NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

Mobile Agents

Graduation Project
COM – 400

Student: Ahmed Ali (20000540)

Supervisor: Mr. JAMAL FATIH

Nicosia - 2006

ACKNOWLEDGEMENTS

This project is done under the supervision of Mr. Jamal, I am very grateful to him who gave his technical and emotional support for the creation of this graduation project.

I will also like to thanks my all friends in Cyprus who gave their ever devotion and helped me for their all valuable information to complete this project.

Further I am very thankful to Near East University academic staff and all those teachers who helped me and encouraged me for the completion of my graduation project.

Finally my thanks go to whom my love will never end, my father and my mother, to my brothers and a sister, that help me a lot and their encouragement in my studies, so that I could be successful in my life. Thanks!

INTRODUCTION

The concept of agents is not unfamiliar one. The precepts of agent technology have existed in many of the applications we use today and take for granted. For example, your e-mail client is a type of agent. At your request, it goes about its business of collecting unread e-mail from your mail server. Contemporary e-mail clients will even presort your incoming messages into specified folders based on criteria that you define. In this manner, the software becomes an extension of the user, performing tasks on the user's behalf. Indeed, the computer itself can be considered an agent, as its primary task is to increase productivity through automation.

Recently, intelligent agents have become to vogue. These agents have some degree of artificial intelligence and are capable of taking judicious decisions within their realm of expertise. For example, the e-mail client can exhibit some sort of artificial intelligence to determine the importance of a particular piece of e-mail - possibly by scanning the message body for indicators of urgency. However, not all agents need to be intelligent.

One of the most interesting and much vaunted category of agents is mobile agents. Mobile agents can themselves be intelligent or non-intelligent. Unlike static agents, which are restricted to operate within a single machine or address space, mobile agents have the ability to migrate about the network, executing tasks at each location, potentially interacting with other agents that cross the paths. That makes the mobile agents not safe and needs a lot of work. The agent itself can be harmful to the host and the host can attack the agent. In this survey, I will focus on the concepts behind mobile agent security.

TABLE OF CONTENTS

AKNOWLEDGEMENTS i

INTRODUCTION ii

1 INTRODUCTION TO IP TELEPHONY 1

1.1 Overview 1

1.2 Cost Reductions 2

1.3 Increased Control of Telephone Services 3

1.4 Instant Activation 4

1.5 Real Time Accounting and Billing 5

1.6 Integrated Sales Information and Telephone Systems 6

1.7 Increased Market Presence 6

1.8 Call Routing Control (Intelligent Call Forwarding) 6

1.9 Remote Multimedia Communication 7

1.10 Conclusion 8

2 BASIC IP TELEPHONY SYSTEM OPERATION 9

2.1 Overview 9

2.2 Converting Voice to Data 9

2.3 Digitization 9

2.4 Digital Speech Compression – Gaining Efficiency 10

2.5 Sending Packets 11

2.6 Packet Routing Methods 11

2.7 Packet losses and Effect of Voice Quality 12

2.8 Converting Packets to Telephone Services 13

2.9 Gateways Connect the Internet to Standard Telephones 14

2.10 Managing the Connections 15

2.11 Gatekeepers Control the Call 15

2.10 Conclusion 16

3 IP TELEPHONY SYSTEM PROTOCOL **17**

3.1 Overview 17

3.2 Protocol 17

3.3 H.323 Packet Based Media Communication System 19

3.4 Session Initiated Protocol (SIP) 20

3.5 Media Gateway Control Protocol (MGCP) 21

3.6 Supporting Protocols and Softwares 22

3.7 Skinny Protocol 22

3.8 Telephone Routing over Internet Protocol (TRIP) 23

3.9 Common Open Policy Service (COPS) 23

3.10 Remote Access Dial in User Server (RADIUS) 23

3.11 Real Time Protocol (RTP) 23

3.12 Architecture for Voice, Video and Integrated Data (AVVID) 23

3.13 Vovida Open Communications Application Library (Vocal) 24

3.14 Multiple Protocols 24

3.15 Conclusion 24

4 BASIC IP TELEPHONY COMMUNICATION SERVICES **25**

4.1 Overview 25

4.2 Voice Service 25

4.3 Mobility Management (via Registration) 27

4.4 Call Hold 29

4.5 Call Forwarding 29

4.6 Click to Dial 31

4.7 Conclusion 32

5 TYING IP TELEPHONY TO OTHER INFORMATION SYSTEMS **33**

5.1 Overview 33

5.2 Order Processing Systems 33

5.3 Web Servers 35

5.4 Instant Mesagging (IM) 36

5.5 Web Seminars (Webinar) 38

5.6 Mobile Communication Information Service 39

5.7 Database System 40

5.8 Dispatch Systems 41

5.9 Security Systems 42

5.10 Interactive Television 43

5.11 Conclusion 44

6 IP TELEPHONY COMMUNICATION SERVERS 45

6.1 Overview 45

6.2 IP Telephony Administrators 45

6.3 Call Manager 46

6.4 Gateway Manager 48

6.5 Unit Manager 49

6.6 System Manager 53

6.7 Translation Server 54

6.8 User Manager 54

6.9 Conference Server 55

6.10 Conclusion 56

7 IP TELEPHONY SYSTEM DESIGN AND SETUP 57

7.1 Overview 57

7.2 IP Telephony System Design 57

7.3 Hosted IP Telephony Systems 59

7.4 Internet Telephone Service Provider (ITSP) 60

7.5 IP Centrex Operators 61

7.6 Dial Plan 62

7.7 Direct Inward Dialing (DID) Assignments 64

7.8 Hunt Groups 65

7.9 Automatic Call Distribution (ACD) 65

7.10 Voice Mail 66

7.11 Conclusion 67

8 VOICE QUALITY, SECURITY, RELIABILITY & FIREWALLS 68

8.1 Overview 68

8.2 Audio Quality 68

8.2.1 Toll Quality Audio 68

8.2.2 Echoes 69

8.2.3 Audio Distortion 71

8.3 Security 72

8.3.1 Physical Access 72

8.3.2 Authentication 73

8.3.3 Encryption 74

8.4 Reliability 76

8.4.1 Access Device Reliability 76

8.4.2 Data Network Reliability 77

8.4.3 Data Connection Reliability 78

8.4.4 Call Server Reliability 79

8.4.5 Feature Operation Reliability 80

8.5 Firewalls 80

8.6 Conclusion 81

CONCLUSION 82

REFERENCES 83

1. INTRODUCTION TO IP TELEPHONY

1.1 Overview

There are three key reasons why companies and people are adding to or converting their existing telephone systems to voice over data network capabilities:

1. Much Lower Costs for the Same Service.
2. Better Control of Communications Services.
3. New Revenue Producing Services.

If your company already has a data communications system or high-speed Internet connections, it does not cost you much more to make calls through data networks to reach standard telephones. The cost for equivalent digital voice service through a data network is usually much less than 1 cent per minute and the cost for connection of digital voice calls to the public telephone network can be 1 to 3 cents per minute to almost anywhere in the world.

Most voice over data network systems allow you to directly control your service activation and feature controls through a standard internal or external web page. This means that you don't need to call a customer service representative (CSR) from the telephone company to setup or change your services. You or your staff can directly control your own telephone services and features. In some cases, this control can be performed directly from an Internet web page.

Internet telephone service also can provide you with new revenue producing features and services. These features include the integration of marketing programs with telephone services, providing web pages that have audio links to customer service, and the use of multiple International telephone numbers that directly connect to your call centers at local calling rates.

1.2 Cost Reductions

According to the US department of commerce, corporations spend approximately 3% of gross sales on telecommunications costs. According to the federal communications commission (FCC), the average costs for telephone voice service in the United States in 2002 was:

1. $52.90 per month for business line connected to a PBX system.
2. 9 cents per minute domestic long distance.
3. 53 cents per minute for international calls.

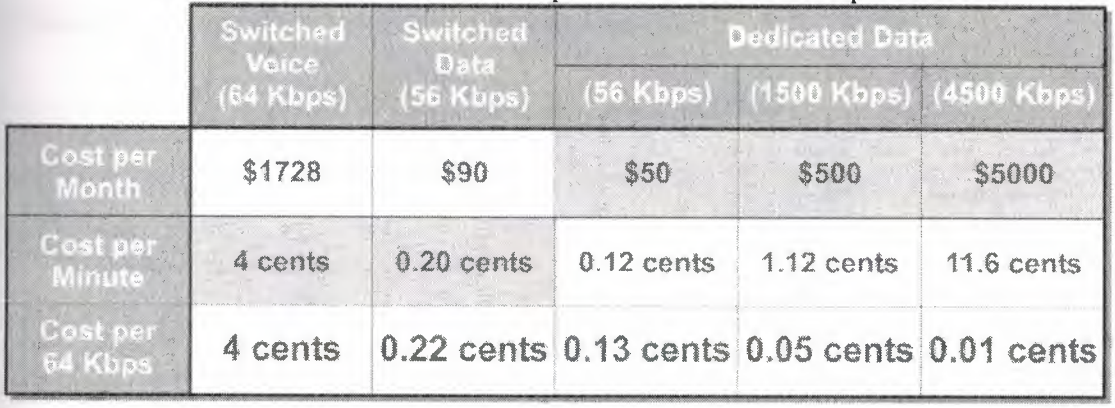
A telephone connection requires approximately 64 kbps of data transmission. Compared to the speed of company data networks, this is a relatively small amount of data transmission. The common data transfer rate for local area networks (LANs) is 100 Mbps (or more). This is almost 2000 times the speed of a typical telephone connection. Even wide area network (WAN) data connections (to connect offices to each other) used by companies typically range from 1 Mbps to 45 Mbps. The cost to send data as opposed to voice is approximately 10 to 20 times less.

Some data connections are temporary (called switched data) and other data connections are continuously connected (called dedicated). Switched data connections may charge by the minute or amount of data that is sent. Switched data connections allow for the rapid setup and disconnection of communication sessions. Dedicated connections usually charge a fixed monthly fee regardless of how much data is sent between two fixed points.

Table 1.1 shows some sample comparisons between traditional charges for voice communication compared to the charges for sending data. This table shows that the average cost per minute for traditional telephone service (called switched voice) is approximately 4 cents per minute. If this service were to remain connected for 24 hours per day and 30 days in a month, this results in a monthly fee of $1,728. A 56 kbps switched connection at 0.2 cents per minute results in a monthly charge of $90. The approximate cost for fixed connections is $50 per month for 56 kbps, $500 per month for 1.5 Mbps (DS1), and $50,000 per month for 45 Mbps connections (DS3).

If you adjust the monthly fee for a 64 kbps voice data rate (64 kbps/data rate divided by 30 days x 24 hours x 60 minutes), the average cost of data connection that is used for voice is 4 cents for switched voice, 0.22 cents for switched data, 0.13 cents for fixed 56 kbps, 0.05 cents for fixed 1.5 Mbps, and 0.016 cents per minute for 45 Mbps.

Table 1.1 Voice over Data and Telephone Service Cost Comparison



	Switched Voice (64 Kbps)	Switched Data (56 Kbps)	Dedicated Data		
			(56 Kbps)	(1500 Kbps)	(4500 Kbps)
Cost per Month	$1728	$90	$50	$500	$5000
Cost per Minute	4 cents	0.20 cents	0.12 cents	1.12 cents	11.6 cents
Cost per 64 Kbps	4 cents	0.22 cents	0.13 cents	0.05 cents	0.01 cents

Another key reason why it may cost so little to use voice over data network service is you may be able to use your existing data network (the data network and/or the Internet) without making many (if any) changes to it. Even if the person you want to call is not directly connected to your network, it is possible to use gateways to connect your voice over data call to the public telephone network. These gateways are located throughout the world at locations that are near the people you want to call. When you do call to the public telephone network, the additional cost of conversion from the data network to the public network is a small fraction of the cost (1 to 2 cents per minute) than if you dialed the call through the public telephone network.

1.3 Increased Control of Telephone Services

Voice over data network systems usually provide you with more direct control over your telephone services. Service is typically activated and changed directly through an internal web page. Instead of using a customer service representative (CSR) from the telephone company, you, or your staff, can setup the services directly. Your changes, such as service activation, can have immediate results.

1.4 Instant Activation

Instant activation is the process of obtaining service immediately after applying for service. If you already have access to a data connection, service activation for services that use the data link for connections (such as Internet telephone service) can be instant. Figure 1.1 shows how it is possible for a user or company system administrator to instantly activate a new voice over data telephone line.

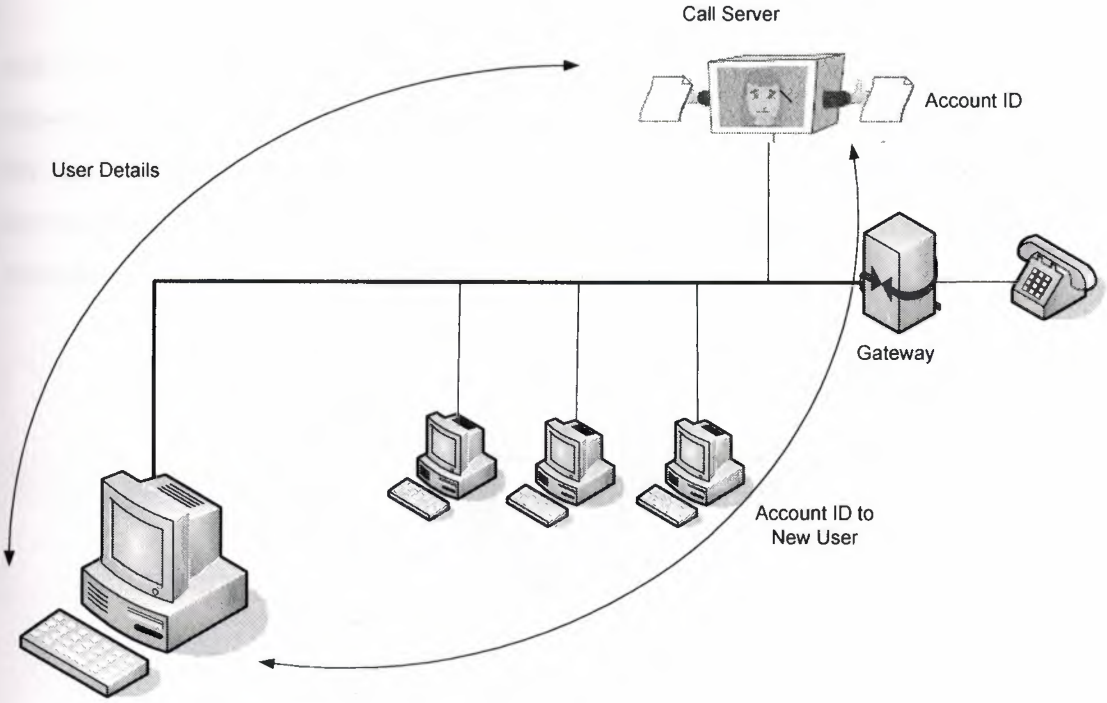


Figure 1.1 VoIP Instant Line Activation

In this example, the system administrator has provided a list of user identification codes and passwords to allow new users to self activate themselves in the Company's telephone network. After the user has entered the correct account identification codes, the user can setup their user details and their feature preferences (such as voice mail and call forwarding options).

1.5 Real Time Accounting and Billing

Real time accounting and billing is the process of gathering, rating, and displaying (posting) of account information either at the time of service request or within a short time afterwards (may be several minutes). Voice over data telephone service commonly allows for real time billing for tracking of voice over data telephone calls.

Figure 1.2 shows how voice over data service can provide real time accounting and billing records immediately after they are created (in real time). This example shows how the call server keeps track of each call as it processes each call setup. It uses the call setup and termination information to adjust the accounting and billing information. In this example, these charges or usage amounts can be displayed immediately through an Internet web page.

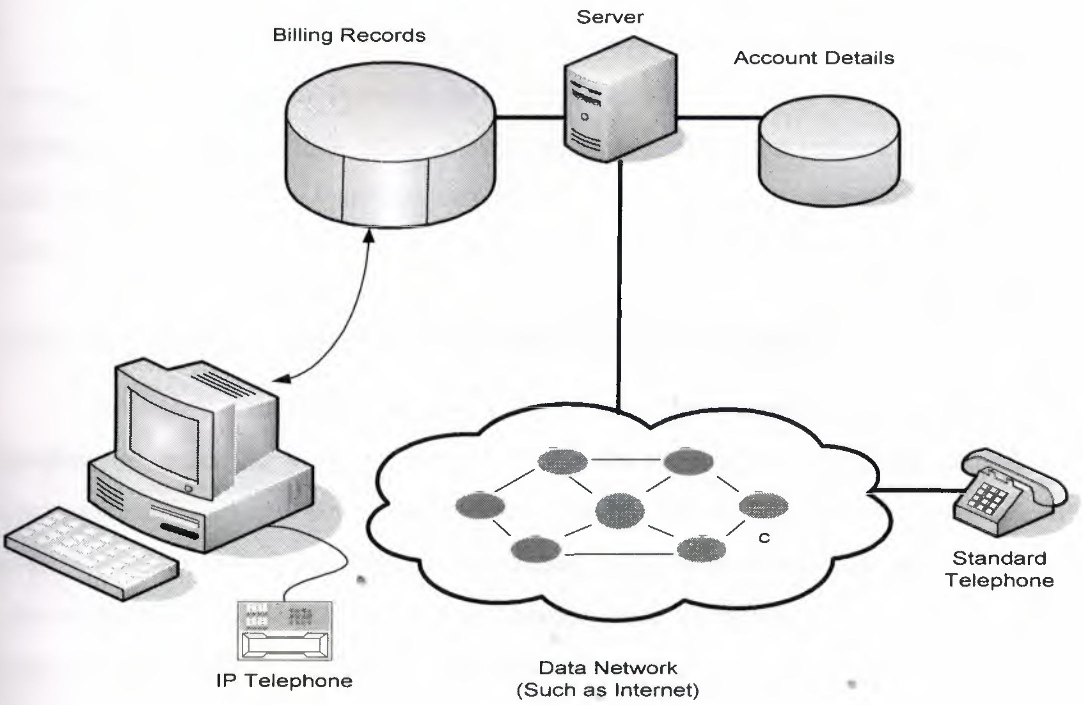


Figure 1.2 Voice over Data Real Time Accounting and Billing

1.6 Integrated Sales Information and Telephone Systems

It is possible to link voice over data network telephone systems with existing information systems. Using the telephone number or other identifying information, information can be gathered about callers and this can be provided to customer service representatives via a "screen pop."

VoIP telephone systems can share the same type of data network, the telephone system can be more easily integrated with the company's information system. In this example, a customer service representative (CSR) is receiving a call from John Doe. The screen pop shows that John Doe has already purchased a book. The CSR can use the account information from John Doe to help him find additional products to purchase.

1.7 Increased Market Presence

Companies can connect voice over data networks to telephone systems located throughout the world to increase their market presence. Using telephone numbers located throughout multiple cities allows callers to dial local telephone numbers and calls can be connected to your company through the data network or the Internet at very low cost.

1.8 Call Routing Control (Intelligent Call Forwarding)

Intelligent call forwarding changes the route of incoming calls to alternative destinations based on your preferred settings and the status of a telephone line or communication session when an incoming call is received. Some of the advanced control features include transferring calls based on the time of day, amount of time an unanswered line is allow to ring before transfer (such as transfer to voice mail), or to transfer the call to another number where you last made a call (call following).

Figure 1.3 shows an example of intelligent call forwarding that allows the destination of the call forwarding number to be changed based on time of day and location of the caller. In this example, these changes are made via web pages. This diagram shows that the user has setup intelligent call for- warding via a web page.

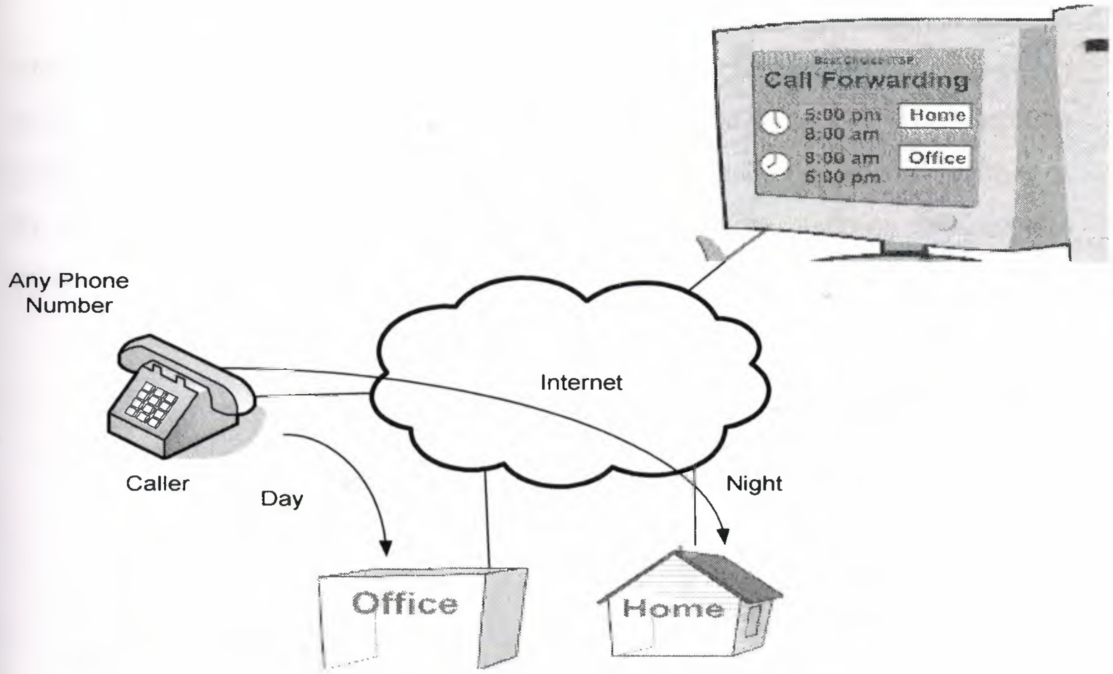


Figure 1.3 Intelligent Call Forwarding

1.9 Remote Multimedia Communication

Multimedia is a term that is used to describe the delivery of different types of information such as voice, data or video. Because Internet telephone service is often used with broadband (high-speed) data services, it is possible to send multiple types of information at the same time.

Figure 1.4 shows how a company can use remote multimedia to provide for corporate training or to conduct fully interactive inter-company meetings linking different people at different locations. This diagram shows that multiple forms of media can be sent during a voice over data network telephone call. This example shows a single broadband connection can simultaneously allow telephone calls (voice over data Telephone service), transfer data (such as a PowerPoint presentation), and allow the display of video (such as video images of the presenter).

In this type, a team leader in New York is presenting a new product to employees in Paris and London. Each participant can see the team leader on his or her monitor in a window box and hear the presenter on their voice over data telephone (using speakerphone). They can also see the course presentation on another window in the computer monitor along with hearing the professor by the audio on the computer speakers or telephone.

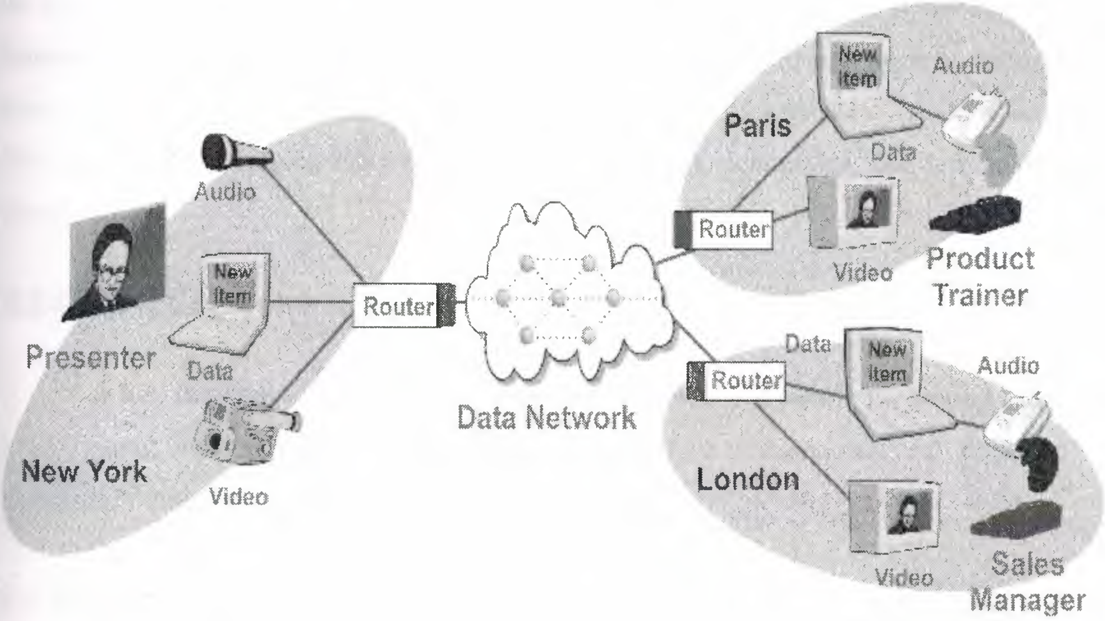


Figure 1.4 Remote Multimedia Communications

1.10 Conclusion

According to the nature of the IP network used, we may speak of two major categories for voice transmission over IP networks. In this chapter the importance of IP telephony has been discussed that has made a vital role in our lives. By using this technology the data rate has increased and the billing cost has decreased. At the end of the chapter call routing process has explained.

2. BASIC IP TELEPHONY SYSTEM OPERATION

2.1 Overview

Understanding the basics of how Voice over Data and IP Telephony service works will help you make better choices and may help you to solve problems that can be caused by selecting the wrong types of equipment and services. IP Telephony and Internet Telephone service operates by converting voice signals to data packets, sending these data packets through the Internet, converting these packets back into telephone like signals, andmanaging the overall cal setup (dialing), connection, and termination (hang-up).

2.2 Converting Voice to Data

A key first step in providing IP Telephony service is converting the analog audio voice signal into a digital form (digitize it) and then compressing the digitized information into a more efficient form.

2.3 Digitization

Digitization is performed because digital information can provide for better voice quality and digital signals are easier to work with than their analog counterparts digitization is the conversion of analog signals (continually varying signals) into digital form (signals that have only two levels). To convert analog signals to digital form, the analog signal is sampled and digitized by using an analog-to-digital (pronounced A to D) converter. The A/D converter periodically senses (samples) the level of the analog signal and creates a binary number or series of digital pulses that represent the level of the signal. Analog signals are converted into digital signals because they are more resistant to noise (distortion) and they are easier to manipulate than analog signals. For the older analog systems (continuously varying signals), it is not easy (and sometimes not possible) to separate the noise from the analog signals. Because digital signals can have two levels, the signal can be regenerated and during this regeneration process, the noise is removed.

Figure 2.1 shows the basic audio digitization process. This diagram shows that a person creates sound pressure waves when they talk. These sound pressure waves are converted to electrical signals by a microphone. When the microphone senses a large sound pressure wave (loud audio), it produces a large (higher voltage) analog signal. To convert the analog signal to digital form, the analog signal is periodically sampled and converted to a number of pulses. The larger the analog signal is, the larger the number of pulses that are produced. The number of pulses can be counted and sent as dig- ital numbers. This example also shows that when the digital information is transmitted, it may acquire distortion during transmission. A digital receiver that detects the high or low signal levels and uses these levels to recreate new digital signals can eliminate this distortion. This conversion process is called regeneration or repeating. This regeneration progress allows digital signals to be sent at great distances without losing the quality of the audio sound. Figure 2.1, Audio Digitization

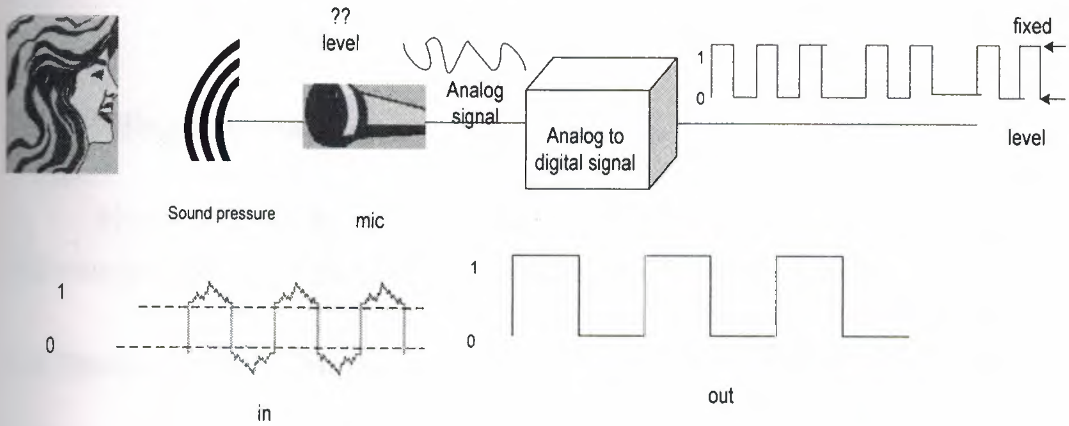


Figure 2.1 Audio Digitization

2.4 Digital Speech Compression – Gaining Efficiency

Digital speech compression is a process of analyzing a digital speech signal (digitized audio) and using the analysis information to convert the high- speed digital signals that represent the actual signal shape into lower-speed digital signals that represent the actual content (such as human voice). This process allows IP Telephony service to have lower data transmission rates than standard telephone service while providing for good quality audio.

Figure 2.2 shows the basic digital speech compression process. In this example, the word "HELLO" is digitized. The initial digitized bits represent every specific shape of the digitized word HELLO. This digital information is analyzed and it is determined that this entire word can be represented by three sounds: "HeH" +"LeL" + "OH."Each of the sounds only requires a few dig- ital bits instead of the many bits required to recreate the entire analog waveform.

