AUDIOVISUAL
PRESERVATION
SOLUTIONS

A Primer on Codecs for Moving Image and Sound Archives

& 10 Recommendations for Codec Selection and Management

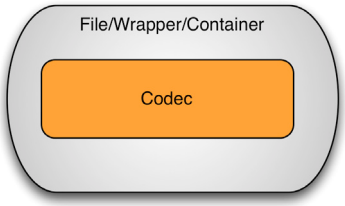
by chris lacinak
president
audiovisual preservation solutions

AudioVisual Preservation Solutions | Media Archiving & Data Management Consultants
350 7th Avenue, Suite 1605 | New York, New York | 10001
917.475.9630 | www.avpreserve.com | info@avpreserve.com

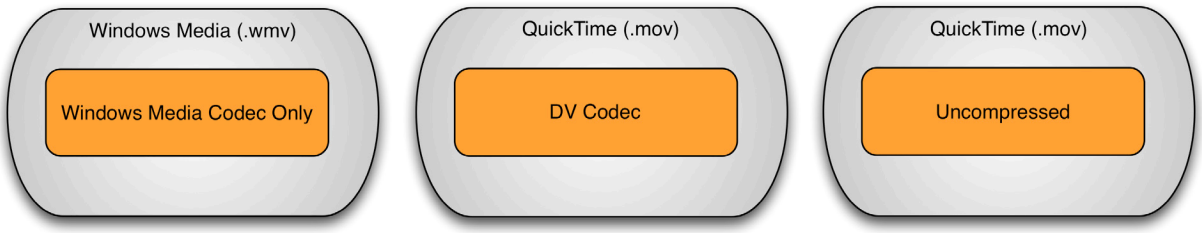
Introduction

The way that the term codec is generally used makes understanding what one actually is a bit confusing. Most people have the general impression that a codec is somehow related to both software and files. Maybe you have run across the term when you find out your computer doesn't have the codec you need to play an audio or video file back, or maybe you have heard it referred to as a method to store data inside of a file or wrapper. Beyond that the details are sketchy for most people. The purpose of this paper is to clarify what a codec is, how it is used and what that means to archives.

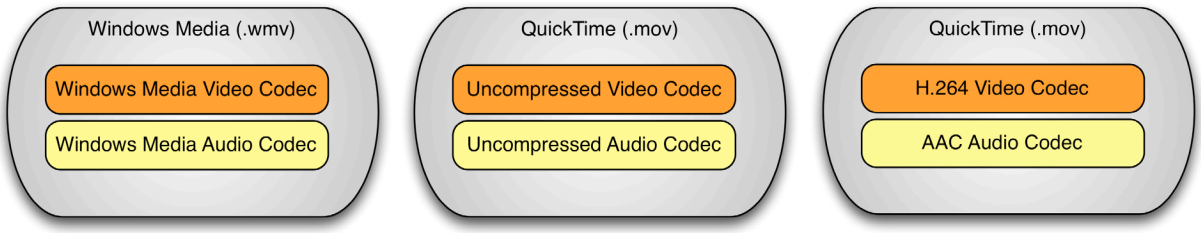
Let's start at the beginning. The terms that are often used when talking about digital media files are 'file format', 'wrapper', 'container' and 'codec'. A codec may be stored inside of a file, a wrapper or a container.



File Formats, Wrappers and Containers are essentially the same, although the terms wrapper and container are generally used to indicate the ability to store different types of codecs as opposed to storing only a single type . For instance Windows Media Files (.wmv) will only store Windows Media codecs. QuickTime and MXF are referred to as wrapper or container formats because they can store many types of codecs including DV, MPEG2, Uncompressed and more.



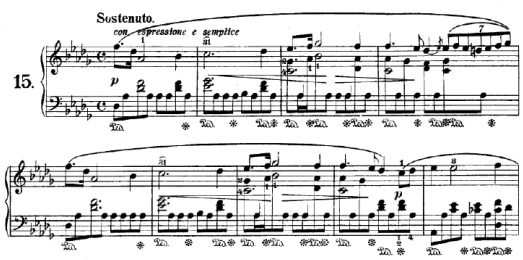
To differentiate further, there are separate codecs stored for audio and video.



Now that we have our terminology settled, let's look closer at codecs. The term codec is derived from the terms encoding/decoding and compression/decompression. This paper is primarily concerned with these terms as they relate to digital audiovisual files, but the encoding and compressing of information can take place in the analog domain as well, which may be a better entryway to understanding the concepts involved.

