ViBE Technology White Paper

Introduction

This document will describe the technical issues facing organisations who wish to deploy voice over IP either on their own IP infrastructure or over WAN links to remote telephony servers. It will also explain how Voipex and ViBE technology can solve all of these problems, and provide the highest density of calls over the smallest available bandwidth available today, and how it can do this on shared voice and data links without compromising call quality.

Briefly, the main barriers to deploying VoIP are:

- Excessive bandwidth consumption of even so-called efficient CODECs such as G.729.
- High load on transit routers due to the large number of packets per second involved where there are many calls being carried.
- Latency and jitter which arises as the result of larger data packets using the same links (even evident if the only data traffic is session data such as that provided by SIP).
- The cost of high bandwidth WAN links which are needed to solve these issues using traditional methods.
- Lack of CODEC support in devices.
- Complexities involved for enterprises wishing to deploy VoIP between sites across the public Internet or non-private links.
- The cost of providing backup solutions in order to avoid the WAN link being a single point of failure.

The features of ViBE which eliminate or significantly reduce these barriers are summarised below:

- Bandwidth used by voice is reduced by as much as five times.
- Jitter introduced by the use of router queues is reduced to virtually zero.
- Classes of data can receive as little as 0.4kbits/s.
- Interactive traffic remains responsive.
- There is no need to reduce the maximum transmission unit (MTU) of the WAN.
- ViBE optionally supports real-time and invisible transcoding of G.711 to more efficient CODECs with higher MOS scores than G.729 (the most widely supported low bandwidth CODEC).
- Backup links can be switched to in less than a second and without losing calls in progress.
- Multiple links can be combined to both increase bandwidth available and eliminate single points of failure.
- Sites can be privately linked across the public Internet. ISPs can create groups of customer sites which form VPNs.

<u>Bandwidth Consumption and Router Load</u>

The Problem

The useful portion of a voice packet is normally very small. For example, a 20mS G.729 encoded voice packet contains only 20 bytes of useful data, transmitted at a rate of 50 packets per second. This results in:

50 x 20 x 8 bits per second = 8000 bits per second
The figure of 8kbit/s agrees with what the books say... but that is only part of the story. In order to be able to carry this packet across an IP network you also need some additional data, which is broken down in the table below:

Protocol	Bytes	Description
IP	20	Standard data required by all Internet traffic. Source/Destination/Length etc.
UDP	8	Source and destination port etc.
RTP	12	Media type, timestamp, source ID, etc.

Adding all of this to the standard 20mS G.729 packet our data rate now becomes:
(20 + 20 + 8 + 12) x 50 x 8 = 24000
Suddenly our voice call is using three times more bandwidth than it actually needs to! Unfortunately, this is still not the end of the story. At layer 2, or the network layer, there is more information required in order to transfer data packets across physical links. At best, this link will be Ethernet, which "only" adds another 14 bytes for every packet (making our G.729 voice channel use 29.6kbits/sec.) However, ATM based technologies such as most broadband links are even less efficient. These networks use fixed sized "cells" of data in order to carry information, and typically these cells are 53 bytes in length. The G.729 packet above would require two such cells, and so now we have:
2 x 53 x 50 x 8 = 42400
Therefore each G.729 voice channel that is carried across a broadband link consumes 42.4kbit/s of bandwidth, only 8kbit/s actually being used for the information that we really want to be transmitted.