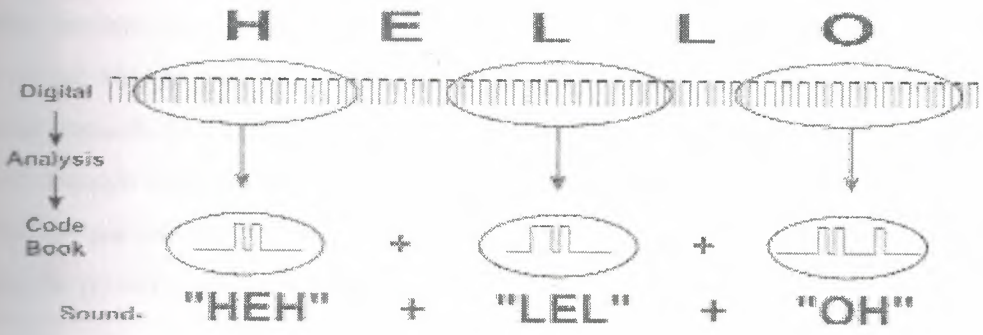


Figure 2.2 Digital Speech Compressions

2.5 Sending Packets

Sending packets through the Internet involves routing them through the network and managing the loss of packets when they can't reach their destination.

2.6 Packet Routing Methods

Packet routing involves the transmission of packets through intelligent switches (called routers) that analyze the destination address of the packet and determine a path that will help the packet travel towards its destination. Routers learn from each other about the best routes for them to select when forwarding packets towards their destination (usually paths to other routers). Routers regularly broadcast their connection information to near- by routers and they listen for connection information from connected routers. From this information, routers build information tables (called routing tables) that help them to determine the best path for them to for- ward each packet to. Routers may forward packets towards their destination simply based on their destination address or they may look at some descriptive information about the packet.

Figure 2.3 shows how blocks of data are divided into small packet sizes that can be sent through the Internet. After the data is divided into packets (envelopes shown in this example), a destination address along with some description about the contents is added to each packet (called in the packet header). As the packet enters into the Internet (routing boxes shown in this diagram), each router reviews the destination address in its routing table and determines which paths it can send the packet to so it will move further towards its destination. If a current path is busy or unavailable (such as shown for packet #3), the router can forward the packets to other routers that can forward the packet towards its destination. This example shows that because some packets will travel through different paths, packets may arrive out of sequence at their destination. When the packets arrive at their destination, they can be reassembled into proper order using the packet sequence number.

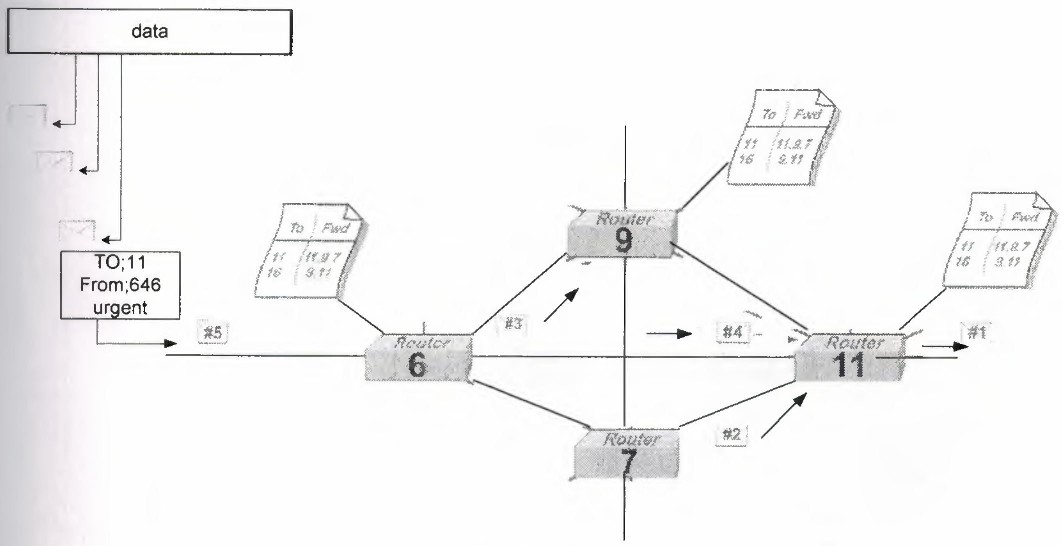


Figure 2.3 Packet Transmission

2.7 Packet Losses and Effects on Voice Quality

Packet losses are the in complete reception or intentional discarding of packets of data as they are sent through a network. Packets may be lost due to broken line connections, distortion from electrical noise (e.g. lightning spike), or through intentional discarding due to congested switch conditions. Packet losses are usually measured by counting the number of data packets that have been lost in transmission compared to the total number of packets that have been transmitted.

Figure 2.4 shows how some packets may be lost during transmission through a communications system. This example shows that several packets enter into the Internet. The packets are forwarded toward their destination as usual. Unfortunately, a lighting strike corrupts (distorts) packet 8 and it cannot be forwarded. Packet 6 is lost (discarded) when a router has exceeded its capacity to forward packets because too many were arriving at the same time.

This diagram shows that the packets are serialized to allow them to be placed in correct order at the receiving end. When the receiving end determines a packet is missing in the sequence, it can request that another packet be re transmitted. If the time delivery of packets is critical (such as for pocketsize voice), it is common that packet retransmission requests are not performed and the lost packets simply result in distortion

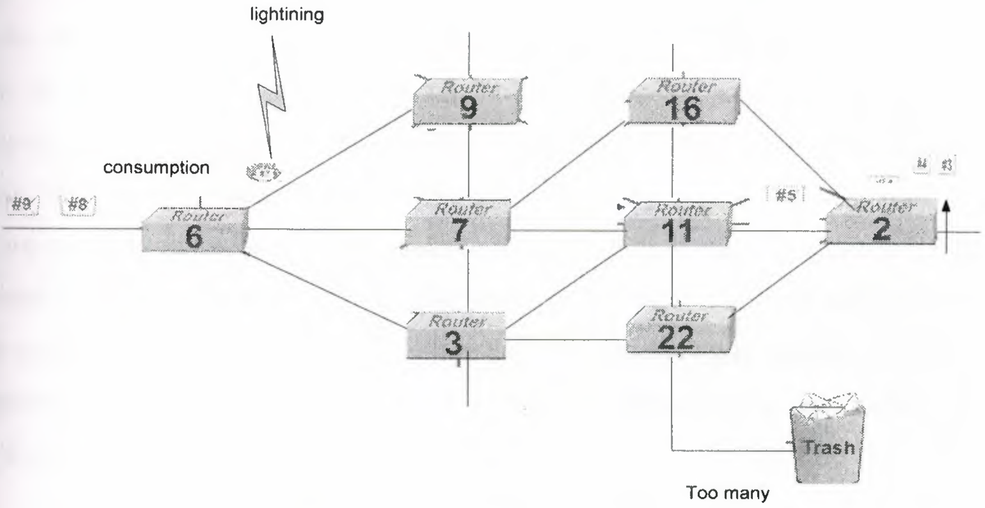


Figure 2.4 Packet Loses

2.8 Converting Packets to Telephone Service

IP telephone data packets are converted back to telephone signals via gate- ways. Gateways may interconnect IP telephone service to the public telephone network or they may simply convert to another format such as a private telephone system (e.g. PBX).

2.9 Gateways Connect the Internet to Standard Telephones

A voice gateway is a communications device or assembly that transforms audio that is received from a telephone device or telecommunications system (e.g. PBX) into a format that can be used by a different network. A voice gateway usually has more intelligence (processing function) than a bridge as it can select the voice compression coder and adjust the protocols and timing between two dissimilar computer systems or voice over data networks.

Figure 2.5 shows how a gateway connects a telephone device to the data network (such as the Internet).This example shows that the gateway must convert both audio and control signals. There are two audio paths through the gateway, one from the caller to the Internet and the other from the Internet to the caller. The gateway converts the audio from the telephone set micro- phone to packets of data that can be sent through the Internet on channel 1. Packets that are received from the Internet are converted to audio on channel 2. The gateway also monitors for control commands to be received from the telephone or the Internet. This example shows that the user has requested to make a three way cal by pushing the flash button on the telephone (or by momentarily pressing the hook-switch). The gateway senses this request and creates a control packet that is sent to the ITSP. When the ITSP receives this request, it sends a command message to the gateway indicating it should create a dial tone and gathers the dialed digits for the three-way call..

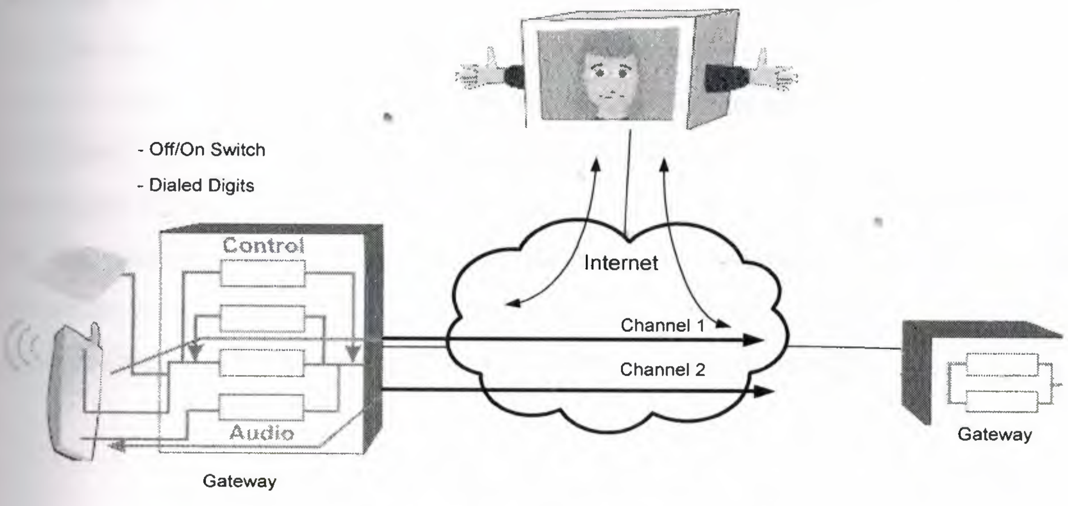


Figure 2.5 Audio Gateways

2.10 Managing the Connections

Gatekeepers control the setup, connection, feature operation, and disconnection of calls through the data network. Gatekeepers can be owned and operated by private companies, or public service providers such as IP Telephony service provider (ITSP).

2.11 Gatekeepers Control the Calls

Gatekeepers are computers that maintain lists of the IP addresses of customers and gateways, process requests for calls and features, and coordinate with the gateways that convert IP telephone calls to standard telephone calls. Gatekeepers perform access control, address translation, services coordination, control signaling coordination, and bill record recording.

Figure 2.6 shows how a gatekeeper sets up connections between IP telephones (IP Telephony's in this example) and telephone gateways. The gate- keeper receives registration messages from IP Telephony when it is first connected to the Internet. This registration message indicates the current Internet address (IP address) of the IP Telephony. When the IP Telephony desires to make a cal, it sends a message to the ITSP that includes the destination telephone number it wants to talk to. The ITSP reviews the destination telephone number with a list of authorized gateways. This list identifies to the ITSP one or more gateways that are located near the destination number and that can deliver the cal. The ITSP sends a setup message to the gateway that includes the destination telephone number, the parameters of the cal (bandwidth and type of speech compression), along with the current Internet address of the calling IP Telephony. The gatekeeper then sends the address of the destination gateway to the calling IP Telephony. The IP Telephony then can send packets directly to the gateway and the gateway initiates a local cal to the destination telephone. If the destination telephone answers, two audio paths between the gateway and the IP Telephony are created. One for each direction and the cal operates as a telephone call.

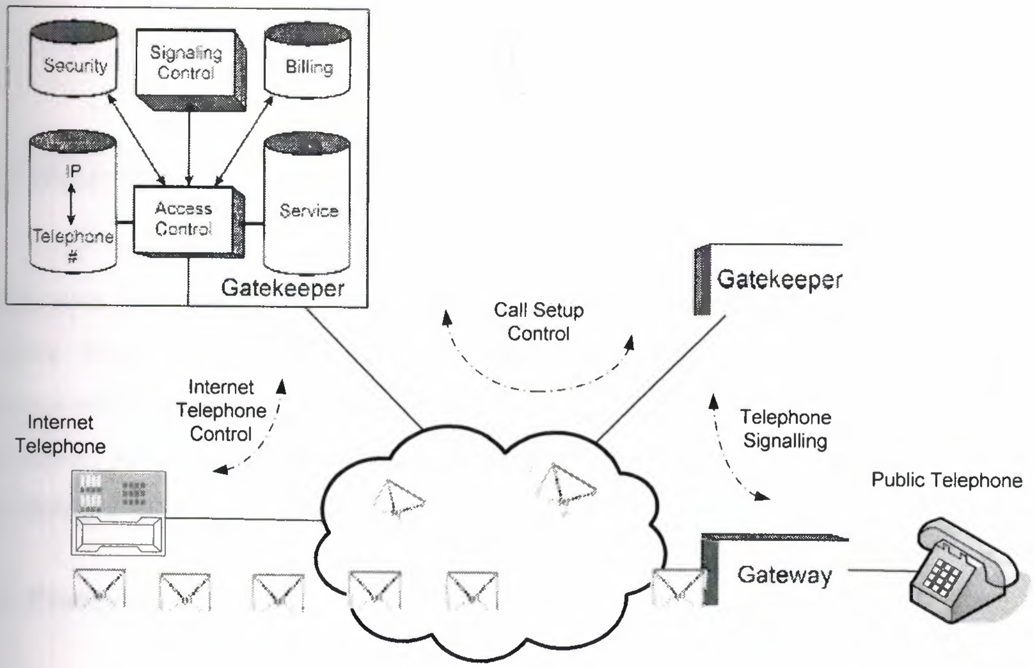


Figure 2.6 Gatekeepers

2.12 Conclusion

This chapter explains the basic function of IP telephony. The larger the analog signal is, the larger the number of pulses that are produced. So, we transmit the data on digital form to make its quality better. Routers learn from each other about the best routes for them to select when forwarding packets towards their destination (usually paths to other routers). Routers regularly broadcast their connection information to near-by routers and they listen for connection information from connected routers.

3. IP TELEPHONY SYTEM PROTOCOLS

3.1 Overview

IP Telephony communication systems use standard Internet protocols and application protocols that were specifically designed for coordinating the IP Telephony system. These protocols are used to control end user devices (called user agents), process cal requests (by the means of proxy servers), authorize user (customer) requests for service access (in databases called registrars), track addresses (in location registers), and forward calls (called redirection servers).

3.2 Protocols

Protocols are the languages, processes, and procedures that perform functions used to send control messages and coordinate the transfer of data. Protocols define the format, timing, sequence, and error checking used on a network or computing system. While several different protocol languages are used for IP telephone services, the underlying processes (setup and disconnection of calls) are fundamentally the same.

Systems can use sets of protocols. There are protocols for cal processes such as cal setup, audio compression, and cal conferencing. Protocols are commonly grouped together into families of protocols to ensure they work together (interoperate) without problems. Protocols are often enhanced and modified over time as new feature needs and problem areas are identified. As a result, protocols may have different revisions and earlier revisions may have more limited features and capabilities.

Figure 3.1 shows how protocols are used to communicate and control each part of an Internet telephone system and how different protocols can be used in different parts of the network. In this diagram, an Internet telephone is communicating with a public telephone. The Internet telephone creates sent to and from a computer (commonly called a gatekeeper, server, or controller) and this computer manages the setup and disconnection of calls and advanced services.

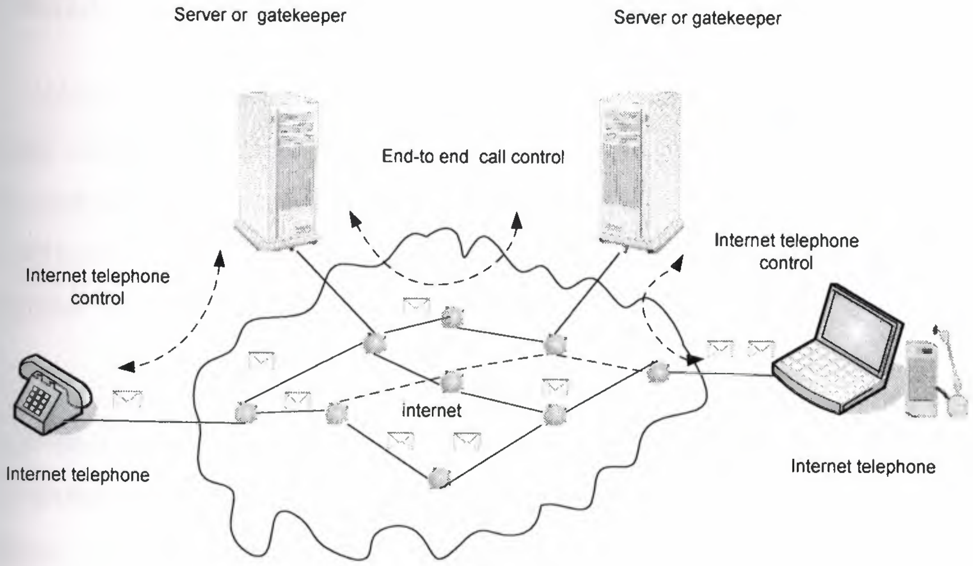


Figure 3.1 IP Telephony Protocols

The controlling computer communicates with other controlling computers in the network to allow calls to be connected. The equipment used for sending voice over data networks most likely con-forms to one or several industry standard protocols. Conforming to specific industry standard protocols helps to ensure reliable operation between devices that are connected through a data network or the Internet. Without standards, features such as caller identification, cal forwarding, or even cal disconnect (hang up) may not work or they may produce very different results than desired.

There are three key industry protocol standards for voice over data (VoIP) telephone service; H.323, SIP, and MGCP. IP Telephony systems and IP Telephony service providers may boast about their conformance and use of one (or several) of these industry standards. The most important thing these standards should mean to the user is the compatibility between the end user access device and the IP Telephony service provider it communicates with. In some systems, devices can translate protocols with systems that use the other protocols.

3.3 H.323 Packet Based Media Communication System

H.323 is a packet based multi-media communication system that combines multiple established protocols (such as telephone protocols) with new proto- cols to allow multimedia communications over data networks such as the Internet and local area networks (LANs). The original name for H.323 was Visual Telephone Systems and Equipment for Local Area Networks.

H.323 can be used to allow independent operation (caller to caller directly through the Internet) or an Internet telephone service provider (ITSP) can use it to setup and manage calls between its customers. The H.323 system has four key components: terminals, gateways, gatekeepers, and multipoint control units (MCUs). Terminals are the access devices such as Internet telephones or PC telephones. Gateways are the conversion devices used to connect the Internet to the public telephone network. Gatekeepers are the controller of the terminals (Internet telephones) and gateways. Multipoint control units (MCUs) may be used to coordinate the simultaneous communication between multiple terminals (conference calls).

H.323 is a well defined, detailed, and somewhat complicated industry specification. This helps to ensure reliable operation of basic and advanced communication services. This system is capable of negotiating compressing and transmitting real-time voice, video, and data between a pair of videoconferencing workstations.

Figure 3.2 shows the basic structure of an H.323 system. This diagram shows that a H.323 terminal can be controlled by an Internet telephone ser- vice provider (ITSP) orbit may be used to directly communicate to other users through the Internet. The terminal is actually a gateway that converts audio and control information into packets. The control packets are sent to and from the gate keeper to request and receive calls. Gatekeepers may communicate with other gatekeepers to setup distant cal connections. This diagram shows how a distant gatekeeper controls a gateway that a lows calls to connect from the Internet to a public telephone. Gatekeepers may also be connected to a multipoint control unit (MCU) to a low for conference calls.

The SIP system has two basic types of components: user agent (UA) clients and servers. Clients are the terminals (Internet telephones) and gateway devices. Servers are the gatekeepers that control the clients. There are several types of severs including proxy servers and redirection (cal control for- warding) servers. Figure 3.3 shows how a SIP system uses relatively simple text messages to setup and control telephone calls. This diagram shows a user agent (UA) Communicates with a cal server that controls a SIP IP telephone the user.

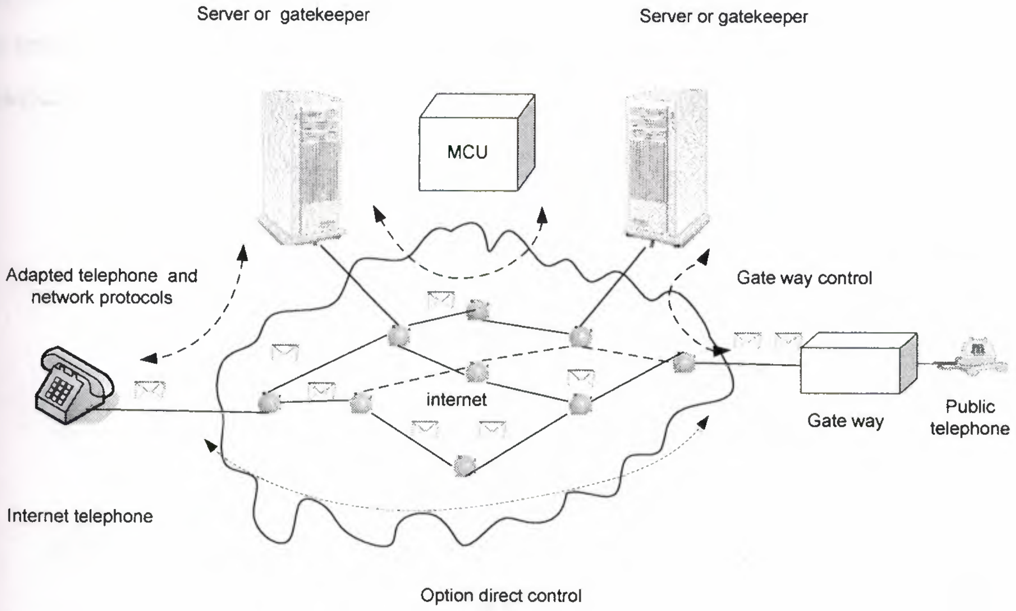


Figure 3.2 H.323 System Overview

3.4 Session Initiated Protocol (SIP)

Session initiated protocol (SIP) is a fairly simple text based Internet telephone communication protocol. SIP uses text-based messages that are similar to Hyper Text Transfer Protocol (HTTP) messages that are used by web applications.

SIP is relatively simple compared to the H.323 protocol because it has created new commands instead of attempting to adapt commands and processes from established telephone protocols. While SIP can allow for the independent operation of cals between users (caller to caller directly through the Internet), SIP is more commonly used by an IPBX system, IP Centrex service provider, or Internet telephone service provider (ITSP) to manage the setup, feature operation, and disconnection of calls.

MGCs are the gatekeepers, the controller of the terminals, and gateways (MGs). Soft switches control the MGCs so calls can be connected between MGs. MGs require connection to specific MGCs to operate. Figure 3.4 shows the basic structure of a MGCP system. This diagram shows a media gateway (MG) that is controlled by a media gateway controller (MGC). The MG converts audio and control information into packets. The control packets are sent to and from the MGC to request and receive calls. MGC communicate with soft switches that keep track of calls through its network. This diagram shows how a distant MGC controls a gateway that allows calls to connect from the Internet to a public telephone. MGCs are connected to a soft switch to allow for coordinated control of al MGCs within its network.

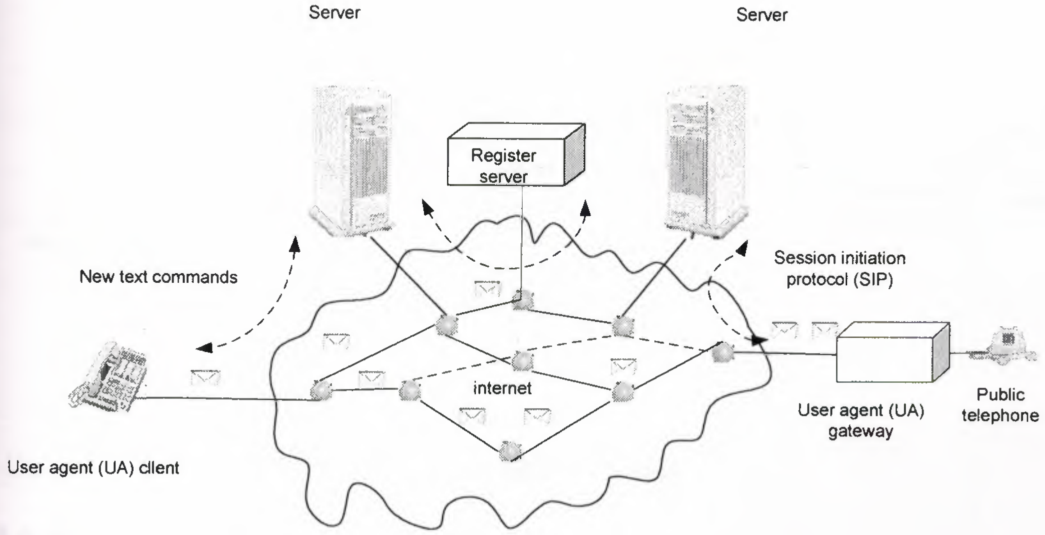


Figure 3.3 SIP System Overview

3.5 Media Gateway Control Protocol (MGCP)

Media gateway control protocol (MGCP) is a control protocol that uses text or binary format messages to setup, manage, and terminate multimedia communication sessions in a centralized communications system. This differs from other multimedia control protocol systems (such as H.323 or SIP) that allow the end points in the network to control the communication session. MGCP is specified in RFC 2705 and It was first drafted in 1998.

MGCP forms the basis of the Packet Cable NCS protocol. The MGCP system has three key components: media gateways (MGs), media gateway controllers (MGCs), and soft switches. MGs are the access devices such as Internet telephones, public telephone, and audio gateways.

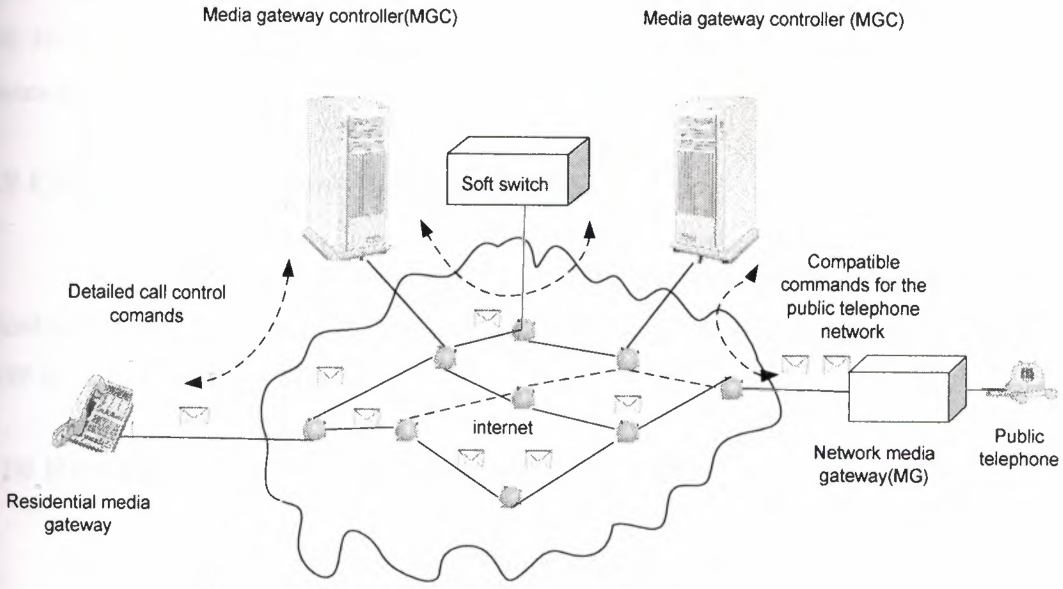

Media gateway controller(MGC)
Media gateway controller (MGC)
Soft switch
Detailed call control comands
Compatible commands for the public telephone network
internet
Residential media gateway
Network media gateway(MG)
Public telephone

Figure 3.4 MGCP System

3.6 Supporting Protocols and Software

In addition to the IP Telephony protocols, additional supporting protocols are used or were developed to efficiently help control IPBX and LAN telephone systems. Some of the important protocols and software solutions include Skinny, TRIP, COPS, RADIUS, RTP, AVVID, and Vocal.

3.7 Skinny Protocol

Skinny protocol was developed by Cisco to support the setup and management of audio calls and conferencing using Internet Protocols (IP). This protocol is a relatively simple IP-Phone protocol that can interoperate with H.323 systems. The simplified protocol provides for reduction in memory size and processing requirements.

3.8 Telephone Routing over Internet Protocol (TRIP)

The use of telephone routing over Internet protocol (TRIP) allows for the dynamic assignment of cal routes through a data network. TRIP accomplishes this by advertising (broadcasting) of the availability of destination devices (such as telephones) and for providing information relatives the available routes and preferences for these routes to reach the destination device(s).

3.9 Common Open Policy Service (COPS)

The COPS protocol allows a system to implement policy decisions by allowing a client to obtain system configuration and parameter information from a policy server. A COP is defined in RFC 2748.

3.10 Remote Access Dial in User Server (RADIUS)

A network device receives identification information from a potential user of a network service, authenticates the identity of the user, validates the authorization to use the requested service and creates event information for accounting purposes. RADIUS is specified in RFC's2138 and2139, RADIUS is a client/server protocol that uses UDP.