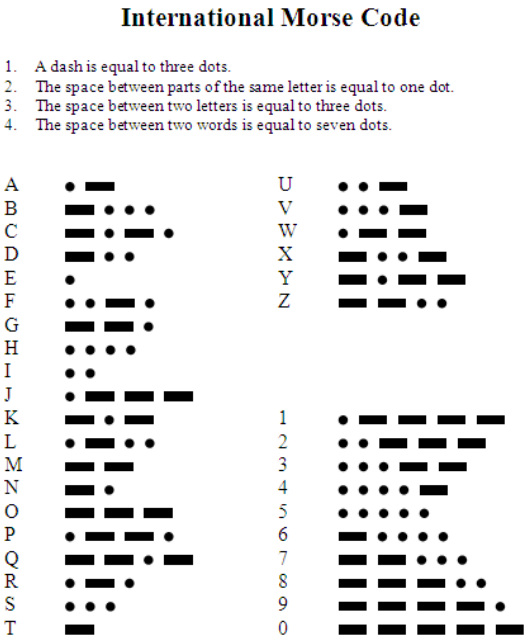
[1] This should not be taken to mean that 'File' or 'File Format' are used to indicate the restriction of codecs to a single stored type.

In that lovely, straight-ahead dictionary speak, encoding is defined as “the conversion of information into code”. In the realm of computer science, code is defined as “the symbolic arrangement of data or instructions in a computer program or the set of such instructions”, but a more general definition is code as a “system used for transmitting messages requiring brevity or secrecy”.



Opening measures to Chopin’s “Raindrop”

Musical notation is an analog world example of code. The system of staff, notes, symbols, and annotations are not music themselves, but they represent a message of embouchure, speed, fingerings, mood, etc. that the musicians decode in order to produce the full message (the song).



Morse is another coding system. Morse code was created to enable more efficient communication by providing a means to express one code (written language) through a seemingly incompatible medium (electronic pulses). In this case, the long/short patterns correspond to specific letters that can be conveyed through sound or image for decoding.

01010111 01101001 01101011
01101001 01110000 01100101
01100100 01101001 01100001

Binary encoding, the basis of modern efficient computing and processing of information, is another popular example.

Compression can be considered as a subtype of coding, as it is defined as “encoding information while reducing the bandwidth or bits required”. It is a form of coding, though not all encoded material is compressed. We can think of compression as a second encoding process, taking coded information and transferring or constraining it to a different, generally more efficient code. Decompression would then be the process of decoding information into the original code where it could be decoded again.

The Unicode Standard for the display of letters and characters is an illustrative example.

Sample text of Russian characters as displayed visually (6 characters):

Москва

Those same characters expressed in Unicode encoding scheme would be represented as the following 6 Unicode code points (6 code points):

041C 043E 0441 043A 0432 0430

Those same Unicode code points could be compressed into a smaller amount of data based on the SC2 compression scheme (7 bytes):

12 9C BE C1 BA B2 B0

In order for the compressed data to decode to the correct characters, it must go through a decompressor that understands the compression scheme and transforms the code back into the Unicode values. The Unicode code points are then decoded by whatever program is reading the text and then are visually displayed as Russian characters.

Compression is constantly being used ‘behind the scenes’ by computers to process information more efficiently as well as being incorporated into tools that are made available to users. A good example of behind the scenes compression is Run Length Encoding . RLE compresses long runs of the same value.

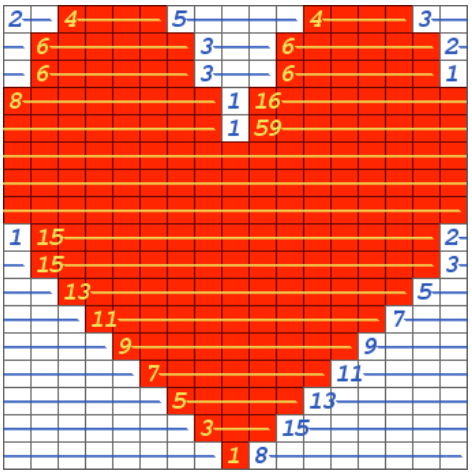


Illustration of Run Length Encoding created by Allon ROTHFARB, posted on Wikimedia Commons

If in the image at left we said that a white box is represented as ‘w’ and a red box as ‘r’, we could represent the first line encoded as:

wwrrrrwwwwwrrrrww

Through Run Length Encoding, we could instead represent the first line as:

2w4r5w4r2w

The compressed value is the same information, but in a form that takes up less disk space and bandwidth. Additionally, as we see in the image, RLE continues counting the entire length of a string of similar content, not ending at each ‘line break’. The fattest part of the heart, making up three lines within the image, could be represented:

59r

instead of as:

rrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrrr

[2] For further explanation see http://www.unicode.org/standard/WhatIsUnicode.html. Examples from “Unicode Technical Standard #6 A Standard Compression Scheme for Unicode” http://unicode.org/reports/tr6/
[3] For further explanation see http://www.fileformat.info/mirror/egff/ch09_03.htm

Also note that RLE is considered mathematically lossless and reversible compression, as it can be decompressed to its exact original value. The input to the encoder and output of the decoder should match exactly bit for bit. This differs from perceptually lossless compression, a scheme for lossless compression whereby there is change to the bit structure of the original data, but that change results in no perceived difference (by most people) between the original and the compressed version.

