Deploying VoIP in Existing IP Networks

Khaled Salah

Department of Information and Computer Science
King Fahd University of Petroleum and Minerals
PO Box 5066
Dhahran 31261, Saudi Arabia
Email: salah@kfupm.edu.sa
Phone: +96638604493
Fax: +96638602174

1 Introduction

Many network managers find it attractive and cost effective to merge and unify voice and data networks. A unified network is easier to run, manage, and maintain. However, the majority of today's existing data networks is Ethernet-based using IP protocols. Such networks are best-effort networks and were not designed to support real-time applications such as VoIP. VoIP requires timely packet delivery with low latency, jitter, packet loss, and sufficient bandwidth. To achieve this goal, an efficient deployment of VoIP must ensure these real-time traffic requirements can be guaranteed over new or existing IP networks.

When deploying a new network service such as VoIP over existing data network, many network architects, managers, planners, designers, and engineers are faced with common strategic, and sometimes challenging, questions. What are the QoS requirements for VoIP? How will the new VoIP load impact the QoS for currently running network services and applications? Will my existing network support VoIP and satisfy the standardized QoS requirements? If so, how many

VoIP calls can the network support before upgrading prematurely any part of the existing network hardware?

Some commercial tools can be utilized to answer some of these challenging questions. A list of the available commercial tools that can be utilized to deploy VoIP is listed in [1,2]. For the most part, these tools use two common approaches in assessing the deployment of VoIP into the existing network. One approach is based on first performing network measurements and then predicting the network readiness for supporting VoIP. The prediction of the network readiness is based on assessing the health of network elements. The second approach is based on injecting real VoIP traffic into existing network and measuring the resulting delay, jitter, and loss.

There is definitely a financial cost associated with the commercial tools. More importantly, none of these commercial tools offers a comprehensive approach for successful VoIP deployment. In particular, none gives any prediction for the total number of calls that can be supported by the network taking into account important design and engineering factors. These factors include VoIP flow and call distribution, future growth capacity, performance thresholds, impact of VoIP on existing network services and applications, and impact background traffic on VoIP. This chapter attempts to address those important factors and layout a comprehensive methodology for a successful deployment of any multimedia application such as VoIP and videoconferencing. However, the chapter focuses on VoIP as the new service of interest to be deployed. The cahpater also contains many useful engineering and design guidelines, and discusses many practical issues pertaining to the deployment of VoIP. These issues include characteristics of VoIP traffic and QoS requirements, VoIP flow and call distribution, defining future growth

capacity, and measurement and impact of background traffic. As a case study, we illustrate how our approach and guidelines can be applied to a typical network of a small enterprise.

The rest of the chapter is organized as follows. Section 2 outlines an eight-step methodology to deploy successfully VoIP in data networks. Each step is described in considerable detail. Section 3 presents a case study of a typical data network of a small enterprise. The methodology is applied to deploy VoIP on such a network. Section 4 summarizes and concludes the study.

2 Step-by-Step Methodology

In this section a step-by-step methodology is described for deploying VoIP. Figure 1 shows a flowchart of a methodology consisting of eight steps for a successful VoIP deployment. The first four steps are independent and can be performed in parallel. Before embarking on the analysis and simulation study, in Step 6 and Step 7, Step 5 must be carried out which requires any early and necessary redimensioning or modifications to the existing network. As shown, both Step 6 and Step 7 can be done in parallel. The final step is pilot deployment. It is worth noting that these steps and methodology can be utilized for the deployment of a variety of network services other than VoIP. Such network services may include videoconferencing, p2p,online gamine, IPTV, ERP or SAP services, etc. Work in [3, 4] show how these steps can be applied in assessing the readiness of IP networks to support and deploy desktop videoconferencing.

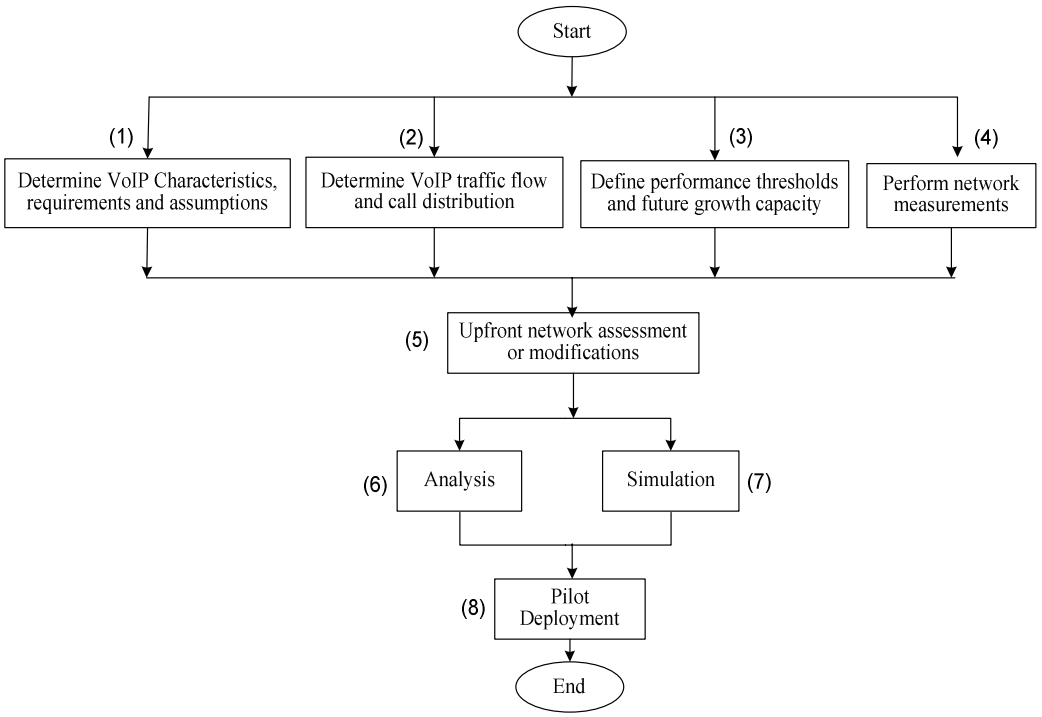


Figure 1. Flowchart of an eight-step methodology

2.1 VoIP Traffic Characteristics, Requirements, and Assumptions

For introducing a new network service such as VoIP, one has to first characterize the nature of its traffic, QoS requirements, and any additional components or devices. For simplicity, we assume a point-to-point conversation for all VoIP calls with no call conferencing. For deploying VoIP, a *gatekeeper* or *CallManager* node has to be added to the network [5,6,7]. The *gatekeeper* node handles signaling for establishing, terminating, and authorizing connections of all VoIP calls. Also a VoIP *gateway* is required to handle external calls. A VoIP *gateway* is responsible for converting VoIP calls to/from the Public Switched Telephone Network (PSTN). As an engineering and design issue, the placement of these nodes in the network becomes crucial. We will tackle this issue in design step 5. Other hardware requirements include a VoIP client terminal, which can be a separate VoIP device, i.e., IP phones, or a typical PC or workstation

that is VoIP-enabled. A VoIP-enabled workstation runs VoIP software such as IP SoftPhones [8-10].

Figure 2 identifies the end-to-end VoIP components from sender to receiver [11]. The first component is the *encoder* which periodically samples the original voice signal and assigns a fixed number of bits to each sample, creating a constant bit rate stream. The traditional sample-based encoder G.711 uses Pulse Code Modulation (PCM) to generate 8-bit samples every 0.125 ms, leading to a data rate of 64 kbps [12]. The *packetizer* follows the *encoder* and encapsulates a certain number of speech samples into packets and adds the RTP, UDP, IP, and Ethernet headers. The voice packets travel through the data network. An important component at the receiving end, is the *playback buffer* whose purpose is to absorb variations or jitter in delay and provide a smooth playout. Then packets are delivered to the *depacketizer* and eventually to the *decoder* which reconstructs the original voice signal.

