Chapter 4

Internet Application Protocols

Internet Applications

- Domain Name System
- Electronic mail
- Remote login
- File transfer
- World Wide Web
- All use client-server model

Names

- Internet communication requires IP addresses
- Humans prefer to use computer names
- Automated system available to translate names to addresses
- Known as Domain Name System (DNS)

DNS Functionality

- Given
  - Name of a computer
- Returns
  - Computer’ s internet address
- Method
  - Distributed lookup
  - Client contacts server(s) as necessary

Domain Name Syntax

- Alphanumeric segments separated by dots
- Examples

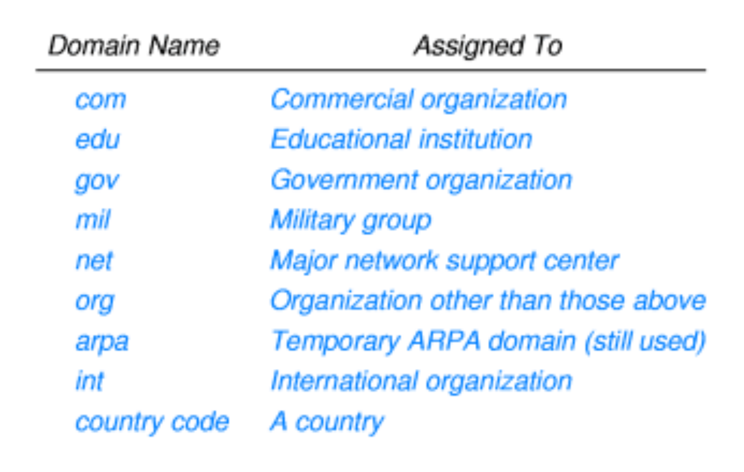
```
www.netbook.cs.purdue.edu
www.eg.bucknell.edu
```

- Most significant part on right

Obtaining a Domain Name

- Organization
  - Chooses a desired name
  - Must be unique
  - Registers with central authority
  - Placed under one top-level domain
- Names subject to international law for
  - Trademarks
  - Copyright

Some Top-Level Domains



Domain Name	Assigned To
com	Commercial organization
edu	Educational institution
gov	Government organization
mil	Military group
net	Major network support center
org	Organization other than those above
arpa	Temporary ARPA domain (still used)
int	International organization
country code	A country

- Meaning assigned to each

Within Organization

- Subdivision possible
- Arbitrary levels possible
- Not standardized
- Controlled locally by organization

Example Name Structure

- First level is `.com`
- Second level is company name
- Third level is division within company
- Fourth level either
  - Company subdivision
  - Individual computer

An Example

- Assume
  - Company is *Foobar*
  - Has two divisions
    - Soap division
    - Candy division
  - Candy division has subdivisions
  - Soap Division has no subdivisions

An Example (continued)