Perceptually lossless compression is considered lossy because it loses bits from the original permanently – the compression cannot be reversed and the complete information and functionality in the original file cannot be re-obtained if the original is discarded or becomes inaccessible. The data is not just repackaged in a more compact form, but it is actively reduced.

The results of lossy compression are more obvious in forms that do not employ a perceptually lossless scheme (think blocky, pixilated photos), but lossy compression does have its purposes. Most access to streaming media is highly dependent on lossy compression schemes, for example, and content that is deployed over multiple platforms will have variable guidelines on size and quality for optimization in each environment.

Loss of data when trying to maintain accuracy, functionality, provenance, and other preservation issues is a major concern, but may not be as important in other situations where that data is not needed or is considered unessential moving ahead. The makeup of the original material, the targeted use of the new file, and the ultimate goals for the persistence and accuracy of the content will all have an influence on making decisions about file creation.

Compression as an Informed Decision

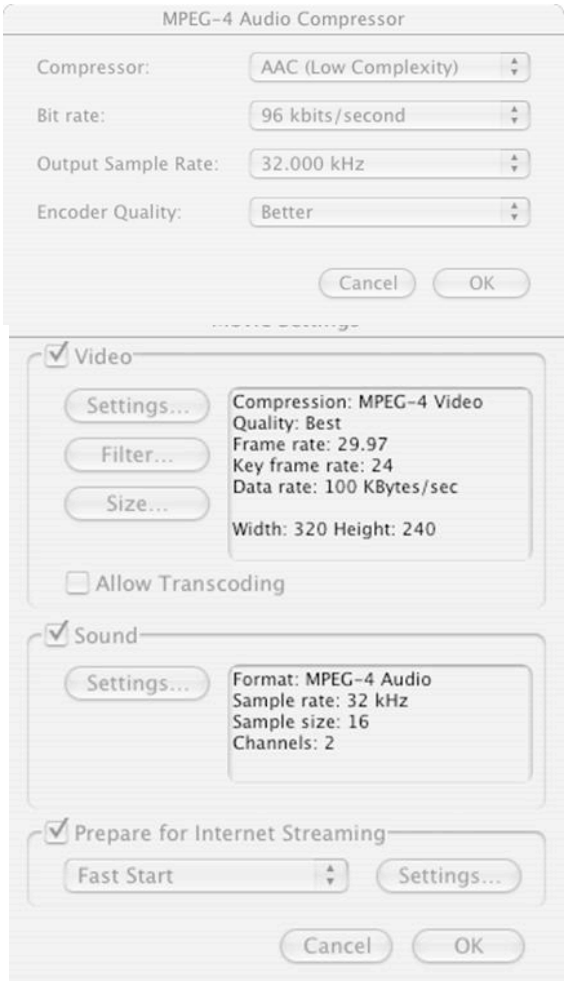
Today a range of encoding and compression choices are put in the hands of the content creator. This could be through the selection of a recording device (audio recorder or video camera):

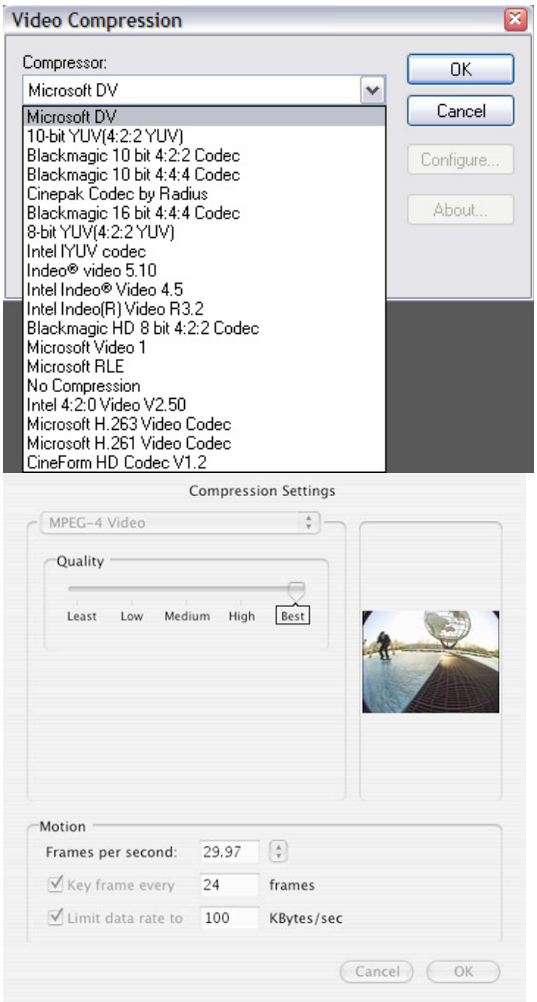
FisherPrice PXL2000 Digital Videocamera posted at http://www.flickr.com/photos/51035560498@N01/493824980/pxl2000