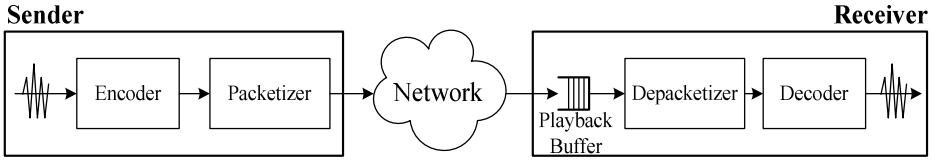


Figure 2. End-to-end components of VoIP

We will follow the widely-adopted recommendations of H.323, G.711, and G.714 standards for VoIP QoS requirements [13,14]. Table 1 compares some commonly-used ITU-T standard codecs and the amount of one-way delay that they impose. To account for upper limits and to meet desirable quality requirement according to ITU recommendation P.800 [15], we will adopt G.711u codec standards for the required delay and bandwidth. G.711u yields around 4.4 MOS

rating. MOS, *Mean Opinion Score*, is a commonly used VoIP performance metric given in a scale of 1 to 5, with 5 is the best [16,17]. However, with little compromise to quality, it is possible to implement different ITU-T codecs that yield much less required bandwidth per call and relatively a bit higher, but acceptable, end-to-end delay. This can be accomplished by applying compression, silence suppression, packet loss concealment, queue management techniques, and encapsulating more than one voice packet into a single Ethernet frame [5,11,18-23].

Table 1. Common ITU-T codecs and their defaults

Codec	Data rate (kbps)	Datagram size (ms)	A/D Conversion delay (ms)	Combined bandwidth (bi-directional) (kbps)
G.711u	64.0	20	1.0	180.80
G.711a	64.0	20	1.0	180.80
G.729	8.0	20	25.0	68.80
G.723.1 (MPMLQ)	6.3	30	67.5	47.80
G.723.1 (ACELP)	5.3	30	67.5	45.80

2.1.1 End-to-End Delay for a Single Voice Packet

Figure 2 illustrates the sources of delay for a typical voice packet. The end-to-end delay is sometimes referred to by M2E or Mouth-to-Ear delay [9]. G.714 imposes a maximum total one-way packet delay of 150ms end-to-end for VoIP applications [14]. In [24] a delay of up to 200ms was considered to be acceptable. We can break this delay down into at least three different contributing components, which are as follows (i) encoding, compression, and packetization delay at the sender (ii) propagation, transmission and queuing delay in the network and (iii) buffering, decompression, depacketization, decoding, and playback delay at the receiver.

2.1.2 Bandwidth for a Single Call

The required bandwidth for a single call, one direction, is 64 kbps. G.711 codec samples 20ms

of voice per packet. Therefore, 50 such packets need to be transmitted per second. Each packet contains 160 voice samples in order to give 8000 samples per second. Each packet is sent in one Ethernet frame. With every packet of size 160 bytes, headers of additional protocol layers are added. These headers include RTP + UDP + IP + Ethernet with preamble of sizes 12 + 8 + 20 + 26, respectively. Therefore, a total of 226 bytes, or 1808 bits, needs to be transmitted 50 times per second, or 90.4 kbps, in one direction. For both directions, the required bandwidth for a single call is 100 pps or 180.8 kbps assuming a symmetric flow.

2.1.3 Other Assumptions

Throughout our analysis and work, we assume voice calls are symmetric and no voice conferencing is implemented. We also ignore the signaling traffic generated by the *gatekeeper*. We base our analysis and design on the worst-case scenario for VoIP call traffic. The signaling traffic involving the *gatekeeper* is mostly generated prior to the establishment of the voice call and when the call is finished. This traffic is relatively small compared to the actual voice call traffic. In general, the *gatekeeper* generates no or very limited signaling traffic throughout the duration of the VoIP call for an already established on-going call [5].

In this chapter, we will implement no QoS mechanisms that can enhance the quality of packet delivery in IP networks. A myriad of QoS standards are available and can be enabled for network elements. QoS standards may include IEEE 802.1p/Q, the IETF's RSVP, and DiffServ. Analysis of implementation cost, complexity, management, and benefit must be weighed carefully before adopting such QoS standards. These standards can be recommended when the cost for upgrading some network elements are high and the network resources are scarce and

heavily loaded.

2.2 VoIP Traffic Flow and Call Distribution

Knowing the current telephone call usage or volume of the enterprise is an important step for a successful VoIP deployment. Before embarking on further analysis or planning phases for a VoIP deployment, collecting statistics about of the present call volume and profiles is essential. Sources of such information are organization's PBX, telephone records and bills. Key characteristics of existing calls can include the number of calls, number of concurrent calls, time, duration, etc. It is important to determine the locations of the call endpoints, i.e., the sources and destinations, as well as their corresponding path or flow. This will aid in identifying the call distribution and the calls made internally or externally. Call distribution must include percentage of calls within and outside of a floor, building, department, or organization. As a good capacity planning measure, it is recommended to base the VoIP call distribution on the busy hour traffic of phone calls for the busiest day of a week or a month. This will ensure support of the calls at all times with high QoS for all VoIP calls. When such current statistics are combined with the projected extra calls, we can predict the worst-case VoIP traffic load to be introduced to the existing network.

2.3 Define Performance Thresholds and Growth Capacity

In this step we define the network performance thresholds or operational points for a number of important key network elements. These thresholds are to be considered when deploying the new service. The benefit is twofold. First, the requirements of the new service to be deployed are satisfied. Second, adding the new service leaves the network healthy and susceptible to future

growth.

Two important performance criteria are to be taken into account. First is the maximum tolerable end-to-end delay; and second is the utilization bounds or thresholds of network resources. The maximum tolerable end-to-end delay is determined by the most sensitive application to run on the network. In our case, it is 150ms end-to-end for VoIP. It is imperative to note that if the network has certain delay-sensitive applications, the delay for these applications should be monitored, when introducing VoIP traffic, such that they do not exceed their required maximum values. As for the utilization bounds for network resources, such bounds or thresholds are determined by factors such as current utilization, future plans, and foreseen growth of the network. Proper resource and capacity planning is crucial. Savvy network engineers must deploy new services with scalability in mind, and ascertain that the network will yield acceptable performance under heavy and peak loads, with no packet loss. VoIP requires almost no packet loss. In literature 0.1% to 5% packet loss was generally asserted [8,23-25]. However, in [26] the required VoIP packet loss was conservatively suggested to be less than 10^{-5}. A more practical packet loss, based on experimentation, of below 1% was required in [24]. Hence, it is extremely important not to utilize fully the network resources. As rule-of-thumb guideline for switched fast full-duplex Ethernet, the average utilization limit of links should be 190%, and for switched shared fast Ethernet, the average limit of links should be 85% [27].

The projected growth in users, network services, business, etc. must be all taken into consideration to extrapolate the required growth capacity or the future growth factor. In our study we will ascertain that 25% of the available network capacity is reserved for future growth

and expansion. For simplicity, we will apply this evenly to all network resources of the router, switches, and switched-Ethernet links. However, keep in mind this percentage in practice can be variable for each network resource and may depend on the current utilization and the required growth capacity. In our methodology, the reservation of this utilization of network resources is done upfront, before deploying the new service, and only the left-over capacity is used for investigating the network support of the new service to be deployed.

2.4 Perform Network measurements

In order to characterize the existing network traffic load, utilization, and flow, network measurements have to be performed. This is a crucial step as it can potentially affect results to be used in analytical study and simulation. There are a number of tools available commercially and non-commercially to perform network measurements. Popular open-source measurement tools include MRTG, STG, SNMPUtil, and GetIF [28]. A few examples of popular commercially measurement tools include HP OpenView, Cisco Netflow, Lucent VitalSuite, Patrol DashBoard, Omegon NetAlly, Avaya ExamiNet, NetIQ Vivinet Assessor, etc.

Network measurements must be performed for network elements such as routers, switches, and links. Numerous types of measurements and statistics can be obtained using measurement tools. As a minimum, traffic rates in bps (bits per second) and pps (packets per second) must be measured for links directly connected to routers and switches. To get adequate assessment, network measurements have to be taken over a long period of time, at least 24-hour period. Sometimes it is desirable to take measurements over several days or a week.

One has to consider the worst-case scenario for network load or utilization in order to ensure good QoS at all times including peak hours. The peak hour is different from one network to another and it depends totally on the nature of business and the services provided by the network.