How ViBE Solves This Problem

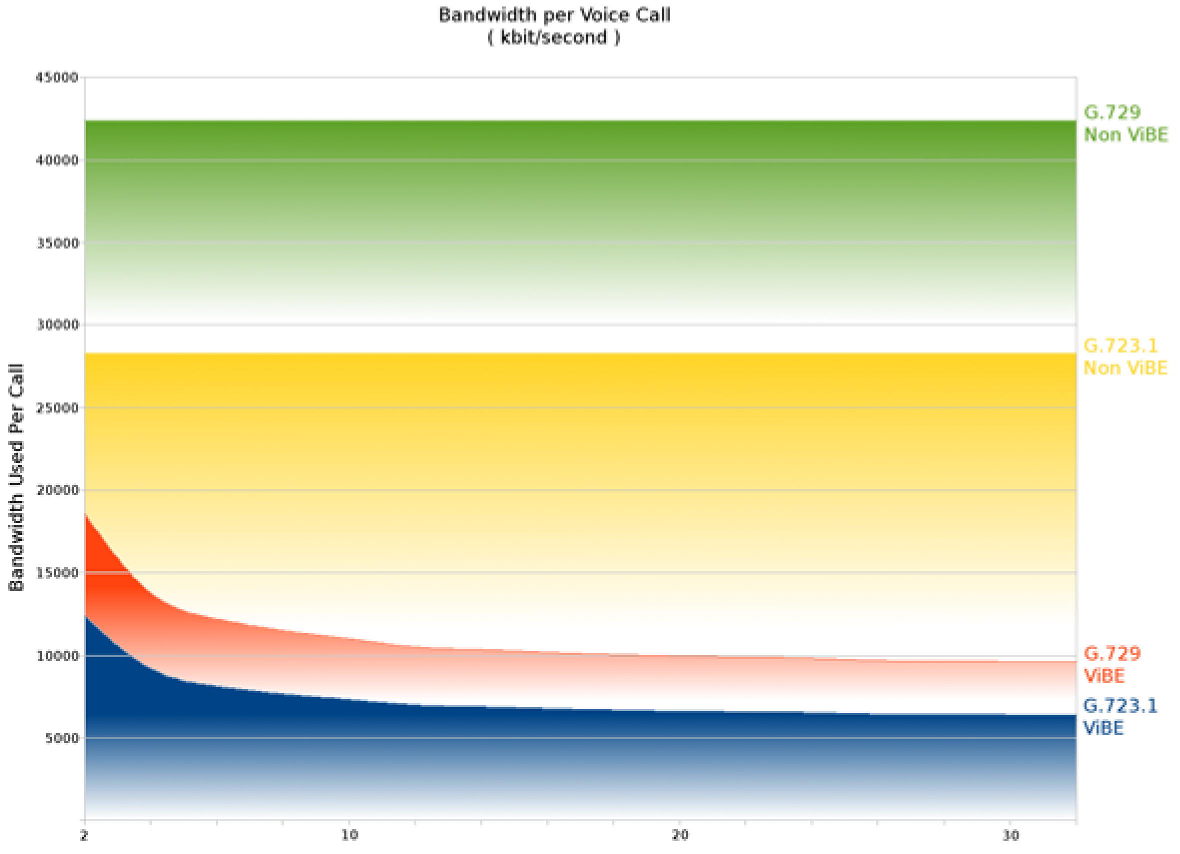
The information that is added to a voice packet in order to transmit it across a network is mostly superfluous. This is because all of it is either fixed for the entire duration of the stream, (such as destination address and port), or can be worked out from previous packets, (such as sequence numbers and time stamps.) ViBE creates a VPN tunnel between two ViBE enabled devices. Within this tunnel, voice packets from multiple channels are combined in to a single data stream, which has several effects:

1. ViBE only needs to send one set of IP and UDP headers for each "super-packet" that it sends.
2. It allows ViBE to not transmit the superfluous information for each voice channel at all. In fact, the ViBE system is so efficient that the total overhead for each channel within the stream is 2.287 bits.
3. The cell padding effect which is present on broadband networks is eliminated.

All of this processing is entirely transparent to other devices on the network since the original data streams are reconstructed before they are sent on their way from the receiving ViBE device. The only effects are a much lower bandwidth consumption and far fewer packets per second across the WAN, reducing load on routers and the possibility of network congestion. The following table illustrates just how much bandwidth is saved using the G.729 CODEC across an ADSL network.

Simultaneous Calls	Bandwidth w/o ViBE	Bandwidth With ViBE
1	42.4 kbit/s	29.2 kbit/s
10	424 kbit/s	110 kbit/s
50	2.12 Mbit/s	468.8 kbit/s
100	4.24 Mbit/s	982.4 kbit/s

To illustrate the bandwidth savings that this represents, below is a graph which shows the average bandwidth used per call, including all overheads, when sent over a broadband network.