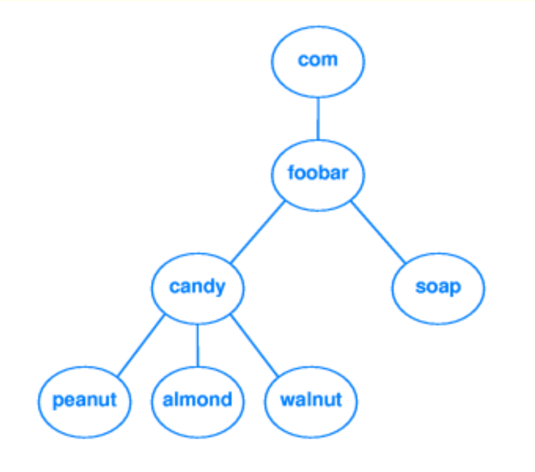
- Names in soap division have form

computer.soap.foobar.com

- Names in candy division have form

computer.subdivision.candy.foobar.com

Illustration of Foobar Naming Hierarchy



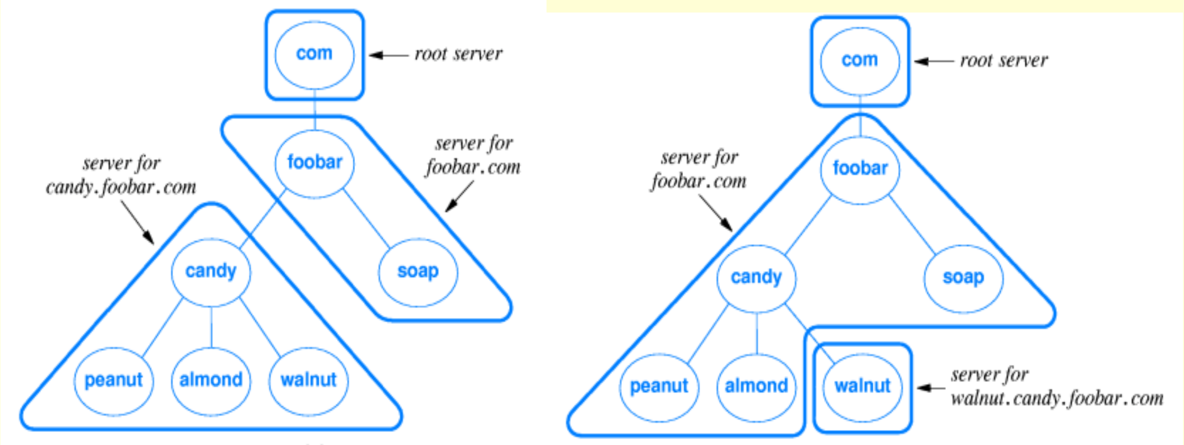
The Point About Names

The number of segments in a domain name corresponds to the naming hierarchy. There is no universal standard; each organization can choose how to structure names in its hierarchy. Furthermore, names within an organization do not need to follow a uniform pattern; individual groups within the organization can choose a hierarchical structure that is appropriate for the group.

DNS Client-Server Interaction

- Client known as resolver
- Multiple DNS servers used
- Arranged in hierarchy
- Each server corresponds to contiguous part of naming hierarchy

Two Possible DNS Hierarchies



♦Choice made by organization

Inter-Server Links

All domain name servers are linked together to form a unified system. Each server knows how to reach a root server and how to reach servers that are authorities for names further down the hierarchy.

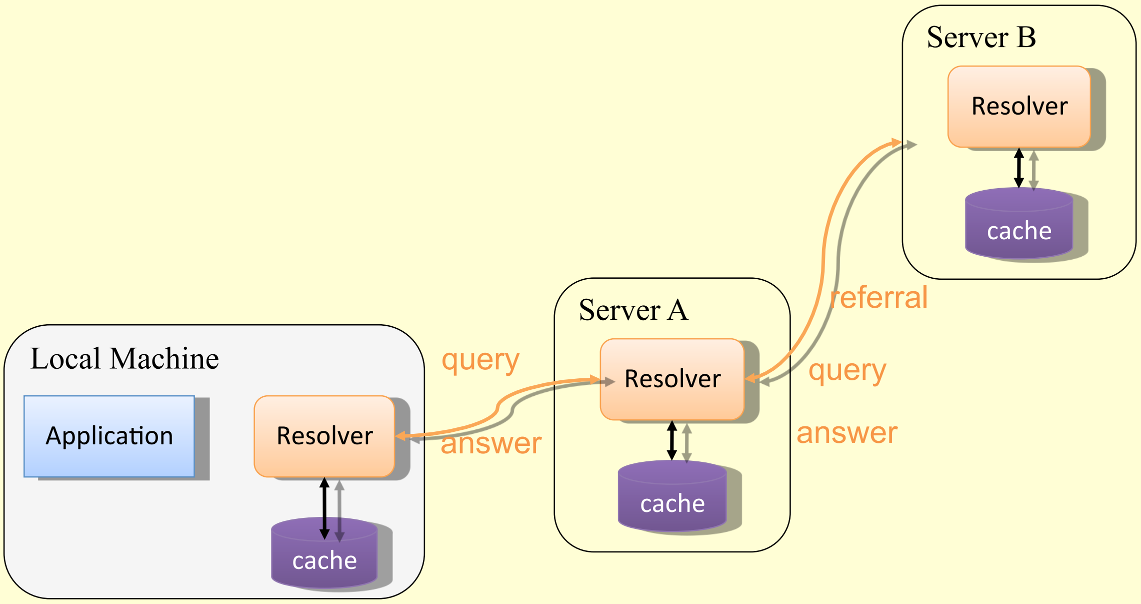
In Practice

- DNS uses backup server(s)
- ISPs and others
  - Offer DNS service to subscribers
- Small organizations and individuals
  - Only need domain names for computers running servers
  - Contract with an ISP for domain service