2.5 Upfront Network Assessment and Modifications

In this step we assess the existing network and determine, based on the existing traffic load and the requirements of the new service to be deployed, if any immediate modifications are necessary. Immediate modifications to the network may include adding and placing new servers or devices (such as VoIP gatekeeper, gateways, IP phones), upgrading PCs, and re-dimensioning heavily utilized links. As a good upgrade rule, topology changes need to be kept to minimum and should not be made unless it is necessary and justifiable. Over-engineering the network and premature upgrades are costly and considered as poor design practices.

Network engineers have to take into account on the existing traffic load. If any of network links is heavily utilized, e.g. 30-50%, the network engineer should decide to re-dimension the link to 1-Gbps link at this stage. As for shared links, the replacement or re-dimensioning of such links must be decided on carefully. Shared Ethernet scales poorly and are not recommended for real-time and delay-sensitive applications. A shared link introduces excessive and variable latency under heavy loads and when subjected to intense bursty traffic [27]. In order to consistently maintain the VoIP QoS, a switched fast full-duplex Ethernet LAN becomes necessary.

2.6 Analysis

VoIP is bounded by two important metrics. First is the available bandwidth. Second is the end-

to-end delay. The actual number of VoIP calls that the network can sustain and support is bounded by those two metrics. Depending on the network under study, either the available bandwidth or delay can be the key dominant factor in determining the number of calls that can be supported.

2.6.1 Bandwidth Bottleneck Analysis

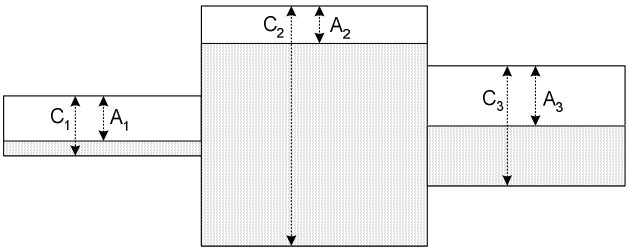


Figure 3. Bandwidth bottleneck for a path of three network elements

Therefore the theoretical maximum number of calls that can be supported by a network element E_i can be expressed in terms of A_i as

$$MaxCalls_i = \frac{A_i(1-growth_i)}{CallBW}, \quad (1)$$

where $growth_i$ is the growth factor of network element E_i, and takes a value from 0 to 1. $CallBW$ is the VoIP bandwidth for a single call imposed on E_i. As previously discussed in design step 2 of Section 2.2, the bandwidth for one direction is given as 50 pps or 90.4 kbps. In order to find the bottleneck network element that limits the total number of VoIP calls, one has to compute the maximum number of calls that can be supported by each network element, as in

equation (1), and the percentage of VoIP traffic flow passing by this element. The percentage of VoIP traffic flow for E_i, denoted as $flow_i$, can be found by examining the distribution of the calls. The total number of VoIP calls that can be supported by a network can be expressed as

$$TotalCallsSupported = \min_{i=1,\ldots,N}\left(\frac{MaxCalls_i}{flow_i}\right). \quad (2)$$

Let us for the sake of illustration compute the $MaxCalls_i$ and $flow_i$ supported by the Router, Switch 1, and uplink from Switch 2 to the Router. Table 3 shows the maximum calls that can be supported by those network elements. For our network example, we choose $growth_i$ to be 25% for all network elements. u_i is determined by Table 2. C_i, for the router and the switch is usually given by the product datasheets. According to [30, 31], the capacity C_i for the router or the switch, is 25,000pps and 1.3M pps, respectively.

2.6.2 Delay Analysis

As defined in Section 2.3 for the existing network, the maximum tolerable end-to-end delay for a VoIP packet is 150 ms. The maximum number of VoIP calls that the network can sustain is bounded by this delay. We must always ascertain that the worst-case end-to-end delay for all the calls must be less than 150 ms. It should be kept in mind that our goal is to determine the network capacity for VoIP, i.e. the maximum number of calls that existing network can support while maintaining VoIP QoS. This can be done by adding calls incrementally to the network while monitoring the threshold or bound for VoIP delay. When the end-to-end delay, including network delay, becomes larger than 150 ms, the maximum number of calls can then be known.

As described in Section 2.1, there are three sources of delay for a VoIP stream: sender, network,

and receiver. An equation is given in [26] to compute the end-to-end delay *D* for a VoIP flow in one direction from sender to receiver.

$$D = D_{pack} + \sum_{h \in Path} (T_h + Q_h + P_h) + D_{play} ,$$

where D_{pack} is the delay due to packetization at the source. At the source, there is also D_{enc} and $D_{process}$. D_{enc} is the encoder delay of converting A/D signal into samples. $D_{process}$ is the PC of IP phone processing that includes encapsulation . In G.711, D_{pack} and D_{enc}, are 20 ms and 1ms, respectively. Hence, it is appropriate for our analysis to have a fixed delay of 25 ms being introduced at the source, assuming worst case situation. D_{play} is the playback delay at the receiver, including jitter buffer delay. The jitter delay is at most 2 packets, i.e. 40ms. If the receiver's delay of $D_{process}$ is added, we obtain a total fixed delay of 45 ms at the receiver. $T_h + Q_h + P_h$ is the sum of delays incurred in the packet network due to transmission, queuing, and propagation going through each hop *h* in the path from the sender to the receiver. The propagation delay P_h is typically ignored for traffic within a LAN, but not for a WAN. For transmission delay T_h and queueing delay Q_h we apply queueing theory. Hence the delay to be introduced by the network, expressed as $\sum_{h \in Path} (T_h + Q_h)$, should not exceed (150 – 25 – 45) or 80 ms.

We utilize queueing analysis to approximate and determine the maximum number of calls that the existing network can support while maintaining a delay of less than 80ms. In order to find the network delay, we utilize the principles of Jackson theorem for analyzing queueing networks. In particular, we use the approximation method of analyzing queueing networks by

decomposition discussed in [32]. In this method, the arrival rate is assumed to be Poisson and the service times of network elements are exponentially distributed. Analysis by decomposition is summarized in first isolating the queueing network into subsystems, e.g., single queueing node. Next, analyzing each subsystem separately, considering its own network surroundings of arrivals and departures. Then, finding the average delay for each individual queueing subsystem. And finally, aggregating all the delays of queueing subsystems to find the average total end-to-end network delay.

For our analysis we assume the VoIP traffic to be Poisson. In reality, the inter-arrival time, $1/\lambda$, of VoIP packets is constant, and hence its distribution is deterministic. However, modeling the voice arrival as Poisson gives adequate approximation according to [26], especially when employing a high number of calls. More importantly, the network element with a non-Poisson arrival rate makes it difficult to approximate the delay and lead to intractable analytical solution. Furthermore, analysis by decomposition method will be violated if the arrival rate is not Poisson.

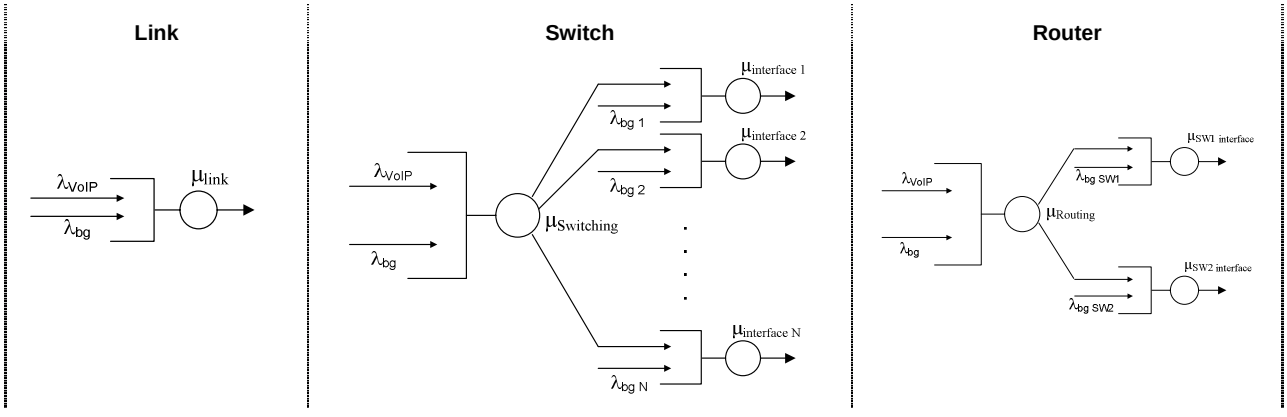


Figure 4. Queueing models for a network link, switch, and router

Figure 4 shows queueing models for three network elements of the router, switch and link. The

