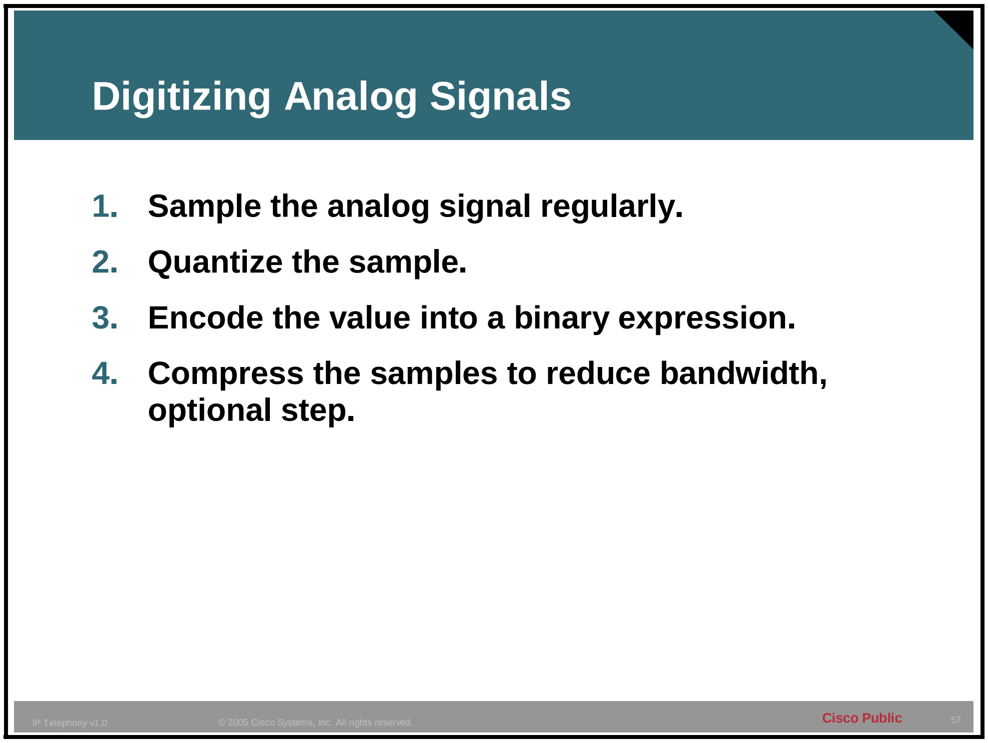
Analog-to-Digital Voice Encoding

Basic Voice Encoding: Converting Analog to Digital

This topic describes the process of converting analog signals to digital signals.



Digitizing speech was a project first undertaken by the Bell System in the 1950s. The original purpose of digitizing speech was to deploy more voice circuits with a smaller number of wires. This evolved into the T1 and E1 transmission methods of today.

To convert an analog signal to a digital signal, you must perform these steps:

Analog-to-Digital Signal Conversion

Step	Procedure	Description
1.	Sample the analog signal regularly.	The sampling rate must be twice the highest frequency to produce playback that appears neither choppy nor too smooth.
2.	Quantize the sample.	Quantization consists of a scale made up of eight major divisions or chords. Each chord is subdivided into 16 equally spaced steps. The chords are not equally spaced but are actually finest near the origin. Steps are equal within the chords but different when they are compared between the chords. Finer graduations at the origin result in less distortion for low-level tones.
3.	Encode the value into 8-bit digital form.	PBX output is a continuous analog voice waveform. T1 digital voice is a snapshot of the wave encoded in ones and zeros.
4.	(Optional) Compress the samples to reduce bandwidth.	Although not essential to convert analog signals to digital, signal compression is widely used to reduce bandwidth.