Flip Digital Videocamera posted at http://www.flickr.com/photos/39435232@N00/2513981045/Recording

Sony HDRFX1 Digital Videocamera

and/or through settings within hardware and software:





» Adobe Premiere Compression and Codec options from blogs.adobe.com/bobddv/2006/09/

« Final Cut Pro Movie & Compression Settings from www.vx1000.com/fcpwebcompress.htm

The preceding discussion leads to a couple of fundamental observations.

1. Encoding can take place with or without compression.

2. Compression is not a bad word, but understanding how and where it is used, and how to manage it, does have implications to accuracy and accessibility. Mathematically lossless compression is almost always incorporated into computer data processing routines. The ability of computers to process and store such large amounts of information is dependent upon encoding and compression. It is not the mere existence of encoding and compression schemes that is the concern but rather the type of encoding or compression we as content producers and caretakers choose that can have positive or negative effects on the persistence of data objects and perception of their content. This is why it is so important that we understand these decisions and processes as well as be aware of all points in the lifecycle of an item where choices must be made, from equipment selection and settings at the point of creation to the point of digitization in the reformatting of analog audio or video.

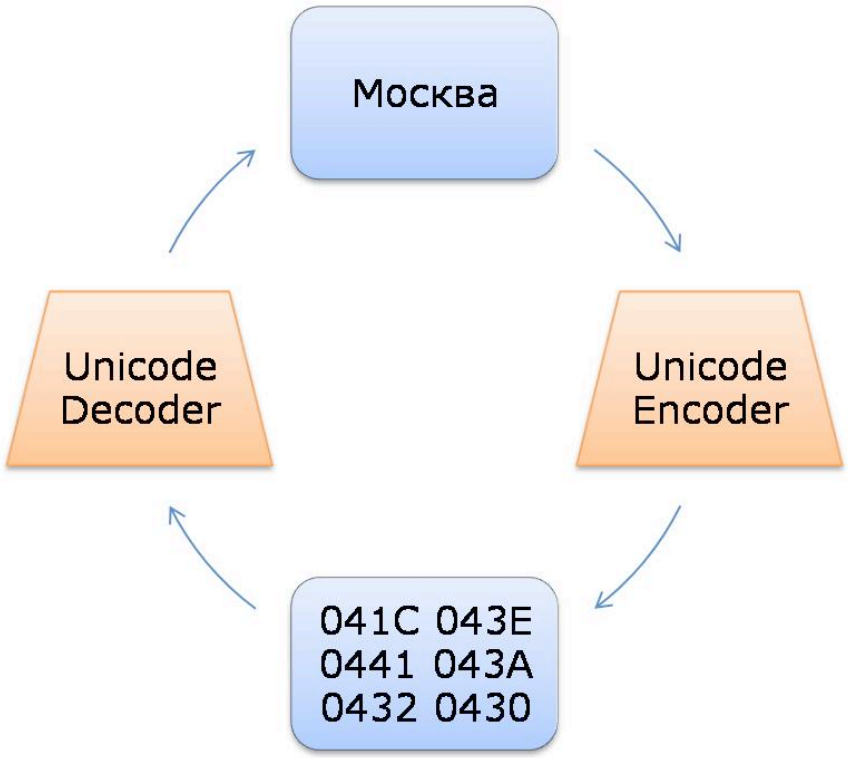
How Encoding Works

Approached schematically, the processes are quite basic.