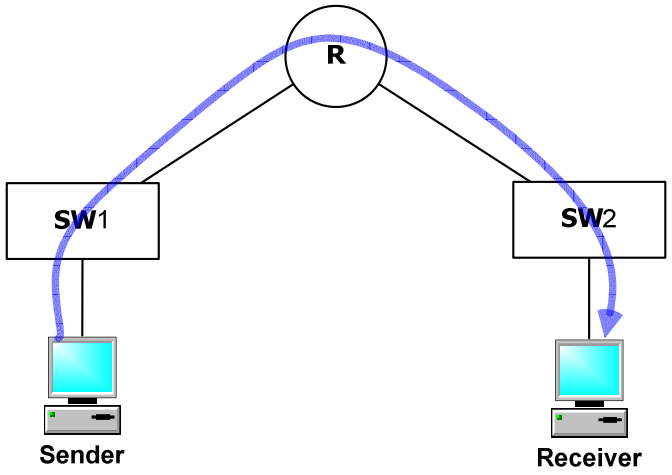
queueing model for the router has two outgoing interfaces: an interface for SW1 and another for SW2. The number of outgoing interfaces for the switches are many, and such a number depends on the number of ports for the switch. We modeled the switches and the router as *M/M/1* queues. Ethernet links are modeled as *M/D/1* queues. This is appropriate since the service time for Ethernet links is more of a deterministic than variable. However, the service times of the switches and the router are not deterministic since these are all CPU-based devices. According to the datasheet found in [30,31], the switches and the router (which are used for our case study in Section 3) have somewhat similar design of a store-and-forward buffer pool with a CPU responsible for pointer manipulation to switch or route a packet to different ports. [33] provides a comprehensive models of common types of switches and routers. According to [35], the average delay for a VoIP packet passing through an *M/M/1* queue is basically $1/(\mu-\lambda)$, and through an *M/D/1* queue is $\left(1-\frac{\lambda}{2\mu}\right)/(\mu-\lambda)$, where λ is the mean packet arrival rate and μ is the mean network element service rate. The queueing models in Figure 4 assume Poisson arrival for both VoIP and background traffic. In [26], it was concluded that modeling VoIP traffic as Poisson is adequate. However and in practice, background traffic is bursty in nature and characterized as self-similar with long range dependence [35]. For our analysis and design, using bursty background traffic is not practical. For one thing, under the network of queues being considered an analytical solution becomes intractable when considering non-Poisson arrival. Also, it is important to remember that in order to ensure good QoS at all times, we base our analysis and design on the worst-case scenario of network load or utilization, i.e., the peak of aggregate bursts. And thus in a way our analytical approach takes into account the bursty nature of traffic.

It is worth noting that the analysis by decomposition of queueing networks in [32] assumes exponential service times for all network elements including links. But [36] proves that acceptable results with adequate accuracy can be still obtained if the homogeneity of service times of nodes in the queueing network is deviated. [36] shows that the main system performance is insensitive to violations of the homogeneity of service times. Also, it was noted that when changing the models for links from *M/D/1* to *M/M/1*, a negligible difference was observed. More importantly, as will be demonstrated in this chapter with simulation, our analysis gives a good approximation.

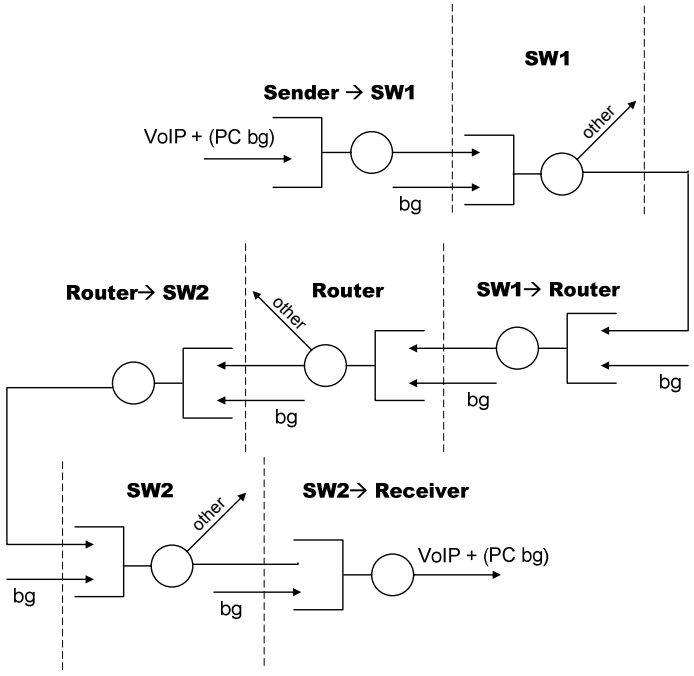
The total end-to-end network delay starts from the Ethernet outgoing link of the sender PC or IP phone to the incoming link of receiver PC or IP phone. To illustrate this further, let us compute the end-to-end delay encountered for a single call initiated between two building floors. Figure 5 shows an example of how to compute the network delay. Figure 5a shows the path of a unidirectional voice traffic flow going from one floor to another. Figure 5b shows the corresponding networking queueing model for such a path.

For Figure 5b, in order to compute the end-to-end delay for a single bi-directional VoIP call, we must compute the delay at each network element. We show how to compute the delay for the switches, links, and router. For the switch, $\mu = (1-25\%)\times 1.3$ Mpps, where 25% is the growth factor. We assume the switch has a capacity of 1.3 Mpps. $\lambda = \lambda_{VoIP} + \lambda_{bg}$, where λ_{VoIP} is the total added new traffic from a single VoIP in pps, and λ_{bg} is the background traffic in pps. For an uplink or downlink, $\mu = (1-25\%)\times 100$Mbps, $\lambda = \lambda_{VoIP} + \lambda_{bg}$. Since the service rate is in bps, λ_{VoIP} and λ_{bg} must be expressed in bps. Similarly for the router, $\mu = (1-25\%)\times 25{,}000$pps

and $\lambda = \lambda_{VoIP} + \lambda_{bg}$. Both λ_{VoIP} and λ_{bg} must be expressed in pps. Remember for a single bi-directional VoIP call, λ_{VoIP} at the router and switches for a single call will be equal to 100pps. However, for the uplink and downlink links, it is 90.4 kbps. One should consider no λ_{bg} for the outgoing link if IP phones are used. For multimedia PCs which equipped with VoIP software, a λ_{bg} of 10% of the total background traffic is utilized in each floor. In Figure 5, we use multimedia PCs.



(a) Unidirectional voice traffic flow path from Floor 1 to Floor 3



(b) Corresponding network queueing model of the entire path

Figure 5. Computing network delay

The total delay for a single VoIP call of Figure 5b, can be determined as follows:

$$D_{path} = D_{Sender-SW1\ Link} + D_{SW1} + D_{SW1-Router\ Link} + D_{Router} + D_{Router-SW2\ Link} + D_{SW2} + D_{SW2-Receiver\ Link}$$

In order to determine the maximum number of calls that can be supported by an existing network while maintaining VoIP delay constraint, we devise a comprehensive algorithm that basically determines network capacity in terms of VoIP calls. Algorithm 1 is essentially the analytical simulator's engine for computing the number of calls based on delay bound. Calls are added iteratively until the worst-case network delay of 80 ms has reached.