The three components in the analog-to-digital conversion process are further described as follows:

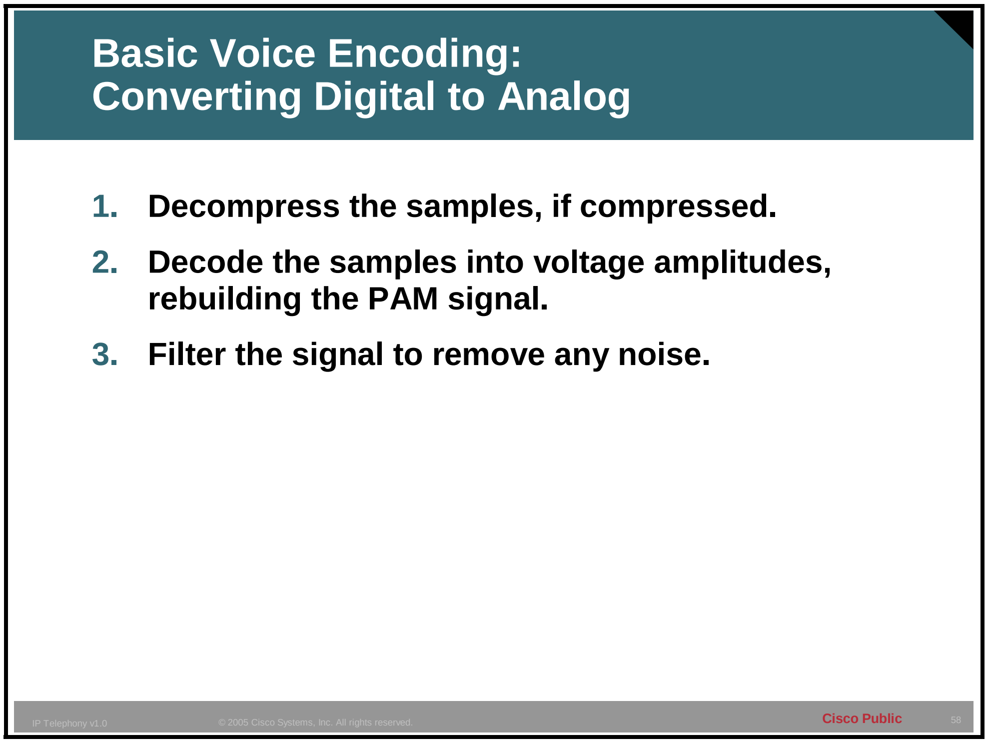
- **Sampling:** Sample the analog signal at periodic intervals. The output of sampling is a pulse amplitude modulation (PAM) signal.
- **Quantization:** Match the PAM signal to a segmented scale. This scale measures the amplitude (height) of the PAM signal and assigns an integer number to define that amplitude.
- **Encoding:** Convert the integer base-10 number to a binary number. The output of encoding is a binary expression in which each bit is either a 1 (pulse) or a 0 (no pulse).

This three-step process is repeated 8000 times per second for telephone voice-channel service. Use the fourth optional step—compression—to save bandwidth. This optional step allows a single channel to carry more voice calls.

Note The most commonly used method of converting analog to digital is pulse code modulation (PCM).

Basic Voice Encoding: Converting Digital to Analog

This topic describes the process of converting digital signals back to analog signals.