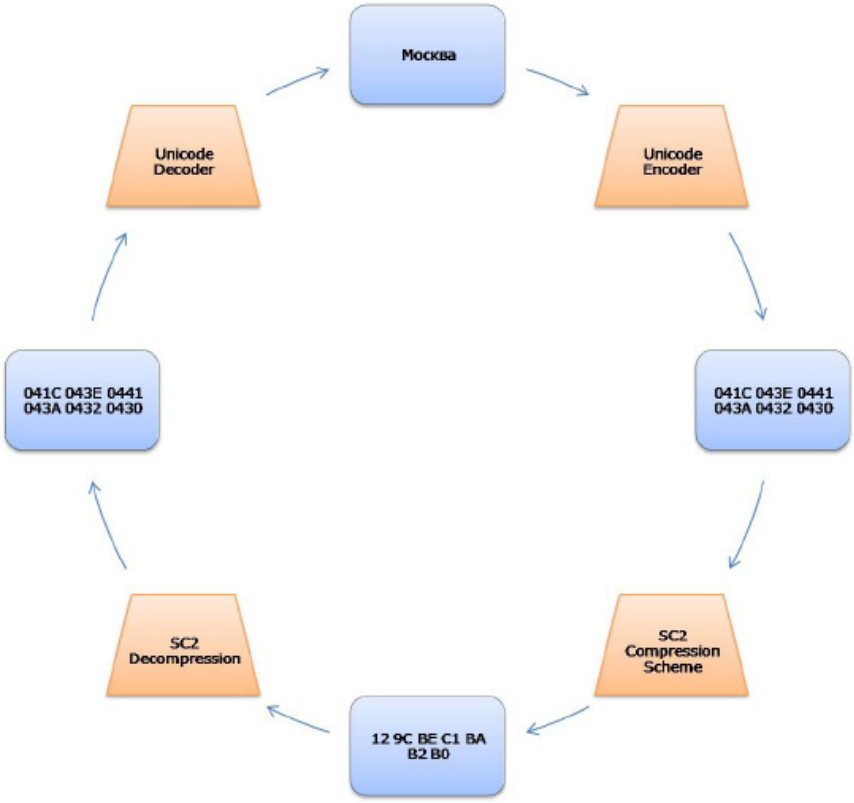
Using the Unicode example from above, we see that in this process we have four primary components - the original data, the encoder, the encoded data, and the decoder.

Encoding Process:

If the encoder uses no compression or lossless compression we retain the ability to recreate original data (as long as the technology and knowledge to do so is sustained).



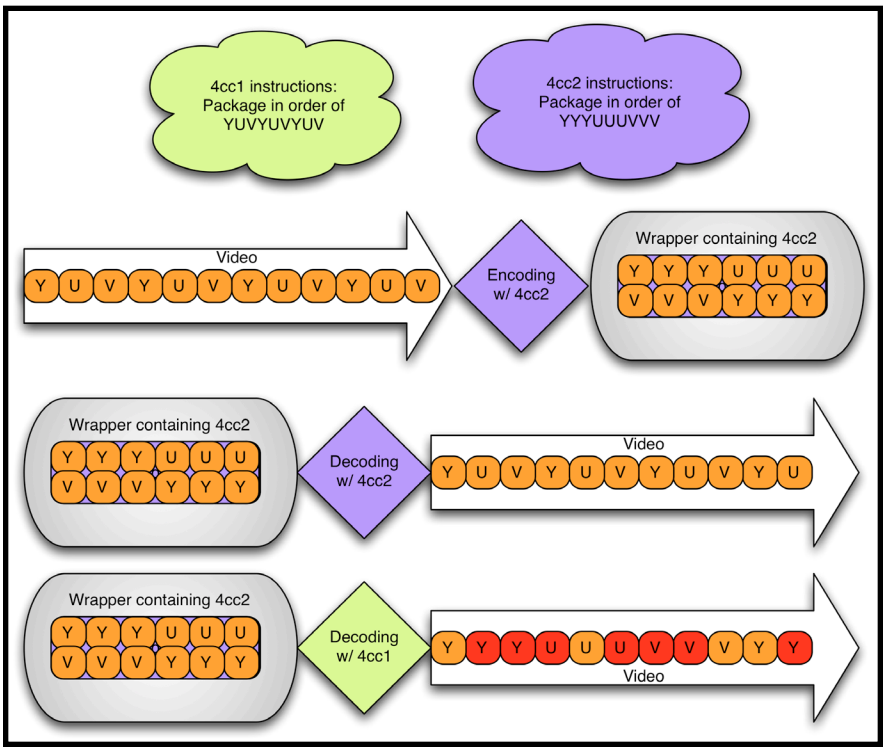
Compression Process: Compression adds another set of transformation points to the cycle.

