

Solutions Products Ordering Support Partners Training Corporate

White Papers

# Understanding Delay in Packet Voice Networks

[TAC Notice: What's Changing on TAC Web](#)

## Contents

### [Introduction](#)

### [Basic Voice Flow](#)

### [How Voice Compression Works](#)

### [Standards for Delay Limits](#)

### [Sources of Delay](#)

[Coder \(Processing\) Delay](#)

[Packetization Delay](#)

[Serialization Delay](#)

[Queuing/Buffering Delay](#)

[Network Switching Delay](#)

[De-Jitter Delay](#)

### [Build the Delay Budget](#)

[Single-Hop Connection](#)

[Two Hops on a Public Network with a C7200 that Acts as a](#)

[Tandem Switch](#)

[Two-Hop Connection over a Public Network with a PBX](#)

[Tandem Switch](#)

[Two-Hop Connection over a Private Network with a PBX](#)

[Tandem Switch](#)

### [Effects of Multiple Compression Cycles](#)

### [Considerations for High-Delay Connections](#)

### [NetPro Discussion Forums - Featured Conversations](#)

### [Related Information](#)

### Help us help you.

Please rate this document.

Excellent

Good

Average

Fair

Poor

This document solved my problem.

Yes

No

Just browsing

Suggestions for improvement:

(256 character limit)

## Introduction

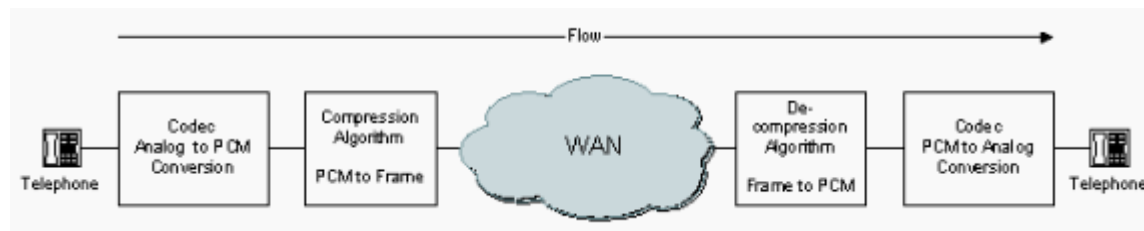
When you design networks that transport voice over packet, frame, or cell infrastructures, it is important to understand and account for the delay components in the network. If you account correctly for all potential delays, it ensures that overall network performance is acceptable. Overall voice quality is a function of many factors that include the compression algorithm, errors and frame loss, echo cancellation, and delay. This paper explains the sources of delay when you use Cisco router/gateways over packet networks. Though the examples are geared to Frame Relay, the concepts are applicable to Voice over IP (VoIP) and Voice over ATM (VoATM) networks as well.

## Basic Voice Flow

The flow of a compressed voice circuit is shown in this diagram. The analog signal from the telephone is digitized into pulse code modulation (PCM) signals by the voice coder-decoder (codec).

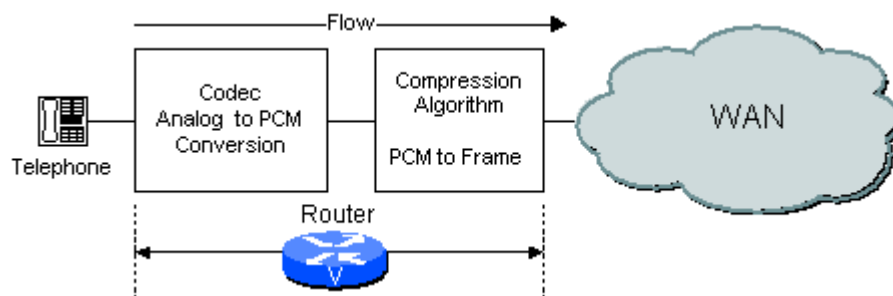
The PCM samples are then passed to the compression algorithm which compresses the voice into a packet format for transmission across the WAN. On the far side of the cloud the exact same functions are performed in reverse order. The entire flow is shown in Figure 2-1.

**Figure 2-1 End-to-End Voice Flow**



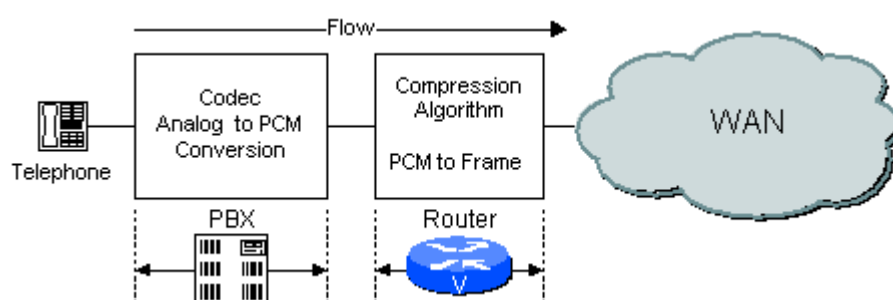
Based on how the network is configured, the router/gateway can perform both the codec and compression functions or only one of them. For example, if an analog voice system is used, then the router/gateway performs the CODEC function and the compression function as shown in Figure 2-2.

**Figure 2-2 Codec Function in Router/Gateway**



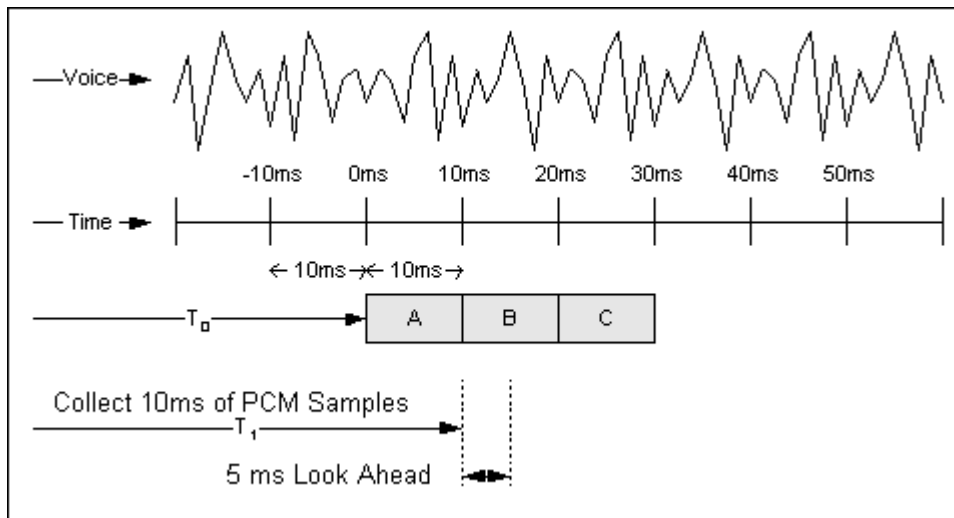
If a digital PBX is used, the PBX performs the codec function and the Router processes the PCM samples passed to it by the PBX. An example is shown in Figure 2-3.