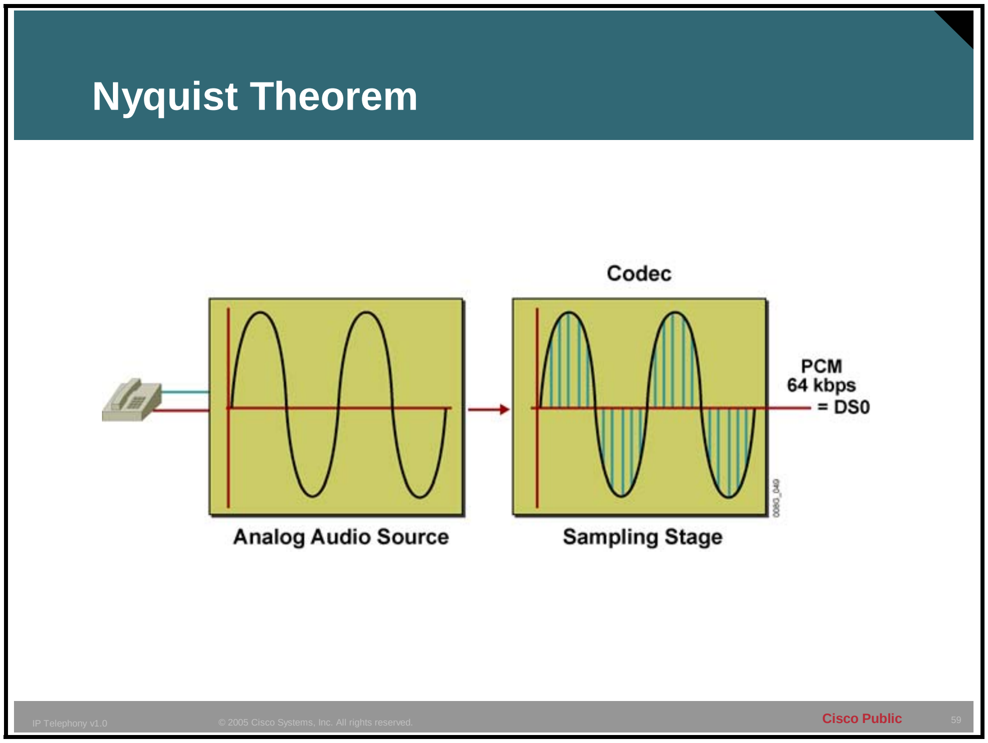
After the receiving terminal at the far end receives the digital PCM signal, it must convert the PCM signal back into an analog signal.

The process of converting digital signals back into analog signals includes the following two steps:

- **Decoding:** The received 8-bit word is decoded to recover the number that defines the amplitude of that sample. This information is used to rebuild a PAM signal of the original amplitude. This process is simply the reverse of the analog-to-digital conversion.
- **Filtering:** The PAM signal is passed through a properly designed filter that reconstructs the original analog wave form from its digitally coded counterpart.

The Nyquist Theorem

This topic describes the Nyquist Theorem, which is the basis for digital signal technology.



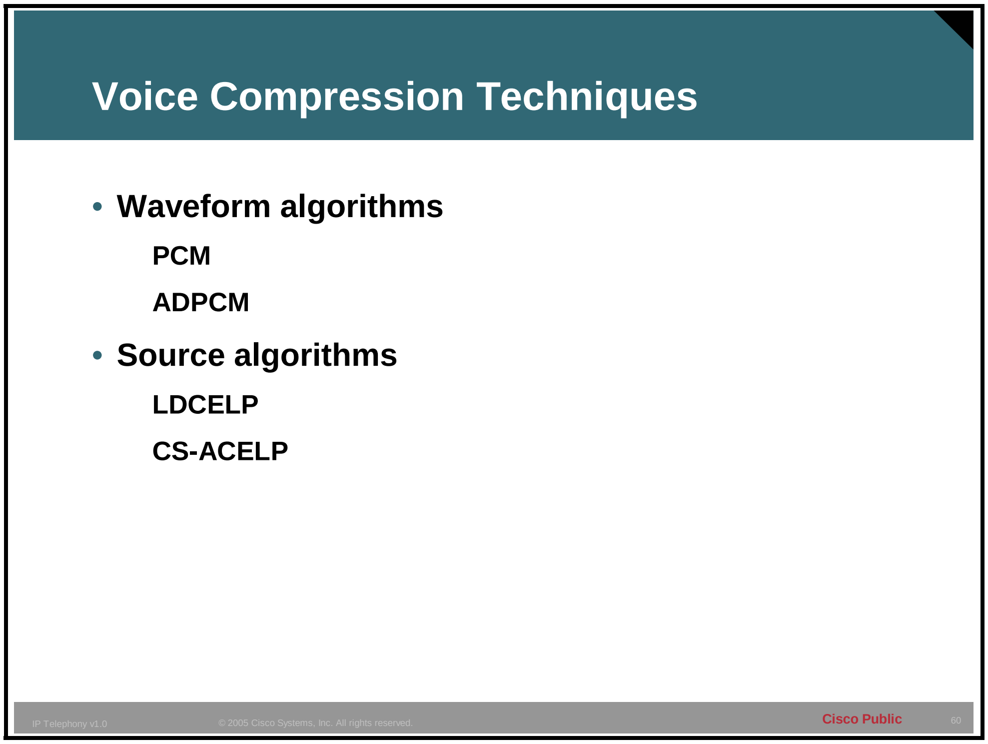
Digital signal technology is based on the premise stated in the Nyquist Theorem: when a signal is instantaneously sampled at the transmitter in regular intervals and has a rate of at least twice the highest channel frequency, then the samples will contain sufficient information to allow an accurate reconstruction of the signal at the receiver.