Algorithm 1: Compute maximum number of calls based on VoIP delay constraint

Input: n : number of network elements
$\lambda[1..n]$: background traffic for network elements $1,2,..n$
$Delay[1..n]$: delay for network elements $1,2,..n$
P : set of call-flow paths (p) where p is a subset of $\{1,2,..n\}$
Output: MaxCalls: maxmimum number of calls

$\lambda_{VoIP} \leftarrow$ 100pps, or 180.8kbps;
VoIP_MaxDelay ← 80; // network delay for VoIP call in ms
MaxDelay ← 0;
MaxCalls ← -1;
$Delay[1..n] \leftarrow$ 0;

while MaxDelay < VoIP_MaxDelay **do**

1. MaxCalls ← MaxCalls + 1
2. Generate a call according to call distribution and let p_c be its flow path
3. **for each** element i in p_c do
   $\lambda_i \leftarrow \lambda_i + \lambda_{VoIP}$
   if i is a link **then**
   $Delay_i \leftarrow (1-\lambda_i/2\mu_i)/(\mu_i-\lambda_i)$
   Else
   $Delay_i \leftarrow 1/(\mu_i-\lambda_i)$
   end if
   end for
4. **for each** p in P where $p \cap p_c \neq \phi$ **do**
   PathDelay(p) $\leftarrow \sum_i Delay_i$, where i is a network element in path p
   if PathDelay(p) > MaxDelay **then**
   MaxDelay ← PathDelay(p)
   end if
   end for

end while

It is to be noted that in Step 2 of Algorithm 1 that a uniform random number generator is used to generate VoIP calls according to the call distribution. Call distribution must be in the form of values from 1 to 100%. Also, the delay computation for the link in Step 3 is different than other network elements such as switches and routers. For the links, it is more appropriate to use the average delay formula for *M/D/1* as the service rate μ is almost constant. However for the

switches and routers, it is more appropriate to use the average delay formula for *M/M/1* as the service rate μ is variable since the routers and switches are CPU-based. For the links, the average delay per packet is calculating first using the average bit delay and then multiplying it by the packet size which is 1808 bits to get the delay per packet. For this, the link service rate and incoming rate have to be all in bps. However for switches and routers, the calculation is all done in pps. In the algorithm above, it is worth noting that the link delay calculation in Algorithm 1 is for a unidirectional link. The total bandwidth that will be introduced as a result of adding one call on link is 50pps in one direction, and another 50pps in the opposite direction. However for switches and routes, the extra bandwidth introduced per call will be 100pps.

2.7 Simulation

The object of the simulation is to verify analysis results of supporting VoIP calls. There are many available simulation packages that can be used including commercial and open source. A list and classification of such available network simulation tools can be found in [39]. In our case study in Section 3, we used the popular MIL3's OPNET Modeler simulation package, Release 8.0.C [40]. OPNET Modeler contains a vast amount of models of commercially available network elements, and has various real-life network configuration capabilities. This makes the simulation of real-life network environment close to reality. Other features of OPNET include GUI interface, comprehensive library of network protocols and models, source code for all models, graphical results and statistics, etc. More importantly, OPNET has gained considerable popularity in academia as it is being offered free of charge to academic institutions. That has given OPNET an edge over DES NS2 in both market place and academia.

2.8 Pilot Deployment

Before embarking on changing any of the network equipment, it is always recommended to build a pilot deployment of VoIP in a test lab to ensure smooth upgrade and transition with minimum disruption of network services. A pilot deployment comes after training of IT staff. A pilot deployment is the place for the network engineers, support and maintenance team to get firsthand experience with VoIP systems and their behavior. During the pilot deployment, the new VoIP devices and equipment are evaluated, configured, tuned, tested, managed, monitored, etc. The whole team needs to get comfortable with how VoIP works, how it mixes with other traffic, how to diagnose and troubleshoot potential problems. Simple VoIP calls can be set up and some benchmark testing can be performed.

3 Case Study

In this section we present a case study of a typical IP network of a small enterprise residing in a high-rise building. We briefly describe the methodology of successfully deploying VoIP in this network. The network is shown in Figure 6. The network is Ethernet-based and has two Layer-2 Ethernet switches connected by a router. The router is Cisco 2621, and the switches are 3Com Superstack 3300. Switch 1 connects Floor 1 and Floor 2 and two servers; while Switch 2 connects Floor 3 and four servers. Each floor LAN is basically a shared Ethernet connecting employee PCs with workgroup and printer servers. The network makes use of VLANs in order to isolate broadcast and multicast traffic. A total of five LANs exist. All VLANs are port based. Switch 1 is configured such that it has three VLANs. VLAN1 includes the database and file servers. VLAN2 includes Floor 1. VLAN3 includes Floor2. On the other hand, Switch 2 is

configured to have two VLANs. VLAN4 includes the servers for E-mail, HTTP, Web & cache proxy, and firewall. VLAN5 includes Floor 3. All the links are switched Ethernet 100Mbps full duplex except for the links for Floor 1, Floor 2, and Floor 3 which are shared Ethernet 100Mbps half duplex.

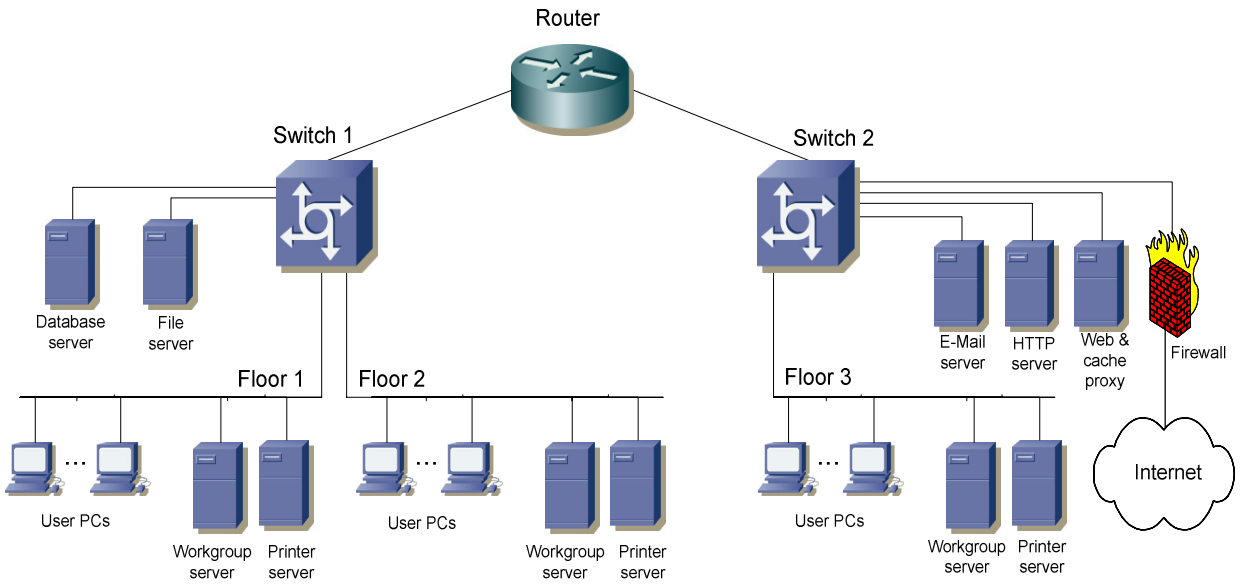


Figure 6. Topology of a small enterprise network

For background traffic, we assume a traffic load not exceeding 10% of the capacity of links. For precise values are described in [41,42]. The values are those of peak-hour utilization of traffic of links in both directions connected to the router and the two switches of the network topology of Figure 6. These measured results will be used in our analysis and simulation study. For call distributions, thresholds, and projected growth, we used those described in [41,42].

For upfront assessment and based on the hardware requirement for deploying VoIP of Step 5, two new nodes have to be added to the existing network: a VoIP *gateway* and a *gatekeeper*. As

a network design issue, an appropriate node placement is required for these two nodes. Since most of the users reside on Floor 1 and Floor 2 and connected directly to Switch 1, connecting the *gatekeeper* to Switch 1 is practical in order to keep the traffic local. For the VoIP *gateway*, we connect it to Switch 2 in order to balance the projected load on both switches. Also it is more reliable and fault-tolerant not to connect both nodes to the same switch in order to eliminate problems that stem from a single point of failure. For example, if Switch 2 fails, only external calls to/from the network will be affected. It is proper to include the gatekeeper to be a member of VLAN1 of Switch 1 which includes the database and file servers. This isolates the gatekeeper from multicast and broadcast traffic of Floor 1 and Floor 2. In addition, the *gatekeeper* can access locally the database and file servers to record and log phone calls. On the other hand, we create a separate VLAN for the *gateway* in order to isolate the *gateway* from multicast and broadcast traffic of Floor 3 and the servers of switch 2. Therefore, the network has now a total of six VLANs. Figure 7 shows the new network topology after the incorporation of necessary VoIP components. As shown, two new *gateway* and *gatekeeper* nodes for VoIP were added and the three shared Ethernet LANs were replaced by 100Mbps switched Ethernet LANs.