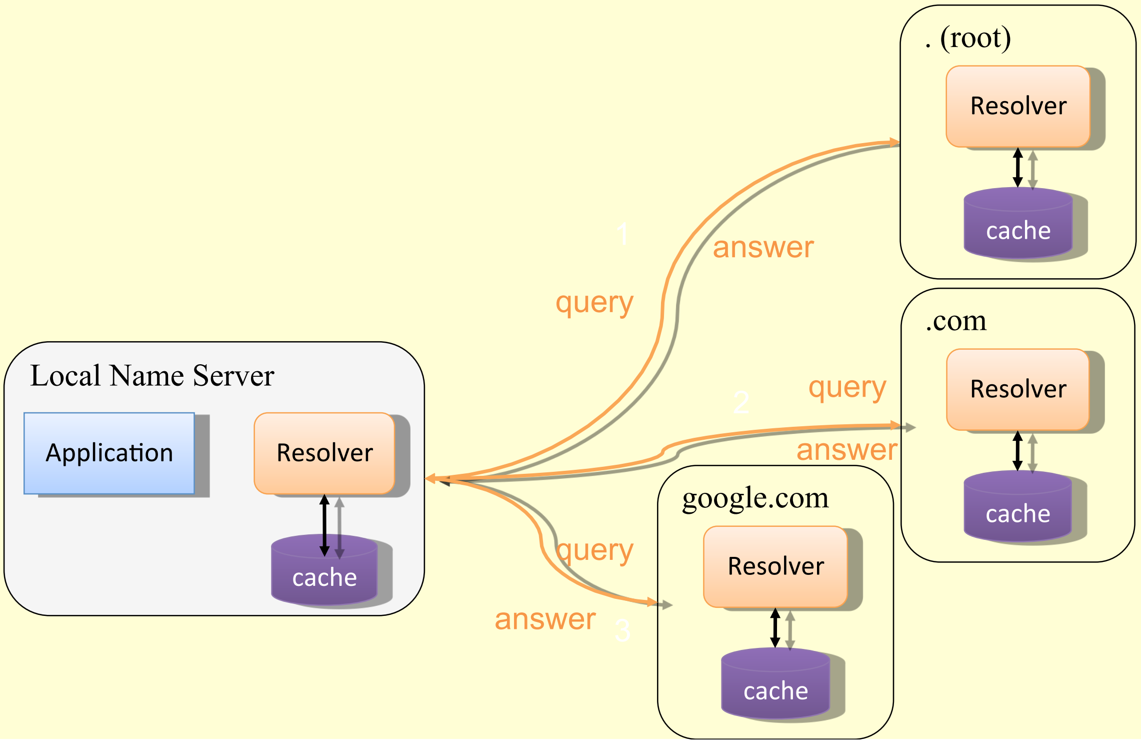
DNS Lookup

- Application
  - Becomes DNS client
  - Sends request to local DNS server
- Local server
  - If answer known, returns response
  - If answer unknown
    - Starts at top-level server
    - Follows links
    - Returns response
  - Called name resolution