Example: Nyquist Theorem

While the human ear can sense sounds from 20 to 20,000 Hz, and speech encompasses sounds from about 200 to 9000 Hz, the telephone channel was designed to operate at about 300 to 3400 Hz. This economical range carries enough fidelity to allow callers to identify the party at the far end and sense their mood. Nyquist decided to extend the digitization to 4000 Hz, to capture higher-frequency sounds that the telephone channel may deliver. Therefore, the highest frequency for voice is 4000 Hz, or 8000 samples per second; that is, one sample every 125 microseconds.

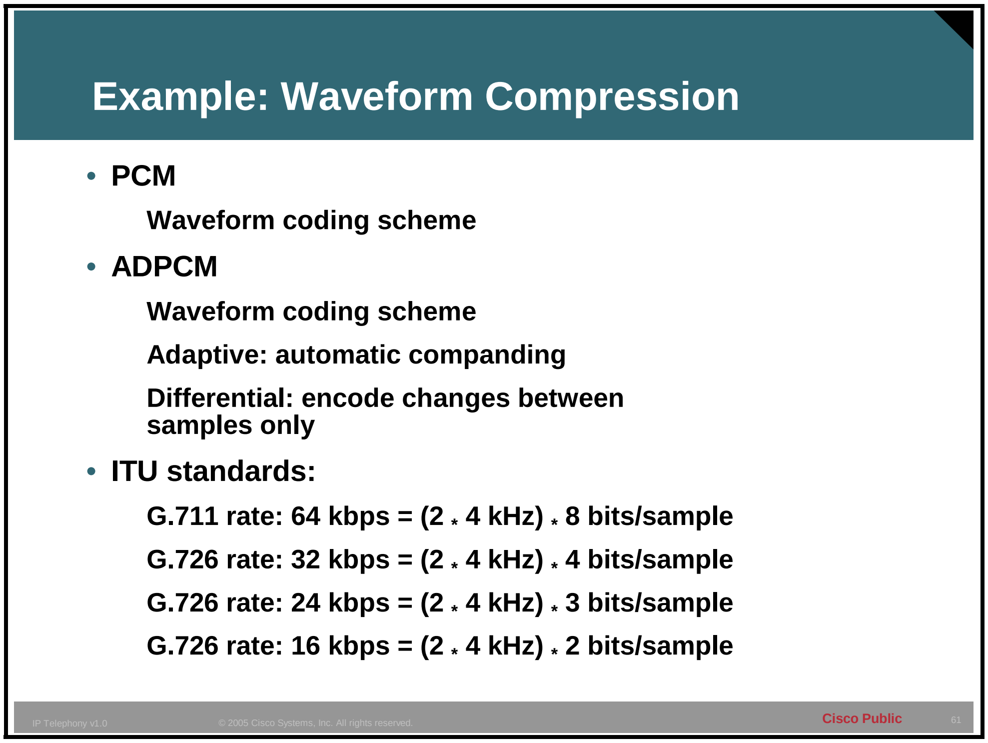
Voice Compression and Codec Standards

This topic describes two types of voice-compression schemes: waveform coding and source coding.