3.11 Real Time Protocol (RTP)

RTP is a packet based communication protocol that adds timing and sequence information to each packet to allow the reassembly of packets to reproduce real time audio and video information. RTP is defined in RFC 1889. Secure Real Time Protocol (SRTP) is a version of real time protocol (RTP) that provides increased security (e.g. confidentiality and message authentication).

3.12 Architecture for Voice, Video and Integrated Data (AVVID)

AVVID is a network structure standard that defines the types of devices used in a voice over data (multimedia) network and how they are interconnected and used within the network. The AVVID structure allows for system expansion, efficient feature deployment, security, and increased reliability. AVVID was developed by Cisco.

3.13 Vovida Open Communications Application Library (Vocal)

Vocal is a group of software applications that are used by developers to create telephone systems that use Internet protocols. Vocal uses open source software that provides the original source code to developers to make them make changes to the software to meet their specific application needs.

3.14 Multiple Protocols

Because there are several different protocols and each protocol can have different revisions of the protocol, many products and system equipments come with the ability to use several different protocol formats. Internet telephones may also be capable of receiving updated protocol information directly from the Internet. This allows for the upgrading of features and correction of software problems (bugs) after the unit has been purchased and connected. Figure 3.5shows how an IP telephone maybe capable of using multiple protocols. This diagram shows an IP telephone that is receiving messages in SIP protocol format. When the message is received, the Internet telephone first determines that the message is in SIP format and it decodes the message accordingly. When messages are received in H.323 protocol format, they are decoded according to that format.

3.15 Conclusion

Protocols are the set of rules of instruction which are follow to make the operation better. There are three key industry protocol standards for voice over data (VoIP) telephone service; H.323, SIP, and MGCP. The H.323 system has four key components: terminals, gateways, gatekeepers, and multipoint control units (MCUs). Terminals are the access devices such as Internet telephones or PC telephones. Gateways are the conversion devices used to connect the Internet to the public telephone network.

4. BASIC IP TELEPHONY COMMUNICATION SERVICES

4.1 Overview

IP Telephony communication services are the setup, management, and disconnection of communication sessions between two or more users of information. IP Telephony communication services permit the independent or combined transfer of voice, data, and video signals.

To provide IP Telephony communication services through a communication network, the session initiation protocol (IP Telephony) was developed. The IP Telephony protocol is intentionally quite simple in it's operation, yet capable of providing a range of services including basic voice telephony but also more advanced call features such as user mobility, supplemental call processing features such as call hold and call forwarding, and integrated services such as click to dial.

IP Telephony is intended to support a full range of multimedia sessions between users and therefore once a IP Telephony call is established, the same connection that is used for voice service can be used to transfer other information such as images, sounds, or a combination of any media that can be transferred through the communication network (such as the Internet).

4.2 Voice Service

Voice service is a type of communication service where two or more people can transfer information in the voice frequency band (not necessarily voice signals) through a communication network. IP Telephony based voice service involves the setup of communication sessions between two (or more) users that allows for the real time (or near real time) transfer of voice type signals between users. In an IP Telephony system Voice services is established by specific types of call processing steps.

The quality of voice services provided by IP Telephony systems can vary dependent on a variety of factors including the amount of data transmission channel quality, and compression method used. The data transmission channel quality can vary based on the transmission delay and the amount and type of errors. The compression methods supported by IP Telephony range from standard 64 kbps pulse coded modulation (PCM) voice to 8 kbps (highly compressed) G.729 speech coding.

The speech coding method is negotiated on call setup. The standard 64 kbps speech coder can provide for both voice and data modem (e.g. fax) transmission. The highly compressed G.729 speech coder cannot used to transfer fax signals or dual tone multi-frequency (DTMF) tones.

Figure 4.1 illustrates a simplified sequence for an IP Telephony call between two users. In this example Larry is going to place a call to Susan over the Internet, Larry has an IP Telephony telephone whilst Susan is using a soft phone, (a piece of software running on a multimedia PC). Larry and Susan belong to different domains and each domain contains an IP Telephony proxy server that manages that domain. The call would be initiated by Larry dialing a number for Susan, or alternatively by selecting an entry from an address book or even a link on web page, (called click to dial).

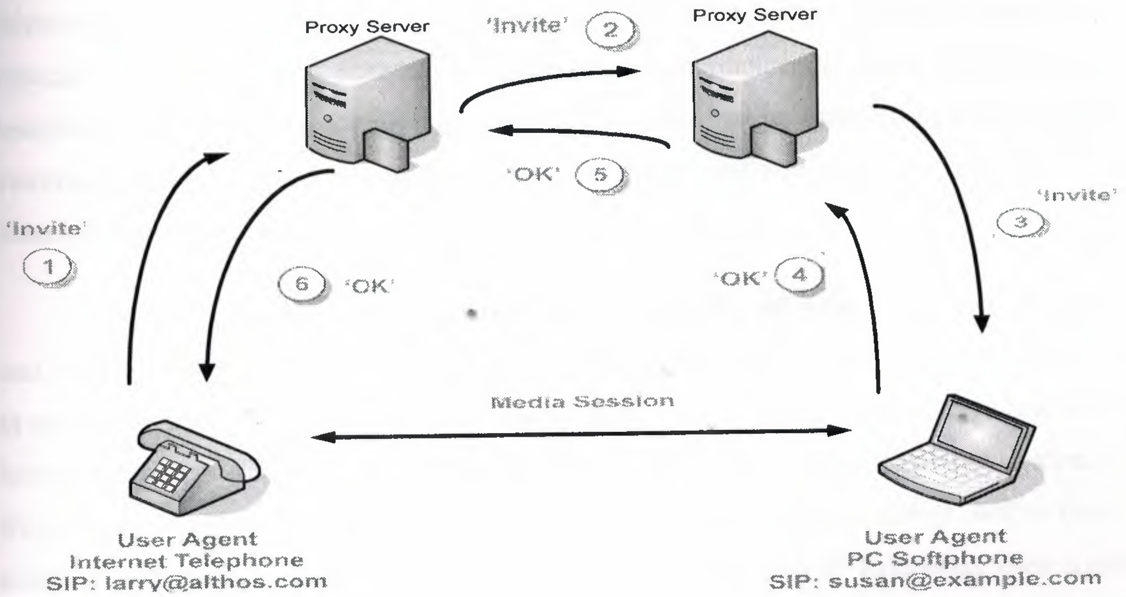


Figure 4.1. IP Telephony Telephone Call

When the call is initiated Larry's phone sends a message to the proxy server for his domain, this proxy server will send a message on to the proxy server in Susan's domain. The Althos.com server may use a form of Domain Name Server (DNS) lookup to obtain the address of the example.com proxy server. If necessary the example.com proxy consults a database known as a location server to identify the current address being used by Susan and forwards the message on to Susan's User Agent, which then generates a message in response that is sent back via the two proxies.

Larry's User Agent will respond with an acknowledgement, but note that this acknowledgement is not necessarily sent via the proxy servers. A two way media session is now established between the users. When the session is complete the 'connection' can be released by means of a simple handshake between the two telephones. It is important to note in the call example of Figure 4.1 that the IP Telephony protocol does not define the media format to be used during the call. Instead the IP Telephony messages will convey information from another protocol to define the media to be used during the communications session. In most cases this additional protocol is likely to be the Session Description Protocol (SDP).

4.3 Mobility Management (via Registration)

Mobility management is the processes of continually tracking the location of telephones or devices that are connected to a communication system. Mobility management typically involves regularly registering telephones or communication access devices. Mobile telephones typically automatically register when they are first powered on or attached to the communication systems. Some devices may also register and when they are powered off or detached from the system.

The IP Telephony protocol supports user mobility, by allowing a user to both initiate and receive sessions on different terminals within a domain, also a user is able to participate in session on terminals outside of their home domain (such as being attached anywhere to the Internet). Servers known as Registrars provide mobility in an IP Telephony system. A Registrar has an associated database, known as the Location Service, which is used to bind a user's IP Telephony address to a current location (IP address). An IP Telephony User Agent can be setup to register with the IP Telephony Registrar when it is first connected to a data network. This allows the Registrar to maintain the latest address (IP Address) where the User Agent is located.

Figure 4.2 shows an example how an IP Telephony system can allow users to attach their devices anywhere within the data network and maintain their ability to make and receive calls. In this example, a User Agent is registering with its Registrar. The User Agent at which Susan is currently located sends a registration message to the Registrar and the Registrar sends this data to be stored in the Location Service database. This creates what is known as a binding between Susan's IP Telephony address and the User Agent she is currently utilizing. When another user, in this example Larry, attempts to establish a session with Susan, the proxy server for Susan's domain will make a query to the Location Service that will return the binding information. This allows the invitation request to be routed from the proxy to the User Agent for Susan.

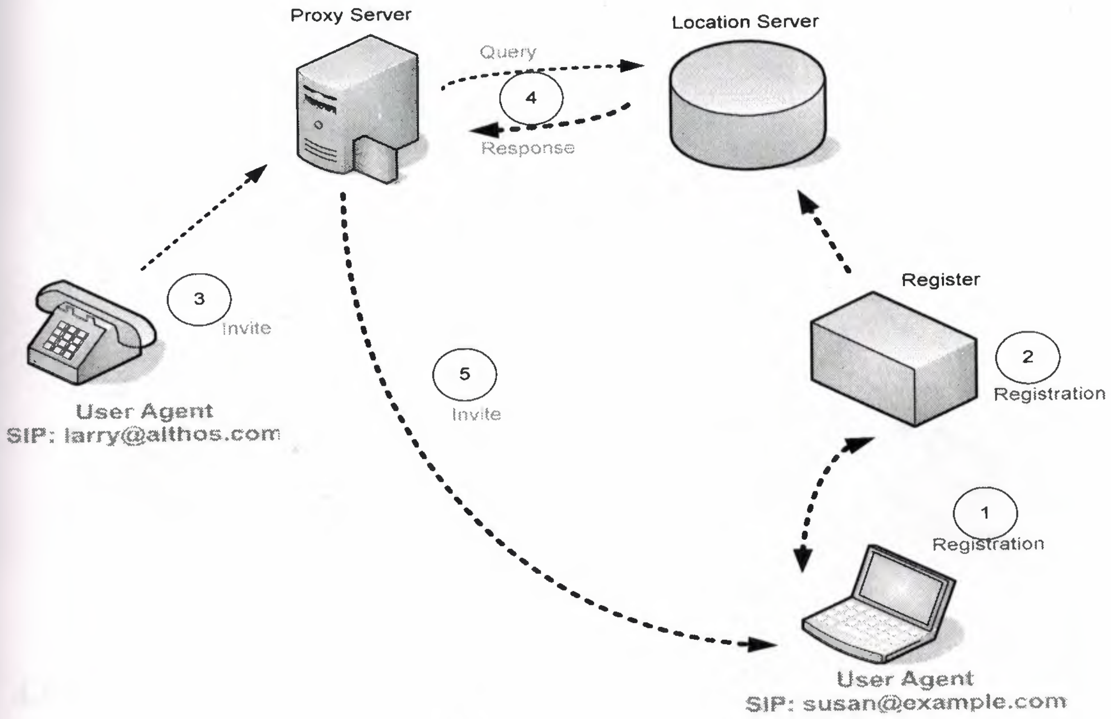


Figure 4.2 IP Telephony Mobility Management

In addition to supporting the basic facility of establishing a call between two, or more users, IP Telephony supports a range features that most users will familiar with from their existing telephony systems. These features include call hold, call forwarding, three-way calling and automatic redial.

4.4 Call Hold

Call hold is a feature that allows a user to temporarily hold an incoming call, typically to use other features such as transfer or to originate a 3rd party call. During the call hold period, the caller may hear silence or music depending on the network or telephone feature. Figure 4.3 shows how an IP Telephony call can be temporarily placed on hold so the call can stay connected without the user having to continue conversation with the caller. During call hold, the media streams in both directions are normally halted. However, IP Telephony can redirect a communication session to provide music on hold. In this case, the party that is placed in the hold condition will be sent a media stream that contains music. A different media server might provide the music.

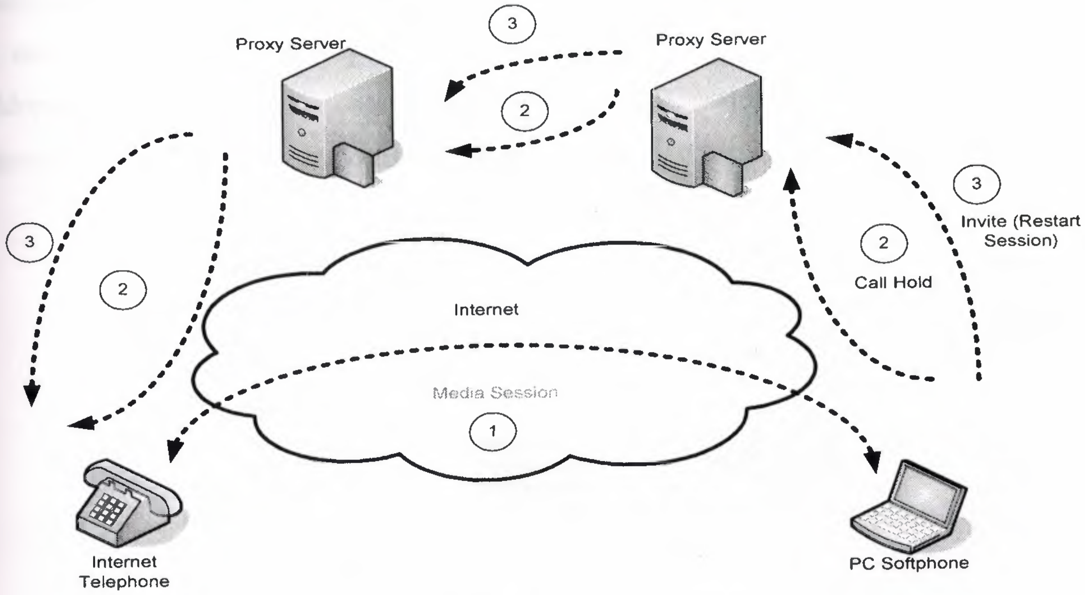


Figure 4.3 IP Telephony Call Hold

4.5 Call Forwarding

Call forwarding is a call processing feature that allows a user to have telephones calls automatically redirected to another telephone number or device (such as a voicemail system). There can be conditional or unconditional reasons for call forwarding. If the user selects that all calls are forwarded to another telephone device (such as a telephone number or voice mailbox), this is unconditional call forwarding. Conditional reasons for call forwarding include if the user is busy, does not answer or is not reachable (such as when a mobile phone is out of service area).

The support of call forwarding in any system requires that at least one network element is aware of the user's call handling preferences. For example, under what conditions, if any, should forwarding be triggered and when it is triggered where is the call to be forwarded to? In an IP Telephony system, the user's proxy server contains the call forwarding parameters.

Figure 4.4 illustrates how an IP Telephony system can provide conditional call-forwarding services. In this example, Susan has called Larry and the proxy server invites Larry's User Agent to join the session. Assuming Larry has set the condition 'call forward on no answer' and he is not available to answer this call, the User Agent would ring, or give some other form of alert, for a short (configurable) period of time. When this time expires, the proxy server for the call will forward (by sending an invite request) to a predefined number. In our example, Larry has requested that the call be forwarded to a second IP Telephony address. Call forwarding on busy is very similar to this example except that Larry's User Agent would return a busy indication to the proxy when it received the session invitation.

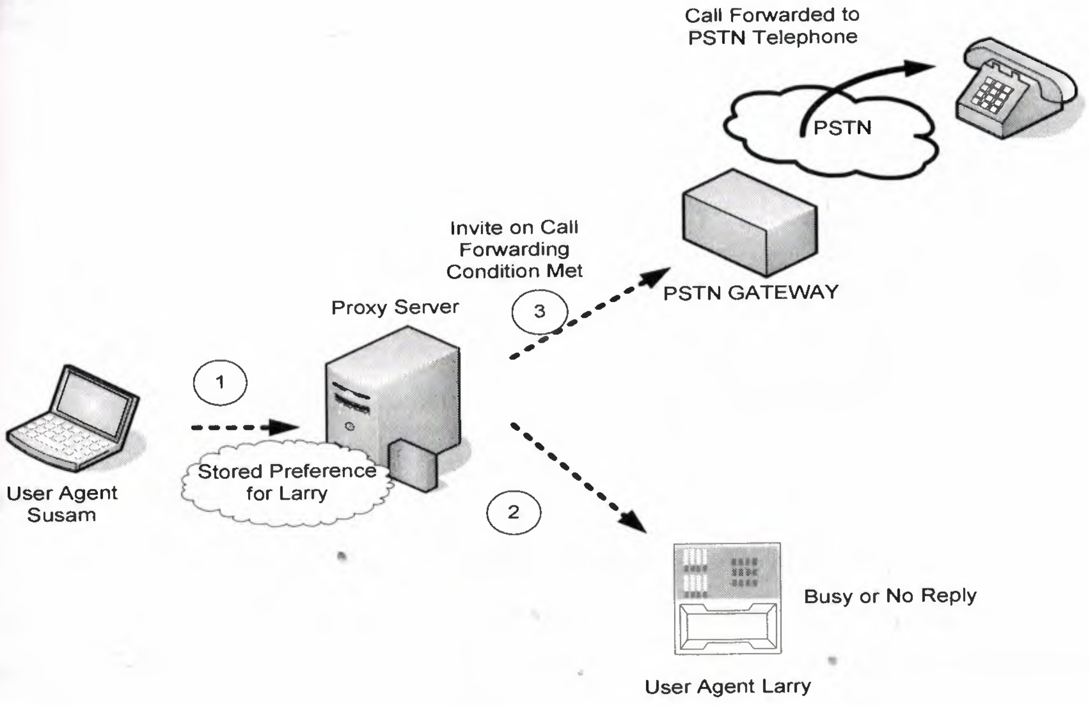


Figure 4.4 IP Telephony Conditional Call Forwarding

Figure 4.5 shows that to implement unconditional call forwarding, a proxy server simply forwards the invitation request to the diverted address previously specified. In this example, Susan has called Larry and the proxy server invites Larry's User Agent to join the session and Larry has set the unconditional call forwarding to a second IP Telephony address. Because the call forwarding has been setup as unconditional call forwarding, the proxy server immediately sends (forwards) the invite request to the designated recipient (forwarded number).

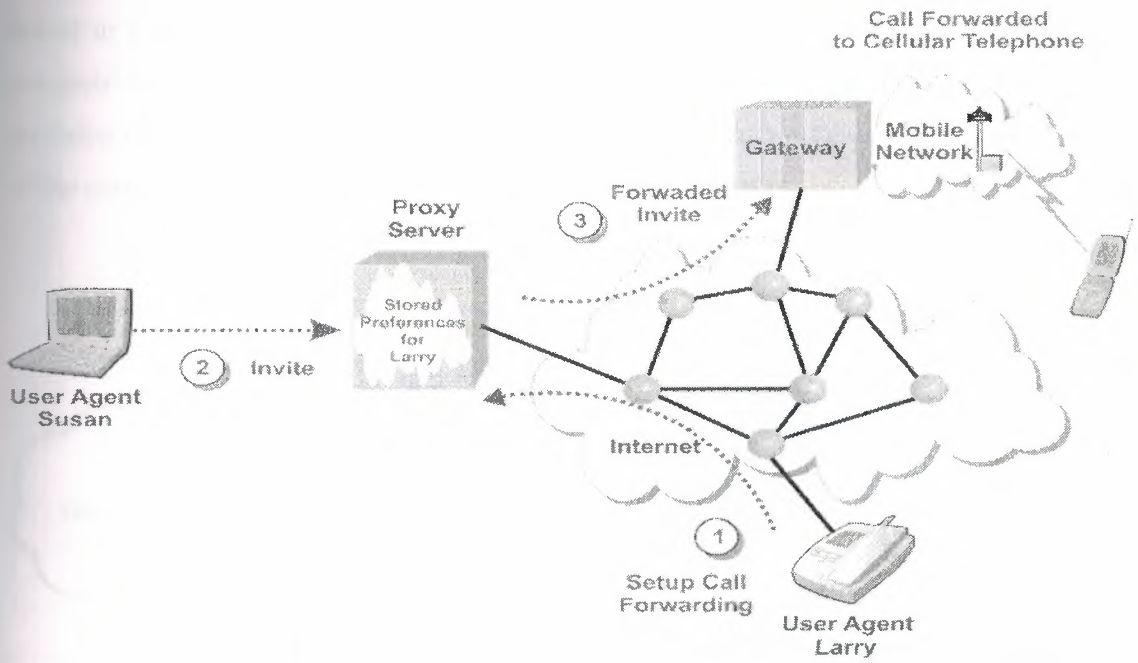


Figure 4.5 IP Telephony Unconditional Call Forwarding

4.6 Click to Dial

Click to Dial is an IP Telephony service that allows a user who is viewing a web page to click a link on that web page to initiate a voice over Internet call. The link contains an embedded address (URL or IP address) that connects to a call server along with the necessary software (such as IP Telephony) that allows for the setup and connection of the call. Click to dial service is similar in concept to the 'mailto:' link that can launch a user's email software when selected.

IP Telephony integrates well with web pages to provide click to dial service because IP Telephony is a text-based protocol that interacts with a web browser in similar processes as well established protocols such as Hypertext Transfer Protocol (HTTP) and Simple Mail Transfer Protocol (SMTP). Like these other protocols IP Telephony employs a request-response transaction between entities, such as a User Agent (UA) and Proxy Server.

Figure 4.6 illustrates how click to dial IP Telephony service operates. In this example, a user Susan is viewing a web page that contains a "Click to Dial" button. This button is linked to Larry's IP Telephony address. When Susan clicks on this click to dial the link, it connects her User Agent (also on Susan's PC) to the server at the IP Telephony address provided by the Click to Dial button. The User Agent can then establishes the call by means of the normal IP Telephony call processing (signaling) sequences.

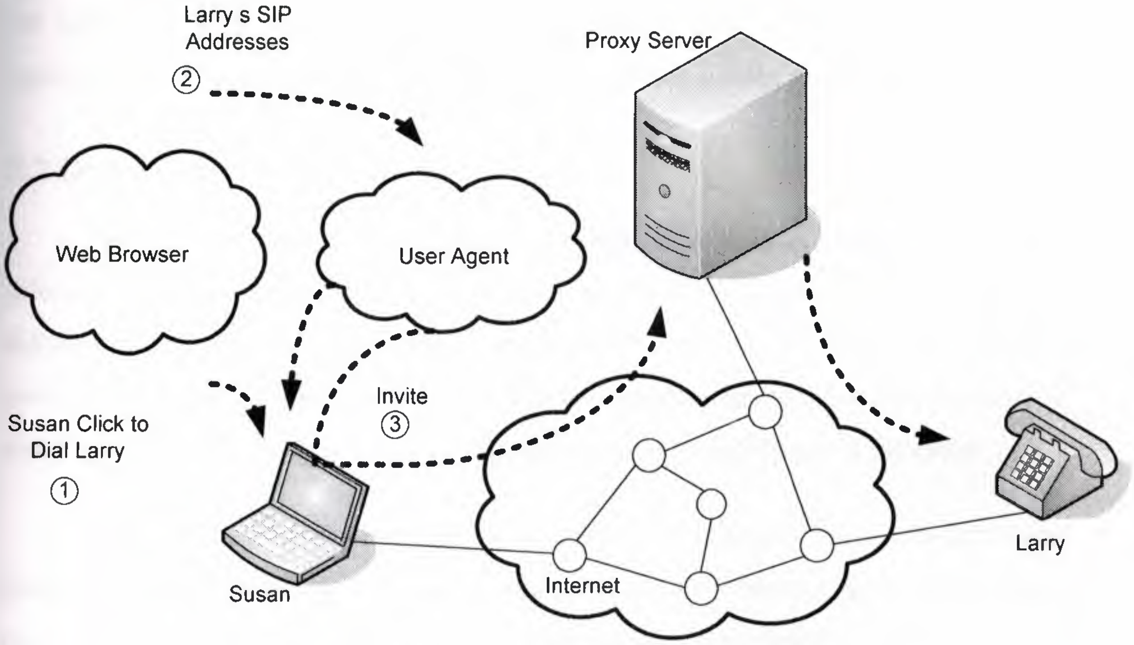


Figure 4.6 Click to Dial Service

4.7 Conclusion

In this chapter the IP telephony services are explained briefly. The quality of voice services provided by IP Telephony systems can vary dependent on a variety of factors including the amount of data transmission channel quality, and compression method used. Servers known as Registrars provide mobility in an IP Telephony system.

5. TYING IP TELEPHONY TO OTHER INFORMATION SYSTEMS

5.1 Overview

IP Telephony systems can be easily integrated with multiple types of information systems and other communication networks to produce advanced communication services. IP Telephony systems can be interconnected to other information systems through the use of specialized application servers (AS) and packet data networks. Application servers are computers and associated software that are connected to a communication network to provide information services (applications) for clients (users). Application servers are usually optimized to provide specific applications such as database information access or sales contact management.

5.2 Order Processing Systems

Order processing systems gather information related to orders, process the information into specific orders, and create actionable information that allows the fulfillment of the orders. IP Telephony telephone systems can be integrated with order processing systems to allow interactive control with customers to allow the capturing of order information directly from customers and to assist in fulfillment of the order.

Order processing systems within companies are typically limited to data entry from user terminals or by computers connected to the Internet. As a result, order processing systems may require a customer service representative to talk to the customer and enter the order information. This limits the order processing capabilities to the availability of a customer service representative and the potential errors that may result from poor communication skills of the customer service representative and the customer. The use of multimedia IP Telephony telephones allows the user to initiate and enter new orders into an order processing system without the need of a customer service representative.

To enable order processing systems to operate from telephone systems, additional servers or new software on existing servers are added that convert the information from the telephone user (such as keypad entries or audio commands) into commands that can be understood by the order processing system.

Figure 5.1 shows how IP Telephony based hotel telephone system can be integrated with a hotel's room service order processing system. This example shows how the hotel system has installed IP Telephony-based telephones in each room of the hotel and that each IP Telephony telephone has a display screen. The IP Telephony server and hotel information system (the hotel's order processing system) is connected through the same local area network (LAN) of the hotel. An IP Telephony server is setup to allow users to select and create orders from the room service menu from their IP Telephony based telephone. The IP Telephony server can reformat and deliver the menu order direct to the order processing system. In this example, the order is displayed in the kitchen and the IP Telephony system is used to alert the room service waiter when the order is ready for delivery to the room.

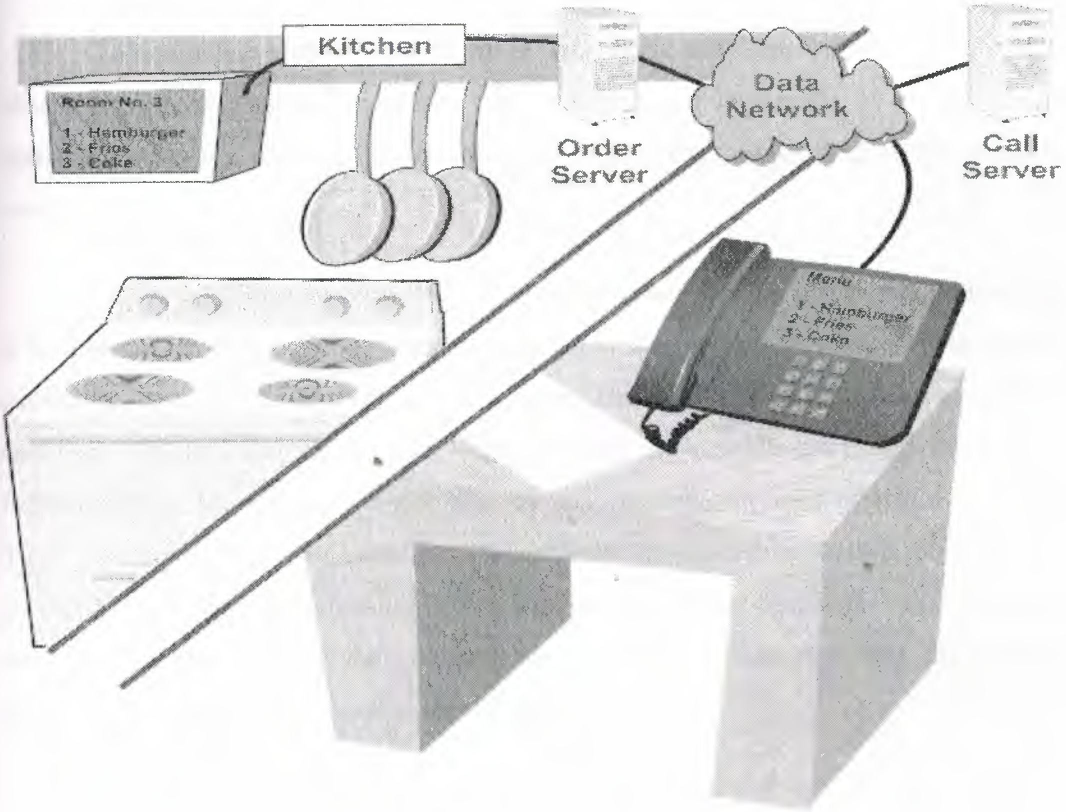


Figure 5.1 IP Telephony Order Processing Operation

5.3 Web Servers

Web servers are computer systems that are used provide access to data that is stored and retrieved by commands in Hypertext Transfer Protocol (HTTP). HTTP is a protocol that is used to request and coordinate the transfer of documents between a web server and a web client (user of information). The typical use of web servers is to allow web browsers (graphical interfaces for users) to request and process information through the Internet.