Recursive Name Resolution



Iterative Name Resolution



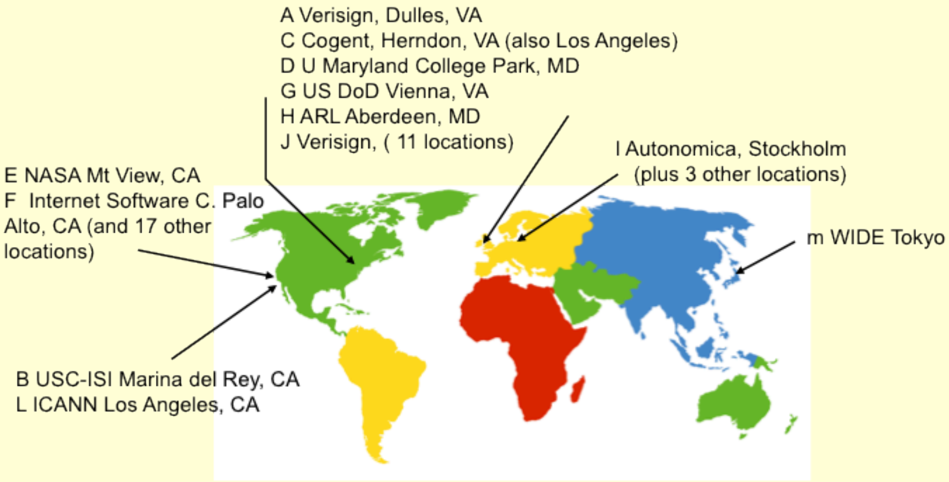
Root name servers

- The root domain contains all top-level domains of the Internet.
- As of March 2013, there are 22 generic top-level domains (gTLDs) and about 250 country code top-level domains (ccTLDs) in the root domain.
- This may soon change dramatically since you can now buy your own gTLDs!

Root name servers (cont.)

- There are currently 13 root name servers specified, with names in the form *letter*.root-servers.net, where letter ranges from *A* to *M*.
- This does not mean there are 13 physical servers; each operator uses redundant computer equipment to provide reliable service even if failure of hardware or software occur.
- Additionally, nine of the servers operate in multiple geographical locations using anycast, providing increased performance and even more fault tolerance.
- The number of root server instances is 356 as of 6 March 2013

Root name servers (cont.)



A Verisign, Dulles, VA
C Cogent, Herndon, VA (also Los Angeles)
D U Maryland College Park, MD
G US DoD Vienna, VA
H ARL Aberdeen, MD
J Verisign, (11 locations)

I Autonomica, Stockholm
(plus 3 other locations)

E NASA Mt View, CA
F Internet Software C. Palo Alto, CA (and 17 other locations)

m WIDE Tokyo

B USC-ISI Marina del Rey, CA
L ICANN Los Angeles, CA

Caching in DNS

- Server always caches answers
- Host can cache answers
- Caching
  - Improves efficiency
  - Eliminates unnecessary search
  - Works well because high locality of reference

DNS Types

- Each entry in server consists of
  - Domain name
  - DNS type for name
  - Value to which name corresponds
- During lookup, client must supply
  - Name
  - Type
- Server
  - Matches both name and type

The Point About Types

The domain name system stores a type with each entry. When a resolver looks up a name, the resolver must specify the type that is desired; a DNS server returns only entries that match the specified type.

Example DNS Types

- Type A (Address)
  - Value is IP address for named computer
- Type MX (Mail eXchanger)
  - Value is IP address of computer with mail server for name
- Type CNAME (Computer NAME)
  - Value is another domain name
  - Used to establish alias (www)