The following describes the two voice compression techniques:

- **Waveform algorithms (coders):** Waveform algorithms have the following functions and characteristics:
  - Sample analog signals at 8000 times per second
  - Use predictive differential methods to reduce bandwidth
  - Highly impact voice quality because of reduced bandwith
  - Do not take advantage of speech characteristics
- **Source algorithms (coders):** Source algorithms have the following functions and characteristics:
  - Source algorithm coders are called vocoders, or voice coders. A vocoder is a device that converts analog speech into digital speech, using a specific compression scheme that is optimized for coding human speech.
  - Vocoders take advantage of speech characteristics.
  - Bandwidth reduction occurs by sending linear-filter settings.
  - Codebooks store specific predictive waveshapes of human speech. They match the speech, encode the phrases, decode the waveshapes at the receiver by looking up the coded phrase, and match it to the stored waveshape in the receiver codebook.



Standard PCM is known as ITU standard G.711.

Adaptive differential pulse code modulation (ADPCM) coders, like other waveform coders, encode analog voice signals into digital signals to adaptively predict future encodings by looking at the immediate past. The adaptive feature of ADPCM reduces the number of bits per second that the PCM method requires to encode voice signals.

ADPCM does this by taking 8000 samples per second of the analog voice signal and turning them into a linear PCM sample. ADPCM then calculates the predicted value of the next sample, based on the immediate past sample, and encodes the difference. The ADPCM process generates 4-bit words, thereby generating 16 specific bit patterns.

The ADPCM algorithm from the Consultative Committee for International Telegraph and Telephone (CCITT) transmits all 16 possible bit patterns. The ADPCM algorithm from the American National Standards Institute (ANSI) uses 15 of the 16 possible bit patterns. The ANSI ADPCM algorithm does not generate a 0000 pattern.

The ITU standards for compression are as follows:

- **G.711 rate:** 64 kbps = (2 * 4 kHz) * 8 bits/sample
- **G.726 rate:** 32 kbps = (2 * 4 kHz) * 4 bits/sample
- **G.726 rate:** 24 kbps = (2 * 4 kHz) * 3 bits/sample
- **G.726 rate:** 16 kbps = (2 * 4 kHz) * 2 bits/sample

Note	CCITT is now called International Telecommunication Union Telecommunication Standardization Sector (ITU-T).

G.729, G.729 Annex A (G.729A), G.729 Annex B (G.729B), and G.729A Annex B (G.729AB) are variations of CS-ACELP.

There is little difference between the ITU recommendations for G.729 and G.729A. All of the platforms that support G.729 also support G.729A.

G.729 is the compression algorithm that Cisco uses for high-quality 8-kbps voice. When properly implemented, G.729 sounds as good as the 32-kbps ADPCM. G.729 is a high-complexity, processor-intensive compression algorithm that monopolizes processing resources.

Although G.729A is also an 8-kbps compression, it is not as processor-intensive as G.729. It is a medium-complexity variant of G.729 with slightly lower voice quality. G.729A is not as high-quality as G.729 and is more susceptible to network irregularities, such as delay, variation, and tandeming. Tandeming causes distortion that occurs when speech is coded, decoded, and then coded and decoded again, much like the distortion that occurs when a videotape is repeatedly copied.