Web servers are limited to providing information to the user in a form and sequence that has been predetermined. As a result, users sometimes need to contact a customer service representative to provide information in an interactive form. Unfortunately, the customer service representative is traditionally limited to audio form. This means the customer has traditionally been limited to using the web page or the telephone to gather the information necessary to purchase a product. IP Telephony systems can be combined with a web server to allow the customer view web pages and communicate through the telephone at the same time.

To combine a web server with an IP Telephony telephone system, the customer should have a multimedia computer with software that is capable of IP Telephony communication and the web server must be modified to establish a communication session with the customer.

Figure 5.2 shows how an IP Telephony telephone system can be integrated with a web server. In this example, a user (potential customer) that has multimedia (audio) capability is accessing a company web page. The user has identified a product the company sells that may satisfy their needs however the user has not found some of the information on the company's web site. In this example, the user selects a "Click to Talk" button and they are connected to a customer service representative. This initiates (invites) a communication session between the user (potential customer) and a customer service representative for the company. may show the customer that the product performs the necessary features to satisfy their needs.

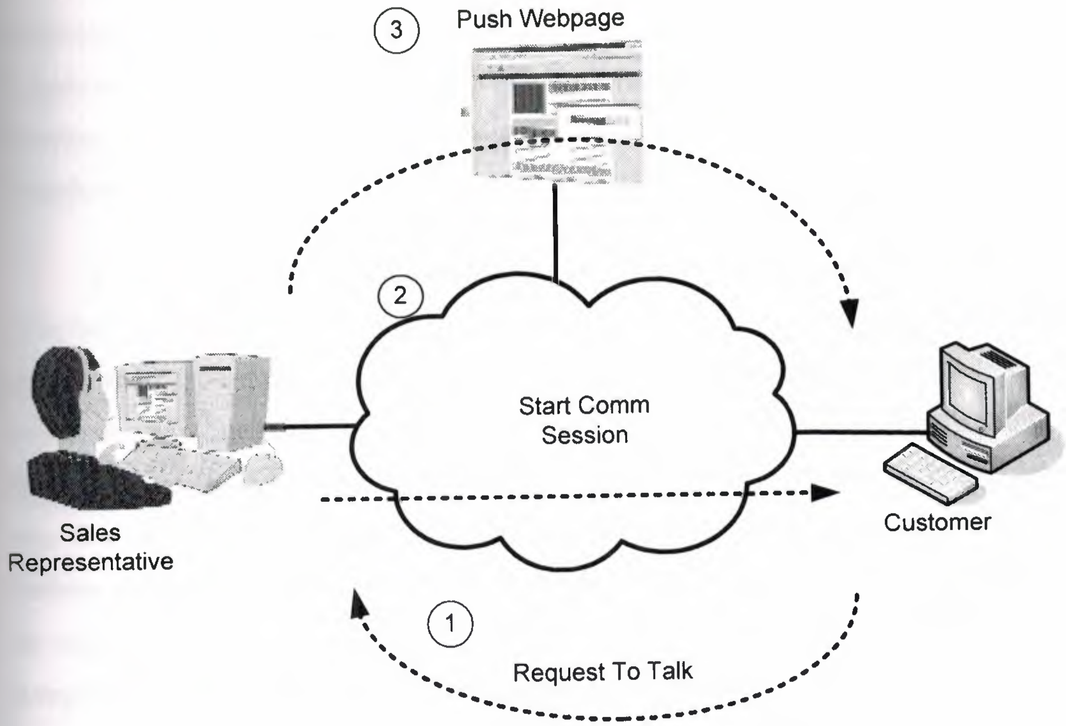


Figure 5.2 IP Telephony Web Server Integration

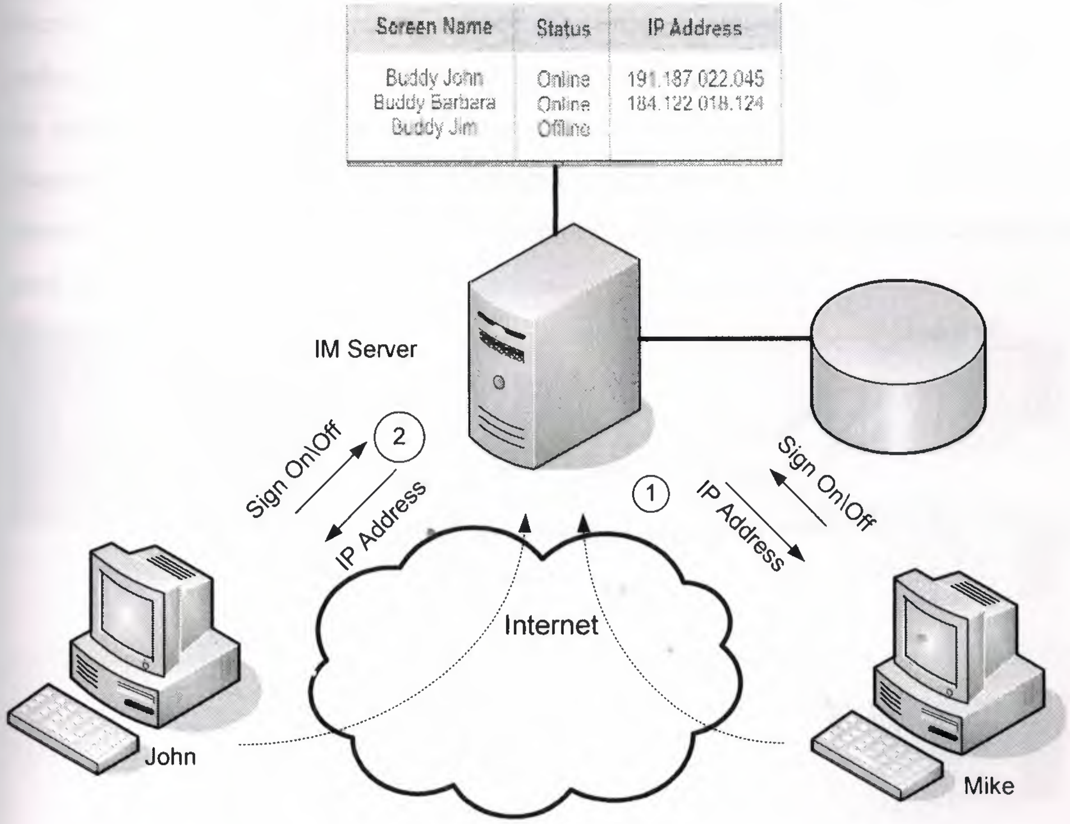
5.4 Instant Messaging (IM)

Instant messaging (IM) is a process that provides for direct messaging connections between computers that are connected to a data communications network. Instant messaging (IM) service usually includes client software that is located on the communicating computers and an instant messaging server that tracks and maintains a list of alias names and their communication status. The IM server usually registers each client and links an address (usually an internet protocol address) so the clients can directly communicate with each other. The client software controls the presentation of information as it is sent directly between each computer.

Instant messaging systems have been traditionally limited to text messages. Because instant messaging systems obtain and share the active IP addresses assigned to the users, it makes it possible to setup voice communication between two or more instant messaging users. Many instant messaging users are familiar with instant messaging software and they have multimedia capable computers so it is a relatively simple process to introduce them to the ability to initiate a voice session.

To upgrade an instant messaging system to use IP Telephony protocol to permit voice communication direct between users, the user's software is upgraded to include IP Telephony protocols. To obtain additional IP Telephony services (such as company directory listings), the user's software must be setup to communicate with the IP Telephony server.

Figure 5.3 shows how an instant messaging system can be integrated with IP Telephony based communication to provide voice service. This diagram shows two people that are instant messaging each other have IP Telephony based voice communication capability. This diagram shows that the IM system has already allowed the participants to discover the IP addresses of the other users. In this example, John uses Barbara's instant messaging address (IP address) to send an invite message to Barbara that requests her to participate in a voice conversation. Barbara acknowledges the request and the instant messaging software negotiates the voice parameters (speech compression in this example) and voice communication session is established.



Screen Name	Status	IP Address
Buddy John	Online	191.187.022.045
Buddy Barbara	Online	184.122.018.124
Buddy Jim	Offline	

Figure 5.3 Instant Messaging

5.5 Web Seminars (Webinar)

Webinars are a seminar or instruction session that uses the Web as a real time presentation format along with audio channels (via web or telephone) that allow participants to listen and possibly interact with the session. Webinars allow people to participate in information or training sessions from any location that has Internet access.

Until recently, the provision of interactive information (such as a training session) to multiple people in different locations required expensive video conferencing facilities or it required participants to travel to common location (such as a conference facility). The use of web seminars allows for the simultaneous provision of audio, video, and data along with the controls necessary for participants to interact with the information moderator.

Figure 5.4 shows how IP Telephony systems can be used to provide web seminar (webinar) service. This example shows how a presenter can invite several people to participate in a training session. Once the communication sessions (logical paths) are established, the instructor can create additional channels of communication for other multimedia services. This example shows that one of the communications channels is used for audio from the instructor to the students. Another communication channel is used for sending presentation graphics, and a final communication channel is used for sending datfiles.

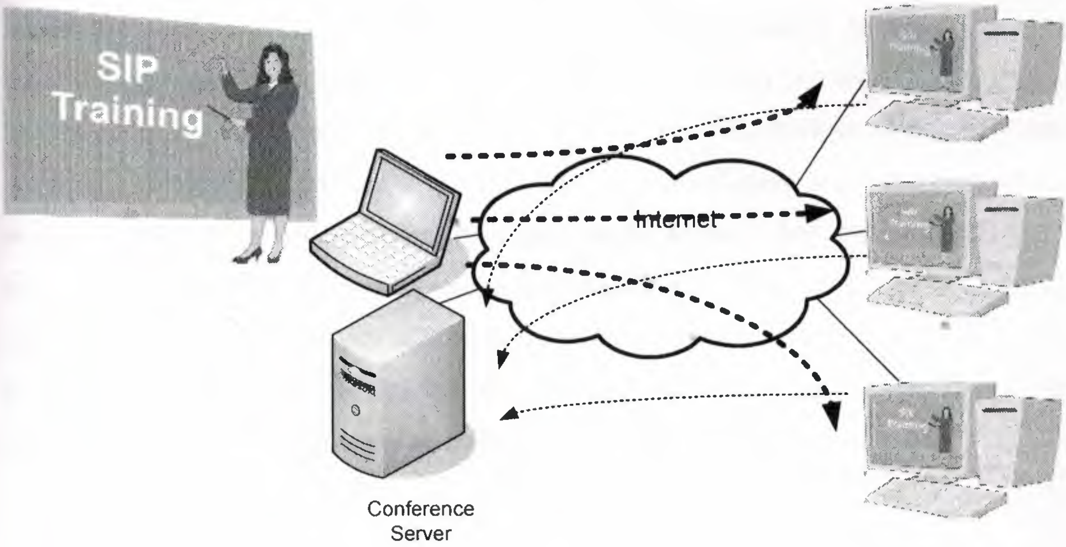


Figure 5.4 IP Telephony Web Seminar Operation

5.6 Mobile Communication Information Service

Mobile telephone service (MTS) is a type of service where mobile radio telephones connect users to the Public T Network (PSTN) or to other mobile telephones. Mobile telephone service includes cellular, PCS, specialized and enhanced mobile radio, air-to-ground, marine, and railroad telephone services.

Users desire to have access to multimedia services similar to their computers without the need to connect to wires. Until recently, mobile communication systems have had expensive and limited data transmission capability and the data connection methods have been proprietary. This has forced the system operator (the carrier) to purchase expensive system upgrades and it has not been easy to customize information services for the customers.

To add multimedia services to a mobile communication system, access to a media server is provided by the packet data system. This media server maybe controlled by the service provider (the carrier) or it maybe managed by an independent vendor (information service provider).

Figure 5.5 shows how a mobile phone network can use IP Telephony to add multimedia information services to wireless voice communications. This example shows how a mobile phone with a graphics display can communicate on the high-speed packet data communication channel to an IP Telephony server to obtain driving direction information. In this example, a mobile telephone user has requested a session with a company that provides driving direction information services. When this user requests a connection (sends an invite), the user is first validated as a subscriber of the map information provider. After the customer's account has been validated, a communication service (logical connection) is established between the mobile device and the media server. The user will browse through a menu that allows them to set the parameters (starting and destination address) and the media server can create the information and graphics that are transferred to the mobile device.

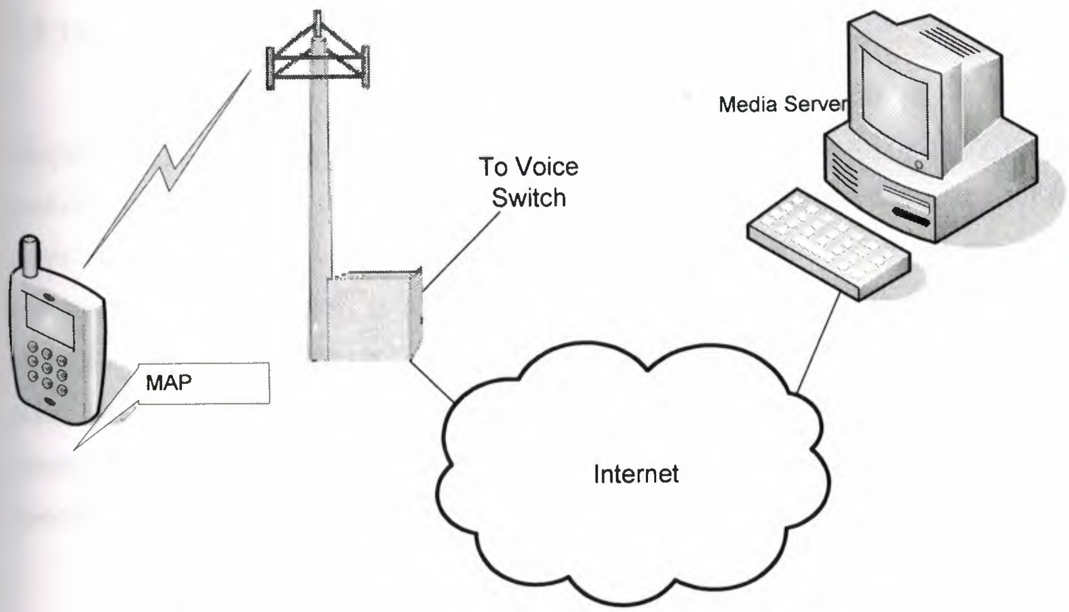


Figure 5.5 IP Telephony Mobile Communication Operations

5.7 Database System

Databases are collections of data that is interrelated and stored in memory (disk, computer, or other data storage medium.) Database systems are typically accessed and controlled by computer terminals that are connected to the same data network as the database system.

The information contained in corporate database systems is typically only accessible through computer terminals. However, database information is commonly stored in a standard form such as Structured Query Language (SQL) format. This allows other servers to access, sort, and retrieve information by using commands that use the standard data storage format. To integrate a database system (such as a customer database) with an IP Telephony based telephone system, various types of servers (such as a voice recognition server) can be added, these may allow for telephone users to control the access to the database by voice commands.

5.8 Dispatch Systems

Dispatch systems provide radio service that allows a central controller (dispatcher) to send dispatch assignments to one or more receivers (typically many mobile radios). Dispatch radio systems normally involve the coordination of a fleet of users via a dispatcher. All mobile units and the dispatcher can usually hear all the conversations between users in a dispatch group by setting the users to a channel (or channel code) that is shared by all the users in the group. Dispatch operation involves push-to-talk operation by a group of users on the dispatch system. Using radio trunking (multi channel access) technology, there can be several different dispatch groups that operate (share) on the same system.

Figure 5.6 shows how multiple dispatch radio networks can be integrated using an IP Telephony system. This example shows the different types of dispatch networks that are interconnected by high-speed packet data networks. The analog land mobile radio dispatch systems are connected through a gateway that converts the analog voice and control signals to digital messages. Digital land mobile radios that use a proprietary communication system are connected through a gateway that converts the digital audio from one format to a format that is compatible with the IP dispatch system.

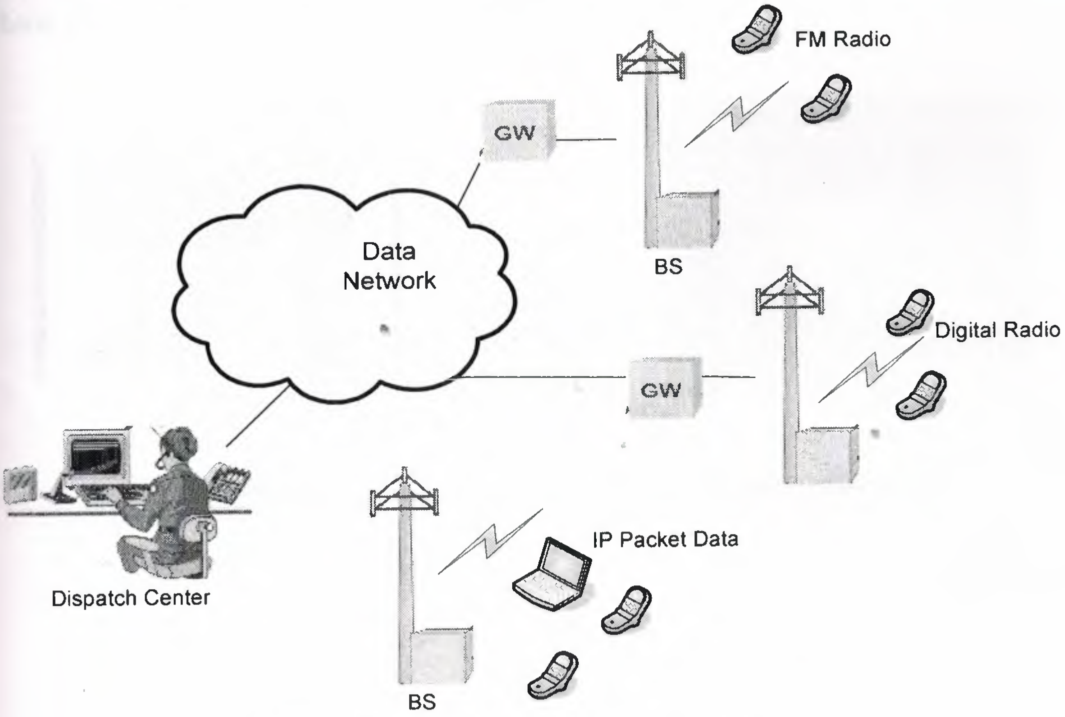


Figure 5.6 IP Telephony Dispatch Operations

5.9 Security Systems

Security systems are monitoring and alerting systems that are configured to provide surveillance and information recording for protection from burglary, fire, water hazard, and other types of losses.

Traditional (legacy) security systems use proprietary sensing and transmission equipment, have limited control processing capabilities, and have interconnections that are limited to local geographic areas. The use of IP Telephony systems connected through standard data networks allows for the sending of media (such as digital video), powerful security system processing in a server, and wide area connectivity (such as through the Internet).

Figure 5.7 shows how a variety of security accessories can be integrated into an IP Telephony based communication system. This example shows how a police station can monitor multiple locations (several banks) through the addition of digital video and alarm connections. This example shows that when a trigger alarm occurs at a bank (such as when a bank teller presses a silent alarm button), the police can immediately see what is occurring at the bank in real-time. Because the images are already in digital format, it may be possible to send these pictures to police cars in the local area to help identify the bank robbers.

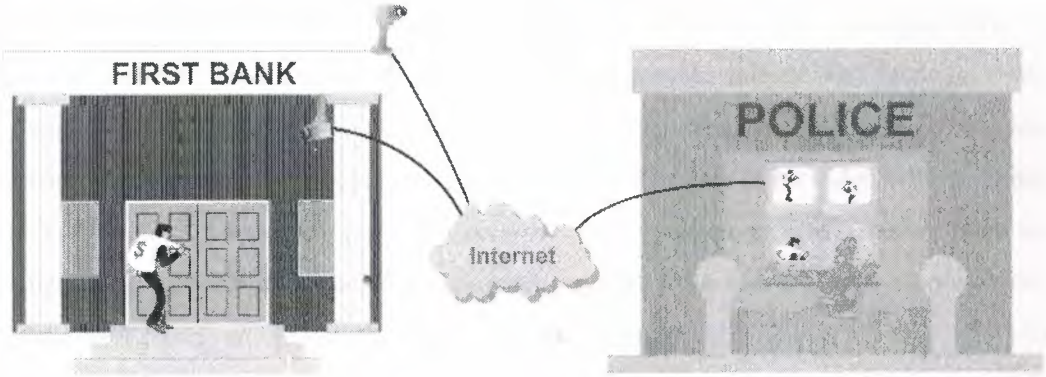


Figure 5.7 IP Telephony Security System Integration

5.10 Interactive Television

Interactive television (iTV) is the combination of video service with the ability to dynamically alter the content or flow of a media program that is provided to a user. Until recently, television and cable television have been delivered in a one-way broadcast process. One-way analog cable television video systems have been transitioning to two-way high-speed digital communication systems.

Cable television systems are composed of a head-end system (the network television receivers), the distribution network, and end user equipment (set top boxes). Until recently, the set top boxes have been proprietary and the control of the system has been limited to sequential programming and nonchangeable media. The use of IP Telephony based systems and a digital media server allows the user to change the programming dynamically based on user inputs and/or predetermined preferences. This can provide preferences such as the elimination or skipping of sections of bad language, nudity, violence or even the selection of different outcomes to a movie (positive or negative). To add IP Telephony capability to digital cable television systems, the users need to have terminals with IP Telephony capability (Multimedia MPEG to NTSC or PAL) and a media server at the head end of the cable television network.

Figure 5.8 shows how a cable television network can use IP Telephony to add interactive media services to video communication networks. This example shows a video on demand (VOD) system that uses IP Telephony to initiate and manage video delivery from a digital movie storage system. In this example, the cable television network has two-way digital communication service. The user is provided with a digital set-top box and the cable television network has installed a media server and an application server that can receive and process requests from users. This example shows that the user has a remote control that can send commands (IP Telephony messages) to the media server at the head-end of the cable television network. These commands allow the end user to control the media source (such as play, stop, or rewind).

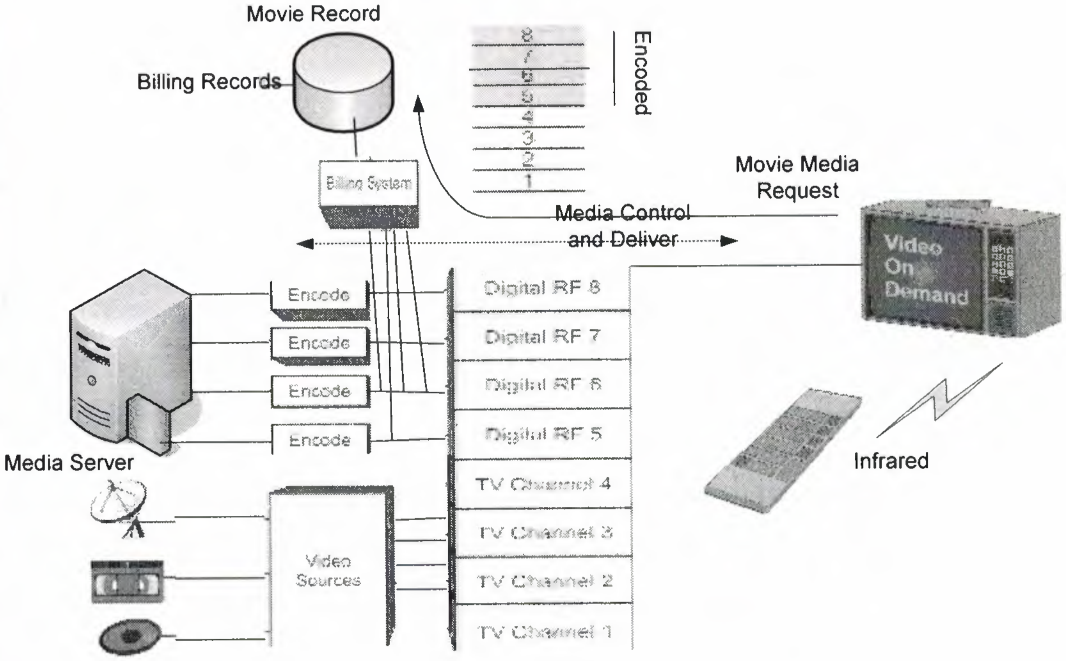


Figure 5.8 IP Telephony Cable Television Operations

5.11 Conclusion

There are several advantages of using IP telephony technology; one of them is that we can establish a communication link for other system as well. This technology is compatible with the internet based application like HTTP (Hypertext Transfer Protocol) and Instant messaging (IM) which is a process that provides for direct messaging connections between computers that are connected to a data communications network. A variety of security accessories can be integrated into IP Telephony based communication system.

6. IP TELEPHONY COMMUNICATION SERVERS

6.1 Overview

A communication server is a computer that can receive, process, and respond to an end user's (client's) request for communication services. Communication servers are application servers that provide common telephony features and/or specialized telephony capabilities.

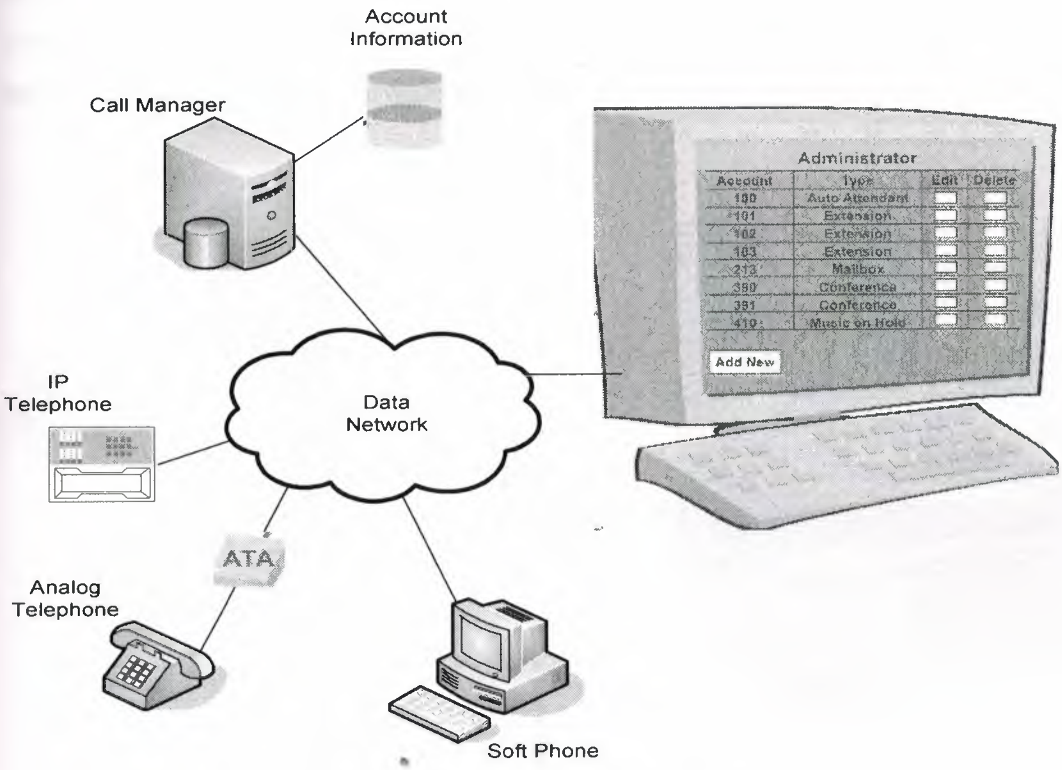
IP Telephony communication servers are at the core of the VoIP communication system. IP Telephony servers within a network can either be located on different machines or can be located on a single machine. When the communication servers are co-located on the same computer, they use different port addresses (separate logical channels). Servers are computers with a server operating system. Popular server software systems include Windows NT, Windows 2000 Server, or Linux. While it is possible to setup an IP Telephony-based communication system on a computer that is used for other functions, it is common to have a dedicated computer for the IP Telephony communication server. IP Telephony communication servers receive request for calls, assist in the setup of calls, perform call feature operations during the call, and process requests in the termination (ending) of calls. IP Telephony communication servers may translate standard telephone numbers into their associated IP addresses, look for available gateways to complete calls between the PSTN and IP Telephony devices, and manage advanced communication features. Manufacturers and vendors commonly use server names that may be different than the names assigned by the industry standards committees.

6.2 IP Telephony Administrators

IP Telephony administrator software is a graphic user interface (GUI) for the system administrator to setup and manage accounts on the IP Telephony system. The administrator has the ability to move, add, and change (MAC) accounts. The setup of the system administrator is one of the first steps in building an IP Telephony system.

IP Telephony communication systems are interconnected by IP data networks that are typically connected through the Internet allowing administrators to setup and manage their system from any place they can connect to the Internet.

Figure 6.1 shows the basic functions of the IP Telephony system administrator. This diagram shows that the IP Telephony system administrator is responsible for adding new accounts to the IP Telephony system, assigning the dialing rights (such as international and long-distance dialing), and the features authorized (such as conference calling). This diagram also shows that the IP Telephony system administrator can perform these assignments and changes from a web portal.