**Figure 2-3 Codec Function in PBX**



## How Voice Compression Works

The high complexity compression algorithms used in Cisco router/gateways analyze a block of PCM samples delivered by the Voice codec. These blocks vary in length based on the coder. For example, the basic block size used by a G.729 algorithm is 10 ms whereas the basic block size used by the G.723.1 algorithms is 30ms. An example of how a G.729 compression system works is shown in Figure 3-1.

**Figure 3-1 Voice Compression**

The analog voice stream is digitized into PCM samples and delivered to the compression algorithm in 10 ms increments. The look ahead is discussed in Algorithmic Delay.

## Standards for Delay Limits

The International Telecommunication Union (ITU) considers network delay for voice applications in Recommendation G.114. This recommendation defines three bands of one-way delay as shown in Table 4.1.

**Table 4.1 Delay Specifications**

Range in Milliseconds	Description
0-150	Acceptable for most user applications.
150-400	Acceptable provided that administrators are aware of the transmission time and the impact it has on the transmission quality of user applications.
Above 400	Unacceptable for general network planning purposes. However, it is recognized that in some exceptional cases this limit is exceeded.

**Note:** These recommendations are for connections with echo adequately controlled. This implies that echo cancellers are used. Echo cancellers are required when one-way delay exceeds 25 ms (G.131).

These recommendations are oriented for national telecom administrations. Therefore, these are more stringent than when normally applied in private voice networks. When the location and business needs of end users are well-known to the network designer, more delay can prove acceptable. For private networks 200 ms of delay is a reasonable goal and 250 ms a limit. All networks need to be engineered such that the maximum expected voice connection delay is known and minimized.

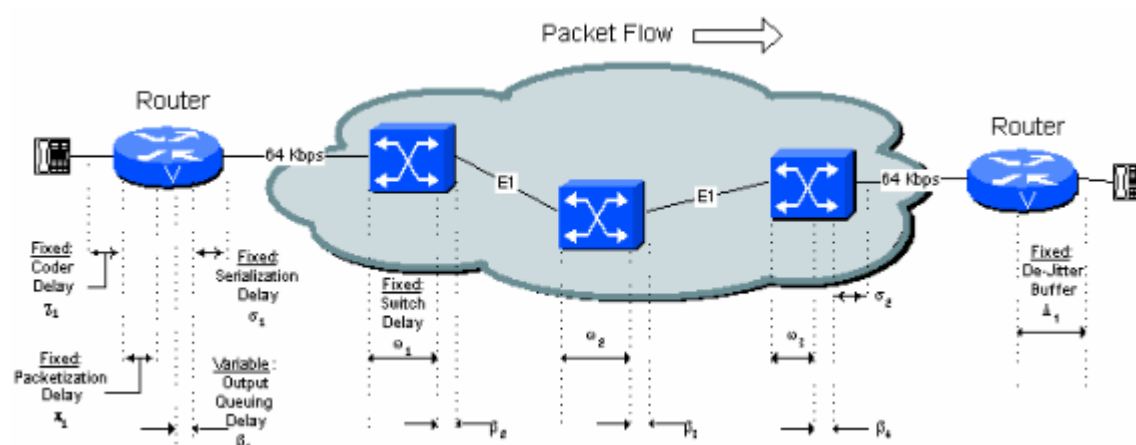
## Sources of Delay

There are two distinct types of delay called fixed and variable.

- Fixed delay components add directly to the overall delay on the connection.
- Variable delays arise from queuing delays in the egress trunk buffers on the serial port connected to the WAN. These buffers create variable delays, called jitter, across the network. Variable delays are handled through the de-jitter buffer at the receiving router/gateway. The de-jitter buffer is described in the De-jitter Delay ( $\Delta_n$ ) section of this document.

Figure 5-1 identifies all the fixed and variable delay sources in the network. Each source is described in detail in this document.

**Figure 5-1: Delay Sources**



## Coder (Processing) Delay

Coder delay is the time taken by the digital signal processor (DSP) to compress a block of PCM samples. This is also called processing delay ( $\chi_n$ ). This delay varies with the voice coder used and processor speed. For example, algebraic code excited linear prediction (ACELP) algorithms analyze a 10 ms block of PCM samples, and then compress them.

The compression time for a Conjugate Structure Algebraic Code Excited Linear Prediction (CS-ACELP) process ranges from 2.5 ms to 10 ms based on the loading of the DSP processor. If the DSP is fully loaded with four voice channels, the Coder delay is 10 ms. If the DSP is loaded with only one voice channel the Coder delay is 2.5 ms. For design purposes use the worst case time of 10 ms.

Decompression time is roughly ten percent of the compression time for each block. However, the decompression time is proportional to the number of samples per frame because of the presence of multiple samples. Consequently, the worst case decompression time for a frame with three samples is 3 x 1 ms or 3 ms. Usually, two or three blocks of compressed G.729 output are put in one frame while one sample of compressed G.723.1 output is sent in a single frame.

Best and worst case coder delays are shown in Table 5.1.

**Table 5.1 Best and Worst Case Processing Delay**

Coder	Rate	Required Sample	Best Case Coder	Worst Case Coder
-------	------	-----------------	-----------------	------------------

		<b>Block</b>	<b>Delay</b>	<b>Delay</b>
ADPCM, G.726	32 Kbps	10 ms	2.5 ms	10 ms
CS- ACELP, G.729A	8.0 Kbps	10 ms	2.5 ms	10 ms
MP-MLQ, G.723.1	6.3 Kbps	30 ms	5 ms	20 ms
MP- ACELP, G.723.1	5.3 Kbps	30 ms	5 ms	20 ms

### Algorithmic Delay

The compression algorithm relies on known voice characteristics to correctly process sample block N. The algorithm must have some knowledge of what is in block N+1 in order to accurately reproduce sample block N. This look ahead, which is really an additional delay, is called algorithmic delay. This effectively increases the length of the compression block.