Example: Codec Complexity

On Cisco IOS gateways, you must use the variant (G.729 or G.729A) that is related to the codec complexity configuration on the voice card. This variant does not show up explicitly in the Cisco IOS command-line interface (CLI) codec choice. For example, the CLI does not display **g729r8** (alpha code) as a codec option. However, if the voice card is defined as medium-complexity, then the **g729r8** option is the G.729A codec.

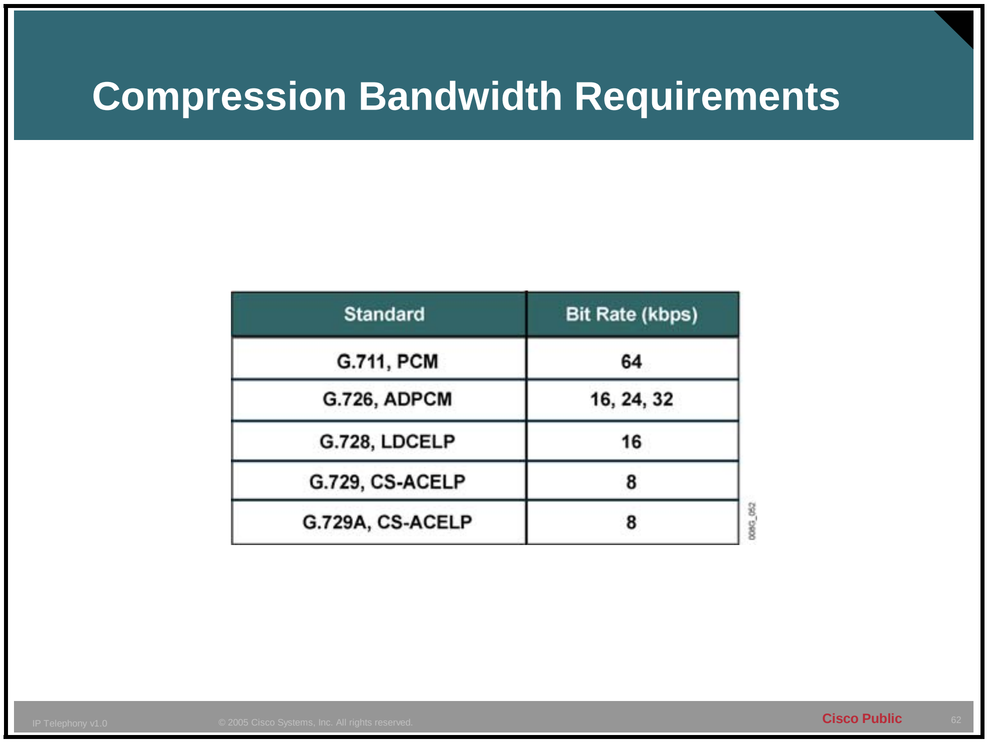
G.729B is a high-complexity algorithm, and G.729AB is a medium-complexity variant of G.729B with slightly lower voice quality. The difference between the G.729 and G.729B codec is that the G.729B codec provides built-in Internet Engineering Task Force (IETF) VAD and comfort noise generation (CNG).

The following G.729 codec combinations interoperate:

- G.729 and G.729A
- G.729 and G.729
- G.729A and G.729A
- G.729B and G.729AB
- G.729B and G.729B
- G.729AB and G.729AB

Compression Bandwidth Requirements

This topic lists the bandwidth requirements for various ITU compression standards.



Standard	Bit Rate (kbps)
G.711, PCM	64
G.726, ADPCM	16, 24, 32
G.728, LDCELP	16
G.729, CS-ACELP	8
G.729A, CS-ACELP	8

The following three common voice compression techniques are standardized by the ITU-T:

- **PCM:** Amplitude of voice signal is sampled and quantized at 8000 times per second. Each sample is then represented by one octet (8 bits) and transmitted. For sampling, you must use either a-law or μ-law to reduce the signal-to-noise ratio.
- **ADPCM:** The difference between the current sample and its predicted value (based on past samples). ADPCM is represented by 2, 3, 4, or 5 bits. This method reduces the bandwidth requirement at the expense of signal quality.
- **CELP:** Excitation value and a set of linear-predictive filters (settings) are transmitted. The filter setting transmissions are less frequent than excitation values and are sent on an as-needed basis.