Administrator

Account	Type	Edit	Delete
100	Auto Attendant	□	□
101	Extension	□	□
102	Extension	□	□
103	Extension	□	□
213	Mailbox	□	□
390	Conference	□	□
391	Conference	□	□
410	Music on Hold	□	□

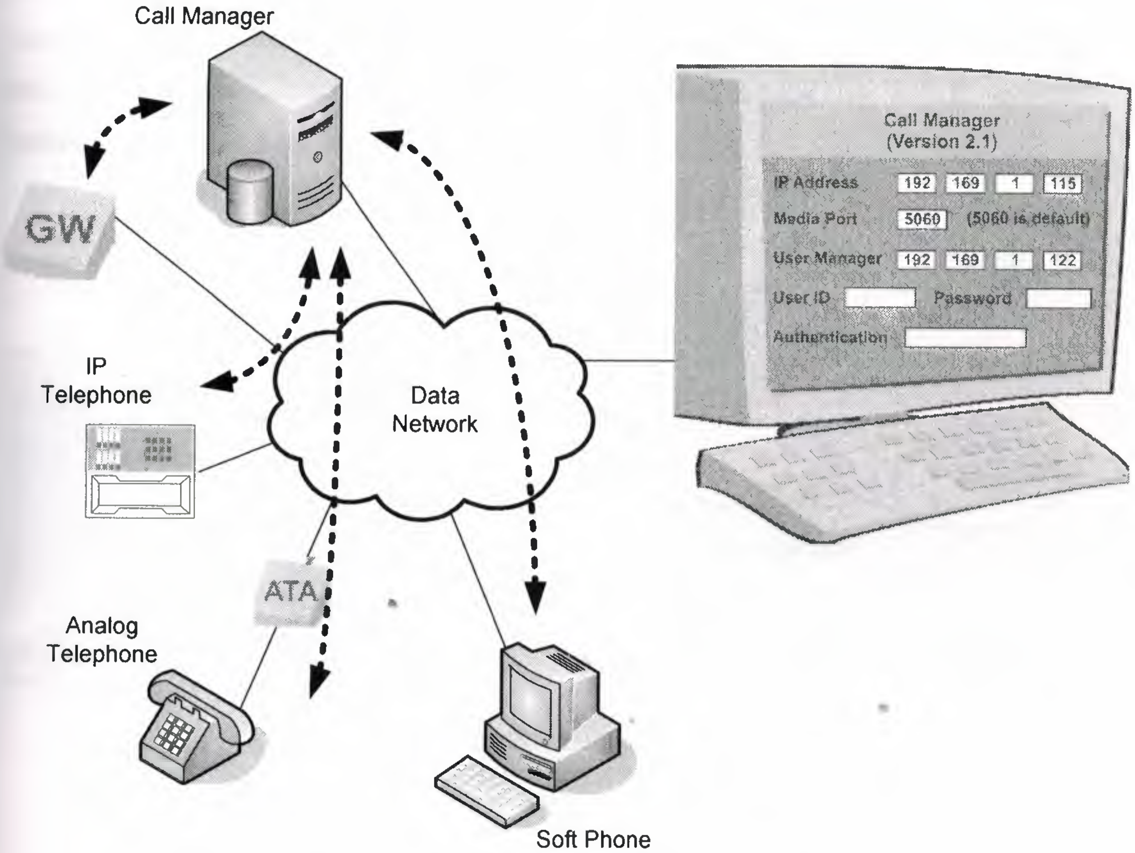
Figure 6.1 IP Telephony Administrator Software

6.3 Call Manager

A call manger (IP Telephony proxy server) is the call control processing software that can receive call requests (call invites) from users and assist in (proxy on behalf of) the setup of connections between communication devices. Call managers only help setup the call connections, they do not actually transfer the call data.

The call manager server may translate IP Telephony commands to other protocols such as Media Gateway Control Protocol (MGCP) or H.323. This allows for an IP Telephony system to use devices (such as Voice Gateways) Figure 6.2, IP Telephony Administrator Software that do not use IP Telephony protocol. In some cases, a separate translation server may be used to convert protocols to and from devices that use other protocols.

Figure 6.2 shows common setup of the call manager (call proxy server) software for an IP Telephony communication server. This diagram shows that the IP Telephony server port (logical) channel is commonly set to 5060 (the default). This diagram also shows that the IP Telephony call manager can be setup to use an alternative server that coordinates the end-user devices. This identifies the IP address of the user manager server. The example shows that authentication security can be setup on another server.



Call Manager
(Version 2.1)

IP Address 192 169 1 115
Media Port 5060 (5060 is default)
User Manager 192 169 1 122
User ID Password
Authentication

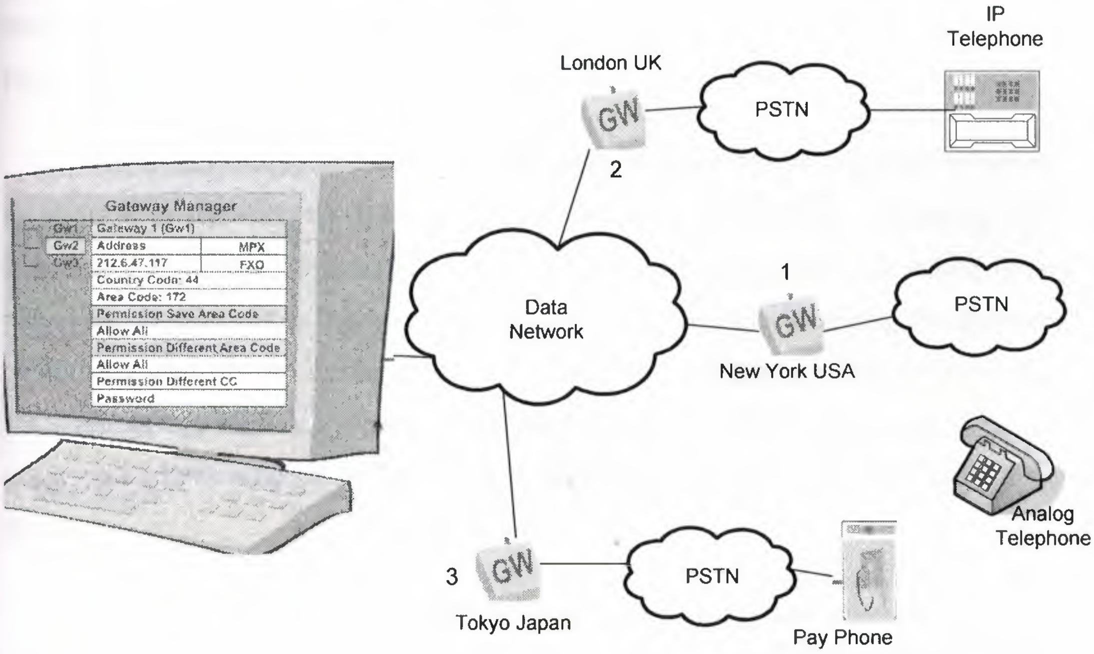
Figure 6.2 IP Telephony Call Manager Server (Proxy Server) Software

6.4 Gateway Manager

A gateway manager is used to configure gateways that connect data networks to other networks such as the public switched telephone network (PSTN). The gateway manager keeps track of the available gateways and what devices in the system are allowed to connect to them. Gateways maybe owned by the user or they may be available through other companies on a fee basis

The gateway manager will setup the authentication and access rights for each gateway. To allow services through the gateway, the IP address of the calling device or system may be used along with account codes and passwords.

Figure 6.3 shows how gateway manager software can be used to configure and manage gateways that connect the IP Telephony network to other networks such as the public switched telephone network. This example shows Figure 6.3 IP Telephony Gateway Manager Software how the gateway manager contains the configuration information for the gateway including the IP address and capabilities such as speech coders, protocols, and access control information.



Gatoway Manager

Gateway 1 (Gw1)

Address	MPX
212.6.47.117	FXO

Country Code: 44

Area Code: 172

Permission Save Area Code

Allow All

Permission Different Area Code

Allow All

Permission Different CC

Password

Figure 6.3 IP Telephony Gateway Manager Software

6.5 Unit Manager

A unit manager server is used to setup (configure) and map (identify) the communication units (IP Telephony telephones and media gateways) to the system. IP Telephony units require configuration so they can initially find the server (their addresses) to coordinate with and to know the protocols (such as IP Telephony) and processes to use (such as the type of speech compression used).

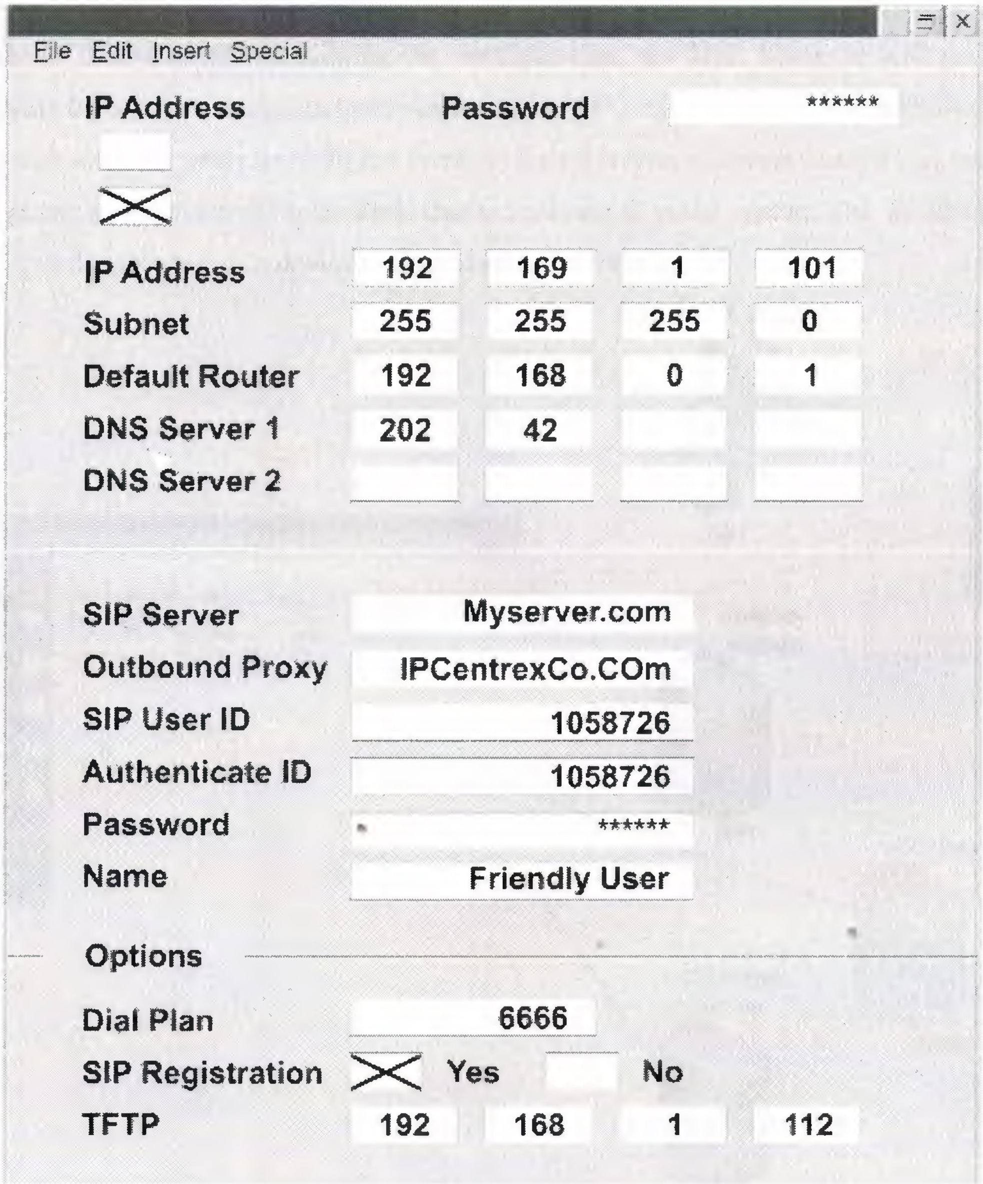
Units within the network may be manually configured through the unit manager or they may be configured automatically through automatic detection (device discovery) and downloading of configuration files. Unit configuration may also be done without the use of a unit manager server by logging onto the units IP address using a web browser (passwords are usually required) or by editing parameters of the Management Information Base (MIB) configuration file of the Unit using Simple Network Management Protocol (SNMP) commands.

Units are uniquely identified by an IP address (network address) and a MAC address (physical address). The IP address assigned to a unit may be fixed (static) or it may be assigned after it is turned on (dynamic addressing). MAC addresses are physically programmed into the device at the time of manufacture.

Units will usually be setup to communicate with one server or a group of servers to allow it to use different types of services (such as internal calls, incoming calls, outgoing calls, and conference calls). The unit may be programmed to search for its servers (to register) when it powers on. Unit configuration includes setting the addresses of servers used by the device (proxy servers), the identification and authentication information, the dial- ing plan and preferred features, when server registration will occur, and how new operating software and configuration data may be transferred to the device.

The unit's operating software (firmware) may be updated through the use of transferred files by means of the Trivial File Transfer Protocol (TFTP). TFTP is a slim version of a file manager system. The downloading of new firmware may be necessary to fix incorrect operations (software bugs) or to provide new feature capabilities for the units.

Figure 6.4 shows a typical unit configuration screen that is used to manually program the settings into an end-user device. This screen shows that the configuration of an IP Telephony unit involves setting the IP address mode (dynamic DHCP or static), setting a default router address, entering the address of the DNS server (for text address to IP address conversion), and the addresses of the call manager (IP Telephony proxy server). The setup also includes identifying information such as the IP Telephony user ID, login ID, and passwords. Optionally, some preference options may be set such as speech coder type (high or low bandwidth), silence suppression, registration periods, and dial strings.



IP Address □ ☒ Password ******

IP Address	192	169	1	101
Subnet	255	255	255	0
Default Router	192	168	0	1
DNS Server 1	202	42		
DNS Server 2				

SIP Server	Myserver.com
Outbound Proxy	IPCentrexCo.COm
SIP User ID	1058726
Authenticate ID	1058726
Password	******
Name	Friendly User

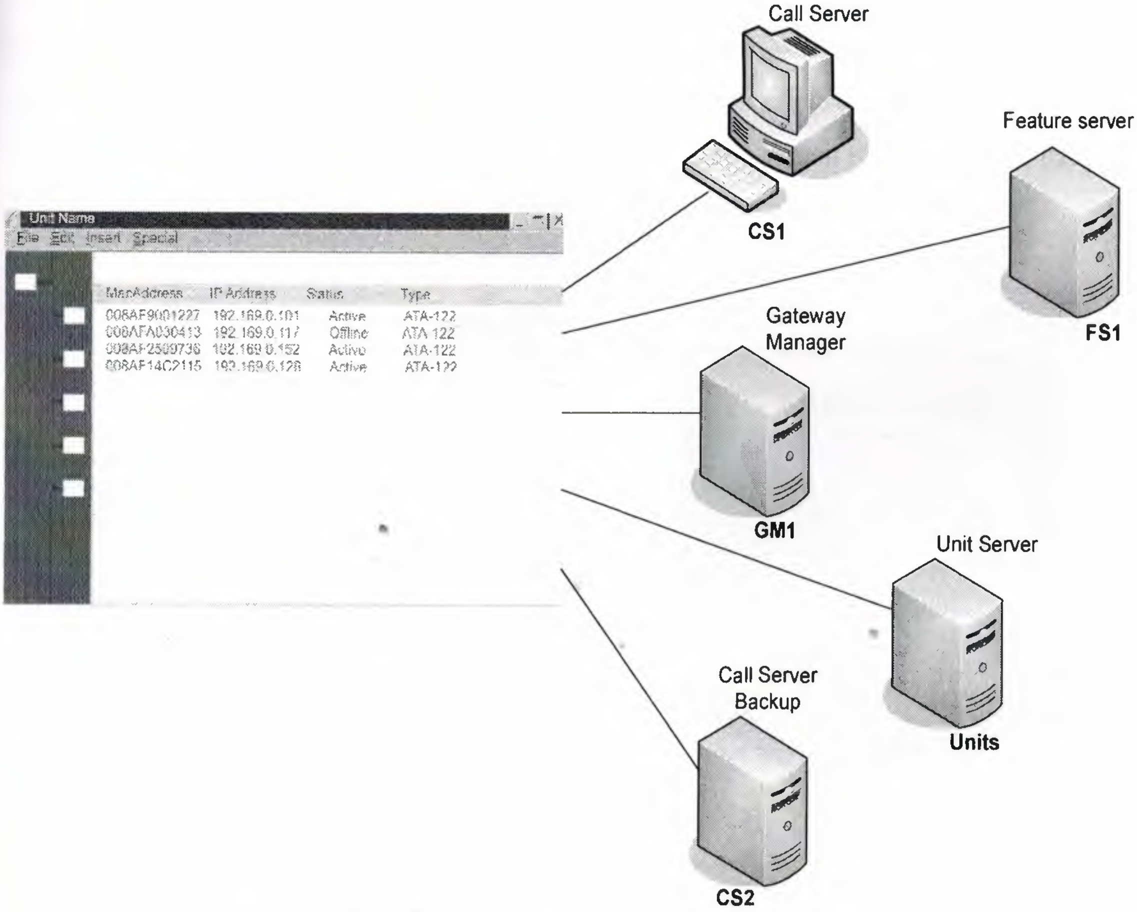
Options

Dial Plan	6666			
SIP Registration	☒ Yes	□ No		
TFTP	192	168	1	112

Figure 6.4 IP Telephony Unit Configuration Screen

When using a unit manager server to control the setup of IP Telephony devices, the devices that are managed usually include their status in the network (configured, off-line, active). The status is usually indicated by different colors. Red usually indicates that the unit is not configured or is inoperable. Blue may indicate that it has received some programming. Yellow might indicate the unit is partially operational. Grey can indicate the unit is ready (correctly configured) to operate.

Figure 6.5 shows how IP Telephony system unit manager software can be used to setup and maintain the configuration settings for end-user and gateway devices that can be managed by the IP Telephony system. In addition Figure 8.6, IP Telephony Unit Manager Software to maintaining the configuration, the Unit Manager will usually indicate the status of each unit that is connected to the network. This diagram shows that this unit manager provides different symbols for each type of device (such as an analog telephone adapter or an IP telephone) that are connected to the system. This system also changes the color of each device icon depending on its status.



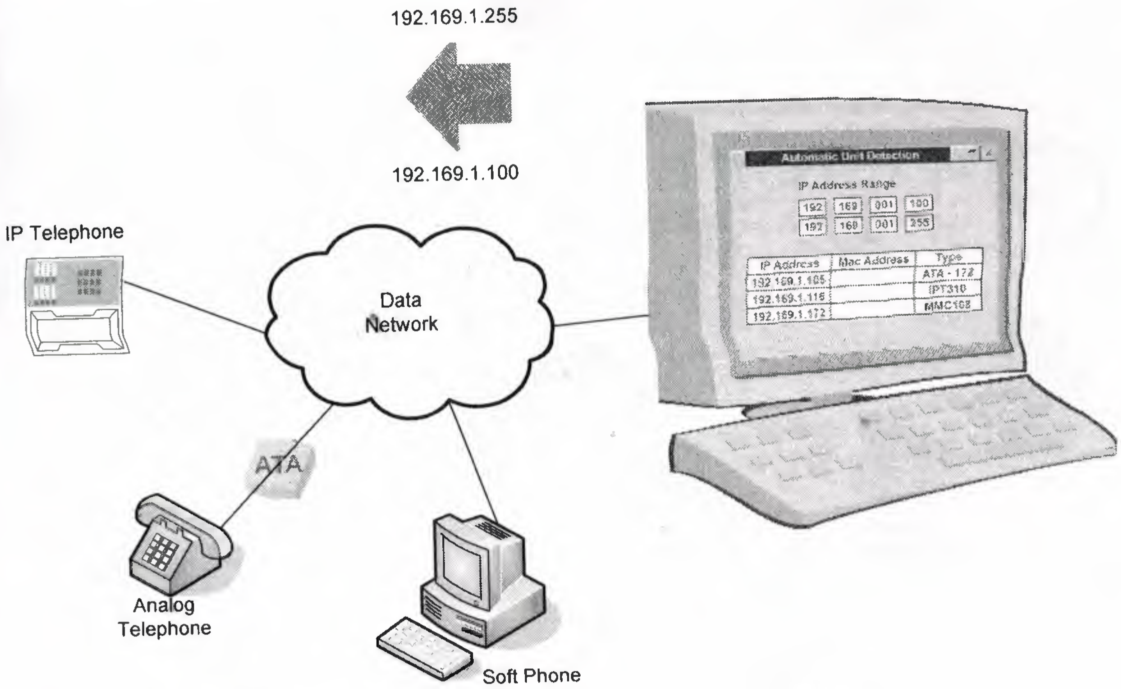
MacAddress	IP Address	Status	Type
008AF9001227	192.169.0.101	Active	ATA-122
008AFA030413	192.169.0.117	Offline	ATA 122
008AF2509736	192.169.0.152	Active	ATA-122
008AF14C2115	192.169.0.128	Active	ATA-122

Figure 6.5 IP Telephony Unit Manager Software

To uniquely identify each unit device, the device has an IP address (that may dynamically change) and a MAC addresses (that is pre-set at the factory and does not change). The type of device may vary from complete IP telephones (hardphones) to software telephones on multimedia computers (softphones).

To identify each device and its capability in the network, all the devices in the network may be requested to register. The addresses of the IP Telephony devices can be manually entered into the unit manager or they can be automatically detected. To automatically detect if an IP Telephony device is connected to the system, the Unit manager can send registration requests to all the devices in its network. To keep the amount of data transmission low and to keep from confusing non-IP Telephony units, registration requests can be sent within a range of addresses. This means that the addresses of end-user devices should only be programmed within a specific range of addresses. For example; 192.169.0.100 to 192.169.0.255.

Figure 6.6 shows how IP Telephony unit device automatic detection software can be used to automatically find end-user and media gateway devices. This example shows that the administrator has entered an IP address search range of 192.169.0.100 to 192.169.0.255 and that the software has found 4 devices. The search result produces the IP address, MAC address, and the type of device that has been found.



IP Address Range

192 169 001 100
192 169 001 255

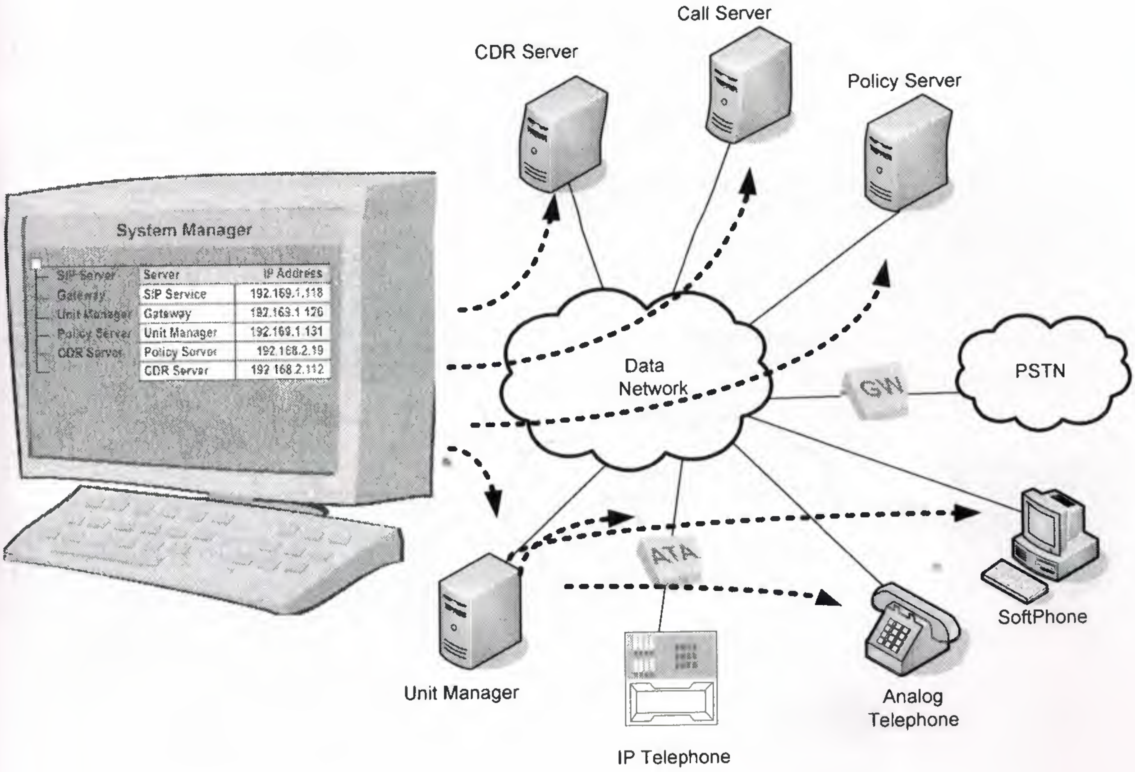
IP Address	Mac Address	Type
192.169.1.105		ATA - 172
192.169.1.116		IPT310
192.169.1.172		MMC108

Figure 6.6 IP Telephony Unit Auto detection Software

6.6 System Manager

An IP Telephony system manager server controls and links all of the IP Telephony system elements (servers) to each other. The system manager coordinates the routing of IP Telephony messages through the system by assigning addresses and communication ports (logical channels) to each server.

Figure 6.7 shows how system manager software is used to link all of the system server components to each other. This example shows a communication server (named CS1) that is operating as a proxy server. A feature server (called FS1) is setup to perform advanced feature processing. The gateway Figure 6.8, IP Telephony System Manager Software manager (named GM1) is used to coordinate calls between the IP Telephony data network and the public telephone network. The unit server (called Units) is used to manage the communication with specific types of end-user equipment. An additional server called CS2 is used as a backup communication server for the CS1 proxy server.



Server	IP Address
SIP Service	192.168.1.118
Gateway	192.168.1.176
Unit Manager	192.168.1.131
Policy Server	192.168.2.19
CDR Server	192.168.2.112

Figure 6.7 IP Telephony System Manager Software

6.7 Translation Server

A translation server in an IP Telephony system is a signaling gateway that can convert from one protocol to another protocol. While the IP Telephony protocols are different (for example IP Telephony, H.323 and MGCP), they perform similar functions. Some of the commands used during the setup, management, and termination of calls require the knowledge of the call status (the state). This means a translation server performs more functions than translating one message to another message.

6.8 User Manager

A user manager stores the features that have been selected by end users. These features may include speed dialing, call forwarding, voice mail options, and other configuration settings the user is allowed to select. To access the user manager feature, the end-user typically logs into the user manager web screen using their account and password. Figure 6.8 shows some of the feature options a user manager may display to a user. This example shows that each user must first log into their account using their account ID and password. The user can then select to setup options such as call forwarding, voice mail, email alerting, and other features.

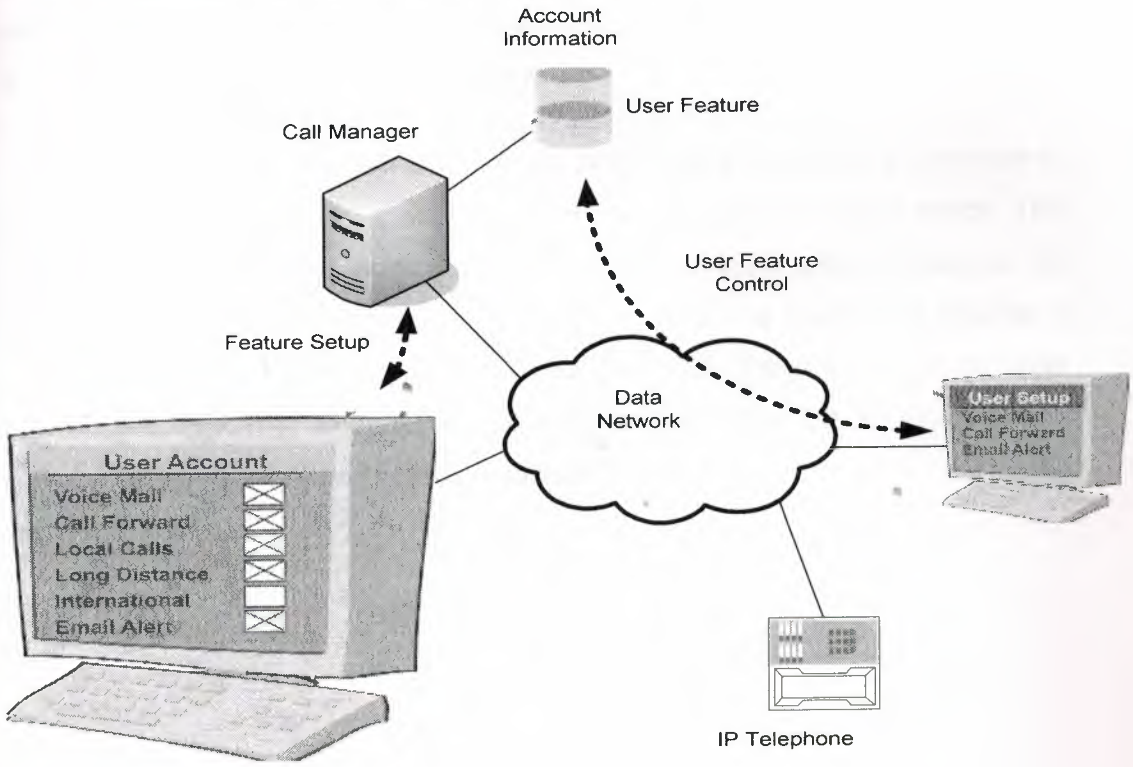


Figure 6.8 IP Telephony User Account Manager Software

6.9 Conference Server

A conference server (bridge) is a telecommunications facility or service that permits callers from several diverse locations to be connected together for a conference call. The conference server in an IP Telephony communication system must perform two key functions: controlling the addition and removal of participants and media mixing.

The conference bridge contains electronics for amplifying and balancing the loudness of each speaker in a conference call, so everyone can hear each other and speak to each other. Background noises are suppressed and typically only the current two or three loudest speakers' voices are retransmitted to other participants by the bridge, while a speaker's own voice audio is not sent back to that speaker to avoid audio feedback, echo or "squealing" self-oscillation.