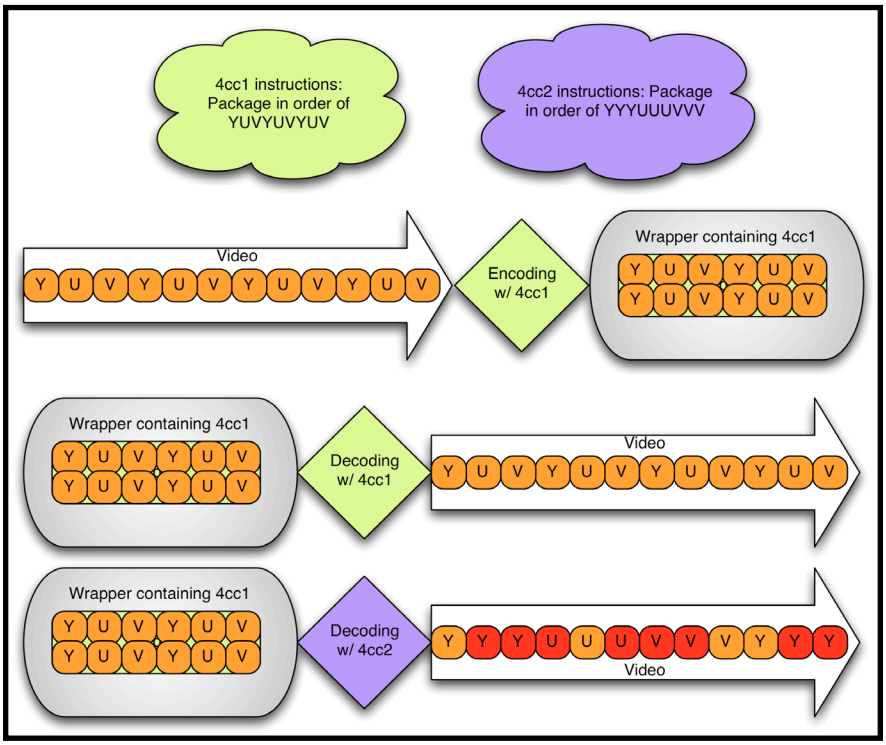


Simple and linear, right? Or it would be, if there were only a single type of data to encode and a single way to encode it. However this is not the case, and there are issues that can arise during the communication and decoding / decompression of the data.

Exchanging Encoded Information

So what happens if the decoder doesn't know how the encoder packaged the data? If the encoder and decoders are human for instance, what happens if they don't speak the same language and they are trying to share Morse code? The encoder, or person reading the text and encoding it as Morse code signals may only speak English. The decoder, or person who receives the Morse code signals and decodes them back into text may only speak Spanish. The encoder can send Morse code signals until their finger goes numb, but the decoder will not be able to entirely decode or understand the message. If the decoder also speaks English or the encoder also speaks Spanish then they share common 'codecs' and will recognize the characters and strings that would allow them to exchange data losslessly.





An encoder is responsible for packaging data according to certain rules. For instance a digital video stream may consist of samples of luma (represented here by 'y') and chroma (represented here by 'u' and 'v') which can be packaged inside a wrapper in variable formations , such as ordering the code as 'yuvyuvyuv' as opposed to 'yyyuuuvvv'. The set of rules which defines how the data is encoded and packaged is what we refer to when we speak about a 'codec', and each codec has a name and an identifier. This identifier consists of four characters, such as 'yuv2', and thus is called a four character code (FourCC). '4cc1' and '4cc2' are used at the left to represent four character codes for hypothetical codecs.

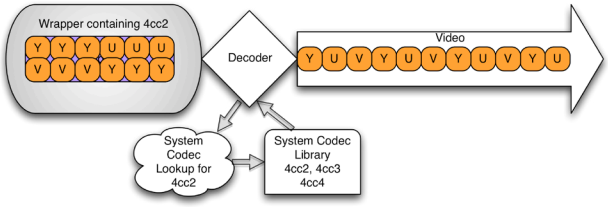
This 4cc is embedded in with the packaged data, either as part of the essence (audio and video) or as part of the container (file or wrapper). In the diagrams we see that the same video stream can be encoded in a different order by the differently defined codecs. This is not a problem as long as the same codec is used to decode the data and unpacks it in the correct order. As the bottom decode in each diagram shows, it is possible for a different codec to access and render the video essence (it, too, understands Y, U, V) and then decode it in the wrong order. Think of the human 'encoder' placing a code before encoding English text saying 'EN-US' to identify it as English US so that the information receiver knows an English 'decoder' is needed to understand it, but also that 'color' is not spelled with a 'u' or that 'football' does not mean soccer. In both cases, the essence of the data is decoded, but it is not rendered faithfully to the original. It is okay to use a different tool to encode and decode the same data (QuickTime/Windows Media, American/British English speaker) as long as the same codec is employed to instruct the tool on proper rendering.