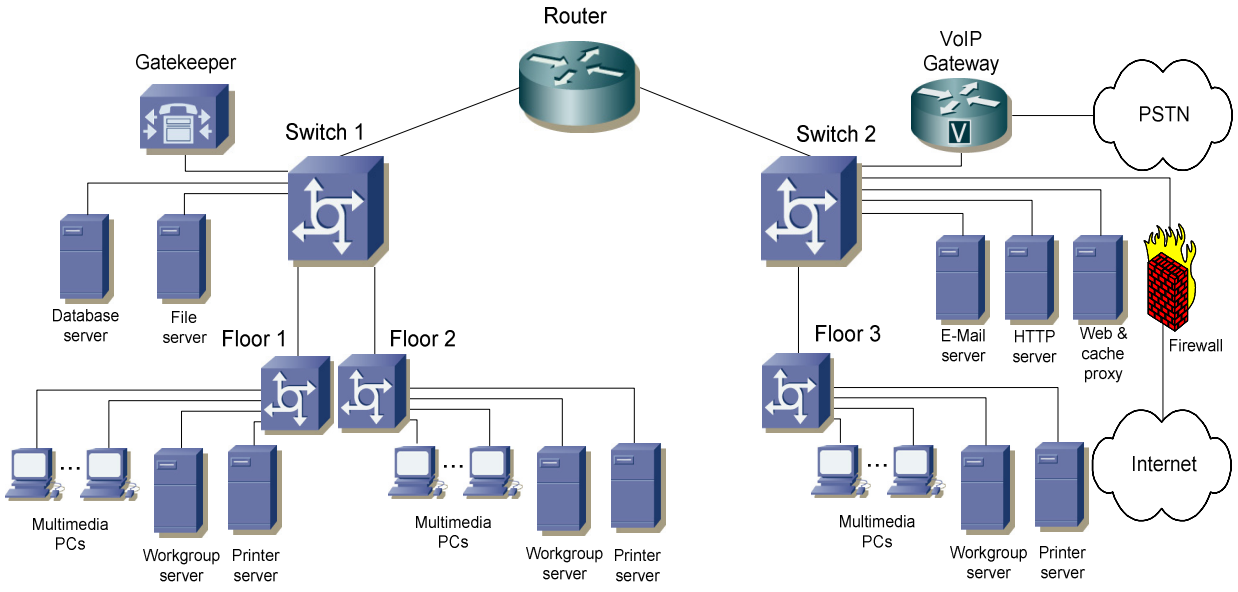


Figure 7. Network topology with necessary VoIP Components

As for Step 6 of analysis, there are two implementation options. One is using MATLAB and the second is using the analytical simulator described in [43]. With first option, MATLAB programs can be written to implement bandwidth and delay analyses described in Section 2.6. Algorithm 1 was implemented using MATLAB and the results for the worst incurred delay are plotted in Figure 8. It can be observed from the figure that the delay increases sharply when the number of calls go beyond 310 calls. To be more precise, MATLAB results shows the number of calls that are bounded by the 80 ms delay is 315. For bandwidth analysis to compute the $MaxCalls_i$ for all network elements, it turns out that the router is the bottleneck element. Hence, $TotalCallsSupported$ is 313 VoIP calls. When comparing the number of calls that network can sustain based on bottleneck bandwidth and worst-delay analysis, we find the number of calls is limited by the available bandwidth more than the delay, though the difference is small. Therefore, we can conclude that the maximum number of calls that can be sustained by the existing network is 313.

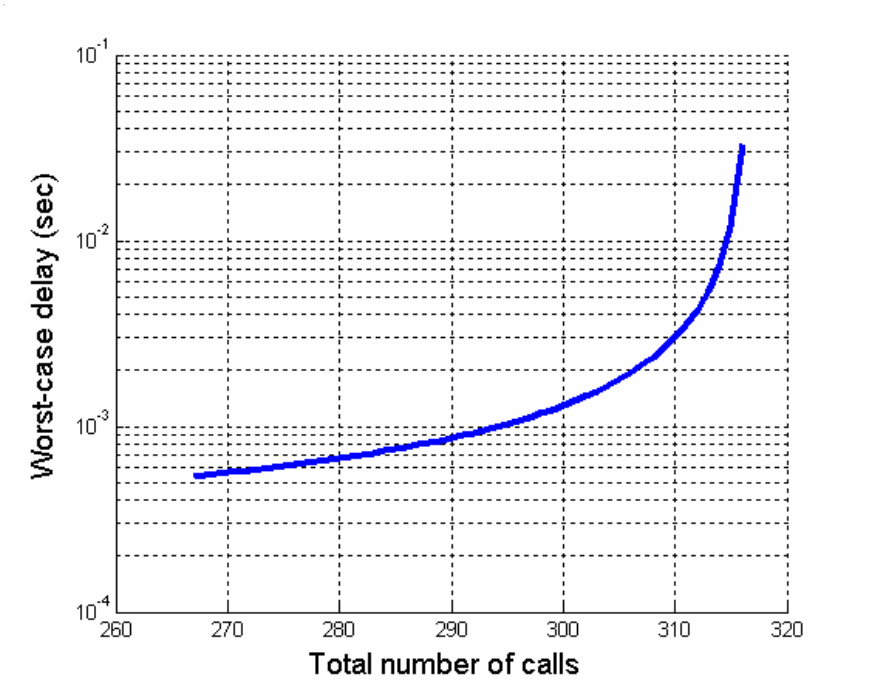


Figure 8. Worst incurred delay vs. number of VoIP calls

The second option is more flexible and convenient as it avoids using MATLAB. It uses a GUI-based analytical simulator which works on any generic network. The analytical simulator is publicly available. The simulator can be downloaded from http://www.ccse.kfupm.edu.sa/~salah/VoIP_Analytical_Simulator.rar. A complete description of the simulator can be found in [43]. The simulator has an interface that is GUI with the ability of building network topology by clicking and dropping (i.e., comparable to building a network in OPNET). In other words the simulator has drag-and-drop features to construct a generic network topology and feed it into the analytical engine. The simulator also allows users to set and configure variety of settings and parameters related VoIP deployment. The analytical engine is based on the analytical approach described in Section 2.6. The simulator gives results on how many VoIP calls can be supported within seconds. The user can easily tune the network configurations and parameters and determine the results within seconds. The results obtained by

the simulator and MATLAB were the same.

Figure 9 shows the corresponding network model constructed by the VoIP Simulator for the network topology of Figure 7. In order to avoid having numerous PC nodes or IP phones per floor representing end-users (and therefore clutter the network topology diagram), Floor LANs have been simply modeled as a LAN that enclose an Ethernet switch and three designated Ethernet PCs used to model the activities of the LAN users. For example, Floor 1 has three nodes (labeled as F1C1, F1C2, and F1C3). F1C1 is a source for sending voice calls. F1C2 is a sink for receiving voice calls. F1C3 is a sink and source of background traffic. This model allows for generating background traffic and also establishing intra-floor calls or paths from F1_C1 and F1C2, and passing through the floor switch of F1SW. The sending and sinking PC nodes of VoIP (e.g., F1C1 and F1C2)have infinite capacity and there is no limit on how many calls can be added or received by them. We ignore the signaling traffic generated by the *gatekeeper*. We base our analysis and design on the worst-case scenario for VoIP call traffic. The signaling traffic involving the *gatekeeper* is only generated prior to the establishment of the voice call and when the call is finished. This traffic is relatively limited and small compared to the actual voice call traffic. In general, the *gatekeeper* generates no signaling traffic throughout the duration of the VoIP call for an already established on-going call.