This happens repeatedly, such that block N+1 looks into block N+2, and so forth and so on. The net effect is a 5 ms addition to the overall delay on the link. This means that the total time required to process a block of information is 10 m with a 5 ms constant overhead factor. See Figure 3-1: Voice Compression.

- Algorithmic Delay for G.726 coders is 0 ms
- Algorithmic Delay for G.729 coders is 5 ms.
- Algorithmic Delay for G.723.1 coders is 7.5 ms

For the examples in the remainder of this document, assume G.729 compression with a 30 ms/30 byte payload. In order to facilitate design, and take a conservative approach, the tables given in the remainder of this document assume the worst case coder delay. The coder delay, decompression delay, and algorithmic delay is lumped into one factor which is called the coder delay.

The equation used to generate the lumped Coder Delay Parameter is:

### Equation 1 : Lumped Coder Delay Parameter

$$\begin{array}{c}
 \text{(Worst Case Compression Time Per Block)} \\
 + \\
 \text{(De-Compression Time Per Block)} \\
 \times \text{(Number of Blocks in Frame)} \\
 + \\
 \text{(Algorithmic Delay)} \\
 \hline
 = \text{"Lumped" Coder Delay Parameter}
 \end{array}$$

The lumped Coder delay for G.729 that is used for the remainder of this document is:

Worst Case Compression Time Per Block: 10 ms

Decompression Time Per Block x 3 Blocks 3 ms

Algorithmic Delay 5 ms -----

Total ( $\gamma$ ) 18 ms

## Packetization Delay

Packetization delay ( $\pi_n$ ) is the time taken to fill a packet payload with encoded/compressed speech. This delay is a function of the sample block size required by the vocoder and the number of blocks placed in a single frame. Packetization delay can also be called Accumulation delay, as the voice samples accumulate in a buffer before they are released.

As a general rule you need to strive for a packetization delay of no more than 30 ms. In the Cisco router/gateways you need to use these figures from Table 5.2 based on configured payload size:

**Table 5 .2: Common Packetization**

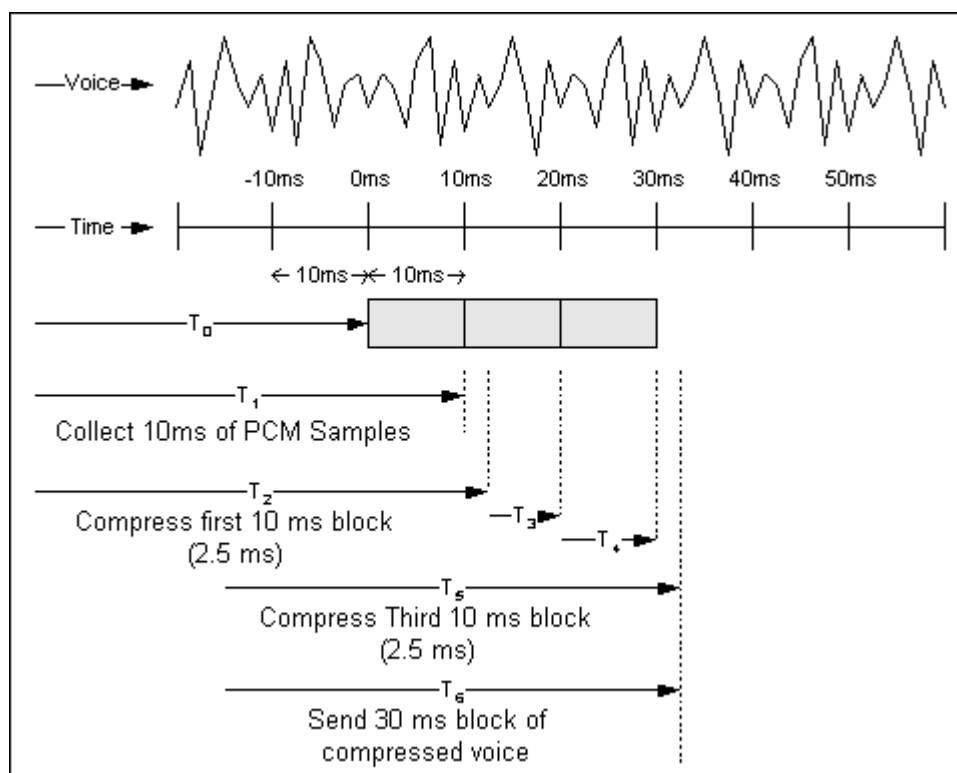
Coder		Payload Size (Bytes)	Packetization Delay (ms)	Payload Size (Bytes)	Packetization Delay (ms)
PCM, G.711	64 Kbps	160	20	240	30
ADPCM, G.726	32 Kbps	80	20	120	30
CS-ACELP, G.729	8.0 Kbps	20	20	30	30
MP-MLQ, G.723.1	6.3 Kbps	24	24	60	48
MP-ACELP, G.723.1	5.3 Kbps	20	30	60	60

You have to balance the Packetization delay against the CPU load. The lower the delay, the higher the frame rate, and the higher the load on the CPU. On some older platforms, 20 ms payloads can potentially strain the main CPU.

## Pipeline Delay in the Packetization Process

Though each voice sample experiences both algorithmic delay and packetization delay, in reality, the processes overlap and there is a net benefit effect from this pipelining. Consider the example shown in Figure 2-1.

## Figure 5-2 : Pipelining and Packetization



The top line of the figure depicts a sample voice wave form. The second line is a time scale in 10 ms increments. At  $T_0$ , the CS-ACELP algorithm begins to collect PCM samples from the codec. At  $T_1$ , the algorithm has collected its first 10 ms block of samples and begins to compress it. At  $T_2$ , the first block of samples has been compressed. In this example the compression time is 2.5 ms, as indicated by  $T_2 - T_1$ .

The second and third blocks are collected at  $T_3$  and  $T_4$ . The third block is compressed at  $T_5$ . The packet is assembled and sent (assumed to be instantaneous) at  $T_6$ . Due to the pipelined nature of the Compression and Packetization processes, the delay from when the process begins to when the voice frame is sent is  $T_6 - T_0$ , or approximately 32.5 ms.

For illustration, this example is based on best case delay. If the worst case delay is used, the figure is 40 ms, 10 ms for Coder delay and 30 ms for Packetization delay.

Note that these examples neglect to include algorithmic delay.

## Serialization Delay

Serialization delay ( $\sigma_n$ ) is the fixed delay required to clock a voice or data frame onto the network interface. It is directly related to the clock rate on the trunk. At low clock speeds and small frame sizes, the extra flag needed to separate frames is significant.

