from an in-plant ad server.

At ingest the content gets transcoded to the format the splicer needs in order to seamlessly insert it into the broadcast feed. Because the splicer can only switch ads in and out on GOP (Group Of Pictures) boundaries, one of the quality checks must ensure that the clip contains an integral number of GOPs; in other words it is “closed” so that there are no references to GOPs preceding or following the clip.

Ultimately the ad server in Figure 6 contains only ads that are in the correct format and ready for insertion. The active broadcast feed passes through the splicer, which inserts local ads at the correct time.

Ad QC

The majority of errors in the Ad Insertion workflow are quality-related. A local merchant simply can't afford the lavish production values of a big national campaign, so compromises are made. A small video house might not have a loudness meter, for example, which means that audio levels may be too high—even to the point of clipping. Or perhaps an inexperienced hand produced graphics with garish, attention-grabbing colors that just happen to be out of gamut. Or there may be over-compression, causing blockiness and artifacts. Clearly there are many pitfalls.

It is the cable operator's prerogative to reject this content or accept it and repair it. In either case it's crucial to have a QC regime that can detect these flaws.

Another job for the QC system is to confirm that the ad content meets submission guidelines. Many operators constrain the delivery formats they will accept, with the intent of reducing complexity. Limitations may include codec type, container type, audio channel assignments, picture size, bit rate, frame rate, and more.

One of the most bothersome errors in ad content is the format mismatch. For example, ads may be submitted in 4:3 standard definition (again the result of cost-cutting production) even though the broadcast will be in HD. Unfortunately this causes letterboxing as the subscriber's equipment tries to make the best of a 4:3 aspect ratio on a 16:9 screen. It is even possible to have letter-boxing and pillar-boxing occurring at the same time; a small picture appears in the center of the screen, surrounded by black on all sides. This is a distracting effect! One way to avoid it is to set up separate ingest paths for SD and HD deliveries. In addition, the QC tool should routinely verify that the video fills the active image and that letter-boxing/pillar-boxing effects are not a permanent part of the file. And it is becoming common practice to check the Active Format Descriptor (AFD) to confirm that the playout code is correct.