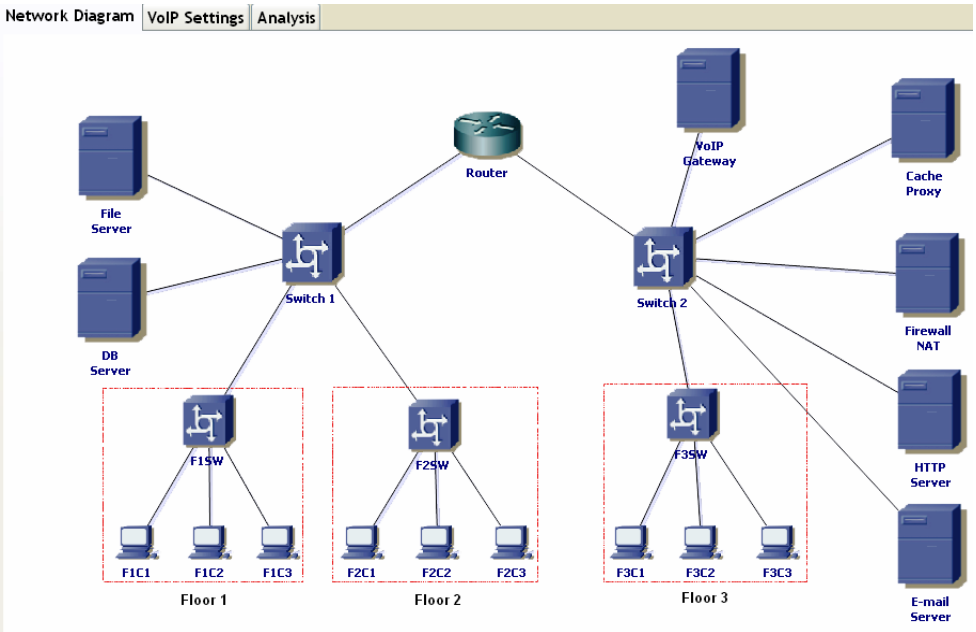
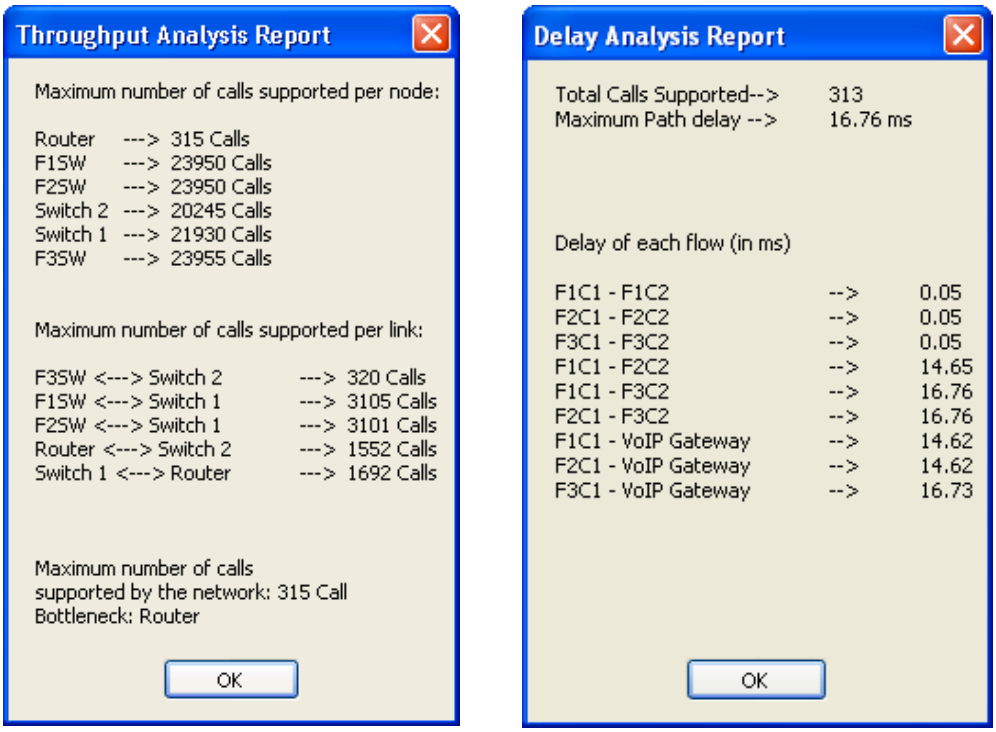


Figure 9. Corresponding network diagram constructed by analytical simulator

Figure 10 shows the reports of throughput and delay analyses. Figure 10(a) reports the number of calls that can be supported based on bandwidth analysis. A total of 315 calls can be supported for the whole network. In order to identify possible bottlenecks, the report also shows individual calls that can be supported per node and per link. It is shown that the router is the bottleneck, and supporting calls more than 315 calls would require definitely a replacement for the router. Figure 10(b) reports the number of calls that can be supported based on network analysis. A total of 313 calls can be supported such that the network delay of any of the specified VoIP flows does not exceed the required 80 ms. The figure shows that with 313 calls, a network delay of 16.76 ms will be introduced. This means when adding one more call, the network delay of a maximum of 80 ms was exceeded. The report of Figure 10(b) also exhibits the network delay per flow or path. In our example, there were a total of nine VoIP flows. As shown, the first triple is for intra-floor flows. The second triple is for inter-floor flows. And the third triple is for

external flows. Such information gives insight on the source of the delays as well as the path that causing most of the delays. As the figure shows, the inter-floor flows of F1C1-F3C2 and F2C1-F3C2 experience the largest delays, as they pass through the router.



(a) (b)

Figure 10. (a) Throughput Analysis Report, (b) Delay Analysis Report

We chose OPNET to verify our analytical approach. Detailed description of the simulation model, configurations, and results can be found in [43]. With OPNET simulation, the number of VoIP calls to be supported was 306. From results of analysis and simulation, it is apparent that both results are in line and give a close match. Based on the analytic approach, a total of 313 calls can be supported. Based on the simulation approach, a total of 306 calls can be supported There is only a difference of 7 calls. The difference can be contributed to the degree of accuracy

between the analytic approach and OPNET simulation. Our analytic approach is an approximation. Also, the difference is linked to the way the OPNET Modeler adds the distribution of the calls. It was found that external and inter-floor calls are added before intra-floor calls. In anyways, to be safe and conservative, one can consider the minimum number of calls of the two approaches.

Therefore, the following network design and engineering decisions can be justified from the analytic and simulation approaches. First, the existing network, with a reserved growth factor of 25%, can safely support up to 306 calls while meeting the VoIP QoS requirements and having no negative impact on the performance of existing network services or applications. Second, the primary bottleneck of the network is the router. If the enterprise under study is expected to grow in the near future, i.e., more calls are required than 306 calls, the router replacement is a must. The router can be replaced with a popular Layer-3 Ethernet switch, and thus relieving the router from routing inter-floor calls from Floor 1 to Floor 2. Before prematurely changing other network components, one has to find out how many VoIP calls can be sustained by replacing the router. To accomplish this, the design steps and guidelines outlined in this chapter must be revisited and re-executed. And finally, the network capacity to support VoIP is bounded more by the network throughput than the delay. This is due to the fact the existing network under study is small and does not have a large number of intermediate nodes. The network delay bound can become dominant if we a have a large-scale LAN or WAN.

4 Summary and Conclusion

This chapter outlined a step-by-step methodology on how VoIP can be deployed successfully in

today's existing IP networks. The methodology can help network designers determine quickly and easily how well VoIP will perform on a network prior to deployment. Prior to the purchase and deployment of VoIP equipment, it is possible to predict the number of VoIP calls that can be sustained by the network while satisfying QoS requirements of all existing and new network services and leaving enough capacity for future growth. In addition, the chapter discussed many design and engineering issues pertaining to the deployment of VoIP. These issues include characteristics of VoIP traffic and QoS requirements, VoIP flow and call distribution, defining future growth capacity, and measurement and impact of background traffic. A case study was presented for deploying VoIP in a small enterprise network. The methodology and guidelines outlined in this chapter were applied. Analysis and OPNET simulation were used to investigate throughput and delay bounds for such a network. Results obtained from analysis and simulation were in line and gave a close match. The methodology and guidelines presented in this chapter can be adopted for the deployment of man other network services (other than peer-to-peer VoIP). These services may include VoIP conferencing and messaging, videoconferencing, IPTV, online gaming, ERP, etc.