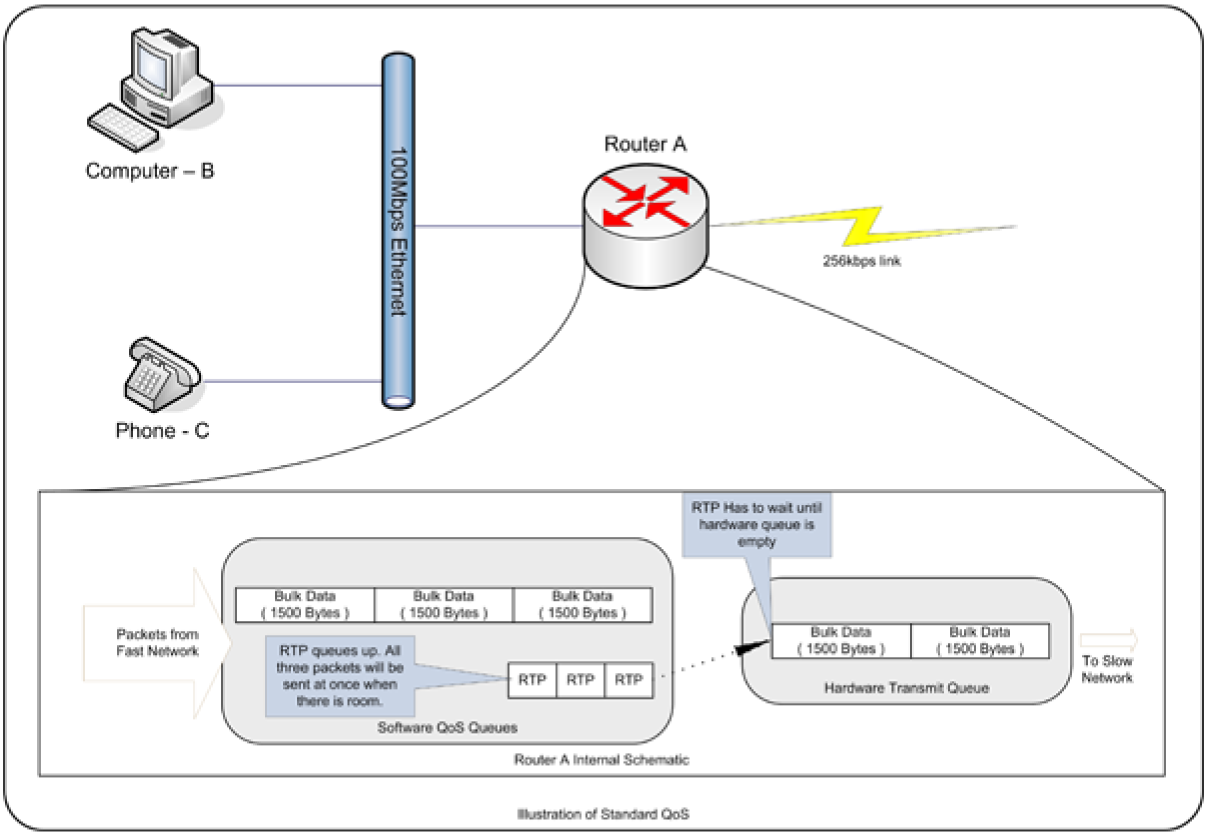


Latency and Jitter Reduction

The Problem

This section could easily be summarised as “traditional traffic shaping does not work at low bandwidths.” It is also the area that all competing technologies fail to address. Essentially, traditional quality of service (QoS) paradigms treat voice as just another class of traffic – yes, they may give it priority over other types of packet as best they can, but they do this in exactly the same way that they would when prioritising any other class. The problem is that voice is not the same as any other class of traffic, it has specific requirements in terms of latency and jitter which need to be honoured.

As an illustration, consider a 256kbit WAN link. This link has typical QoS techniques applied, and so voice packets are prioritised. Look at the illustration below:



Data enters the router, A, from a 100Mbit/s LAN interface. A computer, B, on the LAN is sending an e-mail with a large attachment, and since the LAN is 100Mbit/s his computer sends many large (typically 1500 byte) packets to router A in quick succession. The router will continue to queue packets for sending to the WAN until its buffer fills up.

At the same time, another device C is being used for a voice call. It is sending small (60 byte) packets every 50th of a second. Router A allocates a separate buffer to these packets, and when a decision needs to be made as to which packets to send to its hardware transmit buffer, it will send the packets from C first. However, that decision is only made when there is a space in the hardware buffer, which only happens when a previously queued packet is sent.