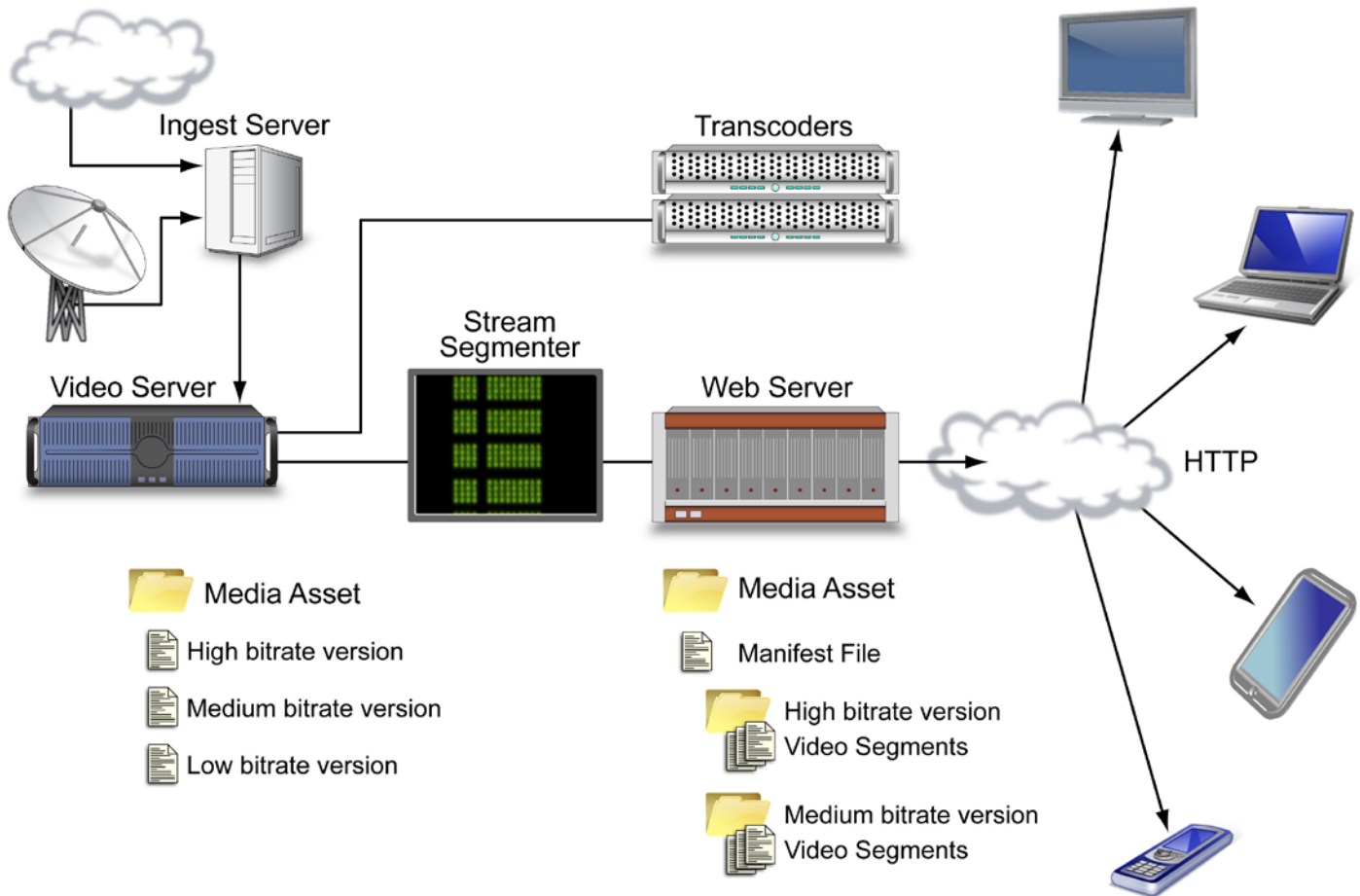


Figure 7. Adaptive Bit Rate (ABR) workflow.

Adaptive Bit Rate

Cable content distribution used to be straightforward, with scheduled programming going out to subscribers via operator-provided set-top boxes. But that market has changed forever. The set-tops are still there, of course, but so are tablets, phablets, phones, laptops and PCs. To be competitive, operators must offer streaming services for all these devices.

The solution for this multi-faceted challenge is Adaptive Bit Rate (ABR) streaming, also known as Over-the-Top (OTT) delivery. HTTP Live Streaming (HLS) and Smooth Streaming are two of the leading streaming architectures in use today. Both HLS and Smooth Streaming rely on transcoding each asset to multiple bit rates. Thus there are several coexisting versions of each asset. Standard HTTP network transport protocols are used for client (subscriber) access.

The multiple bit rates make it possible to optimize transmissions for moment-by-moment network capacity and to tailor the content to the receiving device. An ABR

implementation must be able to repeatedly change the bit rate, switching it higher or lower depending on network conditions. And on the client (receiving) side, a computer with a 21-inch LCD screen requires much more data to support a credible picture than does a phone with a four-inch screen. The client-side player determines the available bit rate and requests the best content available to match that rate.

ABR demands are complex, and optimizing the bit rates is only part of the story. A file must be broken into short segments rather than being transmitted in one full-length delivery to the subscriber. As shown in Figure 7 the “stream segmenter” tool is dedicated to this step. The segments are just a few seconds long, usually ten seconds or so. The reason for this layout is to provide boundary points at which the bit rate can be switched. As bandwidth availability changes, the client player requests the optimum bit rate and the switch occurs at an appropriate Group-of-Pictures (GOP) boundary.

