A new compression technology is required for this new era in media technology. Novel workflows mixing smaller cheaper devices and advanced software that is often cloud based make the requirements of moving bulky imagery more complex and more nuanced. Couple these with new service oriented business models that allow billing per file that is based on size and resources consumed and you end up with the need for a codec that can be locally optimised on a workflow by workflow basis.

We need a high quality video encoder that can extend its range from lossless to high compression ratios. It must be easy to compute, easy to scale in software it must be implemented in repeatable functional blocks in hardware, it must be synergistic with Machine Learning processes, it must be natively multi-resolution, it must be cloud friendly, energy efficient, available now AND an international standard. This white paper explores some of the facts and functionality behind SMPTE ST 2117-1

VC-6 is different

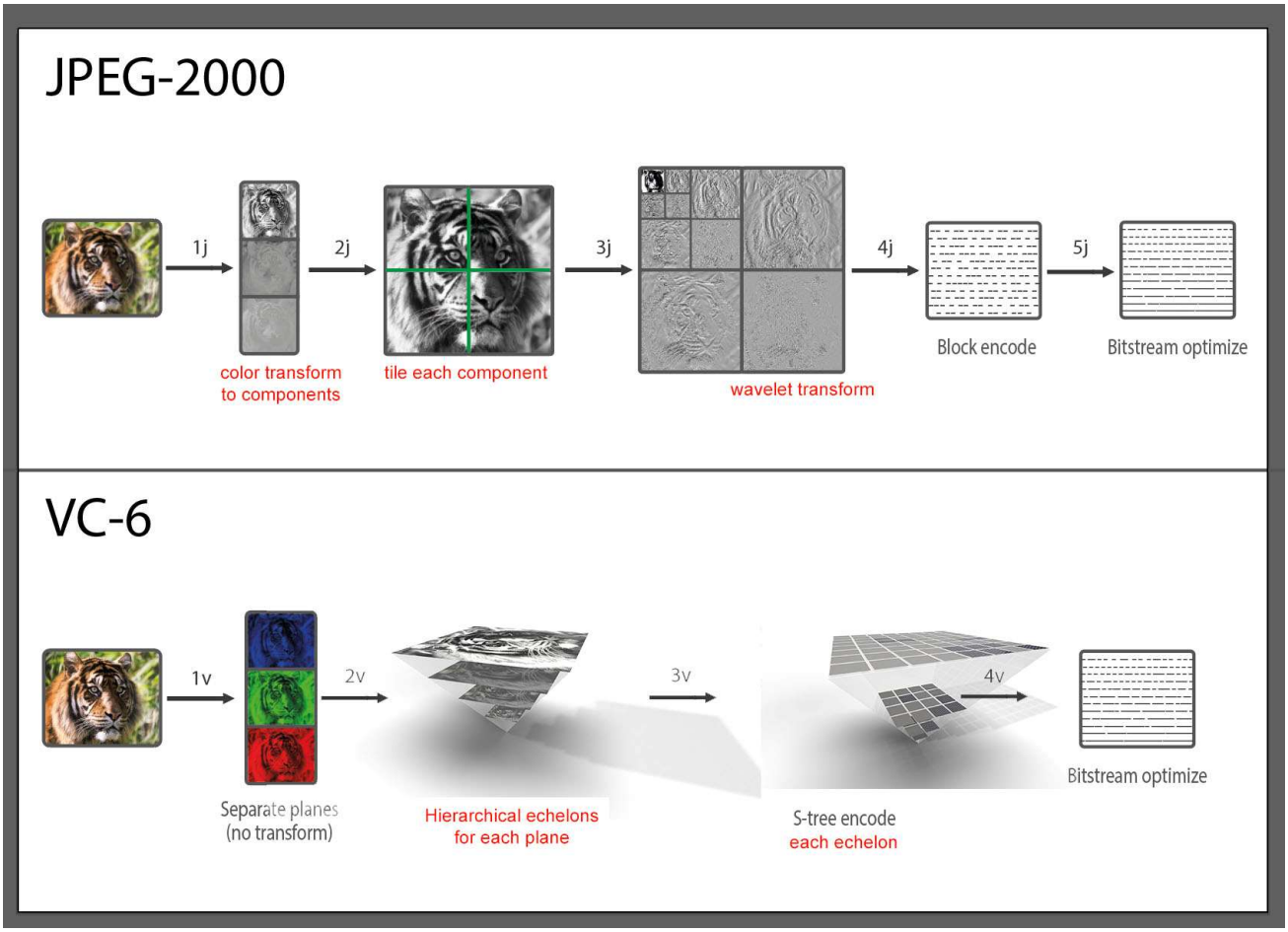

Figure 1 - JPEG2000 vs VC-6

VC-6 shares many features of other coding schemes but also has significant difference. The figure above shows a comparison with JPEG2000 (J2K).

Planes: VC-6 codes the original planes of the image to ensure there is not chance of mathematical error. J2K converts to a luminance plane and color difference planes. This improves . compression ratio but requires two different color transforms - one for lossy and a different one for lossless compression.

Decomposition - VC-6 creates a hierarchical set of images. The tiniest one comtains a grid of pixels. Each higher resolution grid contains differences between its upsampled neighbour and the downsampled original.

J2K takes a different approach. It tiles the image (to improve parallel processing) and then uses a wavelet transform to create frequency-like grids for each hierarchical resolution.

Encode - VC-6 uses a hierarchical s-tree structure to encode each grid. The goal is to use s-tree metadata to predict large numbers of grid values from the lower layers of the tree.

This is intrinsically a highly parallel process. J2K uses a block entropy encode to make sequences of encoded blocks that are as short as possible.

Optimize - Both schemes allow the bitstream to be rearranged in an application specific way so that lower resolutions could be stored in a different file or for proxy data to be at the beginning of each image.