Table 5.3 shows the serialization delay required for different frame sizes at different line speeds. This table uses total frame size, not payload size, for computation.

**Table 5.3: Serialization Delay in Milliseconds for Different Frame Sizes**

Frame	Line Speed (Kbps)
-------	-------------------

Size (bytes)	19.2	56	64	128	256	384	512	768	1024	1544	2048
38	15.83	5.43	4.75	2.38	1.19	0.79	0.59	0.40	0.30	0.20	0.15
48	20.00	6.86	6.00	3.00	1.50	1.00	0.75	0.50	0.38	0.25	0.19
64	26.67	9.14	8.00	4.00	2.00	1.33	1.00	0.67	0.50	0.33	0.25
128	53.33	18.29	16.00	8.00	4.00	2.67	2.00	1.33	1.00	0.66	0.50
256	106.67	36.57	32.00	16.00	8.00	5.33	4.00	2.67	2.00	1.33	1.00
512	213.33	73.14	64.00	32.00	16.00	10.67	8.00	5.33	4.00	2.65	2.00
1024	426.67	149.29	128.00	64.00	32.00	21.33	16.00	10.67	8.00	5.31	4.00
1500	625.00	214.29	187.50	93.75	46.88	31.25	23.44	15.63	11.72	7.77	5.86
2048	853.33	292.57	256.00	128.00	64.00	42.67	32.00	21.33	16.00	10.61	8.00

In the table, on a 64 Kbps line, a CS-ACELP voice frame with a length of 38 bytes (37+1 flag) has a serialization delay of 4.75 ms.

**Note:** The serialization delay for a 53 byte ATM cell (T1: 0.275ms, E1: 0.207ms) is negligible due to the high line speed and small cell size.

### Queuing/Buffering Delay

After the compressed voice payload is built, a header is added and the frame is queued for transmission on the network connection. Voice needs to have absolute priority in the router/gateway. Therefore, a voice frame must only wait for either a data frame that already plays out, or for other voice frames ahead of it. Essentially the voice frame waits for the serialization delay of any preceding frames in the output queue. Queuing delay ( $\beta_n$ ) is a variable delay and is dependent on the trunk speed and the state of the queue. There are random elements associated with the queuing delay.

For example, assume that you are on a 64 Kbps line, and that you are queued behind one data frame (48 bytes) and one voice frame (42 bytes). Because there is a random nature as to how much of the 48 byte frame has played out, you can safely assume, on average, that half the data frame has been played out. Based on the data from the serialization table, your data frame component is  $6 \text{ ms} * 0.5 = 3 \text{ ms}$ . When you add the time for another voice frame ahead in the queue (5.25 ms), it gives a total time of 8.25 ms queuing delay.

How one characterizes the queuing delay is up to the network engineer. Generally, one needs to design for the worst case scenario and then tune performance after the network is installed. The more voice lines available to the users, the higher the probability that the average voice packet waits in the queue. The voice frame, because of the priority structure, never waits behind more than one data frame.

### Network Switching Delay

The public frame relay or ATM network that interconnects the endpoint locations is the source of the largest delays for voice connections. Network Switching Delays ( $\omega_n$ ) are also the most difficult to quantify.

If wide-area connectivity is provided by Cisco equipment, or some other private network, it is possible to identify the individual components of delay. In general, the fixed components are from



propagation delays on the trunks within the network, and variable delays are from queuing delays clocking frames into and out of intermediate switches. In order to estimate propagation delay, a popular estimate of 10 microseconds/mile or 6 microseconds/km (G.114) is widely used. However, intermediate multiplexing equipment, backhauling, microwave links, and other factors found in carrier networks create many exceptions.

The other significant component of delay is from queuing within the wide-area network. In a private network, it can be possible to measure existing queuing delays or to estimate a per-hop budget within the wide-area network.

Typical carrier delays for US frame relay connections are 40 ms fixed and 25 ms variable for a total worst case delay of 65 ms. For simplicity, in examples 6-1, 6-2, and 6-3, any low speed serialization delays in the 40 ms fixed delay are included.

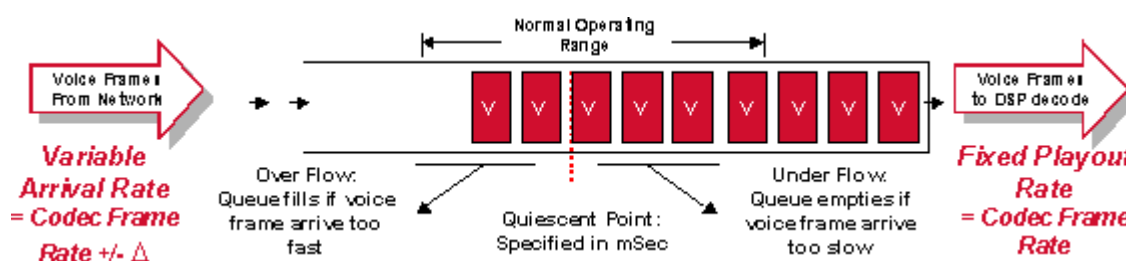
These are figures published by US frame relay carriers, in order to cover anywhere to anywhere coverage within the United States. It is to be expected that two locations which are geographically closer than the worst case have better delay performance, but carriers normally document just the worst case.

Frame relay carriers sometimes offer premium services. These services are usually for voice or Systems Network Architecture (SNA) traffic, where the network delay is guaranteed and less than the standard service level. For instance, a US carrier recently announced such a service with an overall delay limit of 50 ms, rather than the standard service's 65 ms.

## De-Jitter Delay

Because speech is a constant bit-rate service, the jitter from all the variable delays must be removed before the signal leaves the network. In Cisco router/gateways this is accomplished with a de-jitter ( $\Delta_n$ ) buffer at the far-end (receiving) router/gateway. The de-jitter buffer transforms the variable delay into a fixed delay. It holds the first sample received for a period of time before it plays it out. This holding period is known as the initial play out delay.

**Figure 5- 3 : De-Jitter Buffer Operation**



It is essential to handle properly the de-jitter buffer. If samples are held for too short a time, variations in delay can potentially cause the buffer to under-run and cause gaps in the speech. If the sample is held for too long a time, the buffer can overrun, and the dropped packets again cause gaps in the speech. Lastly, if packets are held for too long a time, the overall delay on the connection can rise to unacceptable levels.