[4] Y, U & V are used here for simplicity. Correct notation for digital luma and chroma samples is Y, Cb & Cr

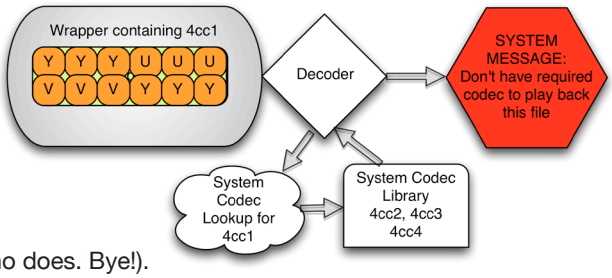
Why Does This Matter for Moving Image and Sound Archives?

Compression and encoding have been used on our video content for years in the form of interlace scanning and the conversion of RGB to YUV for transmission, processing and storage. Just as we need the correct, well-maintained tools to encode/decode this information in order to properly access video, we need to understand codecs and be well aware of their associated best practices for use in order to preserve and maintain access to digital materials.

Generally speaking, when a computer decoder is presented with data one of the things it looks for is the four character code. It then looks to see if it has the associated specification or codec available for application.



To continue the previous analogy, the Spanish speaker seeing EN-US recognizes that he doesn't speak that language. If the Decoder does happen to speak English, the message will be decoded. In this same way, if software/hardware does possess the required codec it follows the associated rules to unpack, or decode the data. If it doesn't you either get nothing or a not-so-helpful message telling you the system doesn't have the required codec and can't play the file back. (I don't speak EN-US. You better go find someone who does. Bye!).



This in and of itself doesn't seem like such a big problem. Feasibly with some available tools (e.g. Exiftool, Dumpster, MediaInfo, ffmpeg, etc) you could inspect the file, discover the four character code and find out what manufacturer creates it by searching around the internet (fourcc.org is always a good place to start).

However this takes a bit of know-how and research. To make matters more complicated no single decoder supports all existing codecs. Even if you find the codec you may not be able to use it with your hardware or software if it does not support – or no longer supports – the particular codec.

When trying to locate the correct codec, it is also reasonable to expect that the more obscure and/or undocumented the codec, the less likely one is to find it available for download. Continuing to use languages as an example, the smaller number of people that speak the language of a given text, the less likely you will be to find the speaker of that language to translate the text in the future. An identifiable codec is not necessarily an available one.

Though no simple feat even for skilled computer scientists, it is feasible that if the codec were well documented you could reverse engineer it and retrieve the data. When dealing with proprietary codecs, this documentation tends to stay obscured or protected from the general public at whatever level a hardware or software manufacturer determines is best to control their rights. However, even in the case of open standards documentation can remain incomplete or insufficiently up to date if it is not fully established or continually maintained. The distinction between proprietary and open codecs is important, but should not be confused with the availability of full, current documentation.

[5] For further explanation see http://en.wikipedia.org/wiki/Interlace
[6] For further explanation see http://en.wikipedia.org/wiki/Chroma_subsampling

What Does This All Mean to Moving Image and Sound Archivists?

This is all very meaningful to the moving image and sound archivist. In almost all cases moving image and sound content is packaged in such a way that it requires a codec for decoding.

The following recommendations should be given serious consideration by anyone overseeing moving image and sound content whether it is born digital or digitized legacy analog materials. (The following recommendations borrow heavily from the overarching framework provided by Carl Fleischhauer and Caroline Arms' Sustainability Factors for Digital Formats. See http://www.digitalpreservation.gov/formats/sustain/sustain.shtml).

These recommendations may be practiced from two different perspectives - one of having influence in the choice of codecs and one of having had no influence. You may use these factors as a guide in cases where you are involved in the selection of codecs for reformatting legacy analog content or new production of born digital content. In other cases you will have no influence, such as accessioning a collection of file-based digital content that you had no involvement with prior to its arrival on your doorstep. In this case you may use these recommendations as a guide to assess risk of the codecs already applied in the accessioned collection.

[7] As a quick pointer VLC and MPlayer are fantastic applications that support a large number of codecs via integration of the library of audiovisual codecs known as libavcodec. These are both freely available and open source.
[8] Generally you are not selecting only codecs, but since that is the topic of this paper they are the focus point of the recommendations here.

Ten Recommendations for Codec Selection and Management

1. Adoption: Choose codecs that are widely adopted by your community and popular manufacturers of hardware and software. Widely adopted codecs may still face obsolescence, but it will tend to occur less rapidly and there will more likely be migration or emulation mechanisms developed out of the wider pool of stakeholders.