What all of this means is that the chances are that there will already be two 1500 byte packets in the hardware buffer at the time when a voice packet really should be sent on the WAN. At 256kbit/s, these 3000 bytes will take 94mS to send, meaning that as many as four voice packets could queue up waiting for a space on the network.
The normal way of making this situation a little better is to reduce the maximum size of packets that can be sent on the WAN. The drawback of this approach is that the WAN is then made less efficient, and in any case the smallest “maximum packet size” allowed is 576 bytes, cutting the router induced jitter to around 36mS, still nearly twice the amount of time allocated to a single voice packet.

How ViBE Solves This Problem

ViBE technology approaches this problem differently. The ViBE tunnel created between enabled devices only sends packets at set intervals, which at 256kbit/s will be the same interval that voice packets are being sent. ViBE knows how much bandwidth is available on the link, and so it also knows how large its packets can be. Any space which is not allocated to voice will simply be used by portions of data packets as required, in a very efficient manner. This is illustrated below.

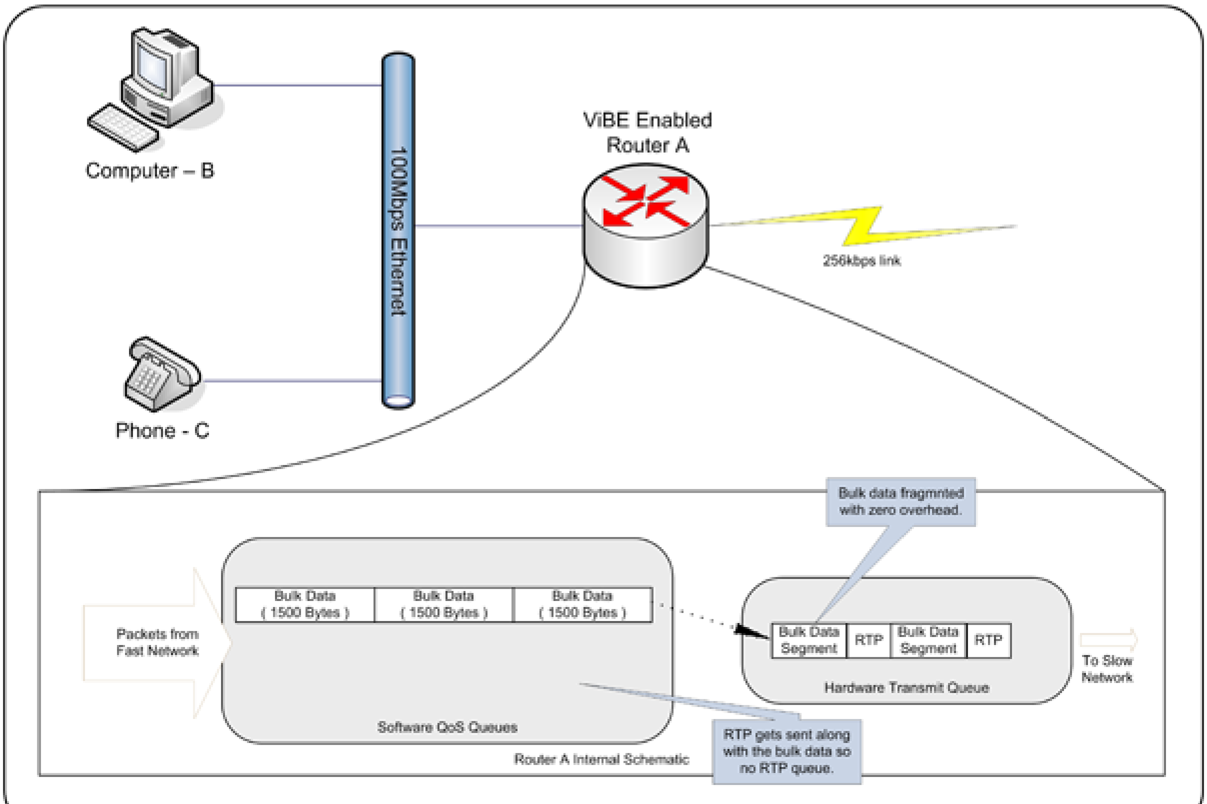


Illustration of ViBE QoS