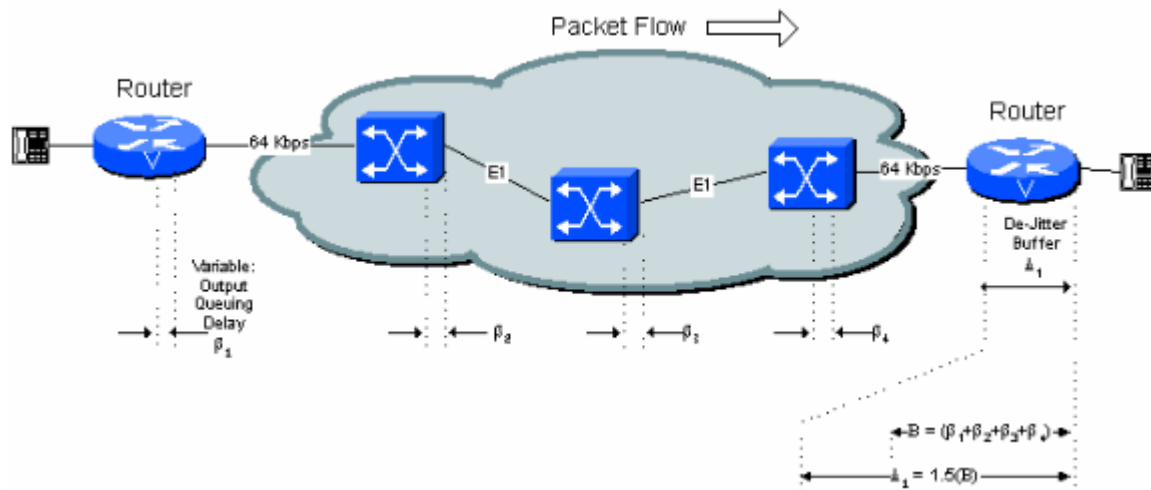
The optimum initial play out delay for the de-jitter buffer is equal to the total variable delay along the connection. This is shown in Figure 5-4.

**Note:** The de-jitter buffers can be adaptive, but the maximum delay is fixed. When adaptive buffers are configured, the delay becomes a variable figure. However, the maximum delay can be used as a

worst case for design purposes.

For more information on adaptive buffers, refer to [Playout Delay Enhancements for Voice over IP](#).

**Figure 5 -4 : Variable Delay and the De-Jitter Buffer**



The initial playout delay is configurable. The maximum depth of the buffer before it overflows is normally set to 1.5 or 2.0 times this value.

If the 40 ms nominal delay setting is used, the first voice sample received when the de-jitter buffer is empty is held for 40 ms before it is played out. This implies that a subsequent packet received from the network can be as much as 40 ms delayed (with respect to the first packet) without any loss of voice continuity. If it is delayed more than 40 ms, the de-jitter buffer empties and the next packet received is held for 40 ms before play out to reset the buffer. This results in a gap in the voice played out for about 40 ms.

The actual contribution of de-jitter buffer to delay is the initial play out delay of the de-jitter buffer plus the actual amount the first packet was buffered in the network. The worst case is twice the de-jitter buffer initial delay (assumption is that the first packet through the network experienced only minimum buffering delay). In practice, over a number of network switch hops, it is probably not necessary to assume the worst case. The calculations in the examples in the remainder of this document increase the initial play out delay by a factor of 1.5 to allow for this effect.

**Note:** In the receiving router/gateway there is delay through the decompression function. However, this is taken into account by lumping it together with the compression processing delay as discussed previously.

## Build the Delay Budget

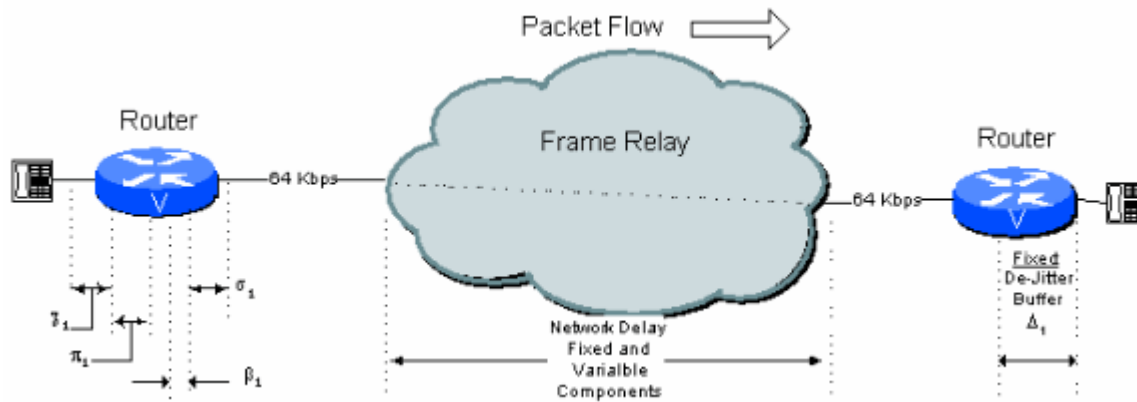
The generally-accepted limit for good-quality voice connection delay is 200 ms one-way (or 250 ms as a limit). As delays rise over this figure, talkers and listeners become un-synchronized, and often they speak at the same time, or both wait for the other to speak. This condition is commonly called talker overlap. While the overall voice quality is acceptable, users sometimes find the stilted nature of the conversation unacceptably annoying. Talker overlap can be observed on international telephone calls which travel over satellite connections (satellite delay is in the order of 500 ms, 250 ms up and 250 ms down).

These examples illustrate various network configurations and the delays which the network designer

needs to take into account.

## Single-Hop Connection

**Figure 6 - 1: Single Hop Example Connection**



From this figure, a typical one-hop connection over a public frame relay connection can have the delay budget shown Table 6.1.