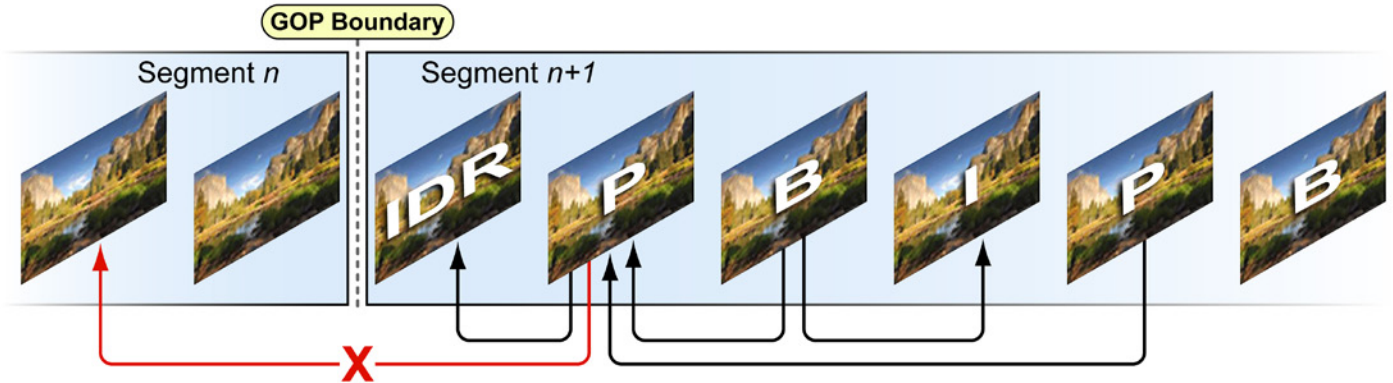


Figure 8. H.264 compressed video content with Groups of Pictures (GOP) made up of I, P, and B frames. An Instantaneous Decoder Refresh (IDR) frame follows each GOP boundary. Frames in segment $n+1$ cannot reference previous segments.

Figure 8 illustrates the frame types and GOP boundary crossings that make up a segmented file. H.264 compressed video provides the basis for this simplified view. The content includes I, P, and B frames. The I frame is the ongoing reference frame and the P and B frames contain incremental changes relative to the I frame. This scheme enables very high compression ratios but it puts constraints on when and where bit rates can be switched.

The H.264 video file in Figure 8 also includes an Instantaneous Decoder Refresh (IDR) frame. In effect this clears the buffers that refer to preceding frames and restarts the I, P, and B sequence to define a new segment. The IDR marks the boundary between any two segments. A frame within the $n+1$ segment in Figure 8 cannot reference a frame from the preceding segment.

“Segmenting” and QC Issues

A key part of the QC regime in an ABR workflow is to verify the content’s readiness for segmenting. It is important to establish early in the workflow that the file can be correctly divided into usable segments. Note that this is not a check on the individual pieces; that is the job of a different tool designed to monitor and measure the ABR stream in real time. At QC time, the task is one of ensuring that smooth segmentation is possible.

In this context “content readiness” implies that IDR frames are embedded at regular timing intervals. These form the segment boundaries. The QC checks should confirm that stored content—whether movies or commercials—has the IDRs in place and timed correctly.

Another quality check relates to picture quality in an environment where content is stored in numerous versions, each with a separate bit rate. Are the lowest bit rates still delivering acceptable picture quality? For that matter, are *all* of the rates providing the expected image quality? This is a test that is not practical to perform on every piece of content in real-time; instead it is best used to guide the design of an effective ABR workflow in the cable plant. Using a picture quality analysis system, it is possible to fine-tune encoder performance to get the most out of each bit rate. In H.264 compressed video, for example, there are numerous settings that can be adjusted to maximize the final picture quality on the receiving device.

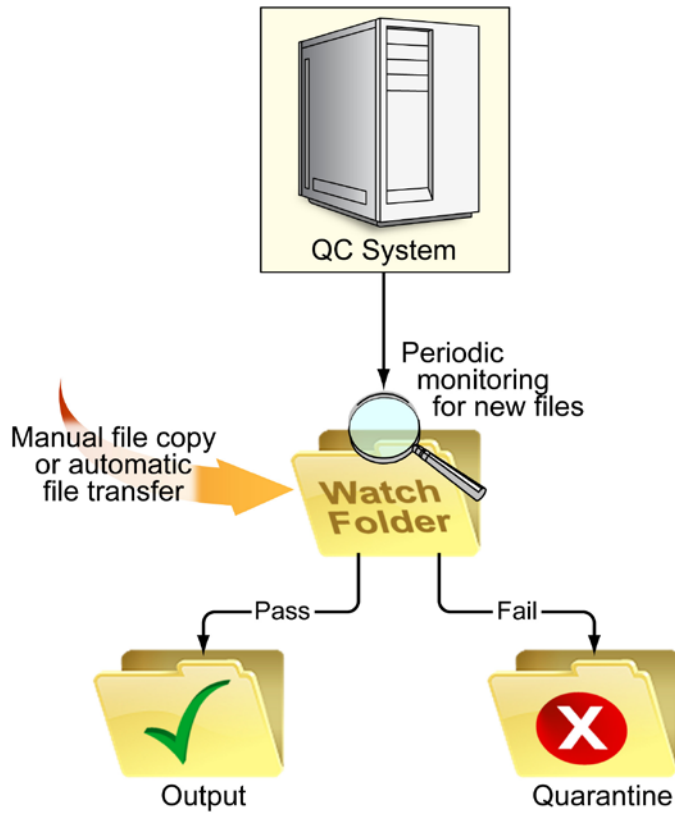


Figure 9. Semi-automated QC flow.