You will be familiar with the generic transform coding block diagram in the panel. Most transform based codecs reversibly convert a block of pixels into a block of data that represents the frequency components of the image. These frequency components are a bit like the building blocks of an image. A small block of pixels might contain only a bit of sky or just the vertical edge of a building. These map to a very small number of frequency components and therefore we can use techniques like quantisation (reducing the number of bits in a value) or entropy encoding (using statistics to send the most common data with the fewest number of bits) to further reduce the bitrate

Transform codecs have dominated the picture compression landscape with standards like MPEG and JPEG delivering decades of advances in deployed implementations. Anyone watching the trends will have spotted that each new generation takes more compute power and has more complexity with greater memory requirements than the generation before. The results are excellent, but in an era where the carbon footprint of 4k encoding might become a KPI for a business, it is probably time to think about the possibility of a new approach. From the panel, you can see that VC-6 is based on hierarchical, repeatable s-tree structures that are similar to a long practiced technique called quadtrees. The compression scheme works by creating a number of different resolutions of an image and then looking at the residual differences between the higher and lower resolutions. The aim of the exercise is to then encode these residual values as cheaply as possible by using small tree-like structures to represent patches of residual values and to match them with other patches of residual values in the difference image. Through extensive research, we know that this is a very efficient form of entropy encoding.

These simple tree structures provide intrinsic capabilities that are well suited to the modern computing landscape, such as massive parallelism and the ability to choose the type of filtering used to convert between image resolutions. This gives a decoder the ability to balance image fidelity and power consumption that is key in mobile and high volume applications.

Where VC-6 shines

The VC-6 codec is optimized for intermediate, mezzanine or contribution coding applications. Typically, these applications involve compressing finished compositions for editing, contribution, primary distribution, archiving and other applications where it is necessary to preserve image quality as close to the original as possible, whilst reducing bitrates, and optimizing processing, power and storage requirements. VC-6, like other codecs in this category uses only intra-frame compressions, where each frame is stored independently and can be decoded with no dependencies on any other frame.