Domain Name Abbreviation

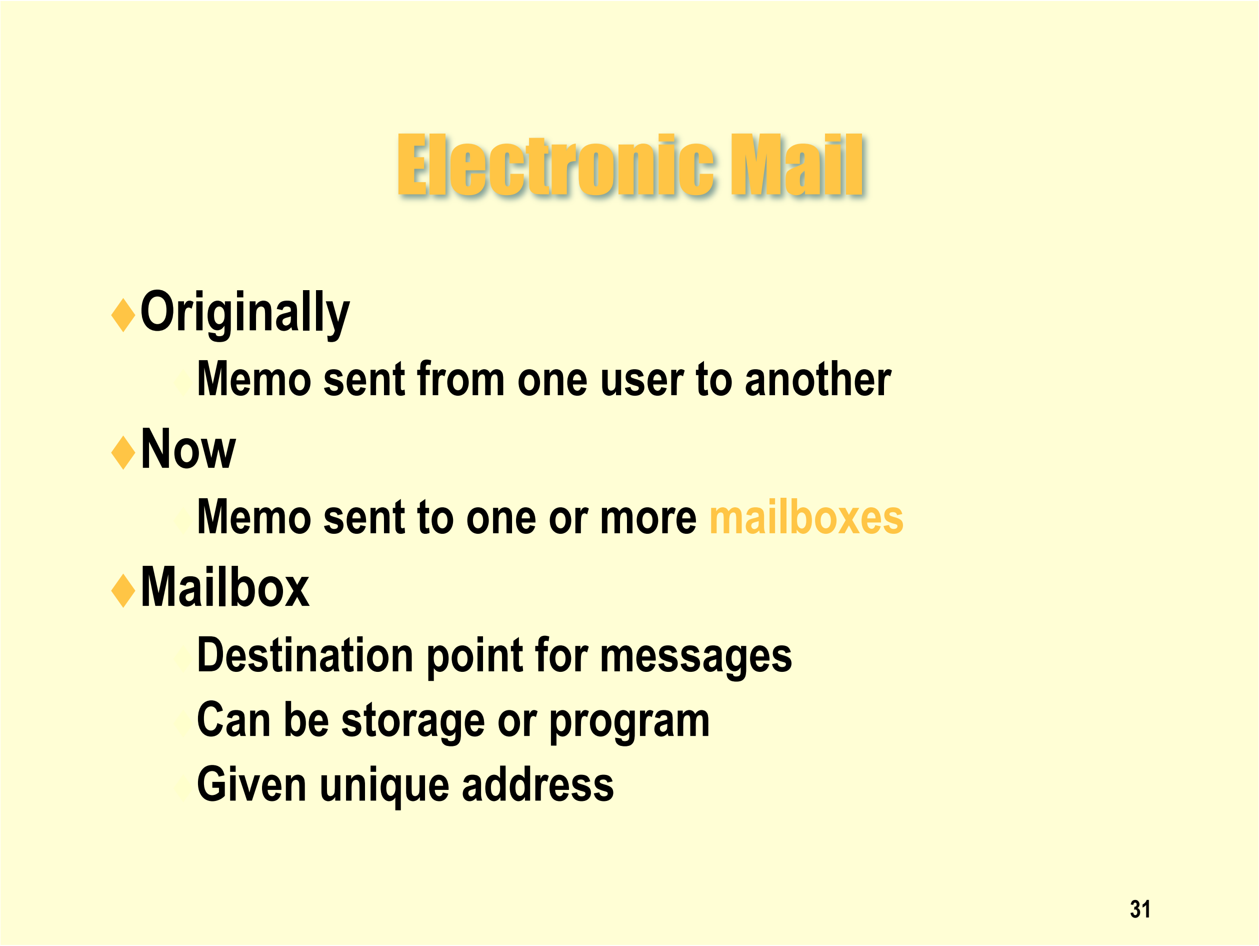
- DNS lookup uses full names
- Users desire abbreviations
- Technique
  - Configure resolver with list of suffixes
  - Try suffixes one at a time

Example of DNS Abbreviations

- Suffixes are
  - foobar.com
  - candy.foobar.com
- User enters name walnut
- Resolver tries
  - walnut
  - walnut.foobar.com
  - walnut.candy.foobar.com