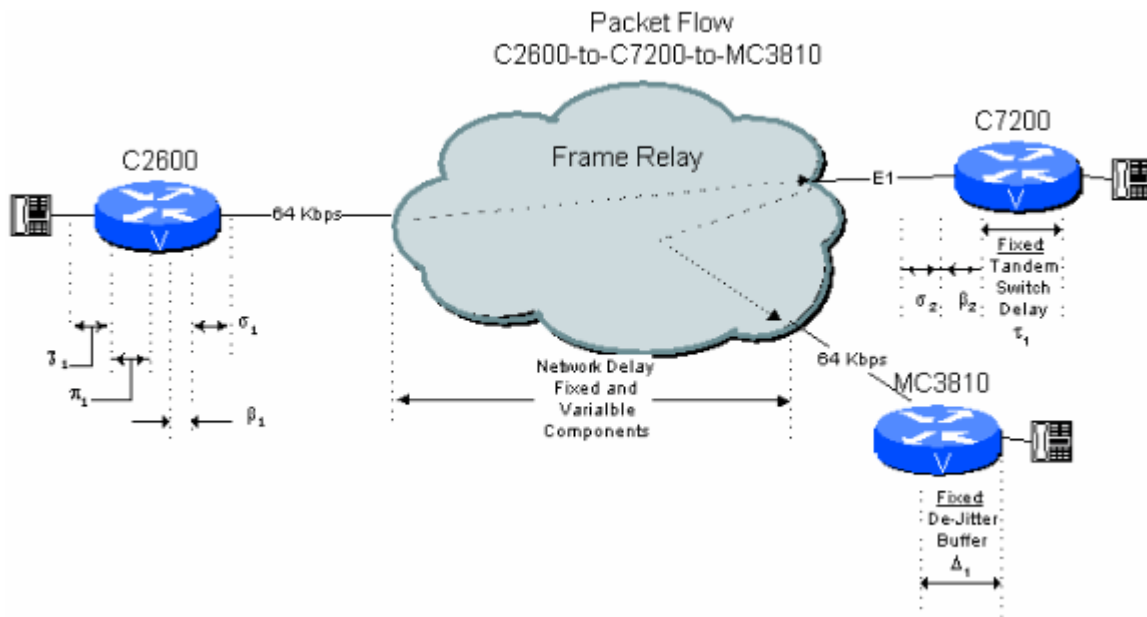
**Table 6 .1: Single Hop Delay Calculation**

Delay Type	Fixed (ms)	Variable (ms)
Coder Delay, $\chi_1$	18	
Packetization Delay, $\pi_1$	30	
Queuing/Buffering, $\beta_1$		8
Serialization Delay (64 kbps), $\sigma_1$	5	
Network Delay (Public Frame), $\omega_1$	40	25
De-jitter Buffer Delay, $\Delta_1$	45	
Totals	138	33

**Note:** Since queuing delay and the variable component of the Network delay is already accounted within the de-jitter buffer calculations, the Total delay is effectively only the sum of all the Fixed Delay. In this case the total delay is 138 ms.

## Two Hops on a Public Network with a C7200 that Acts as a Tandem Switch

**Figure 6 - 2: Two Hops Public Network Example with Router/Gateway Tandem**



Now consider a branch-to-branch connection in a star-topology network where the C7200 in the headquarters site tandems the call to the destination branch. In this case the signal stays in compressed format through the central C7200. This results in considerable savings in the delay budget with respect to the next example, Two-Hop Connection Over A Public Network With A PBX Tandem Switch.

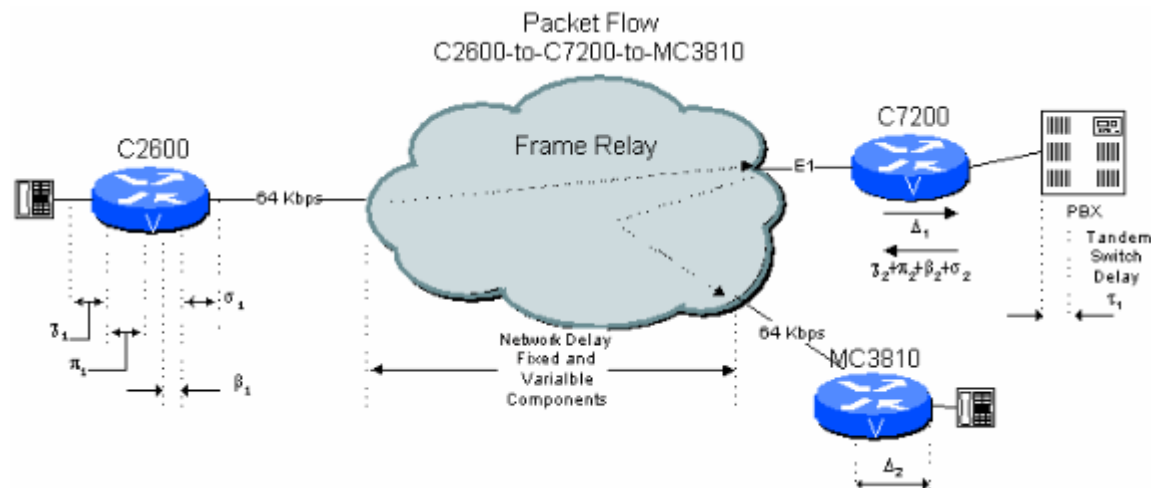
**Table 6.2: Two Hop Public Network Delay Calculation with Router/Gateway Tandem**

Delay Type	Fixed (ms)	Variable (ms)
Coder Delay, $\chi_1$	18	
Packetization Delay, $\pi_1$	30	
Queuing/Buffering, $\beta_1$		8
Serialization Delay (64 kbps), $\sigma_1$	5	
Network Delay (Public Frame), $\omega_1$	40	25
Tandem Delay in MC3810, $\tau_1$	1	
Queuing/Buffering, $\beta_2$		0.2
Serialization Delay (2 Mbps), $\sigma_2$	0.1	
Network Delay (Public Frame), $\omega_2$	40	25
De-jitter Buffer Delay, $\Delta_1$	75	
Totals	209.1	58.2

**Note:** Since queuing delay and the variable component of the Network delay is already accounted within the de-jitter buffer calculations, the Total delay is effectively only the sum of all the Fixed Delay. In this case the total delay is 209.1 ms.

## Two-Hop Connection over a Public Network with a PBX Tandem Switch

**Figure 6-3: Two Hop Public Network Example with PBX Tandem**



Consider a branch-to-branch connection in a branch-to-headquarters network where the C7200 at the headquarters site passes the connection through to the headquarters PBX for switching. Here the voice signal has to be decompressed and de-jittered and then re-compressed and de-jittered a second time. This results in extra delays relative to the previous example. Additionally, the two CS-ACELP compression cycles reduce voice quality (see Effects Of Multiple Compression Cycles).