A key feature of the codec is the ability to navigate spatially within the VC-6 bitstream at multiple levels or resolution. In plain English, this means that a single bitstream can be decoded at multiple, different resolutions within the decoder. Another application of the same property would be to store the different resolutions and enhancements as different files (or objects) and vary which ones are used for reconstructing the image based on the properties of the network or environment in which the decider finds itself. Think of it like a mezzanine decoder with a built-in HTTP Streaming ladder for professional applications. Yet another view of the same feature is to provides the ability for a decoding devices to apply more resources to different regions of the image allowing for Region-of-Interest applications to operate on compressed bitstreams without requiring a decode of the full-resolution image.

VC-6 performs really well as images get bigger. 4k, 8k and higher resolution images are easily handled and show great compression ratios for any given quality. The intrinsic simplicity of the S-Tree computations also show that encoding and decoding of a given image at a given fidelity is faster and cheaper than transform coding alternatives. The added benefit of being able to go up or down in resolution within the codec reduces the number of pre- and post-processing filters which further reduces the computation load in multi-resolution and proxy workflows.

VR and AR applications are particularly suited to this kind of compression where an encoder might compress a particular scene with a single bitstream covering the overall field of view. A low latency decoder can navigate within the bitstream to locate the portion of the high resolution master that the viewer is inspecting and apply more resources to decoding that portion of the image than it would apply to the peripheral vision of the viewer. As the viewer moves their head, the decoder continues to perform the same algorithm and allow cheaper, lower power decoders to perform with excellent fidelity in a system where the master bitstream is the best representation of the scene.

Contribution links from venues have typically been created at the highest resolution available and local upconverters and downconverter have been used to match the resolution of transmitted image to local requirements. As the world moves towards IP links, a codec like VC-6 removes the need for this extra processing step as the received bitstream can be multicast to many end points through multiple trust boundaries with the required image resolution being determined at the point of usage and not by some upstream decision mandated by the availability of down-convertors. This added level of flexibility by combining the properties

of VC-6 with IP routing brings an extra level of dynamic creativity to event coverage where ultimate image quality can be maintained to more devices for less cost.