Conference servers can be setup to manage one-step or two-step conference connections. One-step conference connections allow a caller to be connected directly to a conference by a calling a single number or IP address associated with the conference session. The two-step conference connection occurs when the caller first dials a universal phone number or IP address associated with conferencing and then uses an access code to connect to a specific conference.

Figure 6.9 shows how an IP Telephony system using a conference server can be used to receive requests to allow connections to a conference media server. This example shows a two-step process that has assigned a single telephone number for conference calls. When the conference server receives the connection request, it redirects the call to an interactive voice response (IVR) unit. The IVR prompts the caller to enter the conference identification information. When the IVR has collected the appropriate conference identification information, it redirects the call to the media server.

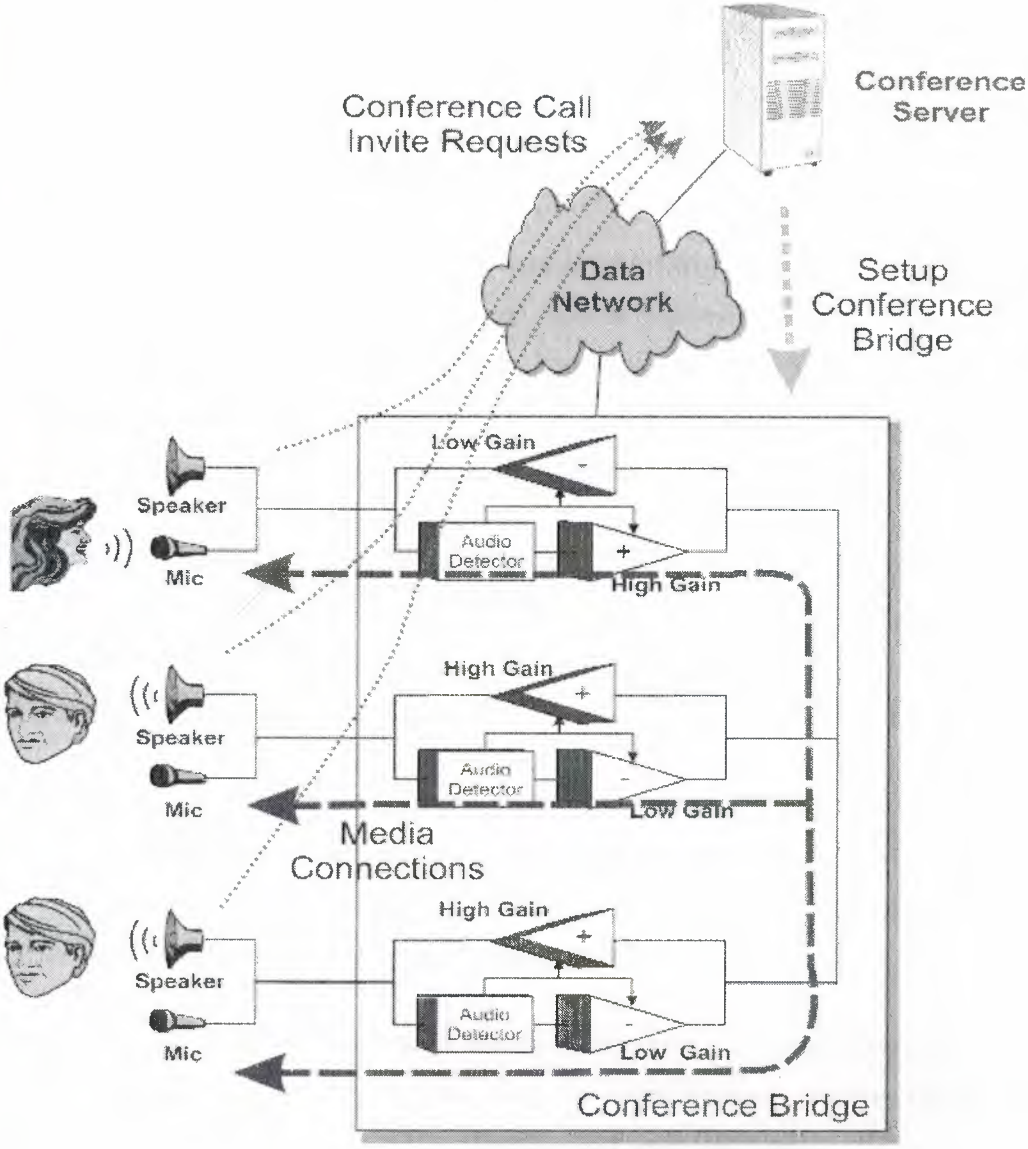


Figure 6.9 IP Telephony Conference Servers

6.10 Conclusion

This chapter explains the internal process of communication and its services. A gateway manager is used to configure gateways that connect data networks to other networks such as the public switched telephone network (PSTN). To configure and identify the communication units, unit manager server is used. Unit manager software can be used to setup and maintain the configuration settings for end-user and gateway devices that can be managed by the IP Telephony system.

7. IP TELEPHONY SYSTEM DESIGN AND SETUP

7.1 Overview

The design set up of an IP Telephony system involves the evaluation of existing communication networks, identifying service and feature requirements, purchasing IP Telephony equipment and core software, and configuration of servers and systems. It may also involve the purchase of IP Telephony applications software (such as voice mail modules) and possibly the development of software for custom system features.

Some of the reasons for installing a new IP Telephony-based communication system include replacing equipment that is near its end of use, the extension of telephone systems to remote sites, the installation of equipment at new locations, or the integration of telephone systems with information systems.

Designing an IP Telephony system can range from a simple turn-key solution to a complicated system integration process. A key option for the setup of an IP Telephony system is to use (or start with) a hosted IP Telephony communications system (called IP Centrex) instead of purchasing and operating a complete IP Telephony system.

The setup of an IP Telephony system usually includes the creation of a dial plan, assigning DID numbers to IP phones, the setup of phone groups, creation of an attendant console(s), configuration of call distribution, and voice mailbox.

7.2 IP Telephony System Design

IP Telephony system design involves identifying the key functional requirements of the desired system, evaluation of the data network requirements, review of the existing telephone system, analysis of available options, and the layout of the new system. One of the first steps in IP Telephony system design should be the definition of the functional business requirements for the new communication system. These functional requirements usually include traditional PBX voice and advanced call-processing features. They also may include unified messaging, call center, and security system integration features.

The first stage of system design is to make diagrams of your existing data and telephone system(s). This should include cable diagrams along with the types of cables (number of lines and category of lines) that are used. This will allow for the reuse of existing cabling when possible.

If an existing data network is going to be used as part of the IP Telephony system, it should be evaluated to determine if it has the available capacity (bandwidth), reliability, and quality of service (QoS) capability desired for the new IP Telephony system. The bandwidth should be evaluated at concentration points (routers and switches) along with the capacity of wide area connections that will be used as part of the IP Telephony communication system. The reliability objectives of the system should be considered and alternative routes and backup power supplies may need to be added. The quality of service (QoS) requirements of data communications and voice systems may be considered and a policy server may be used to give priority to voice packets.

The analysis of available options include which protocols will be deployed (such as IP Telephony, MGCP, or H.323), the use of a partial or complete hosted (IP Centrex) solution, the use of separate networks (independent phone and data), or an integrated network (shared data system).

Once the existing systems have been evaluated and the system options of protocols, server platforms, security firewalls, and end-user equipment types have been selected, layout of the system can begin. The system layout design will usually include the servers (computers) and their names, cabling types, end-user devices, and possibly software programs. The layout may also include connections and references to other carriers (such as an IP Centrex provider).

Figure 7.1 shows the basic design of an IP Telephony system. This diagram shows that the IP Telephony system is composed of servers (computers), end user devices (IP telephones), gateways (media adapters), and interconnecting data networks. This diagram shows a call manager server that coordinates the origination and reception of calls between telephone devices in its area (it's domain). An administrative server is attached to the network that is used to setup and manage user accounts on the system. This system includes multiple types of phones including IP telephones, analog telephones (through an analog telephone adapter), and softphones (on multi- media computers). The call manager can complete calls through a wide area data network or through a voice gateway that connects this system to the public switched telephone network.

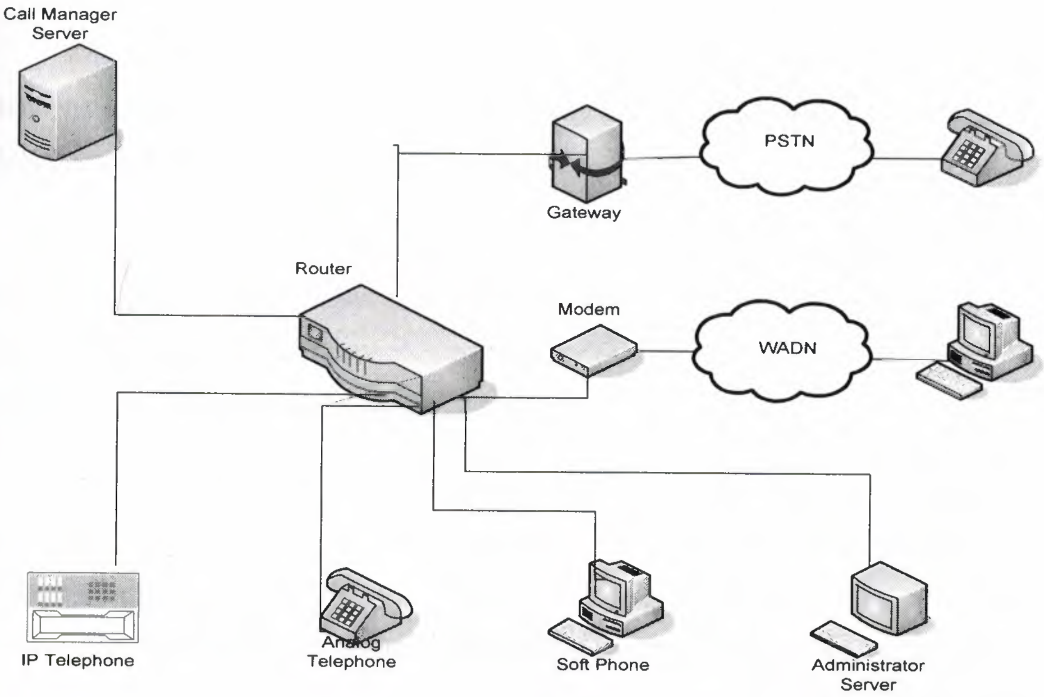


Figure 7.1 Basic IP Telephony System Design

7.3 Hosted IP Telephony Systems

Hosted IP Telephony systems are communication systems that divide between end user terminals and the call processing hardware and software that is managed by an external company. The use of a hosted IP Telephony system allows a company to benefit from the flexibility of IP Telephony technology and integration without the need to understand or manage the call processing. There are two basic types of hosted systems: Internet Telephone Service Provider (ITSP) and IP Centrex.

The use of a hosted system simplifies the development of IP Telephony based communication systems as the IP Centrex provider performs much of the call control and equipment management functions while allowing the customer to add, modify, and delete users from the system. Hosted systems typically provide an administrator access point (a web portal) to allow account management for the system services.

7.4 Internet Telephone Service Provider (ITSP)

Internet Telephony Service Providers (ITSPs) are companies that provide telephone service using the Internet. ITSPs setup and manage calls between Internet telephones and other telephone type devices.

An ITSP coordinates Internet telephone devices so they can use the Internet as a connection path between other telephones. ITSPs are commonly used to connect Internet telephones or PC telephones to telephones that are connected to the public telephone network at remote locations.

Figure 7.2 shows how an ITSP sets up connections between Internet telephones and telephone gateways. The ITSP usually receives registration messages from an Internet telephone when it is first connected to the Internet. This registration message indicates the current Internet Address (IP address) of the Internet telephone. When the Internet telephone makes a call, it sends a message to the ITSP that includes the destination telephone number it wants to connect to. The ITSP reviews the destination telephone number with a list of authorized gateways. This list identifies to the ITSP one or more gateways that are located near the destination number and that can deliver the call. The ITSP sends a setup message to the gateway that includes the destination telephone number, the parameters of the call (bandwidth and type of speech compression), along with the current Internet address of the calling Internet telephone. The ITSP then sends the address of the destination gateway to the calling Internet telephone. The Internet telephone then can send packets directly to the gateway and the gateway initiates a local call to the destination telephone. If the destination telephone answers, two audio paths between the gateway and the Internet telephone are created. One for each direction and the call operates as a telephone call.

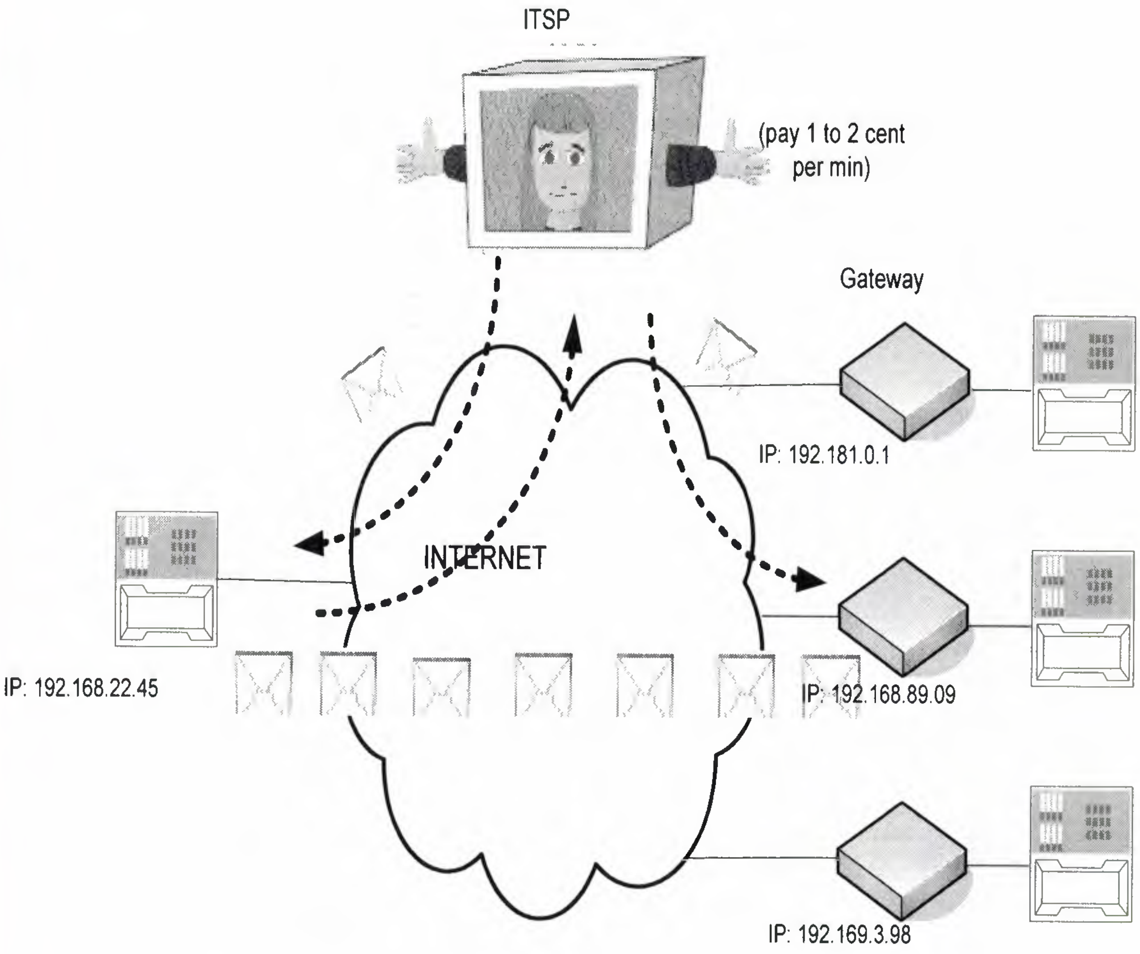


Figure 7.2 Internet Telephone Service Provider (ITSP)

7.5 IP Centrex Operators

Centrex, (a contraction of the term Centralized Exchange), is a telecommunications service that allows customers to get the full range of features available on a PBX, without actually owning or renting PBX hardware. The Centrex services are delivered to the customer by a dedicated partition of the central office or local exchange that behaves like the customer's PBX. IP Centrex operators provide Centrex like services to customers using Internet Protocol (IP) connections.

Figure 7.3 shows a basic IP Centrex system that allows a Local Exchange Company (LEC) in New York City to provide Centrex services to a company in Los Angeles. In this diagram, the LEC in New York City uses a Class 5 Switch to provide for Plain Old Telephone Services (POTS) and Centrex services to their local customers.

To provide Centrex services to new customers located outside the geographic area, the LEC has installed a network gateway in New York that can communicate with the customer gateway in Los Angeles. Because the network gateway converts all Figure 9.3, IP Telephony Hosted (IP Centrex) Communication System the necessary signaling commands to control and communicate with the customer gateway, the Class 5 switch does not care if the customer gateway is in Los Angeles or Tokyo. It simply provides the Centrex services as the user's request.

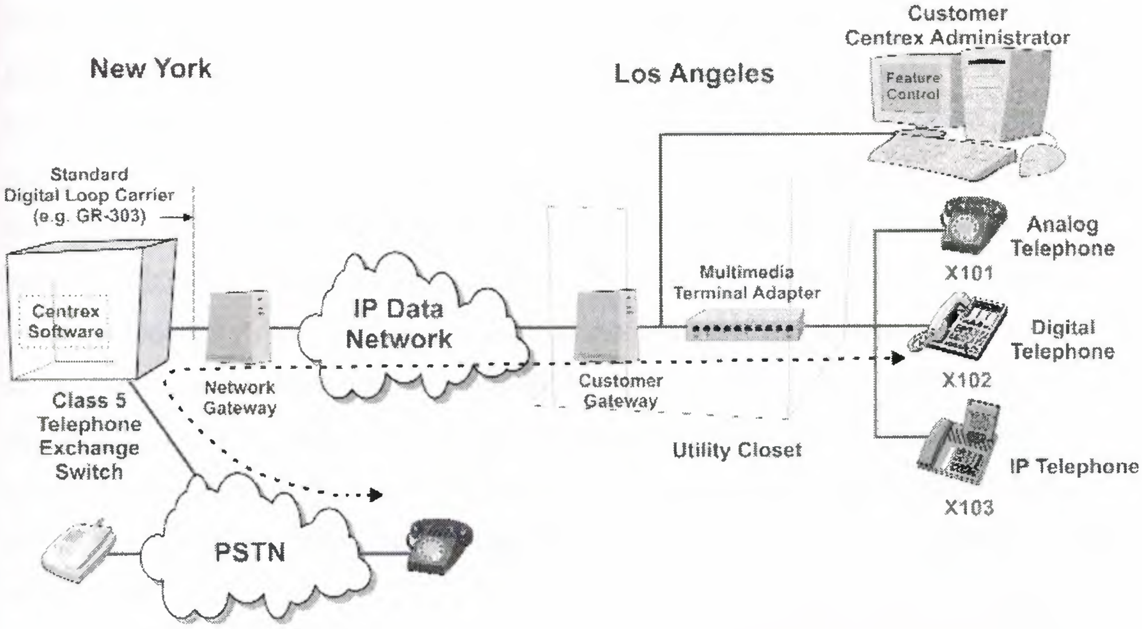


Figure 7.3 IP Telephony Hosted (IP Centrex) Communication System

7.6 Dial Plan

A dial plan (also called a dialing scheme) is the numbering system that is used by a company to identify devices within their network by unique numbers. After a system has been setup, a dialing plan is developed for each communication unit.

To implement a dial plan, a dial map is used. The dial map is the systematic use of certain prefix digits to dial a destination via user selected routing. An example is the use of the dialed prefix "9" from within a PBX to first select an outside local telephone line so that the originator can then dial a (typically 7 digit) local city telephone number. Similarly, a PBX may use the dialed prefix "8" to select a tie line to another PBX.

The dialing map must take into account dialing for emergency services request. Different countries use different numbers for emergency services (such as 911 in the USA and 999 in the UK). Each device within an IP Telephony system can be setup to use a different calling map to adjust for different calling patterns.

Figure 7.4 shows how a basic dial map operates. This diagram shows that there are several dial plan rules that are used each time a number is dialed. The first step in the dial map is to determine if the first digit is a 0, 3, 8, or 9. This first rule allows the system to determine if the caller desires to reach the attendant (0), is calling an internal number (3+), long distance (8), or outside line or emergency services (9). This rule changes how the next digit is processed. If the first digit is a 3, it is an internal call (4 digits for this system) and the system will wait for 3 more digits before attempting to connect the call to another unit in the system. If the first digit is 8, the system will capture multiple digits and analyze the call as a long distance public telephone number (country code, city code, exchange code, and extension). If the first digit is a 9, the system will analyze the following digits to determine if it is an emergency call (for example 911) or a local telephone call. If it is a local telephone call, the system will wait *until it has sufficient digits and connect the call to a local gateway. If the next 3 digits* were 911, it would connect the call to a local gateway and route to the emergency services number.

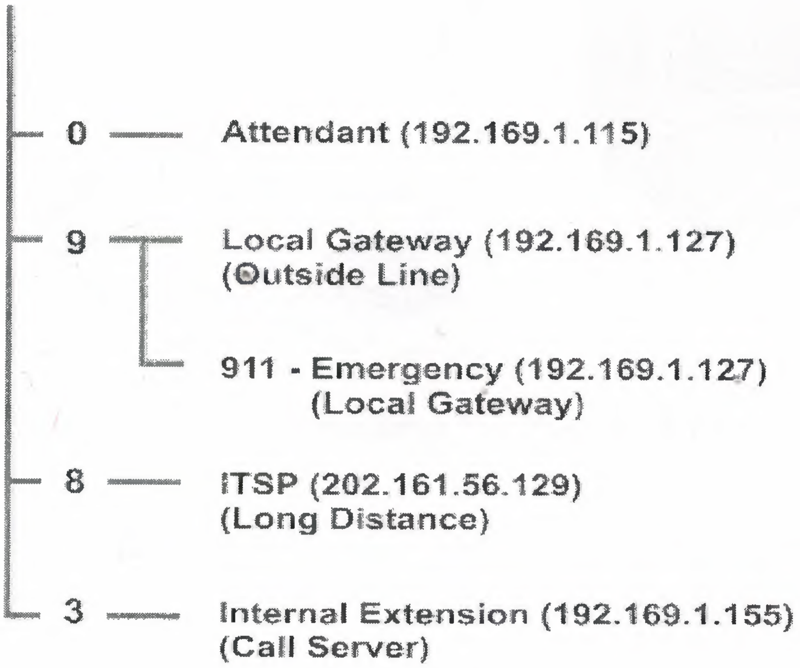
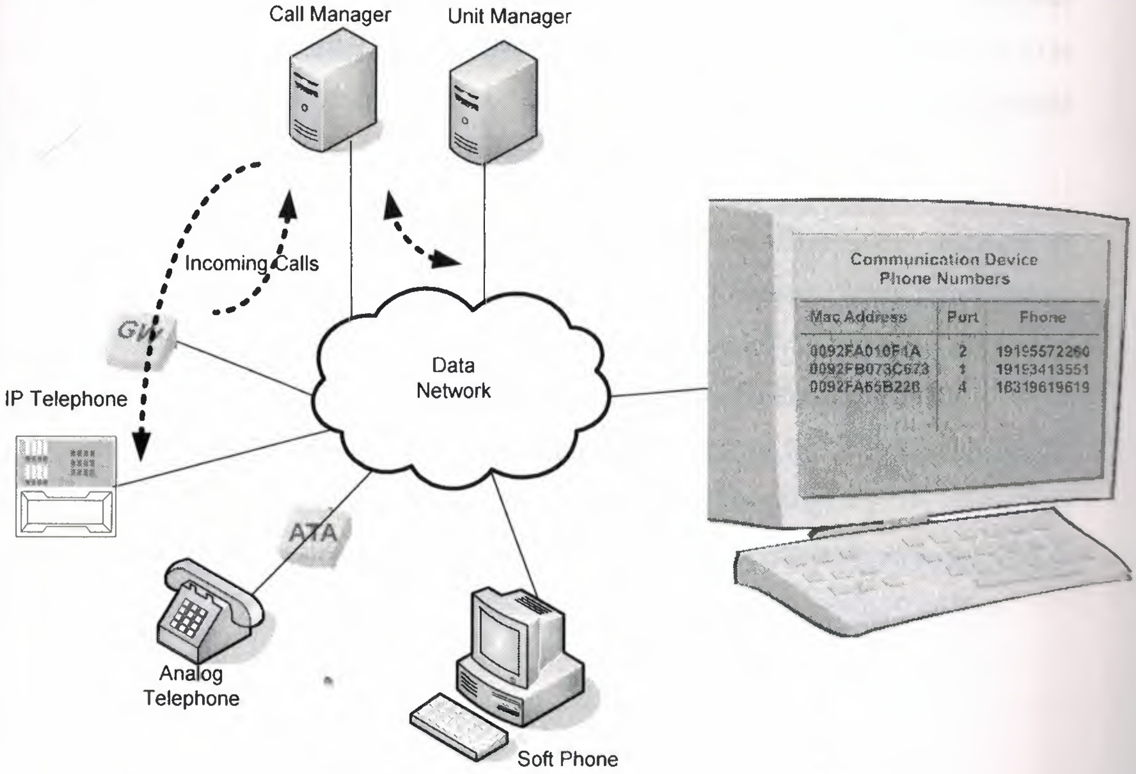

0
Attendant (192.169.1.115)
9
Local Gateway (192.169.1.127)
(Outside Line)
911 - Emergency (192.169.1.127)
(Local Gateway)
8
ITSP (202.161.56.129)
(Long Distance)
3
Internal Extension (192.169.1.155)
(Call Server)

Figure 7.4 Dial Map Operations

7.7 Direct Inward Dialing (DID) Assignments

Direct Inward Dialing (DID) assignments, map incoming phone numbers to specific extensions or groups within the communication system. Examples of DID assignments include assigning the main phone line (main telephone number) to attendant's console or to the auto attendant and assigning the fax line (fax telephone number) to the fax machine extension.

Figure 7.5 shows how the telephone administrator may assign telephone numbers to IP Telephony communication units. This diagram shows that each device that can be assigned a telephone number has a MAC address and potentially a port number.



Communication Device Phone Numbers

Mac Address	Port	Phone
0092FA010F1A	2	19195572260
0092FB073C673	1	19193413551
0092FA65B226	4	16319619619

Figure 7.5 IP Telephony Communication Unit Phone Number Assignment

7.8 Hunt Groups

A hunt group is a list of telephone numbers that are candidates for use in the delivery of an incoming call. When any of the numbers of the hunt group are called, the telephone network sequentially searches through the hunt group list to find an inactive (idle) line. When the system finds an idle line, the line will be alerted (ringing) of the incoming call. Hunt groups are sometimes called rollover lines.

7.9 Automatic Call Distribution (ACD)

Automatic Call Distribution (ACD) is a system that automatically distributes incoming telephone calls to specific telephone sets or stations based on the characteristics of the call. These characteristics can include an incoming phone number or options selected by a caller using an interactive voice response (IVR) system. ACD is the process of management and control of incoming calls so that the calls are distributed evenly to attendant positions.

Figure 7.6 shows a sample automatic call distribution (ACD) system that uses an interactive voice response (IVR) system to determine call routing. When an incoming call is initially received, the ACD system coordinates with the IVR system to determine the customer's selection. The ACD system then looks into the databases to retrieve the customers' account or other relevant information and transfer the call through the IP PBX to a qualified customer service representative (CSR). This diagram also shows that the ACD system may also transfer customer or related product information to the CSR.