**Table 6.3: Two Hop Public Network Delay Calculation with PBX Tandem**

Delay Type	Fixed (ms)	Variable (ms)
Coder Delay, $\chi_1$	18	
Packetization Delay, $\pi_1$	30	
Queuing/Buffering, $\beta_1$		8
Serialization Delay (64 kbps), $\sigma_1$	5	
Network Delay (Public Frame), $\omega_1$	40	25
De-jitter Buffer Delay, $\Delta_1$		40
Coder Delay, $\chi_2$	15	
Packetization Delay, $\pi_2$	30	
Queuing/Buffering, $\beta_2$		0.1
Serialization Delay (2 Mbps),	0.1	

$\sigma_2$		
Network Delay (Public Frame), $\omega_2$	40	25
De-jitter Buffer Delay, $\Delta_2$	40	
Totals	258.1	58.1

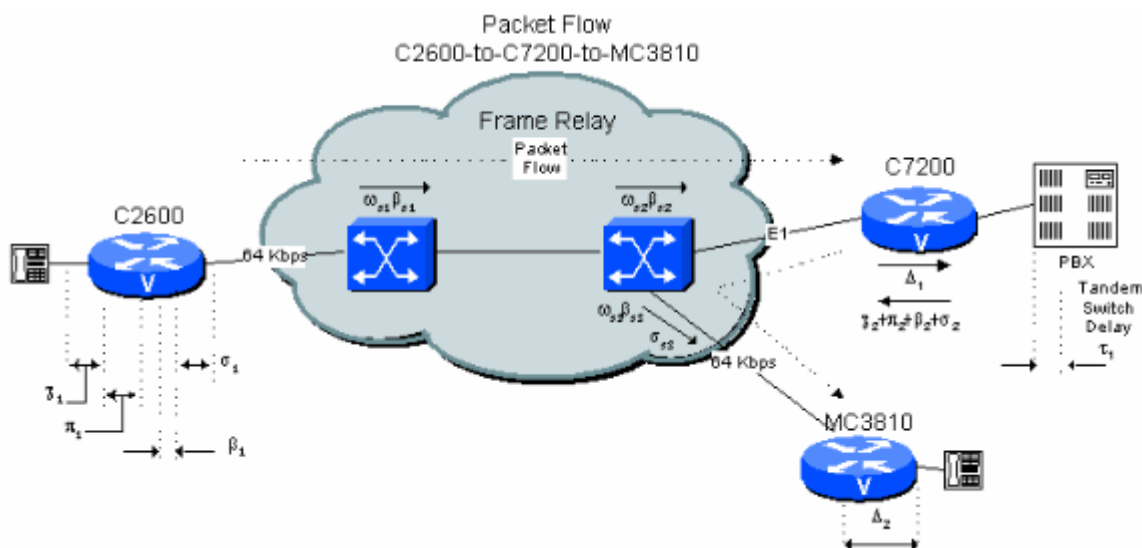
**Note:** Since queuing delay and the variable component of the Network delay is already accounted within the de-jitter buffer calculations, the Total delay is effectively only the sum of all the Fixed Delay plus the de-jitter buffer delay. In this case the total delay is 258.1 ms.

If you use the PBX at the central site as a switch, it increases the one-way connection delay from 206 ms to 255 ms. This is close to the ITU limits for one-way delay. This type of network configuration requires the engineer to pay close attention to design for minimum delay.

The worst case is assumed for variable delay (although both legs on the public network do not see maximum delays simultaneously). If you make more optimistic assumptions for the variable delays, it only minimally improves the situation. However, with better information about the fixed and variable delays in the frame relay network of the carrier, the calculated delay can be reduced. Local connections (for instance intra-State) can be expected to have much better delay characteristics, but carriers are often reluctant to give delay limits.

## Two-Hop Connection over a Private Network with a PBX Tandem Switch

**Figure 6-4: Two Hop Private Network Example with PBX Tandem**



Example 4.3 shows that, with the assumption of worst case delays, it is very difficult to keep the calculated delay under 200 ms when a branch-to-branch connection includes a PBX tandem hop at the central site with public frame-relay network connections on either side. However, if the network topology and traffic is known, it is possible to substantially reduce the calculated figure. This is because the figures generally given by carriers are limited by the worst case transmission and queuing delay over a wide area. It is much easier to establish more reasonable limits in a private network.

The generally accepted figure for transmission delay between switches is of the order of 10

microseconds/mile. Based on the equipment, the trans-switch delay in a frame relay network needs to be in the order of 1 ms fixed and 5 ms variable for queuing. These figures are equipment and traffic dependent. The delay figures for the Cisco MGX WAN Switches is less than 1 ms per switch total if E1/T1 trunks are used. With the assumption of 500 miles of distance, with 1 ms fixed and 5 ms variable for each hop, the delay calculation becomes:

**Table 6 .4: Two Hop Private Network Delay Calculation with PBX Tandem**

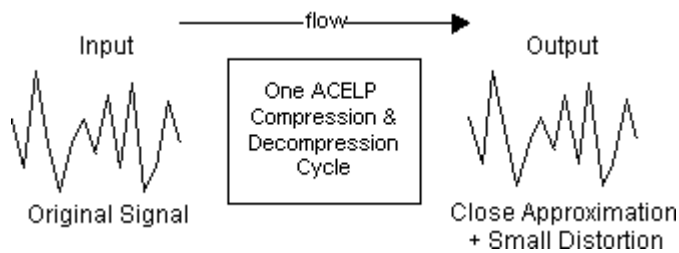
Delay Type	Fixed (ms)	Variable (ms)
Coder Delay, $\chi_1$	18	
Packetization Delay, $\pi_1$	30	
Queuing/Buffering, $\beta_1$		8
Serialization Delay (64 kbps), $\sigma_1$	5	
Network Delay (Private Frame), $\omega_{S1} + \beta_{S1} + \omega_{S2} + \beta_{S2}$	2	10
De-jitter Buffer Delay, $\Delta_1$	40	
Coder Delay, $\chi_2$	15	
Packetization Delay, $\pi_2$	30	
Queuing/Buffering, $\beta_2$		0.1
Serialization Delay (2 Mbps), $\sigma_2$	0.1	
Network Delay (Private Frame), $\omega_{S3} + \beta_{S3}$	1	8
Serialization Delay (64 kbps), $\sigma_{S3}$	5	
De-jitter Buffer Delay, $\Delta_2$	40	
Transmission/distance delay (not broken down)	5	
Totals	191.1	26.1

**Note:** Since queuing delay and the variable component of the Network delay is already accounted within the de-jitter buffer calculations, the Total delay is only the sum of all the Fixed Delay. In this case the total delay is 191.1 ms.