Automating File-Based QC

Content readiness means consistent high quality every day, in every asset that is stored for distribution. And that may add up to many thousands of hours of material, particularly in a VOD service. Manpower is a costly and admittedly fallible resource for checking content quality on this scale of quantity.

Semi-Automated QC Workflows

A far better solution to the problem is some degree of automation in the QC effort. The more automation the better, of course, but a partially automated workflow can ensure content quality very effectively. Figure 9 illustrates this scheme. A semi-automated work flow is optimal for small-to-medium operations and tasks like ad insertion.

In a semi-automated workflow, the QC System is in charge of passing files from acquisition to storage. On the input side, the QC System periodically polls one or more Watch folders and creates a directory listing to detect content that needs to be ingested. Both manual file copy operations and automated transfers via the “catcher” can go into the Watch Folder(s) anytime during the day or night.

When a new asset arrives, the QC System runs tests to measure the quality and compliance of the material. Based on the Pass/Fail results, the file goes to either an Output folder or a Quarantine folder. In the latter case an e-mail alert can be sent to an operator who can repair or reject the content as he/she sees fit.

Asset Management and the Fully Automated Workflow

A fully automated QC workflow is the solution of choice for enterprises that handle VOD libraries or other material in great volume. The amount of content that must be reviewed for aberrations and errors makes manual evaluation impracticable. The overarching term for this content is media assets.

The heart of a media asset is known as the “essence,” which consists of video and/or audio material. When metadata is added to this, the sum is “content,” but the package is still not complete. With the addition of “usage rights” the viewable media asset is complete. Figure 10 is a symbolic view of the hierarchy.

A media asset has value because, like any inventory item in any company, it costs money to acquire it. The cost is embodied in the usage rights. Cable providers pay for the right to show a particular program a limited number of times, and usually for a limited time span.

The need to manage media assets is pervasive across the workflow. Asset management tasks include searching for and retrieving assets and importantly, keeping track of the usage rights and the payouts accrued for each asset. A media Asset Management System (AMS) is the clearing house for all such transactions. The key to making an AMS architecture work is the quality of the metadata stored with the assets.

With an AMS and effective metadata, it is feasible to automate the operation and move beyond semi-automated model where the QC tools must look for files and then test them. Under the control of an Asset Management System, QC is based on where an asset is in the workflow.

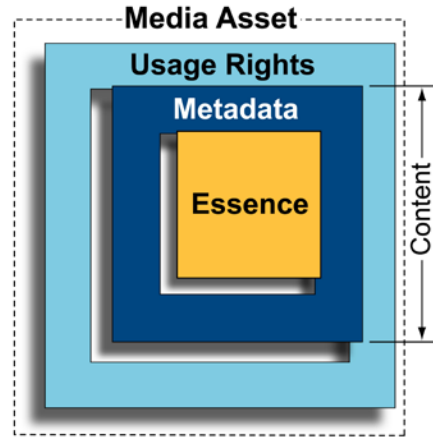


Figure 10. A hierarchy of elements makes up the media asset.