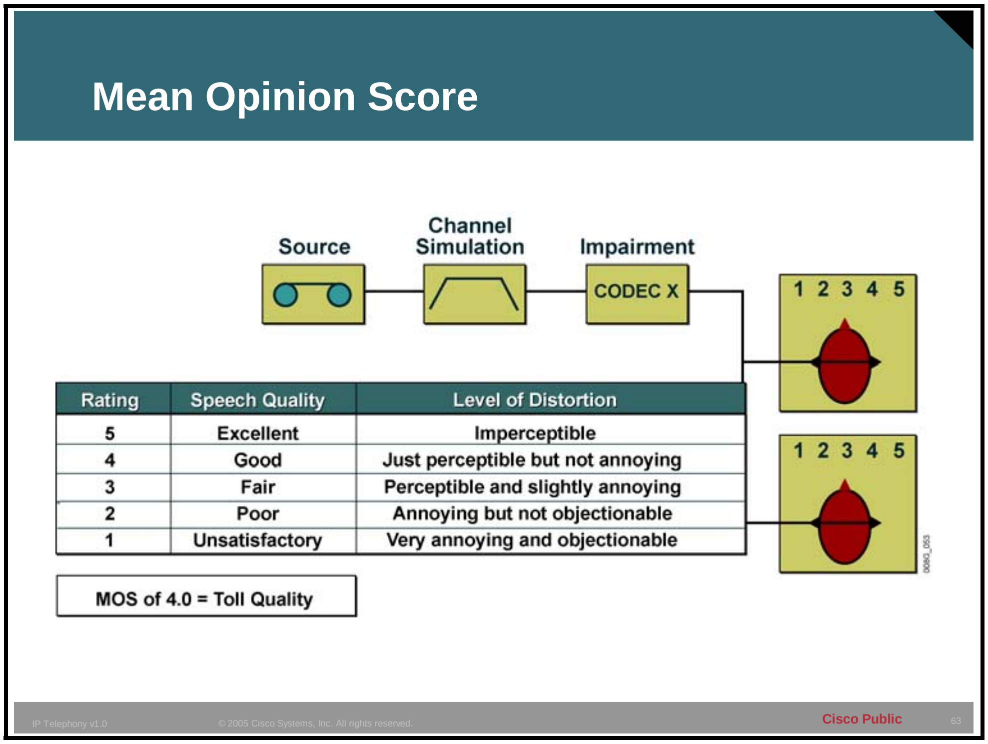
The table describes the codecs and compression standards:

Codecs and Compression Standards

Codec	Compression Technique	Bit Rate (kbps)
G.711	PCM	64
G.726	ADPCM	16, 24, 32
G.728	LDCELP	16
G.729	CS-ACELP	8
G.729A	CS-ACELP	8

Voice Quality Measurement

This topic describes two methods that are used to subjectively measure the quality of voice transported on a telephone line. Because different compression schemes produce different quality results, a method of comparing them is necessary.