When you run over a private frame relay network, it is possible to make a spoke-to-spoke connection through the PBX at the hub site and stay within the 200 ms figure.

## Effects of Multiple Compression Cycles

The CS-ACELP compression algorithms are not deterministic. This means that the input data stream is not exactly the same as the output data stream. A small amount of distortion is introduced with each compression cycle as shown in Figure 7-1.

**Figure 7-1: Compression Effects**

Consequently, multiple CS-ACELP compression cycles quickly introduce significant levels of distortion. This additive distortion effect is not as pronounced with adaptive differential pulse code modulation (ADPCM) algorithms.

The impact of this characteristic is that in addition to the effects of delay, the network designer must consider the number of CS-ACELP compression cycles in the path.

Voice quality is subjective. Most users find that two compression cycles still provide adequate voice quality. A third compression cycle usually results in noticeable degradation, which can be unacceptable to some users. As a rule, the network designer needs to limit the number of CS-ACELP compression cycles in a path to two. If more cycles must be used, let the customer hear it first.

In the previous examples, it is shown that when a branch-to-branch connection is tandem switched through the PBX (in PCM form) at the headquarters site, it experiences significantly more delay than if it were tandem-switched in the headquarters C7200. It is clear that when the PBX is used to switch, there are two CS-ACELP compression cycles in the path, instead of the one cycle when the framed voice is switched by the central C7200. The voice quality is better with the C7200-switched example (4.2), although there can be other reasons, such as calling plan management, that can require the PBX to be included in the path.

If a branch-to-branch connection is made through a central PBX, and from the second branch the call is extended over the public voice network and then terminates on a cellular telephone network, there are three CS-ACELP compression cycles in the path, as well as significantly higher delay. In this scenario, quality is noticeably affected. Again, the network designer must consider the worst-case call path and decide whether it is acceptable given the users network, expectations, and business requirements.

## Considerations for High-Delay Connections

It is relatively easy to design packet voice networks which exceed the ITU generally accepted 150 ms one-way delay limit.

When you design packet voice networks, the engineer needs to consider how often such a connection is used, what the user demands, and what type of business activity is involved. It is not uncommon for such connections to be acceptable in particular circumstances.

If the frame relay connections do not traverse a large distance, it is quite likely that the delay performance of the network is better than that shown in the examples.

If the total delay experienced by tandem router/gateway connections becomes too great, an alternative is often to configure extra permanent virtual circuits (PVCs) directly between the terminating MC3810s. This adds recurring cost to the network as carriers usually charge per PVC, but it can be necessary in some cases.





## NetPro Discussion Forums - Featured Conversations

Networking Professionals Connection is a forum for networking professionals to share questions, suggestions, and information about networking solutions, products, and technologies. The featured links are some of the most recent conversations available in this technology.

NetPro Discussion Forums - Featured Conversations for Voice
Service Providers: Voice over IP
<a href="#">PGW2200-MGCP-2811-PBX</a> - Jul 30, 2008 <a href="#">Change of DNS</a> - Jul 29, 2008 <a href="#">Match unknown number with analogue FXO interface</a> - Jul 29, 2008 <a href="#">Hold and Transfer soft keys missing on some calls</a> - Jul 29, 2008 <a href="#">How the VoIP Phone works</a> - Jul 28, 2008
Voice & Video: Voice over IP
<a href="#">Cisco VT Advantage video issues</a> - Jul 30, 2008 <a href="#">voice</a> - Jul 30, 2008 <a href="#">Cisco DMM server - Network Teaming?</a> - Jul 30, 2008 <a href="#">Conditional Trust Policies</a> - Jul 30, 2008 <a href="#">Resource unavailable in video calls</a> - Jul 30, 2008
Voice & Video: IP Telephony
<a href="#">Callmanager 4.0 - Call routing problem</a> - Jul 31, 2008 <a href="#">local conference without CCM</a> - Jul 31, 2008 <a href="#">Logged out of hunt group</a> - Jul 31, 2008 <a href="#">Weight of device on CCM4.1(3)</a> - Jul 31, 2008 <a href="#">LDAP integration with CUCM</a> - Jul 31, 2008
Voice & Video: IP Phone Services for End Users
<a href="#">Survey: How many avoid 9 for outside dialing due to 911 calls</a> - Jul 30, 2008 <a href="#">Set the custom background images to default?</a> - Jul 30, 2008 <a href="#">Extension Mobility: logged out user and external call</a> - Jul 30, 2008 <a href="#">CCME - Fastdial name not seen when calling??</a> - Jul 30, 2008 <a href="#">Extention Mobility service problem under CCM 6.1.2</a> - Jul 29, 2008
Voice & Video: Unified Communications
<a href="#">Attendant Console behaviour</a> - Jul 31, 2008 <a href="#">How to get prompt wav file using AvT</a> - Jul 30, 2008 <a href="#">911 Applications</a> - Jul 30, 2008 <a href="#">moh file</a> - Jul 30, 2008 <a href="#">Error accessing 7941 and 7961 via Web Access</a> - Jul 30, 2008
Voice & Video: IP Phone Services for Developers
<a href="#">AXL Request?</a> - Jul 30, 2008 <a href="#">SoftKeyItem Problem</a> - Jul 30, 2008 <a href="#">Cisco TSP Newbie Question</a> - Jul 30, 2008 <a href="#">Invoking Service Programmatically</a> - Jul 30, 2008 <a href="#">Retrieve Data from CUCM 6.x in an XML file using Webservice</a> - Jul 30, 2008
Voice & Video: General
<a href="#">New PRI installation - Utilizing Dialers</a> - Jul 30, 2008 <a href="#">VG224 SRST</a> - Jul 30, 2008 <a href="#">Trying to register an IP phone via VPN with 877.</a> - Jul 30, 2008 <a href="#">COS trust</a> - Jul 30, 2008 <a href="#">Setting up a Job/Announcement Extension</a> - Jul 30, 2008

## Related Information

- [International Telecommunication Union](#) 
- [Voice Technology Support](#)
- [Voice and Unified Communications Product Support](#)
- **Recommended Reading:** [Troubleshooting Cisco IP Telephony](#) 
- [Technical Support - Cisco Systems](#)

---

<a href="#">Home</a>	<a href="#">How to Buy</a>	<a href="#">Login</a>	<a href="#">Profile</a>	<a href="#">Feedback</a>	<a href="#">Site Map</a>	<a href="#">Help</a>
----------------------	----------------------------	-----------------------	-------------------------	--------------------------	--------------------------	----------------------

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2007 - 2008 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)