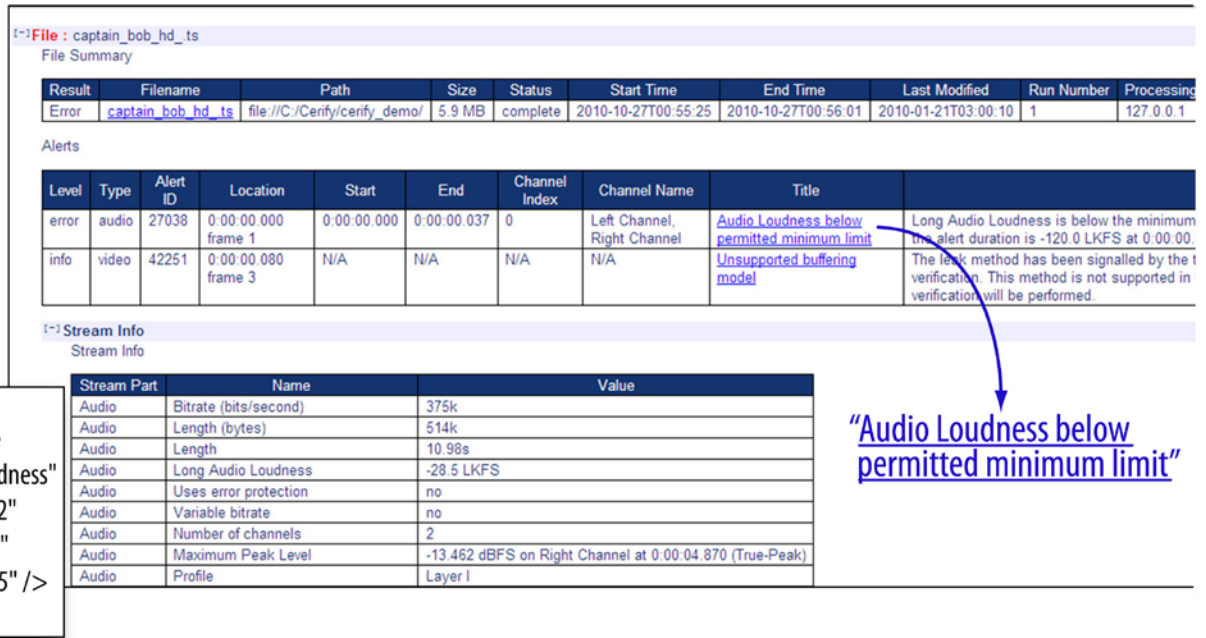


Figure 11. A human-readable QC system report.

QC Reporting

A QC system produces a report upon completion of its tests on each file or set of files. That report can be saved with the asset and stored in the Asset Management System's database. It includes both errors and measured values such as play duration, peak levels, and more. The human-readable components of the report enable operators to respond efficiently to any detected errors that were found in the file. To implement a fully automated QC process, it is of course necessary to include machine-readable data in the report.

The technology that supports these exchanges between machines and also facilitates human intervention is the XML text format, a solution borrowed from the IT world. The box on the left of Figure 11 depicts a short segment of XML code as seen by the AMS. Adapted to a style sheet as shown in the figure, this same information is human-readable.

How is a Workflow Like a Web Browser?

How is all this data exchanged among the various systems in a workflow? The reality of many workflow elements is that they are simply software tools residing within computers. All this software needs communicate without adding a lot of complexity to the flow. Increasingly, conventional web services are being used to integrate software tools from diverse sources (vendors) into coherent systems.

These web services are actually the same proven protocols commonly used for web browsing. The functional needs of the video workflow are very similar, with local clients (say, the AMS) sending requests to remote servers and receiving responses. The Simple Object Access Protocol (SOAP) submits requests and receives responses over HTTP in a server-client architecture. The Web Services Description Language (WSDL) is, as its name implies, a machine-readable description of the operations available by web services. WSDL can automatically generate library code and user documentation for the services.

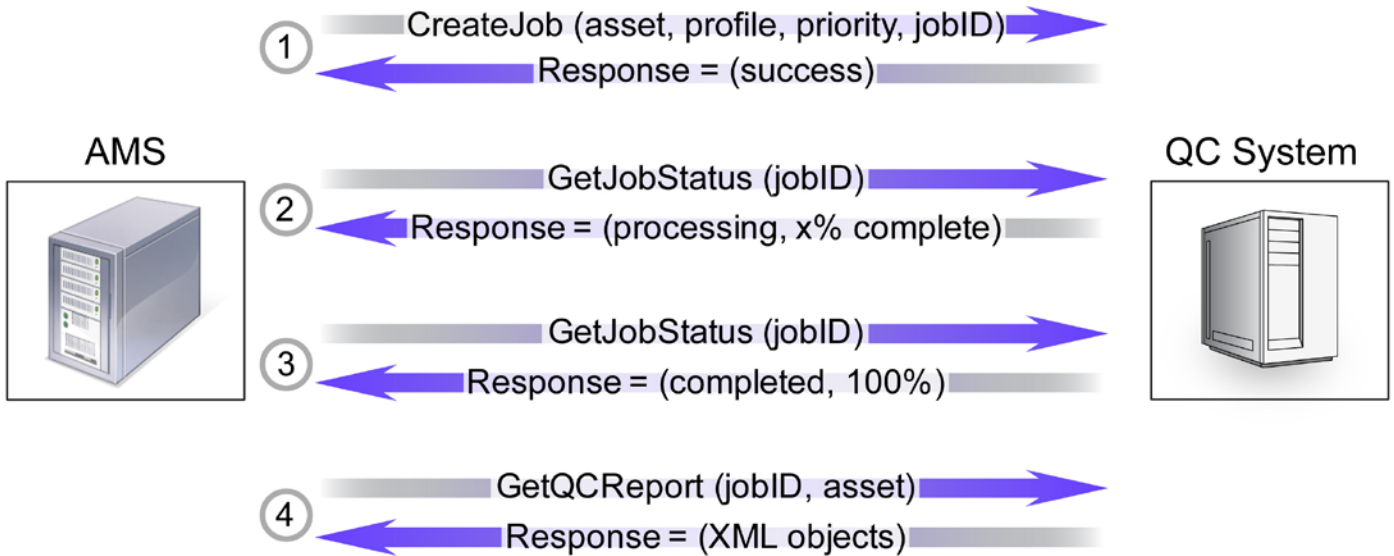


Figure 12. Steps in a transaction between the AMS and the QC system.