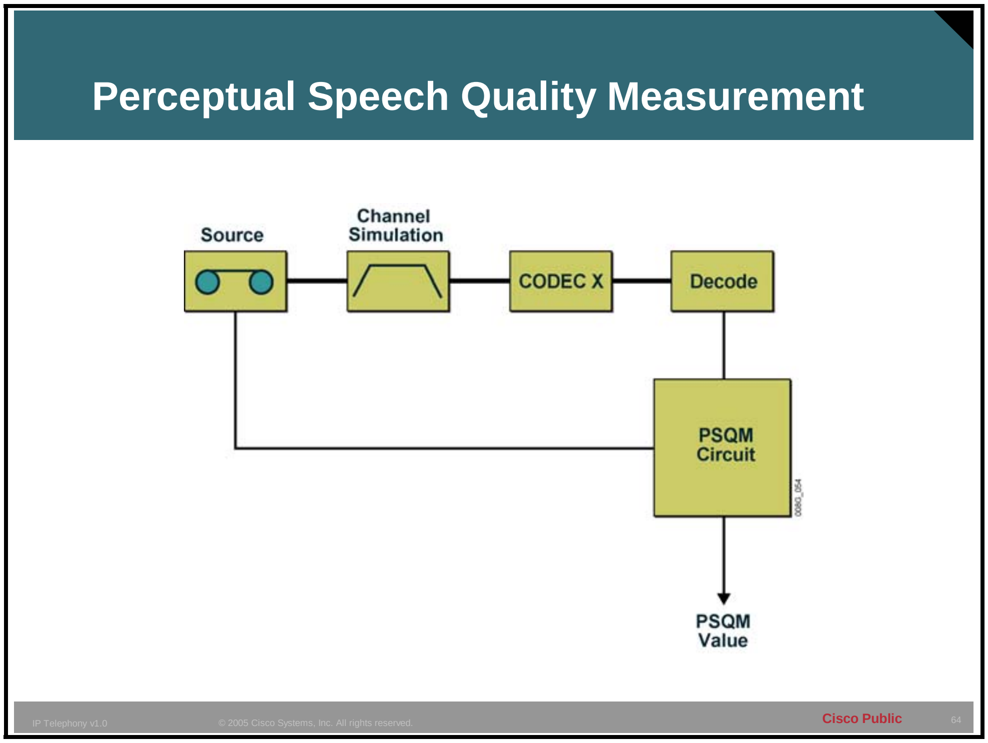


The figure depicts mean opinion score (MOS). MOS is a system of grading the voice quality of telephone connections. The MOS is a statistical measurement of voice quality derived from the judgments of several subscribers.

Graded by humans and very subjective, the range of MOS is 1 to 5, where 5 is direct conversation.

Voice Quality of Telephone Connections

Rating	Speech Quality	Level of Distortion
5	Excellent	Imperceptible
4	Good	Just perceptible but not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying but not objectionable
1	Unsatisfactory	Very annoying and objectionable



A newer, more objective measurement is available that is quickly overtaking MOS scores as the industry quality measurement of choice for coding algorithms. Perceptual Speech Quality Measurement (PSQM), as per ITU standard P.861, provides a rating on a scale of 0 to 6.5, where 0 is best and 6.5 is worst.

PSQM is implemented in test equipment and monitoring systems that are available from vendors other than Cisco. Some PSQM test equipment converts the 0-to-6.5 scale to a 0-to-5 scale to correlate to MOS. PSQM works by comparing the transmitted speech to the original input and yields a score. Various vendor test equipment is now capable of providing a PSQM score for a test voice call over a particular packet network.

In 1998, British Telecom developed a predictive voice quality measurement algorithm called Perceptual Analysis Measurement System (PAMS). PAMS can predict subjective speech quality measurement methods, such as MOS, when fidelity is affected by such things as waveform codecs, vocoders, and various speaker dependencies, such as language. PAMS, unlike PSQM, includes automatic normalization for levels.

ITU standard P.862 supercedes P.861 and describes a voice quality measurement technique that combines PSQM and PAMS. Originally developed by KPN Research, the Netherlands, and British Telecommunications (BT), Perceptual Evaluation of Speech Quality (PESQ) is an objective measuring tool that can “predict” results of subjective measuring tests, such as MOS. PESQ can be found in test equipment from a variety of vendors.

Example: Measuring Voice Quality

Cisco voice equipment does not perform voice quality measurements. There are a number of vendors who offer voice quality measurement products, some of which are designed to work with Cisco CallManager and CiscoWorks.