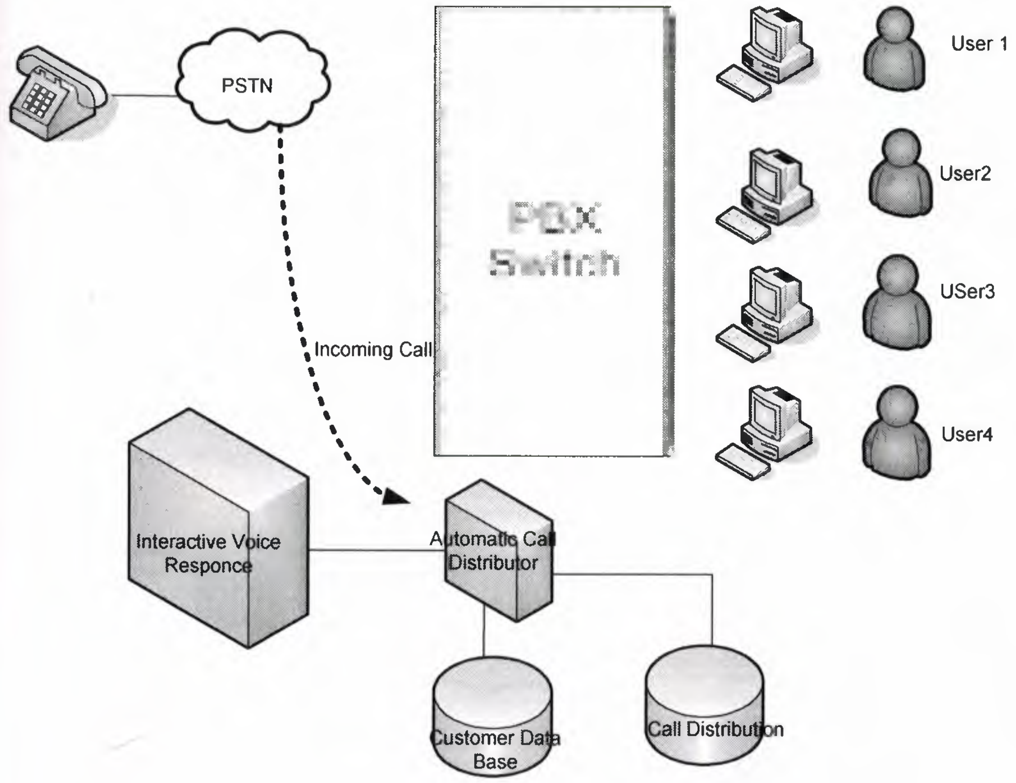


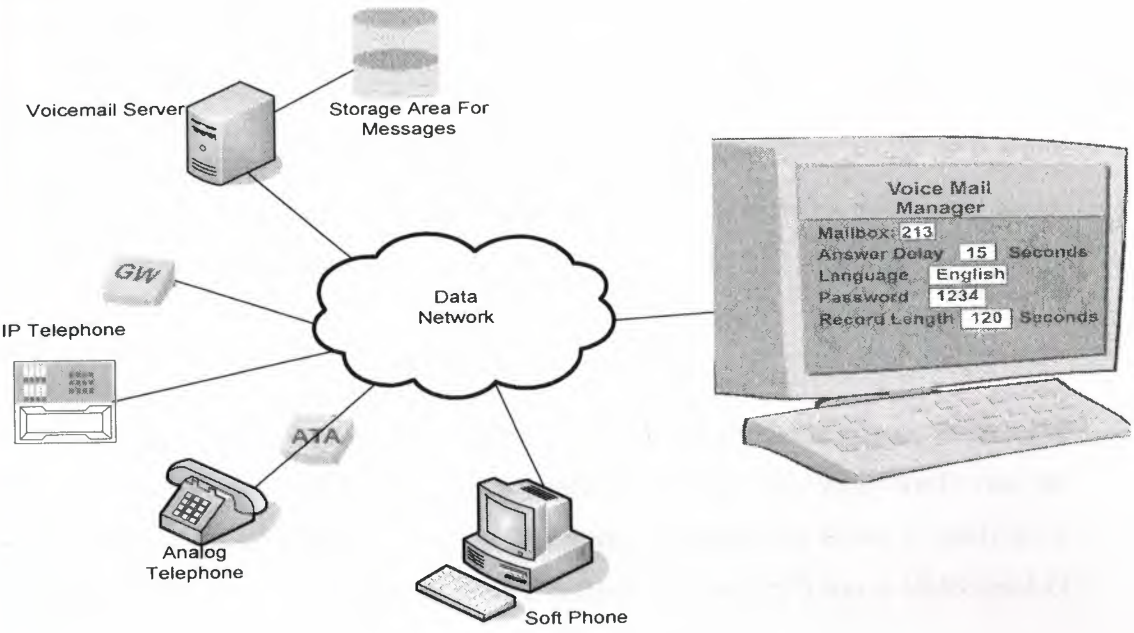
Figure 7.6 Automatic Call Distributions

7.10 Voice Mail (VM)

Voice mail is a service that provides a telephone customer with an electronic storage mailbox that can answer incoming calls and store voice messages. Voice mail systems use interactive voice response (IVR) technology to prompt callers and customers through the options available from voice mailbox systems. Voice mail systems offer advanced features not available from standard answering machines including message forwarding to other mailboxes, time of day recording and routing, special announcements and other features.

The administrator of the voice mail systems can setup and delete voice mailbox accounts for users. The users are provided with the tools to change their greeting, passwords, message forwarding, and other options.

Some of the available options for IP Telephony based voice mail systems include message waiting notification and forwarding voice mail to an email address. Voice mail notification can be sent to an email address or paging addresses. The voice mail contents can be sent as an attachment to any email address allowing the user to retrieve their messages via any computer terminal that has audio capability and is connected to the Internet. Figure 7.7 shows a voice mail server screen that allows an IP Telephony system manager to setup new media accounts on a IP Telephony voice mail system.



Voice Mail Manager
Mailbox: 213
Answer Delay 15 Seconds
Language English
Password 1234
Record Length 120 Seconds

. **Figure** 7.7 IP Telephony Voice Mail Software

7.11 Conclusion

The major aspect of IP telephony is its system design and setup. The first stage of system design is to make diagrams of your existing data and telephone system(s). This should include cable diagrams along with the types of cables (number of lines and category of lines) that are used. IP Telephony based voice mail systems include message waiting notification and forwarding voice mail to an email address. Voice mail notification can be sent to an email address or paging addresses.

8. VOICE QUALITY, SECURITY, RELIABILITY & FIREWALLS

8.1 Overview

The telephone network provides for a fairly high level of quality of service (QoS) and to be successful, Internet telephone service should have similar quality, security, and reliability. The relatively good audio quality of telephone systems is called toll quality audio. To ensure security, public telephone networks restrict physical access to systems and have relatively tight network access control processes. The high reliability of public telephone networks is achieved through extensive standards, tests, and government regulations.

8.2 Audio Quality

The measurement of telephone audio quality is subjective. Some of the key audio quality of service issues for Internet telephone service includes toll quality audio, control of echo, and the handling of audio distortion.

8.2.1 Toll Quality Audio

Mean Opinion Score (MOS) measures the quality of the telephone audio. The MOS is number that is determined by a panel of listeners who subjectively rate the quality of audio on various samples. The rating level varies from 1 (bad) to 5 (excellent). Good quality telephone service (called "toll quality") has a MOS level of 4.0.

Voice over data (VoIP) telephone service can provide toll quality audio (and in some cases better than toll quality). Public telephone networks directly digitize the voice and this provides for excellent voice quality. Voice over data telephone systems may use the same form of digitization as the telephone system (called G.711) or it may you a more efficient form of digital voice called speech compression. If the voice over data telephone service uses uncompressed digitized voice with reliable data connections, the quality of audio will be as good as or better than toll quality telephone service.

It can be better than toll quality because voice over data telephone service digitizes the audio at the caller's location. The telephone network may digitize the audio connection at the switching center that can be miles away. An analog telephone line between the telephone and the switching center can accumulate noise signals that cannot be removed in the way that digital signals can remove the added effects of noise.

An early challenge for voice over data telephone audio quality was the excessive use of speech compression to increase the cost efficiency of connections and the use of low-speed data connections (such as Internet dial-up service). Generally, the more you compress the voice, the lower the audio quality. Recently, innovations in speech compression technology provide similar toll quality service using a much lower data communication (connection) speed. This allows for efficient voice over data telephone service to provide better than toll quality audio.

8.2.2 Echoes

Echoes are a type of transmission impairment in which a signal is reflected (repeated) back to its originating source. In the transmission, the reflected signal often is attenuated (reduced in volume) and delayed, resulting in an echo. Figure 8.1 shows a common cause of echo in voice over data telephone systems. This diagram shows how a telephone is calling an IP telephone that uses speakers and a microphone (speakerphone). Some of the audio from the speaker reaches the microphone and travels back to the caller. Because Internet telephone transmission takes some time, the audio signal that is returned to the caller is slightly delayed.

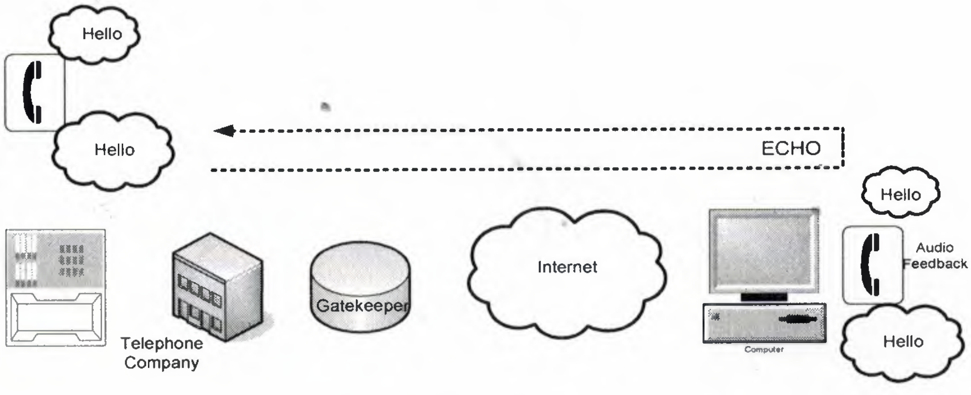


Figure 8.1 Causes of Internet Telephone Echo

8.2.3 Audio Distortion

Audio distortion is the undesired changing of an audio signal and it can come from a variety of sources in Internet telephone service. However, some of the key factors in audio distortion are packet loss, packet corruption, and echo.

Packet loss is the inability of the communication network to deliver a packet to its destination within a prescribed period of time. The effect of packet loss on audio distortion is to temporarily mute or distort the audio signal. Packet loss can result from a variety of events such as network congestion or equipment failures. Because audio communication systems require rapid delivery of packets of data, it is not usually possible or practical to resend packets of data that contain audio information. Because packet loss is infrequent and the packet size is relatively small, the loss of packets usually results in the temporary muting of information. In some cases, when a packet is lost, the missing segment may be recreated from the audio packet that is received from previous packets. Our voice does not change that much from packet to packet so we can repeat the previous packet to fill in the missing audio. As a result, if number of packets that are lost is relatively small, it is unnoticeable by the user.

Packet corruption is the changing of some of the packet data during its transmission. Packet corruption can come from a variety of sources such as poor communication line quality or momentary line loss from lightning spikes. Because voice over data telephone service may use *speech compression (not all do)*, the packet data represents a sound that will be recreated rather than a specific portion of the actual audio signal. As a result, if corrupted data is used, this can create a very different audio sound then expected. This distorted sound is commonly called "Warble."

Echo is a form of transmission signal impairment where some of the transmitted signal is reflected back to the originating source. There are several causes of echo in voice over data telephone systems. To reduce the effects of echo, voice over data telephone system equipment (gateways and IP telephones) may have one or more echo canceling devices to remove echo.

Figure 8.3 shows some of the causes and effects of audio distortion in voice over data telephone systems. This example shows that packet loss results in the temporary muting of the audio signal. Packet corruption results in the creation of a different altered sound than the sound that was previously transmitted. Echo results from some of the caller's audio signal being sent back (audio feedback) by the receiver of the call.

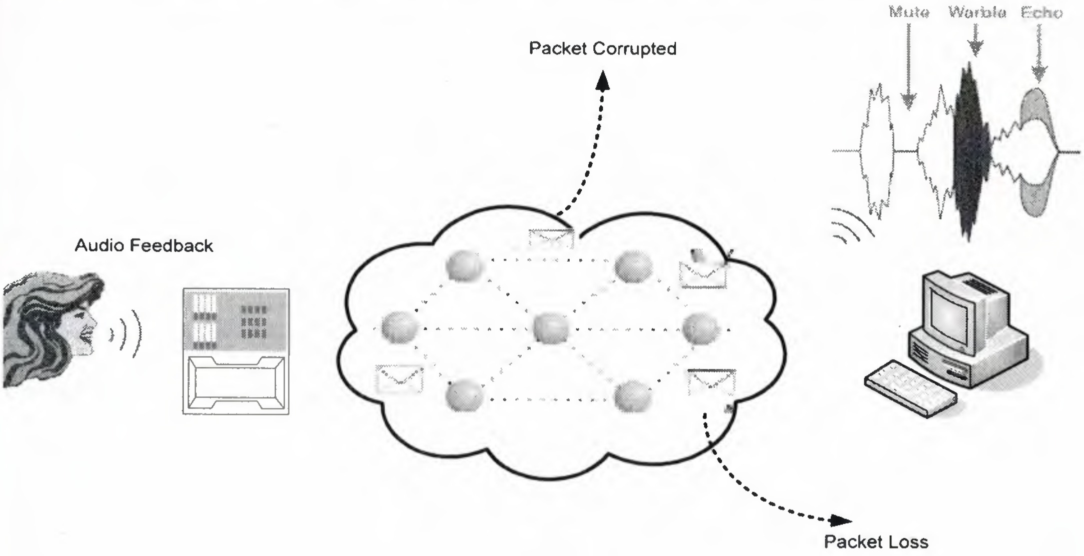


Figure 13.3 Internet Telephone Audio Distortion

8.3 Security

Communication security involves the control of physical access to information, identity validation (authentication), service authorization, and information privacy protection (encryption).

8.3.1 Physical Access

Physical access is the ability of a user or unauthorized user to physically send or receive information with a communication system or device. Gaining physical access to the to voice over data telephone service involves gaining physical access to the companies data network. If the company uses an unmanaged public data network such as the Internet, physical access must be done before or after entrance or exit from the Internet connection. When packets travel through the Internet, they can take many different paths through the network so it would be very difficult to gather all the packets.

Figure 8.4 shows that the typical physical access to Internet telephone service is usually limited to the connection points to and from the Internet. In this example, a cordless telephone is going through an analog telephone adapter gateway box. The gateway box is connected to the Internet service provider (ISP) through telephone wires that are mounted on the telephone poles. The ISP has a high-speed connection to the Internet that is also located on telephone poles. In this configuration, the radio signals between the cordless telephone and cordless base allow physical access by someone who is located within a few hundred feet within the cordless telephone. The physical access to the lines between the gateways and ISPs and ISPs to the Internet usually requires direction connection to telephone wires or data lines (such as coaxial cable). This example shows that once the Internet telephone call enters into the Internet, packets are usually routed through different paths. The only points that route all of the packets are the entry and exit points for the Internet.

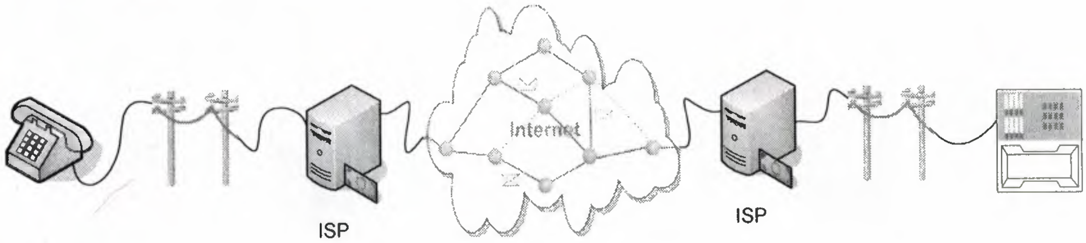


Figure 8.4 Physical Access to Internet Telephone Calls

8.3.2 Authentication

Authentication is a process during where information is exchanged between a communications device (typically a user device such as an IP telephone or mobile phone) and a communications network that allows the carrier or network operator to confirm the true identity of the user (or device). The validation of the authenticity of the user or device allows a service provider to deny service to users that cannot be identified. Thus, authentication inhibits fraudulent use of a communication device that does not contain the proper identification information.

Figure 8.5 shows that Internet telephone authentication is typically divided into at least 2 parts, ISP authentication and ITSP authentication. For step 1, the ISP requires that a user identify them to the ISP prior to being provided access to the Internet. This may involve low security user identification and password or it may involve a more secure authentication that requires the transfer of authorization codes. In step 2, the ITSP requires that the user identify them before being provided services from the ITSP. This also may involve simple identification and password or it may include the transfer of authorization codes.

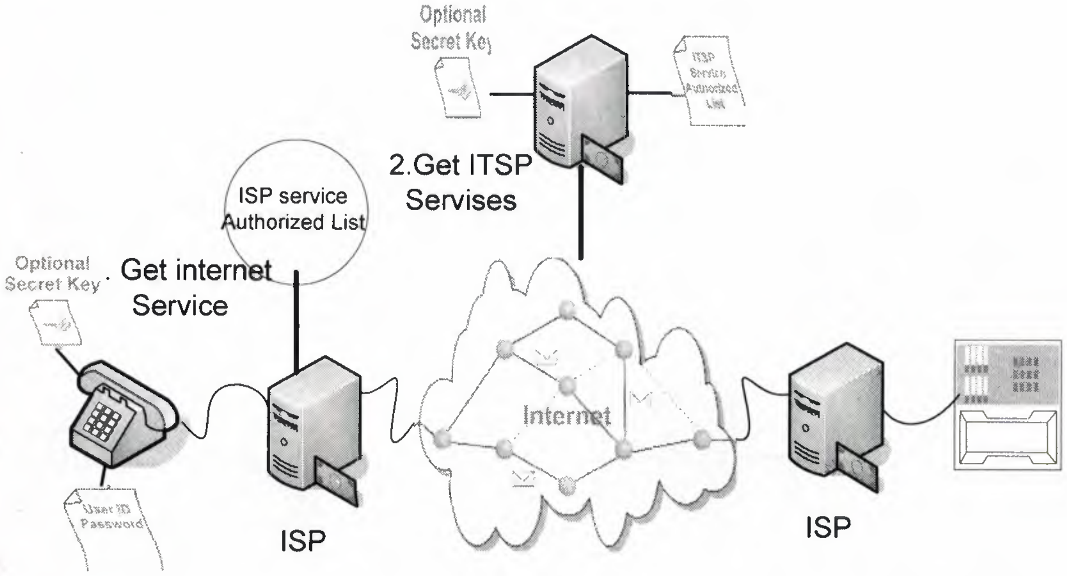


Figure 8.5 Internet Telephone Service Authentication

8.3.3 Encryption

Encryption is a process of a protecting voice or data information from being obtained by unauthorized users. Encryption involves the use of a data processing algorithm (formula program) that uses one or more secret keys that both the sender and receiver of the information use to encrypt and decrypt the information. Without the encryption algorithm and key(s), unauthorized listeners cannot decode the message. When the encryption and decryption keys are the same, the encryption process is known as symmetrical encryption. When different encryption and decryption keys are used (such as in a public encryption system), the process is known as asymmetrical encryption.

Encryption may be automatically provided between two points on a network. For example, on a cable modem, there is usually encryption between the cable modem and the cable network's connection to the Internet. This is important, as several users on the cable system will have physical access to the signals of other users on the cable network.

Figure 8.6 shows how voice over data telephone encryption may be used to protect information as it passes between callers via the Internet. This diagram shows two users that are sending encrypted (protected) packets of data to each other. To encrypt the data, the digital audio is processed (modified) using an encryption program (algorithm) and a key. When the encrypted data is received, the information is decoded using a decryption program (algorithm) and the key. If other users receive the encrypted packets, they cannot decode the information because they do not have the key.

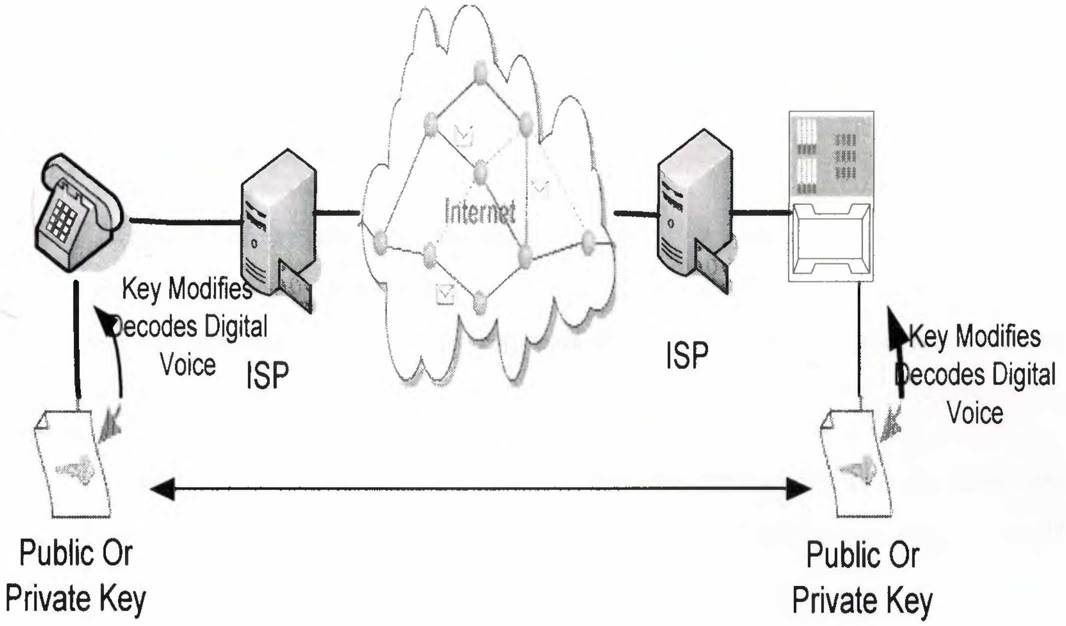


Figure 8.6, Encryption

8.4 Reliability

Reliability is the ability of a network or equipment to perform within its normal operating parameters to provide a specific quality level of service. Reliability can be measured as a minimum performance rating over a specified interval of time. These parameters include bit error rate, minimum data transfer capacity or mean time between equipment failures (MTBF).

The reliability of the public telephone network is 99.999% system reliable (commonly called the "Five 9's"). The public telephone network must be reliable because it is a lifeline service. Lifeline service assures a person can call for assistance or be contacted in the event of an emergency. Reliability factors for Internet telephone service include IP telephone access device reliability, data network connection reliability, data network reliability, call server reliability, and feature operation reliability.

8.4.1 Access Device Reliability

Access device reliability is the ability of device or system equipment to allow a user to gain access to a network within a specific quality level of service. For voice over data telephone service, the access device must be able to setup and receive calls and convert between audio and data packets.

To be effective, access devices must be able to continuously process audio and digital signals during a call. Access device operation may be dedicated (such as an IP telephone) or they may be shared (such as a PC telephone).

Access devices are often connected to a modem or local data network equipment. The reliability of these local data communication devices also affects the reliability of Internet telephone service. Some of these devices may change their configuration during connection and disconnection. If the data communication device does not appear to be working, it is best to turn its power off and restart the equipment.

Figure 8.7 show that the selection of access device can affect the operation and quality of voice over data telephone calls. In this example, an Internet phone is attempting to call a PC telephone through the Internet. The Internet phone is designed to perform one function, Internet telephone service and it always has the resources (processing power) to do this. Unfortunately, the PC telephone is a multipurpose device that is currently running several applications (word processor, spreadsheet, and email). When the computer receives this call, the other processes may cause the computer to miss the call (unlikely) or they may cause the audio to be somewhat distorted.

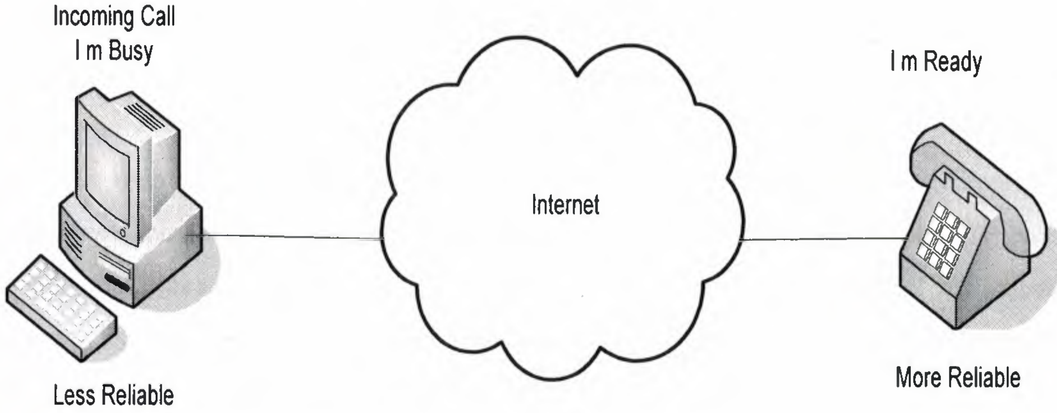


Figure 8.7 Internet Telephone Access Device Reliability

8.4.2 Data Network Reliability

Data network reliability is the ability of the communication network to consistently provide data transmission between points that are connected to the data network. Data networks such as the Internet were designed to successfully operate even if large portions of the network were destroyed. To accomplish this, the Internet was designed as a dumb network that uses smart switches. Each switch in the Internet (called a router) has the ability to dynamically change the path it uses to sending data through based on information it regularly receives from other routers. If a router can no longer send data to a neighboring router, it will automatically start to send data to a router it can communicate with. As a result, the Internet is very reliable as it can repair itself in the event equipment failures.

8.4.3 Data Connection Reliability

Data connection reliability involves the connection from your computer or IP telephone to the data network (such as the Internet). Your data connection may be divided into two parts; access provider and data network provider (such as an ISP). The access provider manages the connection between your equipment and the data network provider converts your data into a format that it can transmit through the data network.

Figure 8.8 shows the key parts of an Internet service provider (ISP) and how they can affect your communications reliability. This diagram shows that an Internet connection can be divided into an ISP portion and an access portion. This example shows an Internet telephone that is connected to a cable modem. The cable modem is connected to the head-end of the cable television system where a gateway adapts the data from the cable network into a format that can be used by the ISP. The ISP has a router that connects the gateway into a format that is sent to the Internet. This diagram shows that this ISP only has on connection to the Internet and if it experiences difficulty, the Internet connection can be lost.

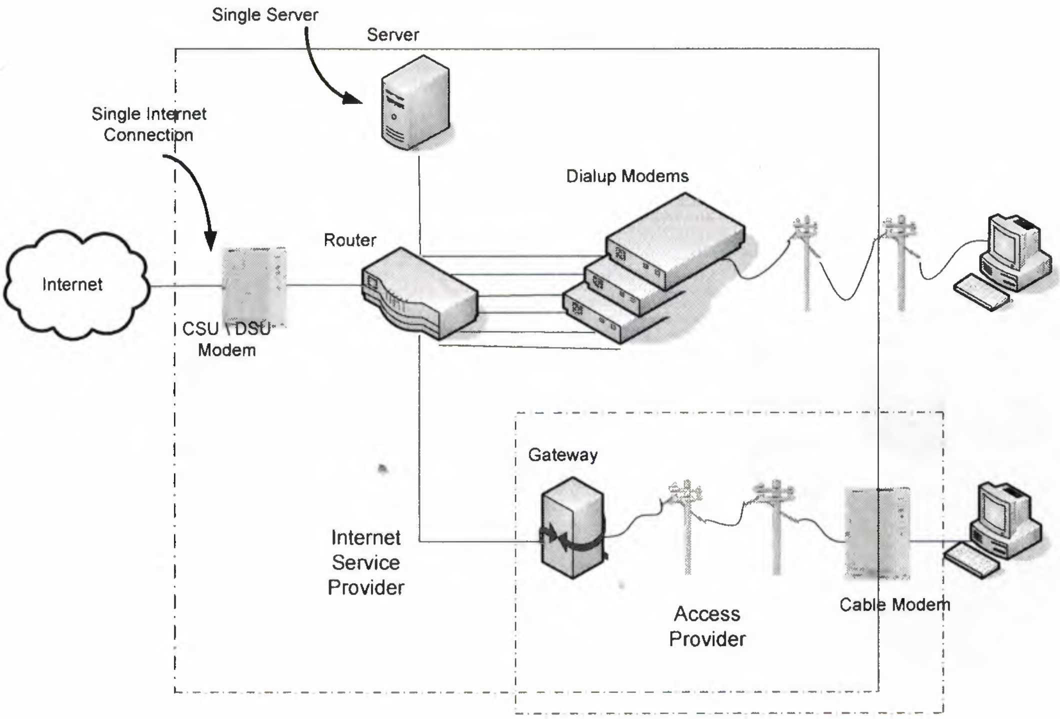


Figure 8.8 ISP Reliability

8.4.4 Call Server Reliability

Call server reliability includes the ability of a call server (call processing computer) to setup and control calls along with selecting and managing gateways. To ensure reliability, call servers may have redundant (duplicate) server equipment, updated lists of audio gateways, and use equipment that confirms to specific and compatible revisions of communication protocols.

Figure 8.9 shows the key parts of a call server that is used to provide Internet telephone service and how the configuration can affect reliability. In this example, the ITSP call server has two call processing centers that are connected to the Internet at different locations. Internet telephones communicate with the ITSP servers to setup and receive calls. Each sever has a gateway list that comes from a company that maintains lists of gateways (a clearinghouse). In the event of a failure of one of the servers, the other server will operate to setup and connect calls.

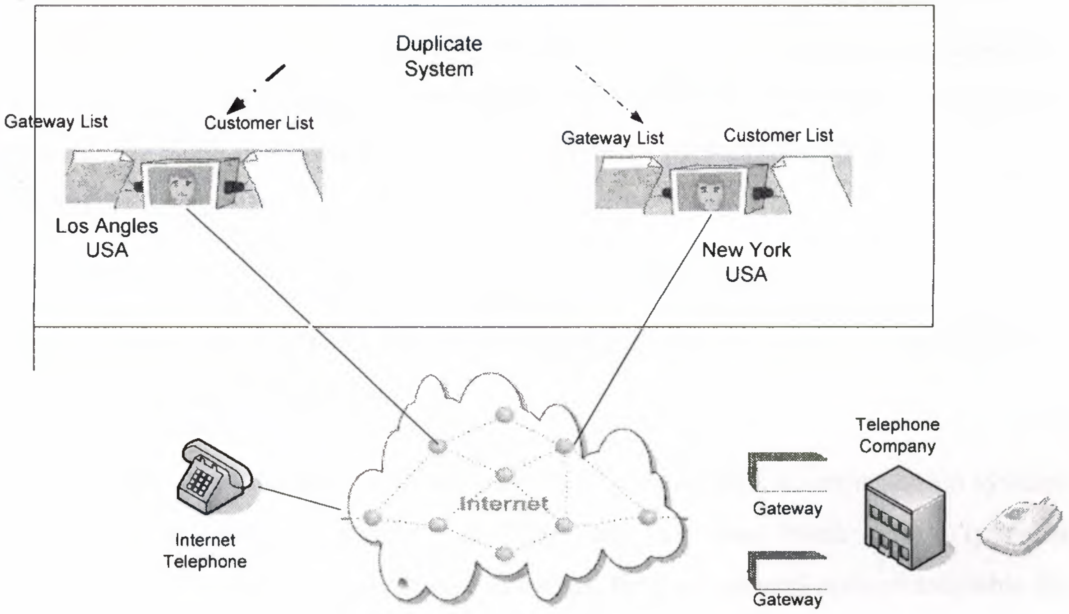


Figure 8.9 ITSP Reliability

8.4.5 Feature Operation Reliability

Feature operation reliability is the ability of the system to recognize and process feature requests. There are many features available in the public telephone networks and these features have been designed and tested to interoperate with each other. These features are usually managed by a single system. When these features are offered via ITSPs, there may be interaction with these features with features offered by different service providers. This can cause challenges with the operation of specific features. For example, if an ITSP provides a free voice mailbox and you have an answering machine, if you do not answer the call, it may be automatically routed to the voice mailbox provided by the ITSP.