5 References

[1] M. Bearden, L. Denby, B. Karacali, J. Meloche, and D. T. Stott, "Assessing Network Readiness for IP Telephony," Proceedings of IEEE International Conference on Communications, ICC02, vol.4, 2002, pp. 2568-2572

[2] B. Karacali, L. Denby, and J. Melche, "Scalable Network Assessment for IP Telephony," Proceedings of IEEE International Conference on Communications (ICC04), Paris, June 2004, pp. 1505-1511.

[3] Salah, K., Calyam, P., and Bukhari, M. *"Assessing readiness of IP networks to support desktop videoconferencing using OPNET,"* International Journal of Network and Computer Applications, Elsevier Science, In Press.

[4] Salah, K., "*An Analytical Approach for Deploying Desktop Videoconferencing,"* IEE Proceedings Communications, Vol. 153(3) (2006), pp. 434-444.

[5] Goode B, "Voice over Internet Protocol (VoIP)," Proceedings of IEEE, vol. 90, no. 9, Sept. 2002, pp. 1495-1517.

[6] P. Mehta and S. Udani, "Voice over IP", *IEEE Potentials Magazine*, vol. 20, no. 4, October 2001, pp. 36-40.

[7] W. Jiang and H. Schulzrinne, "Towards Junking the PBX: Deploying IP Telephony," Proceedings of ACM 11th International Workshop on Network and Operating System Support for Digital Audio and Video, Port Jefferson, NY, June 2001, pp. 177-185

[8] B. Duysburgh, S. Vanhastel, B. DeVreese, C. Petrisor, and P. Demeester, "On the Influence of Best-Effort Network Conditions on the Perceived Speech Quality of VoIP Connections," Proceedings of IEEE 10th International Conference of Computer Communications and Networks, Scottsdale, AZ, October 2001, pp. 334-339.

[9] W. Jiang, K. Koguchi, and H. Schulzrinne, "QoS Evaluation of VoIP End-Points," Proceedings of IEEE International Conference on Communications, ICC'03, Anchorage, May 2003, pp. 1917-1921

[10] Avaya Inc., "Avaya IP voice quality network requirements," http://www1.avaya.com/enterprise/whitepapers, 2001.

[11] A. Markopoulou, F. Tobagi, M. Karam, "Assessing the quality of voice communications over internet backbones", *IEEE/ACM Transaction on Networking*, vol. 11, no. 5, 2003, pp. 747-760

[12] Recommendation G.711, "Pulse Code Modulation (PCM) of Voice Frequencies," ITU, November 1988

[13] Recommendation H.323, "Packet-based Multimedia Communication Systems," ITU, 1997.

[14] Recommendation G.114, "One-Way Transmission Time," ITU, 1996.

[15] ITU-T Recommendation P.800, "Methods for subjective determination of transmission quality," www.itu.in/publications/main_publ/itut.html

[16] L. Sun and E. C.. Ifeachor, "Prediction of Perceived Conversational Speech Quatlity and Effects of Playout Buffer Algorithms," Proceedings of International Conference on Communications, ICC'03, Anchorage, May 2003, pp. 1-6

[17] A. Takahasi, H. Yoshino, and N. Kitawaki, "Perceptual QoS Assessment Technologies for VoIP," *IEEE Communications Magazine*, vol. 42, no. 7, July 2004, pp. 28-34

[18] J. Walker and J. Hicks, "Planning for VoIP," NetIQ Corporation white paper, December 2002, http://www.telnetnetworks.ca/products/netIq/whitepapers/planning_for_voip.pdf

[19] Recommendation G.726, "40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM)," ITU, December 1990.

[20] Recommendation G.723.1, "Speech Coders: Dual Rate Speech Coder for Multimedia Communication Transmitting at 5.3 and 6.3 kbit/s," ITU, March 1996.

[21] Annex to Recommendation G.729, "Coding of Speech at 8kbit/s using Conjugate Structure Algebraic-Code-Excited Linear-Prediction (CS-ACELP)," Annex A: "Reduced Complexity 8 kbit/s CS-ACELP Speech Codec", ITU, November 1996.

[22] W. Jiang and H. Schulzrinne, "Comparison and Optimization of Packet Loss Repair Methods on VoIP Perceived Quality under Bursty Loss," Proceedings of ACM 12th International Workshop on Network and Operating System Support for Digital Audio and Video, Miami, FL, May 2002, pp. 73-81

[23] J. S. Han, S. J. Ahn, and J. W. Chung, "Study of Delay Patterns of Weighted Voice Traffic of End-to-End Users on the VoIP Network," *International Journal of Network Management*, vol. 12, no. 5, May 2002, pp. 271-280 (2002)

[24] J. H. James, B. Chen, and L. Garrison, "Implementing VoIP: A Voice Transmission Performance Progress Report," *IEEE Communications Magazine*, vol. 42, no. 7, July 2004, pp. 36-41

[25] W. Jiang and H. Schulzrinne, "Assessment of VoIP Service Availability in the Current Internet" Proceedings of International Workshop on Passive and Active Measurement (PAM2003), San Diago, CA, April 2003.

[26] M. Karam and F. Tobagi, "Analysis of delay and delay jitter of voice traffic in the Internet," *Computer Networks Magazine*, vol. 40, no. 6, December 2002, pp. 711-726 (2002)

[27] S. Riley and R. Breyer, "*Switched, Fast, and Gigabit Ethernet*," Macmillan Technical Publishing, 3rd Edition, 2000.

[28] CAIDA, http://www.caida.org/tools/taxonomy, April 2004.

[29] R. Prasad, C. Dovrolis, M. Murray, and K.C. Claffy, "Bandwidth Estimation: Metrics, Measurement Techniques, and Tools," *IEEE Network Magazine*, vol. 17, no. 6, December 2003, pp. 27-35

[30] Cisco Systems Inc., "Cisco 2621 Modular Access Router Security Policy," 2001, http://www.cisco.com/univercd/cc/td/doc/product/access/acs_mod/cis2600/secure/2621rect.pdf

[31] 3Com, “3Com Networking Product Guide,” April 2004, http://www.3com.co.kr/products/pdf/productguide.pdf

[32] K. M. Chandy and C. H. Sauer, “Approximate methods for analyzing queueing network models of computing systems,” *Journal of ACM Computing Surveys*, vol. 10, no. 3, September 1978, pp. 281-317.

[33] F. Gebali, *Computing Communication Networks: Analysis and Designs*, Northstar Digital Design, Inc., 3rd Edition, 2005.

[34] L. Kleinrock, Queueing Systems: Theory, vol 1, New York, Wiley, 1975.

[35] W. Leland, M. Taqqu, W. Willinger, D. Wilson, "On the Self-Similar Nature of Ethernet Traffic", *IEEE/ACM Transaction on Networking*, vol. 2, no. 1, February 1994, pp. 1-15

[36] R. Suri, “Robustness of Queueing Network Formulas,” *Journal of the ACM*, vol. 30, no. 3, July 1983, pp. 564-594.

[37] R. Onvural, “Survey of Closed Queueing Networks with Blocking,” *ACM Computing Surveys*, vol. 22, no. 2, June 1990, pp. 83-121

[38] J. Bolot, “End-to-End Packet Delay and Loss Behavior in the Internet,” Proceedings of ACM Conference on Communications, Architectures, Protocols and Applications, San Francisco, CA, October 1993, pp. 289-298

[39] K. Pawlikowski, H. Jeong and J. Lee, "On Credibility of Simulation Studies of Telecommunication Networks", *IEEE Communications Magazine*, vol. 40, no. 1, January 2002, pp. 132-139

[40] OPNET Technologies, http://www.mil3.com

[41] Salah, K., *“On the Deployment of VoIP in Ethernet Networks: Methodology and Case Study,”* International Journal of Computer Communications, Elsevier Science, Vol. 29(8) (2006), pp. 1039-1054.

[42] Salah, K., and Alkhoraidly, A., *“An OPNET-based Simulation Approach for Deploying VoIP,”* International Journal of Network Management, John Wiley, Vol. 16(3-4) (2006, pp. 159-183.

[43] Salah, K., Darwish, N., Saleem, M., and Shaaban, Y., *“An Analytical Simulator for Deploying IP Telephony,”* Accepted for publication to International Journal of Network Management, John Wiley.