Meeting changing requirements

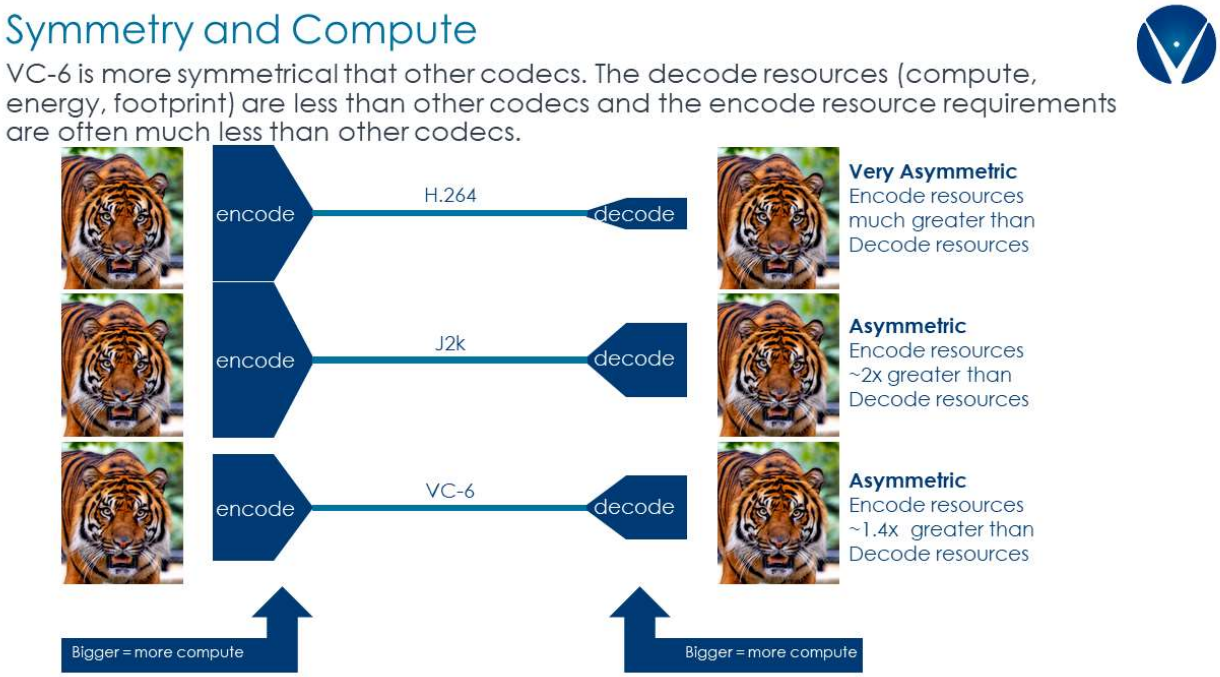

Figure 2 - Symmetry vs Compute

One of the major benefits of transform encoding during its emerging dominance back in the 1990s was the highly asymmetric nature of the encode / decode pipeline. At the time, this was great because the industry was battling semiconductor physics to get chips that were big enough, fast enough and cheap enough to handle SD & HD images in real time.

Now we have passed 2020, the world is different. We carry 4k battery powered decoders in our pocket and we can do real time 4k effects on that pocket device that look great and have the added benefit of warming your hands on a cold day. We regularly have 2 way video calls and the highly asymmetric benefits of the transform coding paradigm no longer look as attractive as they did 30 years ago.

VC-6 is much more symmetric than traditional transform codecs. Its intra-frame nature means that encode and decode delays are small, quality is high, bitrates are low and overall compute costs are small. This makes it a great fit for any bidirectional link whether you're a broadcaster of creating a drone that needs the best pictures without draining the battery.

Applications of the VC-6 multi-resolution features are particularly appropriate for the new revolution in Machine Learning and Artificial Intelligence systems. The ability to decode a particular region of interest at a particular resolution can significantly increase the throughput of image indexing and segmentation systems, whilst simultaneously reducing the costs and bandwidth consumed during the process. This looks particularly appealing in applications that are shot at high resolution in remote locations yet need high quality and rapid metadata indexing.

It is almost too obvious to point out that VC-6 is intrinsically suited to proxy workflows where different users might need to balance their needs against the constraints of networks and devices. Once upon a time you would have to settle for a one-shoe-fits all approach and choose one, or maybe two proxy settings for a facility. With VC-6 and some intelligent bitstream manipulation, the ability to dynamically manage proxies based on current network conditions becomes possible. In effect, the encoder produces a single bitstream and the decoder will decode whatever it's given. A bitstream processor in a closed network environment can then manage the bitstreams based on policies. The editor checking focus and selecting shots has a different technical need to the subtitling department. No more re-encoding needed.

Figure 3 - Hierarchical Encoding creates echelons of different resolutions

Multiple resolutions of the image (called echelons in Figure 3 above) are encoded using S-Tree structures that minimise the amount of data to be sent. The VC-6 encoding scheme has complete freedom to choose the number of echelons and the resolution of each echeleon to optimise the bitrate and the bistream properties for a particular application.

[ ](vc6-s-tree-nodata-shorts2(hd).png

Figure 4 - Each echelon is encoded using an s-tree structure

The S-Trees predict neighbouring pixels as well as pixels in higher resolutions based on what is happening in a local region of the image. This can be done efficiently at scale in software and hardware. Figure 4 above shows an s-tree with no data being carried in any of its nodes.