8.5 Firewalls

A firewall is a data-filtering device that is installed between a computer server or data communication device and a public network (e.g. the Internet). A firewall continuously looks for data patterns that indicate unauthorized use or unwanted communications to the server. Firewalls vary in the amount of buffering and filtering they are capable of providing. An ideal (perfect) firewall is called a "brick wall firewall."

Getting through the firewall is called "Firewall Traversal." Firewalls create small data transmission paths between the inside network (protected area) and other networks (such as the Internet).

Firewalls can cause considerable problems for real time communication systems such as IP Telephony. Firewalls add delays and they may block certain types of protocols. To overcome the problem of firewalls, there are several options available for system designers. These include installing the voice communication system ahead of existing firewalls (bypassing the firewall), updating firewalls to detect and allow IP Telephony or other VoIP protocols, or installing Application Level Gateway (ALG) firewalls that can detect and modify IP telephony packets that enter and leave the system.

Figure 8.10 shows some of the firewall options that are available for IP Telephony systems. Option 1 shows how the IP Telephony system is installed ahead of the firewall that protects the company's information system. Option 2 shows a system that has upgraded the firewall to allow IP Telephony protocols. Option 3 shows a system that has installed an ALG firewall that can detect and modify packets that are transmitted between the external (public) system and internal (private secure) system.

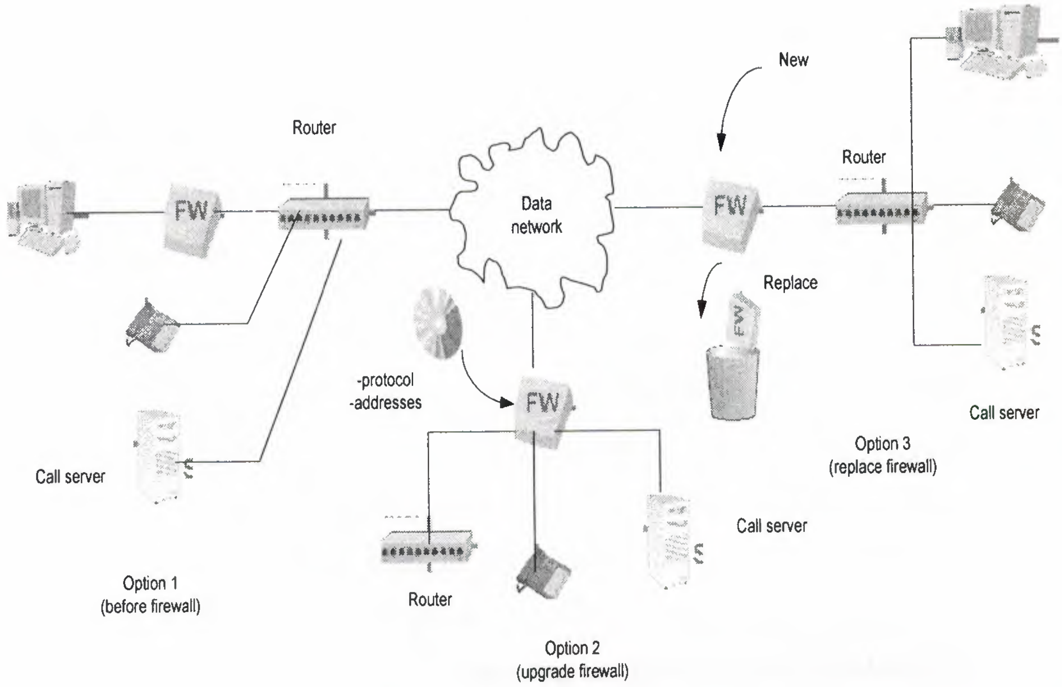


Figure 8.10, IP Telephony Firewall

8.6 Conclusion

While using any network, its reliability, stability and security are important factor. IP telephony provides high quality of voice and data transferring. In recent years, sophisticated audio processing equipment has been developed to allow the removal of echoes. Digital audio is processed (modified) using an encryption program (algorithm). To encrypt the data, the Firewalls vary in the amount of buffering and filtering they are capable of providing. An ideal (perfect) firewall is called a "brick wall firewall."

9. MOBILE AGENTS

9.1 Introduction to Mobile Agents

The term "**agent**" is heard frequently today. While it means a variety of things to a variety of people, commonly it is defined as an independent software program which runs on behalf of a network user. An agent may run when the user is disconnected from the network, even involuntarily. Some agents run on specialized servers, others run on standard platforms. Many examples of agent systems exist, and they are receiving much attention on the World Wide Web ("WWW").

At Mitsubishi Electric Information Technology Center America, they have developed a framework for the deployment of specialized agents called **Mobile Agents.** A Mobile Agent is specialized in that in addition to being an independent program executing on behalf of a network user, it can travel to multiple locations in the network. As it travels, it performs work on behalf of the user, such as collecting information or delivering requests. This mobility greatly enhances the productivity of each computing element in the network and creates a uniquely powerful computing environment well suited to a number of tasks.

Our framework, called **Concordia**, allows the creation of Mobile Agent programs written in the Java language. These programs use Concordia services to move about a network of distributed machines and to access services available on them. Common examples are user GUIs, databases, and other agents. Administrators control which services are available to which agents and users, and full management features are provided. By using Concordia, a new class of simple, easy-to-write and easy-to-run programs is enabled.

An Example of a Mobile Agent Application

A good example of a Mobile Agent application is a database search. Let's imagine a user with access to a corporate database is at some remote location, say a sales person is at a customer site. The sales person needs a price quotation and availability information for a product. Being at a remote location, there is no direct access to the corporate database, and the communications links are problematic. Security is a concern for the corporation, of course. How can agents create the solution to this problem?

The corporation first deploys the agent server. This is a straightforward operation which requires selecting a platform within the corporate Intranet. Perhaps this platform already exists, if not then many systems can provide it, such as a UNIX or Windows NT system. It can be collocated on an existing server machine, such as the database server, which in fact provides advantages of performance and management. The Mitsubishi Electric ITA product, Concordia, uses the Java virtual machine for its runtime environment, making installation particularly simple.

Once the agent server is installed, the sales people need portable computing platforms suitable for agents. In many cases, this platform will be a laptop or palmtop machine with either dialup or wireless communications devices. The software environment will be comprised of a suitable operating system combined with an agent runtime environment. In the case of Mitsubishi Electric ITA's product, Concordia, this again will be Java and may even be pre-installed.

The third step is to define security credentials and permissions which can be used to identify agents and their users to the existing database service, or whatever services are chosen for export by the corporation. An administrative process is then undertaken to manage and map these permissions, and the credentials are assigned to the users.

Finally, the corporation needs to create the agents which will actually perform the requests. This is not necessarily a major undertaking. For instance, Mitsubishi Electric ITA's Concordia system uses existing Java tools combined with a powerful Agent Tools Library to make easy the authoring of useful agents. In any case, the agent application is much easier to create and therefore is in production sooner. This is because it is the agent framework which transparentiy provides security, communications, and distribution so the programmer can focus on the job at hand.

Sales people then go to work productively, using the power of agents to perform their queries!

Advantages of Mobile Agent Programming

The following are the primary advantages of Mobile Agents:

- They facilitate high quality, high performance, economical mobile applications.

  Applications employing Mobile Agents transparently use the network to accomplish their tasks, while taking full advantage of resources local to the many machines in the network. They process data at the data source, rather than fetching it remotely, allowing higher performance operation. They use the full spectrum of services available at each point in the network, such as GUI's at the user and database interface on servers. They make best use of the network as they travel.

- They enable use of portable, low-cost, personal communications devices.

  Network support, including security, is contained in a lightweight server which manages the movement of agents in the network. Coupled with the sophisticated, self-contained programming model afforded by agents, this permits a small footprint to be achieved on user devices, without sacrificing functionality for the application.

- They permit secure Intranet-style communications on public networks.

  Security is an integral part of the Mobile Agent framework, and it provides for secure communications even over public networks. Agents carry user credentials with them as they travel, and these credentials are authenticated during execution at every point in the network. Agents and their data are fully encrypted as they traverse the network. All this occurs with no programmer intervention.

- They efficiently and economically use low bandwidth, high latency, error prone communications-channels.

The agent network employs a store and forward mechanism to transfer agents between nodes. This is well suited to the problematic nature of many communications channels, especially in the mobile arena. Queuing and persistent checkpoints enhance this further, to the point that agents can use such channels *with no degradation in reliability* or *response*. Because the agent data processing takes place locally to the source, the network has no effect on the agent as it executes.

9.2 Breaking the Client/Server Barrier

The Limitations of Client/Server

Historically, distributed applications such as these are created with "client/server" programming. In this model, an operation is split into two parts across a network, with the client making requests from a user machine to a server which services the requests on a large, centralized system. A protocol is agreed upon and both the client and server are programmed to implement it. A network connection is established between them and the protocol is carried out.

The client/server model has the advantage of enabling the removal of the client to smaller, remote machines, and it works well for certain applications. However it breaks down under other situations, including highly distributed systems, slow and/or poor quality network connections, and especially in the face of changing applications.

In a system with a single central server and numerous clients, there is only a problem of simple scaling. When multiple servers become involved, the scaling problems multiply rapidly, as each client must manage and maintain connections with the multiple servers. The use of two-tier systems or proxies only moves this problem to the network: it does not eliminate the basic problem.

With client/server comes a need for good quality network connections. First, the client needs to connect reliably to its server, because only by setting up and maintaining the connection may it be authenticated and secure. Second, the client needs to be assured of a predictable response, since its many requests of the server require full round trips to be completed. Third, it needs good bandwidth, since due to its very nature, client/server must copy data across the network.

Finally, the protocol which a client and server agree upon is by its very nature specialized and static. Often, specific procedures on the server are codified in the protocol and become a part of the interface. Certain classes of data types are bound to these procedures and the end result is a special network version of an application programming interface. This interface is extensible, but only at the high cost of recoding the application, providing for protocol version compatibility, software upgrade, etc. As the applications grow and the needs increase, client/server programming rapidly becomes an impediment to change.

Mobile Agents to the Rescue

Mobile agents overcome all these inherent limitations in client/server.

First and foremost, the Mobile Agent shatters the very notion of client and server. With Mobile Agents, the flow of control actually moves across the network, instead of using the request/response architecture of client/server. In effect, every node is a server in the agent network, and the agent (program) moves to the location where it may find the services it needs to run at each point in its execution. For example, the same agent interacts with the user via a GUI to obtain request keys, then travels to a database server to make its request.

The scaling of servers and connections then becomes a straightforward capacity issue, without the complicated exponential scaling required between multiple servers. The relationship between users and servers is coded into each agent instead of being pieced out across clients and servers. It is the agent itself that creates the system, rather than the network or the system administrators. Server administration becomes a matter simply of managing systems and monitoring local load.

The problem of robust networks is greatly diminished, for several reasons. The hold time for connections is reduced to only the time required to move the agent in or out of the machine. Because the agent carries its own credentials, the connection is simply a conduit, not tied to user authentication or spoofing. No requests flow across the connection, the agent itself moves only once, in effect carrying a greater "payload" for each traversal. This allows for efficiency and optimization at several levels.

Last and most important, no application-level protocol is created by the use of agents. Therefore, compatibility is provided for *any* agent-based application. Complete upward compatibility becomes the norm rather than a problem to be tackled, and upgrading or reconfiguring an application may be done without regard to client deployment. Servers can be upgraded, services moved, load balancing interposed, security policy enforced, without interruptions or revisions to the network and clients.

All in all, a significant advantage in Mobile Agents!

9.3 Operation on Diverse Hardware

To date, it has been very difficult, if not impossible, to provide user interfaces to systems from inexpensive, small, hand held user devices, nor in fact from mobile devices at all. Two things can change all that: the Java language and Mobile Agents.

We have seen how Mobile Agents can be used to create new, lightweight applications which move about the network to accomplish their jobs. Now consider what it means to deploy them on a range of devices from traditional desktop PC's to portables.

The Java language has created many new opportunities in the software world to create truly portable applications. The "skinny client" envisioned by the Java community consists of little more than the Java runtime, a GUI, and a communications path to a server. What better platform than this on which to consider agents? However, with so few local resources, how do we build powerful and useful applications which do not depend on expensive and local communications hardware?

The Desktop System

On the desktop, today the Web Browser is fast becoming the user interface of choice for many applications. Mobile agents are perfectly suited to this environment. With the powerful GUI tools, the integrated Java support, the security credentials and the rich communications across the LAN, all the pieces are in place for the Mobile Agent.

The Portable PC

The laptop or portable PC is basically identical to the desktop system, with the possible reduction in memory and disk resources, and of course the frequent disconnection from the LAN. This environment is where the advantages of Mobile Agents become apparent. While the machine may be able to function in a traditional client/server environment when docked in the office, it becomes much less useful as a remote client when used remotely. However, except for the network, all the software infrastructure is still available. Mobile agents can easily bridge this gap.

The Personal Communicator

The Personal Digital Assistant, or PDA, is not a new phenomenon, but the power of the hardware and software available on hand held, even pocketable devices, is. The Windows CE palmtop, the Apple Newton and the General Magic communicator, are three excellent examples of powerful, portable user computing platforms. All three additionally can run Java, or will someday soon.

The Server

Finally, agents run on servers, such as databases, groupware servers, and virtually any other system of interest. The Java virtual machine is omnipresent on such systems and in many cases is already supporting local access to their services. To such a server, Mobile Agents are simply another standard client. When coupled to the power of the Mobile

Agent network, an entirely different, more powerful system is created without impacting the server at all.

9.4 The Software Infrastructure

Creating a software infrastructure for the agents is the next step. Quite apart from the mechanisms of getting the agents to the various platforms in the network, verifying their identity and permissions, reconstituting their state and running them (all functions of Concordia), there is then the problem of making useful services available to them.

A number of possibilities exist:

Use an existing legacy system. This requires exporting the legacy system's programming interface to the agent runtime. Given that Concordia uses Java, this is straightforward.

Layer an existing legacy system under a standard agent API. This is similar to the first option, but more portable and possibly already provided by the software. A good example is JDBC, Java Data Base Connectivity, which is an open programming layer available for many databases.

Code a new service as an agent. This is not so far-fetched as it may seem, given the power of agent programming. Agents are well suited to many dynamic tasks and can be the framework of choice for a wide variety of operations such as searching, directories, etc.

Use a hybrid of all the above options.

When used as a "wrapper" for legacy systems, Mobile Agents can serve to provide numerous advantages not previously available. They can provide new clients for a fraction of the development cost. They can provide mobility to systems that were never designed with mobility in mind. They can provide management and security in systems over public networks, and a host of other advantages which we will cover when we discuss Concordia in detail.

9.5 The Advantages of Java

The Java language has a number of advantages that make it particularly appropriate for Mobile Agent technology. While Java is by no means the only language being employed by Mobile Agents, it is arguably the best choice. The reasons for this are many.

Java's main appeal for agents is its portability. Its use of bytecodes and its interpreted execution environment mean that any system with sufficient resources can host Java programs. There are even machines being built today that execute Java natively. For Mobile Agents this is a tremendous opportunity. The more platforms capable of executing the agents' code, the better.

A second advantage comes from the ubiquitous nature of Java on the Internet. Because it is embedded in many Web browsers, as well as application servers, there are many platforms deployed already. Application Programming Interfaces such as AWT, the Advanced Windowing Toolkit, and JDBC, Java Data Base Connectivity, are leading toward even more deployment of Java. Additionally, this deployment exactly targets the sort of services that agents can best use.

Another major advantage is the proliferation of tools that support Java programmers. Many programmers are already familiar with C++, which Java resembles in many ways. Added to that is the migration of existing tools to Java and the creation of many more. The net result is an abundance of high quality, easy to use tools for both development and debugging.

Finally, there is the movement of major segments of the software industry to Java. Not only will Java be here for many years to come, it will be employed in ever increasing applications. They at Mitsubishi Electric ITA, as well as others, are committed to making Mobile Agents part of this progression.

9.6 The Mobile Agent System

We now come to see the characteristics of the systems that utilizes Mobile Agents. Starting with a legacy system, or simply the existence of a database, order entry, groupware, or other system, we add software interfaces to these existing services. The language bindings are in Java, perhaps to existing Java definitions such as JDBC. To these straightforward API extensions, we write agents, prototyping them in only a few lines of Java code, and these agents navigate the network transparently to perform the programmer's requests. Users are entered into a security database and under control of a central policy, are allowed to launch these agents. With truly a minimum of work, a secure, distributed, mobile system is up and running!

Concordia

A Framework for Mobile Agents

9.6.1 Introduction to Concordia

Concordia is a full-featured framework for the development and management of network-efficient Mobile Agent applications which extend to any device supporting Java. Concordia consists of multiple components, all written wholly in Java, which combine together to provide a complete, robust environment for applications.

Concordia Overview

A Concordia system, at its simplest, is made up of a Java VM, a Concordia Server, and at least one Agent. The Java VM can be on any machine: it is a standard environment. The Concordia Server is a Java program which runs there, and at any other nodes on the network where agents may need to travel. The agent is also a Java program which the Concordia Server manages, including its code, data, and movement.

Usually, there are many Concordia Servers, one on each of the various nodes of a network, both user and server nodes. The Concordia Servers are aware of one another and connect on demand to transfer agents in a secure and reliable fashion. The agent initiates the transfer by invoking the Concordia Server's methods. This signals the Concordia Server to suspend the agent and to create a persistent image of it to be transferred. The Concordia Server inspects an object called the Itinerary, created and owned by each agent, to determine the appropriate destination. That destination is contacted and the agent's image is transferred, where it is again stored persistently before being acknowledged. In this way the agent is given a reliable guarantee of transfer.

After being transferred, the agent is queued for execution on the receiving node. This happens promptly but possibly subject to certain administrative constraints. When the agent again begins executing, it is restarted on the new node according to the method specified in its itinerary, and it carries with it those objects which the programmer requested. Its security credentials are transferred with it automatically and its access to services is under local administrative control at all times.

The work that the agent performs depends on its purpose, that is, the code which it was programmed to execute. Generally, agents have several components, just as any program has. An agent might start interactively, by prompting the user for search information, then may travel to a server to perform the query. Or, the agent may simply be a kind of remote demon, such as a mailbox filter or notification sender. As its methods complete, the itinerary causes the agent to be moved to other Concordia nodes. Therefore agents with different purposes will typically have different itineraries.

In all cases, the Concordia agent is autonomous and self-determining in its operation. In this way, it is unique since it is in control of its own itinerary.

The Concordia system is made up of numerous components, each of which integrates together to create the full Mobile Agent framework. The Concordia Server is the major building block, inside which the various Concordia Managers reside. Certain Concordia Managers have a user interface component, such as the Administration Manager. In any case, each Concordia Manager is responsible for a component of the Concordia design, in a modular and extensible fashion.

Concordia Components

All Concordia components are coded completely in the Java language.

Concordia Server [Conduit Server]

The Concordia Server, also called the Conduit Server, provides the communications infrastructure that allows for agents to be transmitted from and received by nodes on the network. It abstracts the network interface in order that Agent programmers need not know any network specifics nor need to program any network interfaces. The Concordia Server also manages the life cycle of the agent. It provides for agent creation and destruction, and provides an environment in which the agents execute.

Administration Manager

Administration of the Concordia network is provided by the Administration Manager, in cooperation with Concordia services running on the various nodes under administration. The Administration Manager manages all of the services provided by Concordia, including Concordia Servers, Security Managers, Event Managers, etc. The Administration Manager supports remote administration from a central location, so only one Administration Manager is required in the Concordia network, although more can be employed as desired. The Administration Manager has a user interface component which is its primary means of use.

Security Manager

The Security Manager is responsible for identifying users, authenticating their agents, protecting server resources and ensuring the security and integrity of agents and their accumulated data objects as the agent moves among systems. The Security Manager is also responsible for authorizing the use of dynamically loaded Java classes which satisfy the needs of agents. The Security Manager has a user interface component, in order to configure and monitor the security attributes of the various users and services known to Concordia. This user interface function is integrated into the Administration Manager interface.

Security credentials used by the Security Manager may come from a number of sources. For secure, self-contained systems, it may be that no credentials are needed. For systems that traverse public or semi-public networks, encryption may be required but credentials may need only reflect user identity, such as user name or group id. For fully fledged agent systems deployed on the Internet, strong authentication and security can be provided from external authorities such as Verisign. All these security levels can be supported by Concordia's Security Manager.

Persistence Manager

The Persistence Manager, also called the Persistent Store Manager, maintains the state of agents in transit around the network. As a side benefit, it allows for the checkpoint and restart of agents in the event of system failure. Additionally, it can checkpoint objects upon request by agents, to provide finer granularity of reliability guarantees for critical procedures. The Persistence Manager is completely transparent in its operation, that is, neither the agents nor the administrator need control or monitor its operation. However, management access is available if needed.

Event Manager

The Event Manager handles the registration, posting and notification of events to and from agents. The Event Manager can pass event notification to agents on any node in the Concordia network. The Event Manager works in conjunction with the Concordia Server to distribute events as needed. An important function of the Event Manager is to support Concordia agent collaboration.

Queue Manager

The Queue Manager is responsible for the scheduling and possibly retrying the movement of agents between Concordia systems. These features include the maintenance of agents as they await the opportunity to perform their work, maintaining their persistent state as they enter and leave a system, and retrying as necessary when Concordia systems are disconnected from the network. The Queue Manager provides the mechanism for prioritizing and managing the execution of agents on entry to Concordia nodes.

Agent Tools Library

The ATL is a library which provides all the classes needed to develop Concordia Mobile Agents. This of course includes the ***Agent*** class, and others derived from Java base classes, with interfaces to the Concordia infrastructure.

Advantages of Concordia

Well, if you've read this far, we don't need to sell you again on the advantages of developing applications using Mobile Agents. What, however, are the advantages of Concordia itself?

Concordia is written in Java. Therefore it's portable, even ubiquitous. It runs on platforms large and small, and integrates easily with existing applications and frameworks.

Concordia agents provide for mobile applications. Agents support mobile computing as well as off-line processing and disconnected operation. These applications are in turn written with little or no knowledge of the underlying communications that they will employ. Concordia both hides the details from the programmer and user, as well as allows the agent to adapt to its environment and administration.

Concordia agents are secure. Each agent carries the identity of the user that created it, and the operations the agent requests are subjected to the same user's permissions. Each agent is securely transmitted across the network, and no additional code is required to provide for secure, distributed operation.

Concordia agents are reliable. All Concordia agents are checkpointed before execution by the Persistence Manager, and they may return to these checkpoints if necessary. Objects the agents may create are checkpointed as well. Coupled with the services of the Queue Manager while they are being exchanged across the network, Concordia agents are assured of reliability at every stage of their operation.

Concordia agents can collaborate. The concept of collaboration is important and useful to the agent programmer. It can provide a number of benefits, such as enabling parallel operation over multiple servers or multiple networks. It can divide a task into suitable pieces, and these pieces can be carried out in the most appropriate places. The results of these sub-tasks are then assembled by collaboration. The collaboration framework then permits a decision to made based upon the results, which can be used to determine destination, action, or other appropriate behavior.

Uses of Concordia

Concordia:

- Enables mobilization of legacy applications
- Is a great way to program mobile devices as clients of applications
- Breaks client/server barriers
- Integrates with distributed objects e.g. CORBA
- Integrates with legacy systems e.g. databases.
- Easily piggybacks on Web

- Easily runs standalone

Concordia agents:

- Process data at the data source
- Pull data with them as they travel, i.e. they "learn"
- Can literally run anywhere: Web, desktop, palmtop, etc.
- Enable highly scaleable and parallel programming.
- Hide the network transport from application, developer, and user.
- Hide distribution, scale, parallelism from application.

And Concordia systems:

- Offer rapid prototyping with easy paths to production.
- Offer robust operation via persistent agents.
- Provide security and integrity.
- Support off-line and/or disconnected operation.
- Provide for heterogeneous database access
- Are a natural for software distribution - agents carry code to remote platforms

9.6.2 Deploying Concordia

Deploying Concordia is made substantially easier by Concordia's use of Java as its runtime framework. The portability of the Java virtual machine, coupled with Concordia's advanced administration makes the process an evolutionary one. This is because Java can be integrated into practically any system which runs the services that agents might need to access. In this way, no additional systems need be added to the network.

Next, Concordia provides its own advanced management functions, including administration and security. These are provided with easy to use GUI's and since they are coded completely in Java, they are immediately available wherever Concordia runs, without installation or porting effort.

Finally, Concordia makes use of available networks, it does not impose a protocol or distributed computing service of its own. Normally, Concordia employs existing TCP/IP communications services, widely available and compatible with local area network, dialup networks, and wireless public and private networks. Concordia provides its own security layer to protect the agents and their data as they pass across all such networks.

Management

Management of the many servers in a Concordia network is provided by the Concordia Administration Manager, which operates in conjunction with services available at each Concordia node. These services include the Concordia Servers themselves, along with the various Concordia Managers, including Security, Queue, etc. The Concordia Administration Manager provides a single graphical interface to the administrator for all the Concordia nodes in the network.

Administration

Concordia provides for administration of all its servers and services from any Concordia Administration Manager. The administration available falls into two major area, roughly divided along managing the Concordia Server and managing Concordia Agents.

The Concordia administrator can perform the following operations on Concordia Servers. This is not an exhaustive list but is intended to outline the major features of Concordia administration.

- Start and stop Concordia Servers.

- Upgrade and install Concordia Servers and installed software.
- Monitor Concordia Server performance.
- View Concordia Server logs.
- Manage the Concordia Persistent Store.
- Manage the Concordia Queues.

The Concordia administrator can also perform the following operations on individual Concordia agents. Again, this list is not exhaustive, but representative.

- Install and remove Agent code and libraries
- Manage agent itineraries
- Remotely launch agents
- Terminate, suspend, and resume agents.
- Monitor individual agent operations.

Security

The Concordia Administration Manager additionally manages Concordia security. Among the security aspects are the following.

- Management of trust relationships between Concordia Servers.
- User permission administration: user account, group and access to services.
- Encryption key administration.
- Monitor security logs.
- Monitor security statistics.

Deployment

Given these powerful tools, deploying Concordia is easy to achieve. Starting with an existing system or without, the requirements are few.

First, users must be assigned to devices, and these devices need the Java virtual machine, Concordia software, and a network connection such as TCP/IP.

Second, services must be provided and Java virtual machines must have access to them, either locally or remotely. Generally, the servers will have a local Java virtual machine, and will be connected to a LAN. User devices will be on the LAN or will connect via remote means. User devices could even be assigned accounts on the server itself, and both user and service will share the same Java virtual machine. Concordia software will be installed and configured.

Third, an administration node will be identified and configured. Again, this node can easily be the same as the major service node, or a different one. There can be multiple administration nodes in the network, for redundancy or for partitioning of tasks.

Fourth, agents are deployed in the new software infrastructure. These agents are individual and specially coded to perform their task, so there may be many or few, depending on need. Some agents will serve to execute tasks on user demand, others may reside close to services to perform disconnected service enhancements. Agents are expected to travel as necessary to complete their tasks in the Concordia network.

9.6.3 Programming Concordia Applications

Writing Mobile Agents

Writing a Concordia Mobile Agent is in many ways little different from developing a non-mobile Java program. The difference comes in structuring the application in such a way to take best advantage of the Concordia facilities. This section will give a brief introduction to the concept of writing a Concordia Mobile Agent. Mitsubishi Electric ITA has prepared a paper, ***"Concordia Agent Development Guide",*** which covers the issues of writing Mobile Agents in great detail. It is available upon request.

9.7 Summary

A mobile agent is a program that can migrate from machine to machine in a heterogeneous network. The program chooses when and where to migrate. It can suspend its execution at an arbitrary point, transport to another machine and resume execution on the new machine. In the picture below, an agent carrying a mail message migrates first to a router and then to the recipient's mailbox. The agent can perform arbitrarily complex processing at each machine in order to ensure that the message reaches the intended recipient.

CONCLUSION

A mobile-agent system called D'Agents is under development at Dartmouth College. The ultimate goal of D'Agents is to support applications that require the retrieval, organization and presentation of distributed information in arbitrary networks. Some of the research areas are:

- Security mechanisms
- Support for mobile and partially connected computers
- Navigation, network sensing and resource discovery tools
- Automatic indexing, retrieval and clustering techniques for text and other documents

D'Agents is used in several information-retrieval and workflow applications.

REFRENCES

[1] Curtis D. Johnson, Wireless Networks Technology, 5th ed, Prentice-Hall International Inc., University of Houston, 1997.

[2] D. Raychaudhuri, L. J. French, R. J. Syracuse, S. K.Biswas, R. Yuan P. Norseman, and C. A. Johnston,"WATMnet: A prototype wireless ATM system for multimedia personal communication," IEEE J. Select. Areas Common, vol. 15, pp. 83-95, Jan. 1997.

[3] Raymond Steele, Lajos Hanzo, Mobile Radio Communication, Wiley 1999

[4] http:// www.Google.com

[5] http://www.mks.com