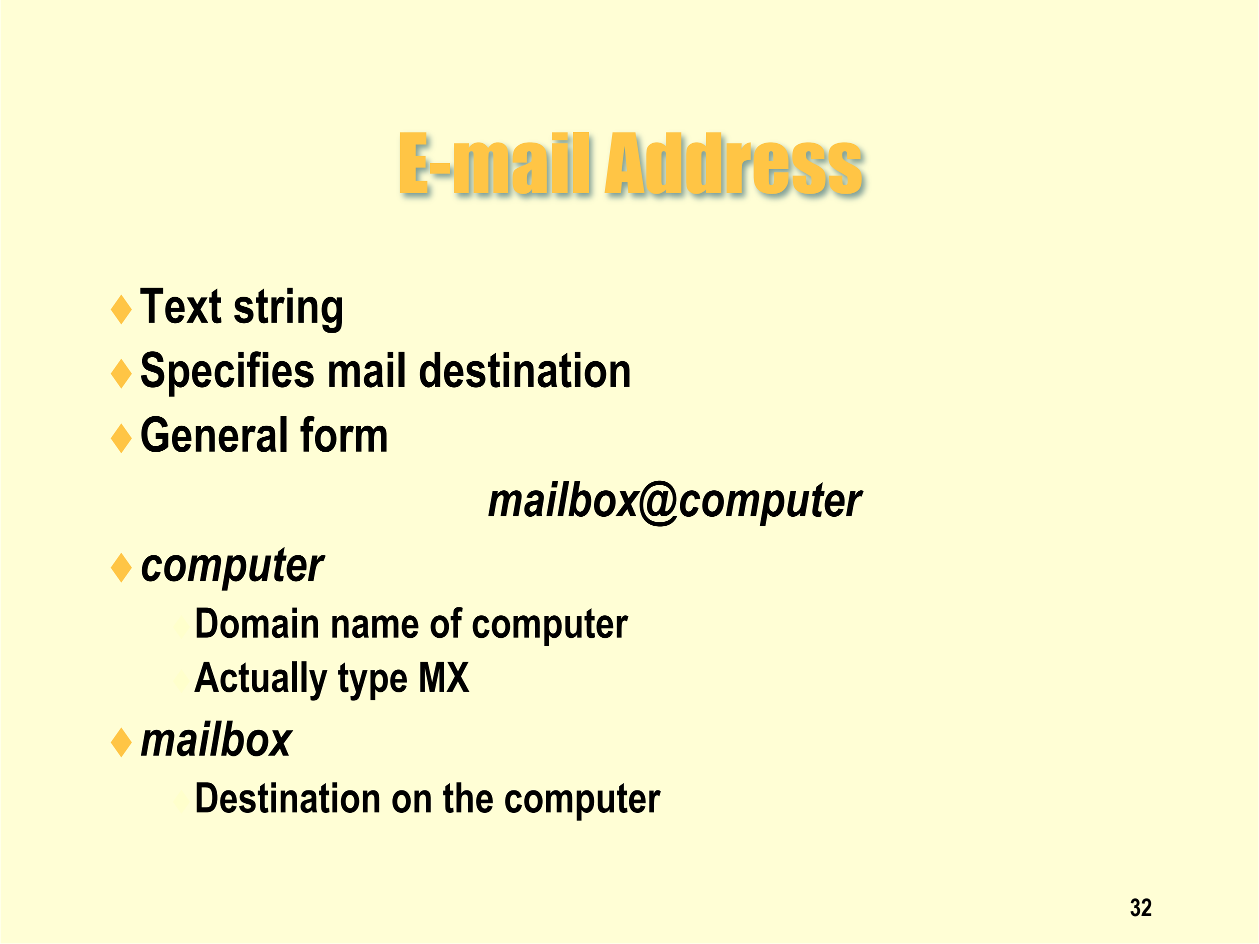
Other Internet Applications

- Invoked directly by user
  - E-mail
  - Remote login
  - File transfer
  - Web browsing



Electronic Mail

- Originally
  - Memo sent from one user to another
- Now
  - Memo sent to one or more mailboxes
- Mailbox
  - Destination point for messages
  - Can be storage or program
  - Given unique address



E-mail Address

- Text string
- Specifies mail destination
- General form

mailbox@computer

- *computer*
  - Domain name of computer
  - Actually type MX
- *mailbox*
  - Destination on the computer

Use of E-mail Address

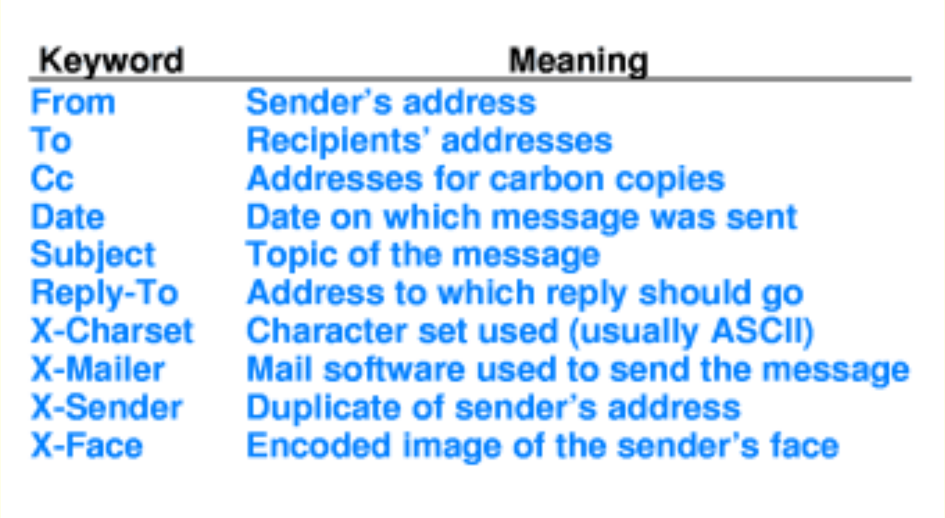
Each electronic mailbox has a unique address, which is divided into two parts: the first identifies a user's mailbox, and the second identifies a computer on which the mailbox resides. E-mail software on the sender's computer uses the second part to select a destination; e-mail software on the recipient's computer uses the first part to select a particular mailbox.