2. Disclosure: Primarily choose codecs that are well documented. This means that the specifications and technical data necessary to understand how the codec works are publicly available. This may be in the form of a standard from standards forming bodies such as the Society of Motion Picture and Television Engineers or the Audio Engineering Society. It may also be published and made publicly available from a non-standards forming organization or company. Secondarily choose open source codecs over proprietary codecs. Open or non-proprietary codecs are generally more fully documented and are more likely to have tools available for analyzing the underlying code, but such is not always the case. The established and continuing availability of full documentation is the overriding consideration here.

3. Transparency: The more transparent your content is in an encoded state, the easier it will be to retrieve in the absence of the original decoder. Transparency is the ability to access the raw data using basic tools such as MediaInfo , Dumpster , JHOVE 2.0 or a hex editor . The ability to view the structure provides the opportunity to analyze and assess the data, thus increasing the ability to rebuild a file in case of corruption, degradation or obsolescence.

   Metadata embedded in the file offers another aspect of transparency. The metadata can provide valuable information related to the creation of the file that can contribute to the analysis of authenticity, provenance, quality, structure, and other concerns. However, this is a more complicated factor. Certain codecs do enable greater self documentation than others. For instance DV stores much more structural and technical metadata than MPEG in the encoded stream. This may be a determining factor for selecting between those two if all else is seen as equal, but there may very well be other overriding factors that make it insignificant. More important is assessing the persistence and need of significant embedded metadata when migrating from, or to, the next codec and wrapper.

4. External Dependencies: Choose codecs which are not dependent on one particular system, one vendor, or one particular piece of hardware. In other words an ideal codec is supported by multiple systems and vendors. Technologies change or end production, and companies change business plans or can go out of business. Being locked in to one vendor or hardware solution based on your codec selection runs a higher risk of that codec losing support or of its functional dependencies not being compatible with future systems when migration or emulation becomes necessary.

5. Documentation & Metadata: Incorporate codec selection into statements of work whether performing the reformatting work in-house or outsourcing. Document the codec as part of your technical metadata. Document the environment necessary for reproduction as part of your preservation metadata, as suggested in PREMIS.

6. Pre-Planning: When buying systems for Digitization (e.g. Blackmagic vs. AJA or FCP vs. Premiere) or capture (e.g. cameras, audio recorders) review codec support as part of purchasing criteria.

[9] http://mediainfo.sourceforge.net/en
[10] http://developer.apple.com/quicktime/quicktimeintro/tools/
[11] https://confluence.ucop.edu/display/JHOVE2Info/Home
[12] http://en.wikipedia.org/wiki/Hex_editor - Note of caution here: Hex editors expose and enable editing to the binary code which higher level software programs interpret in order to display the content properly. Any changes made at this base level will affect the interpretation of the code. Changes should not be made without full awareness of the implications to the integrity of the file.

7. Maintenance: Store and maintain codecs representing any significant amount of encoded content in your archive. Also store associated software which will enable retrieval of the encoded content (e.g. playback, parsing of essence and metadata, decoders and transcoders).

8. Obsolescence Monitoring: Monitor obsolescence of codecs in addition to other things such as wrappers -- awareness and maintaining current knowledge of codecs is an essential aspect of moving image and sound archiving.

9. Maintain the original: Whether applying an overall collection management strategy of obsolescence monitoring (only transcoding to a new codec when an object's current codec is at risk of obsolescence) or a strategy of normalization (transcoding all objects to the same preservation standard codec upon ingest), the original object should always be maintained. As technology advances new tools or options are developed for accessing and maintaining content. As long as it is viable – or becomes viable again after assumed obsolescence – the original object may very well be the best source for migration and/or transcoding.

10. Avoid unnecessary transcoding or re-encoding: Migrating digital content to other storage media with the goal of maintaining the format / codecs should be done in such a way that the content is transferred bit for bit (like Firewire) rather than in a way that subjects the content to any transcoding or compression that then re-encodes in the original format on the new storage device. Transcoding and re-encoding changes the original data – potentially leading to loss of integrity, quality and functionality – and should not be done unnecessarily.

Conclusion

Undoubtedly the question remains: "So which codec should I choose"?

The answer: "The right one".

There is no one choice that will meet the needs of every archive. It is important to recognize and remember that preservation is a strategy. Your selection will be informed by a number of factors which you must have a solid understanding of before you can make the correct preservation decisions. These factors include goals and objectives, mission, existing and planned systems within your organization, budgets, functional requirements and uses, scale of the collection in question, and more. Consideration of these and the ten recommendations discussed in this document will help lead you to the right selection.