Slide 2 - Introduction

Slide notes: Welcome to Scalable Video Coding, a module in the Polycom Fundamentals series. In this short module we will talk some more about how video is coded, and in particular about a technology called Scalable Video Coding, or SVC.

This module is approximately 8 minutes long.
Slide 3 - Coding and Compression

Slide notes: We have already covered how the video and audio are converted to signals and compressed for transmission using standards developed to ensure as many endpoints as possible can decode the signals. What we're going to look at is how the signal is compressed.

A simple way of thinking of compression is to consider transmitting a snapshot of the picture (a frame) taken at a given point in time. From that, a number of snapshots are taken and transmitted, but only of the changes which have occurred. Then after a set time, another full frame is sent. This process is often depicted like this, where the red slides represent the full frames, and the blue represent the picture changes only. The arrows show the picture building in two ways; with the frames adjoining and with each picture change adjoining.

By not sending each picture in its entirety in this way, a lot of resources can be saved. It works because even if all the changes are lost, another full picture is coming shortly (we are talking in fractions of a second here – bear in mind that a video call may be made at up to 60 frames per second) so the picture might freeze but then rebuild – something seen in a video conference when the picture flickers, goes blurry and then sharpens again.
Slide 4 - Coding and Compression 2

Slide notes: This sending of frames to make up a moving picture is known as a stream, and in a point to point call there would be one stream going in each direction. Although a multipoint call might seem more complex, it actually isn’t; the near end sends a stream to the far end (in this case, an MCU). The MCU sends a single stream to each endpoint which is a composite image it creates of all the incoming streams. So technically there is still only one stream going to and from each endpoint.

This technique for sending video is known as Advanced Video Coding (AVC), which is part of the H.264 standard, and it uses a process known as temporal scalability – temporal means to do with time. The reason it is described as scalability is because the higher the frame rate of the call, the more full frames are sent, so the process of AVC scales depending upon the requirement.
Slide 5 - SVC Basics

Slide notes: So that’s what AVC is all about, and what about SVC? Well, SVC, or Scalable Video Coding to use its full name, has since been developed, also as part of the H.264 standard. It’s more complicated to explain than AVC, but the simplest way of describing it is that instead of one stream, SVC sends three. This may sound like it needs more network resources to do, but it’s actually more efficient, and here’s why.

The three streams are all carrying different information, each known as layers and all of which have scalable properties like AVC. The first layer is called the base layer, and provides basic information, let’s say a frame rate at 15 frames per second. In addition to the base layer there are enhancement layers, which can add additional detail. You can see here the first enhancement layer is adding information enabling a frame rate of 30 frames per second, and the second enhancement layer is enabling a frame rate of 60 frames per second.

So the concept is that a device which is SVC-capable can transmit three versions of the same thing, and the capabilities of the device at the far end will control what the far end participants will see.
Slide 6 - SVC Basics 2

Slide notes: The first layer provides temporal scalability, like AVC, controlling transmission in frames per second.

The second layer provides spatial scalability, which enables lower quality pictures to predict what higher quality pictures would look like, controlling picture resolution such as SD and HD.

The third layer provides quality scalability, which means that the picture is created in different qualities. When we talk about quality, what we mean is how like the original image is to the image received. This isn't the same thing as resolution in the second layer, it's the controlling of the sample rate used in transmission. A lower sample rate will use less bandwidth, but will be a lower quality at the far end. The more bandwidth allocated to this, the better quality the image, and the more true to the original image the far end will see. The lower the quality here, the more we see of what we call noise, which is just unwanted data entering the stream, often adding imperfections to the picture.
Slide 7 - AVC vs SVC

Slide notes: So that's what SVC is. Now we'll compare how AVC and SVC work in a video call. We know that in an AVC call one endpoint dials the other, they go through the capabilities exchange and agree upon the codecs, resolution and bandwidth they will use for the call. Following this a single stream is sent between the two endpoints. Even if there is content being shared in the call, it is still a single stream, it's just subdivided to make space for the content.
Slide 8 - AVC vs SVC

Slide notes: SVC is different in that although the endpoints still complete the capabilities exchange, after that things get a bit more interesting. Instead of the single stream sent by the AVC coding method, three streams are sent - temporal, controlling frame rate; spatial, controlling resolution; and quality, controlling noise.
Slide 9 - AVC vs SVC 3

Slide notes: This might on first glance appear more complicated than AVC, but when we start to look at SVC in a multipoint environment we start to see why it's such a powerful method of video transmission. Remember in an AVC multipoint call, all the endpoints dial in and go through the capabilities exchange with the MCU as I said before. A single stream is then established between the MCU and each individual endpoint, with the MCU transcoding where necessary to match the capabilities of endpoints so that they can all join the call.

However, in an SVC environment, although the endpoints all dial in and go through the capabilities exchange, due to the multiple streams being generated already there is no need for transcoding.

This picture shows an example of an endpoint generating layers for four different video resolutions. The MCU takes them in and just feeds the other endpoints whichever layers they can support, with all the layers going to the endpoint which can also support 1080p, and only the lowest layer for the mobile device which can support the lowest resolution. Note that all the layers are required for the 1080p stream; don't forget that the enhancement layers only add additional detail, not an entirely new stream.
Slide 10 - AVC vs SVC 4

Slide notes: Another comparison is in the end-user experience. The picture on the left shows an AVC multipoint conference. The MCU has encoded the four separate participants into a single video stream at 720p resolution and the receiving endpoint is able to display the video in full resolution.

The picture on the right shows an SVC multipoint conference. The same participants are in the call, but each is actually a separate VGA stream delivered directly to the endpoint. This means that as long as the endpoint capabilities allow, these separate streams can be arranged how each participant wants to see them.
Slide 11 - SVC Benefits

Slide notes: So with all this in mind, SVC is a very flexible and efficient way of using the resources available within an endpoint or MCU. Transcoding takes a lot of processing and traditionally MCUs have needed a great deal of power to manage this, as well as the considerable costs associated with upgrading the hardware required to process the audio and video quickly enough to maintain the real-time experience.

Using SVC starts to change the game from an infrastructure point of view; as no transcoding is required, the processing power required by the MCU is not as costly to upgrade, specialist hardware is no longer required so it becomes simpler to use an appropriate server instead.

This is tremendously beneficial as we continue to expand across all types of video collaboration usage, and as this is all part of the H.264 protocol, the existing benefits such as Lost Packet Recovery all still apply.