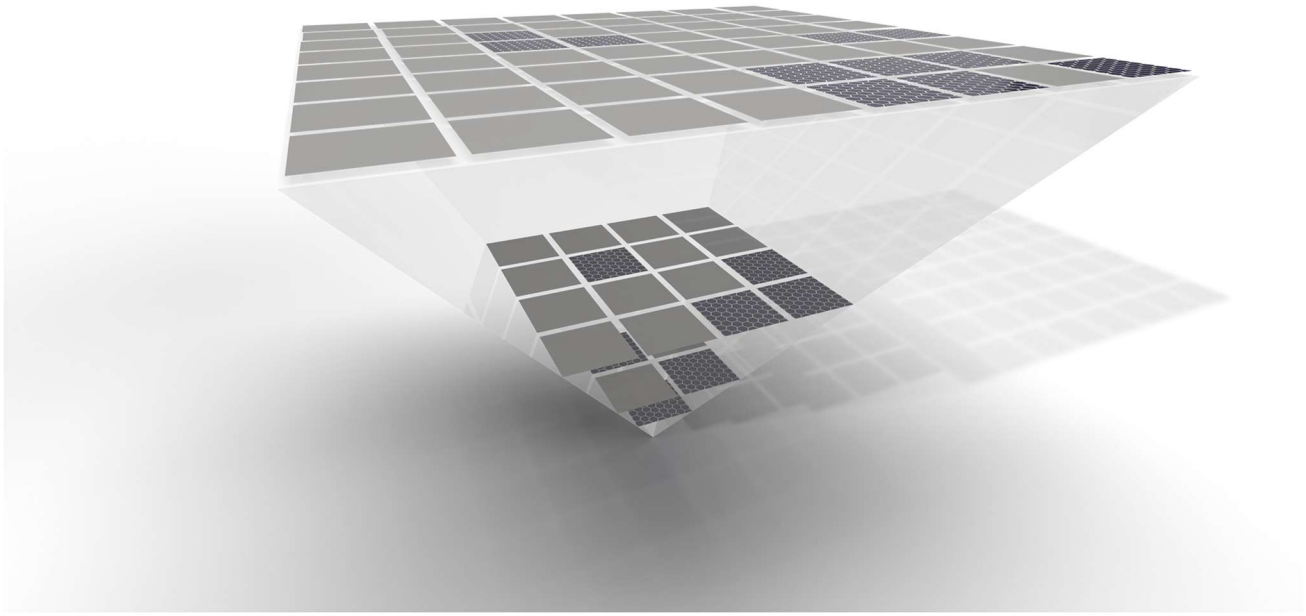

Figure 5 - An s-tree carries data in nodes that cannot be predicted

Figure 5 above shows that some of the nodes now carry data. Within the s-tree, the layers with more data are predicted from the layers with less data. These hierarchical structures are then used to predict echcelons that are finall brought together to predict the final image at a desired resolution.

The best way to understand VC-6 is to consider the standardised bitstream. The job of a decoder is to recreate images from the standard bitstream and the job of an encoder is to create a compliant the bitstream. In each case, there are many different ways that a bitstream can be created and many ways in which processes can be concatenated to recover the images. The bitstream is the common language between those two sets of processes.

VC-6 is an example of intra-frame coding, where each picture is coded without referencing other pictures. It is also intra-plane, where no information from one plane (e.g. Red or Green or Blue or Alpha) is used to predict another plane. As a result, the VC-6 bitstream contains all of the information for all of the planes of a single image. An image sequence is created by concatenating the bitstreams for multiple images, or by packaging them in a container such as MXF or QuickTime or Matroska.

The VC-6 bitstream is defined in the SMPTE ST 2117-1 standard by pseudo code, and a reference decoder has been demonstrated based on that definition. The header is the only fixed structure defined by the standard. Decoding the header allows you to build a number of structures that are used to rebuild a compressed image. The structures and concepts are touched on here:

Planes

A plane is simply one of the components of the image to be reconstructed. Planes might be RGB or RGBA pixels originating in a camera, YCbCr pixels from a conventional TV-centric video source or some other planes of data or metadata. There may be up to 255 independent planes of data, and each plane can have a grid of data values of dimensions up to 65535 x 65535. The SMPTE ST 2117-1 standard focuses on compressing planes of data values, typically pixels.

Figure 6 - An image decomposed into RGB planes

To compress and decompress the data in each plane, VC-6 uses hierarchical representations of small tree-like structure that carry metadata used to predict other trees. There are 3 fundamental structures repeated in each plane.

S-tree

The core compression structure in VC-6 is the s-tree. It is similar to the quadtree structure common in other compression schemes. An s-tree is comprised nodes arranged in a tree structure, where each node links to 4 nodes in the next layer. The total number of layers above the root node is known as the **rise** of the **s-tree**. Compression is achieved in an s-tree by using metadata to signal whether levels can be predicted with selective carrying of enhancement data in the bitstream. The more data that can be predicted, the less information that is sent, and the better the compression ratio. VC-6 has metadata that allows large numbers of these S-trees to be omitted from the most detailed layers. Not sending any data at all for an area of the picture is the best compression that you can get.