When Systems Communicate

Again, the interaction among systems is a process of sending requests and getting responses. An example transaction between the AMS and the QC system might proceed as depicted in Figure 12, with the AMS acting as the client:

1. When the AMS receives a new asset from the VOD catcher or other transfer source, it sends the QC System a new job request to test the file. Crucial parameters about the file are included in the request: where the file is located, priority (high, medium, or low), and a profile that describes the type of asset and the checks to be performed on it. The Job ID is a token that accompanies the file through the remaining steps in the process.
2. Next, query the QC System. The Asset Management System needs to poll the QC system periodically. What is the status of the job with this ID? In a web services model, it is always necessary to send a request to get a response. Without a request there would be no notification when the job gets completed, so frequent polling ensures a timely response. If the job is not finished, the QC system will respond with its current percent of completion.
3. Repeat the status queries until the response indicates that the job is 100% complete.
4. Lastly, get the QC report for the asset. The response comes back in the XML format explained earlier. Now the AMS can interpret the results and make decisions (guided by predetermined failure codes) about what to do if the file failed. For example a file that's missing captions will get treated differently than one that has failed the loudness test.

Conclusion

The spread of services like Video on Demand is a challenge to cable providers who must not only manage a much larger volume of content than ever before, but also deliver that content with consistently high quality. Many cable providers are joining their broadcast industry colleagues in adopting

automated QC inspection tools to speed their file-based workflows. In a field where profits depend on vast queues of vibrant, instantly deliverable entertainment, content readiness is a top priority. Modern integrated solutions like the Tektronix Cerify automated video content verification system are the shortest path toward an efficient content readiness strategy.

Contact Tektronix:

- ASEAN / Australia** (65) 6356 3900
- Austria*** 00800 2255 4835
- Balkans, Israel, South Africa and other ISE Countries** +41 52 675 3777
- Belgium*** 00800 2255 4835
- Brazil** +55 (11) 3759 7627
- Canada** 1 (800) 833-9200
- Central East Europe and the Baltics** +41 52 675 3777
- Central Europe & Greece** +41 52 675 3777
- Denmark** +45 80 88 1401
- Finland** +41 52 675 3777
- France*** 00800 2255 4835
- Germany*** 00800 2255 4835
- Hong Kong** 400-820-5835
- Ireland*** 00800 2255 4835
- India** +91-80-30792600
- Italy*** 00800 2255 4835
- Japan** 0120-441-046
- Luxembourg** +41 52 675 3777
- Macau** 400-820-5835
- Mongolia** 400-820-5835
- Mexico, Central/South America & Caribbean** 52 (55) 56 04 50 90
- Middle East, Asia and North Africa** +41 52 675 3777
- The Netherlands*** 00800 2255 4835
- Norway** 800 16098
- People's Republic of China** 400-820-5835
- Poland** +41 52 675 3777
- Portugal** 80 08 12370
- Puerto Rico** 1 (800) 833-9200
- Republic of Korea** +822-6917-5000
- Russia** +7 495 664 75 64
- Singapore** +65 6356-3900
- South Africa** +27 11 206 8360
- Spain*** 00800 2255 4835
- Sweden*** 00800 2255 4835
- Switzerland*** 00800 2255 4835
- Taiwan** 886-2-2656-6688
- United Kingdom*** 00800 2255 4835
- USA** 1 (800) 833-9200

* If the European phone number above is not accessible,
please call +41 52 675 3777

Contact List Updated June 2013

For Further Information

Tektronix maintains a comprehensive, constantly expanding collection of application notes, technical briefs and other resources to help engineers working on the cutting edge of technology. Please visit www.tektronix.com



Copyright © 2013, Tektronix. All rights reserved. Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this publication supersedes that in all previously published material. Specification and price change privileges reserved. TEKTRONIX and TEK are registered trademarks of Tektronix, Inc. All other trade names referenced are the service marks, trademarks or registered trademarks of their respective companies.