Tableau

VC-6 defines a tableau as the root node, or the highest layer of an **s-tree**, that contains nodes for another s-tree. You can think of tableaux as inter-linked s-tress that are arranged in layers where metadata tells you that higher layers are either predicted or transmitted in the bitstream. The more prediction metadata that is used, the better the compression.

Echelon

The hierarchical **s-tree** and **tableau** structures are used to carry enhancement data (called *resid-vals*) and other metadata that needs to be carried in the bitstream payload. The final hierarchical tool is an ability to arrange the tableaux for different resolutions of each plane of the image. The tableaux for a given plane at a given resolution is known as an **echelon**. Each **echelon** is assigned a numerical **index** value, where a more negative index is a low resolution and a more positive index indicates a higher resolution. The reconstructed pixels for any given resolution are used as predictors for higher resolutions..

The VC-6 standard defines a list of up-samplers to scale-up the results of the dequantization for the echelon above. The upsampler to be used for perfect reconstruction is specified in the bitstream header, but a decoder is free to choose an upsampler that might be more suited to its needs. For example a low power phone might trade a bit of image quality for lower power consumption by choosing a simpler upsampler.

VC-6 compared to other Mezzanines

There are many mezzanine codec options. Some of which are tied to manufacturers, some are high quality variants of consumer codec, Some are old and some, like VC-6 are new. It is a tricky job to compare different codecs and to ensure that the comparison is fair. One way we thought would be fair is to take a well known high quality, compute-efficient codec and compare it to VC-6. Given the choice of JPEG2000, AVC, FFV1, HEVCI, ProRes DNxHR and others, we know that there are lots of comparisons available to anyone skilled at the art of searching the internet. ProRes is known to be a leader in both quality and computational speed.

We took a typical UHD test source (YUV 422 10bit at 3840 x 2160) and ran decoder tests at various resolutions on an Intel i7-8850H with 16GB RAM and with an NVIDIA Quadro P600 Mobile processor with 2GB VRAM. To make the test fair, the images were compressed at 2.47 bits per pixel. The table in the breakout shows the Decode Energy Usage in Joules based on the Total Dissipated Power of the processor for each image. A smaller number indicates less energy consumption

Resolution	ProRes (CPU)	VC-6 (CPU)	VC-6 GPU
3840 x 2160	1845	711	432
1920 x 1080	1125	297	167
960 x 540	900	167	92
480 x 270	900	122	68
240 x 135	N/A	108	64

Like all tests, your results will vary depending on the number of cores, optimisation of each bit of code, other concurrent processes on the Operating System.

How to try VC-6

Get in touch with V-Nova who have all the texhnical resources you need. Alternatively, buy the Standard SMPTE ST 2117-1 from SMPTE. You can also check out this video on Youtube.

About V-Nova

V-Nova, a London based IP and software company, is dedicated to improving data compression by building a vast portfolio of innovative technologies based on the game-changing use of AI and parallel processing for data, video, imaging, point cloud compression, with applications across several verticals.

This is achieved through deep-science R&D (300+ international patents) and the development of products that test, prove and continuously enhance the technology portfolio.

V-Nova LCEVC is the industry's first highly optimized library for encoding and decoding enhanced video streams with MPEG-5 Part 2, Low-Complexity Enhancement Video Coding (LCEVC). V-Nova VC-6 is a high-performance AI-powered software library for SMPTE VC-6 (ST-2117-1) which is used primarily for professional production workflows and imaging applications.

V-Nova has developed multiple award-winning software products to kickstart the ecosystems for its technologies and allow their immediate deployment, addressing use cases in TV, media, entertainment, social networks, eCommerce, ad-tech, security, aerospace, defence, automotive and gaming. V-Nova's business model is to monetize its technologies through software licensing, IP royalties and product sales.