Mail Message Format

- Header
  - Identifies sender, recipient(s), memo contents
  - Lines of form

    keyword:information
- Blank line
- Body
  - Contains text of message

Example E-mail Header Fields



Keyword	Meaning
From	Sender's address
To	Recipients' addresses
Cc	Addresses for carbon copies
Date	Date on which message was sent
Subject	Topic of the message
Reply-To	Address to which reply should go
X-Charset	Character set used (usually ASCII)
X-Mailer	Mail software used to send the message
X-Sender	Duplicate of sender's address
X-Face	Encoded image of the sender's face

- Most header lines optional

Extending E-mail

- Original e-mail
  - SMTP - message restricted to ASCII text
- Users desire to send
  - Image files
  - Audio clips
  - Compiled (binary) programs
- Solution
  - Multi-purpose Internet Mail Extensions (MIME)

MIME

- Allows transmission of
  - Binary data
  - Multimedia files (video/audio clips)
  - Multiple types in single message
  - Mixed formats
- Backward compatible

MIME Encoding

- Sender
  - Inserts additional header lines
  - Encodes binary data in (printable) ASCII
- Sent like standard message
- Receiver
  - Interprets header lines
  - Extracts and decodes parts
- Separate standards for content and encoding

Example of MIME

- Header lines added

```
MIME-Version: 1.0
Content-Type: Multipart/Mixed; Boundary=Mime_sep
```

- Specifies
  - Using MIME version 1.0
  - Line *Mime_sep* appears before each message part

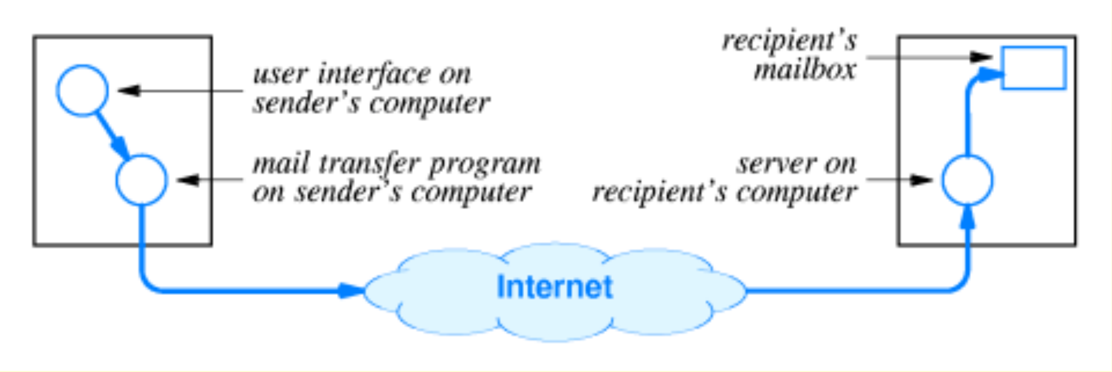
MIME

Although Internet e-mail only transfers text, MIME can be used to transport binary data by encoding it in printed characters. A MIME mail message includes additional information that a receiving application uses to decode the message.

Mail Transfer

- Protocol is Simple Mail Transfer Protocol (SMTP)
- Runs over TCP
- Used between
  - Mail transfer program on sender' s computer
  - Mail server on recipient' s computer
- Specifies how
  - Client interacts with server
  - Recipients specified
  - Message is transferred

Illustration of Mail Transfer

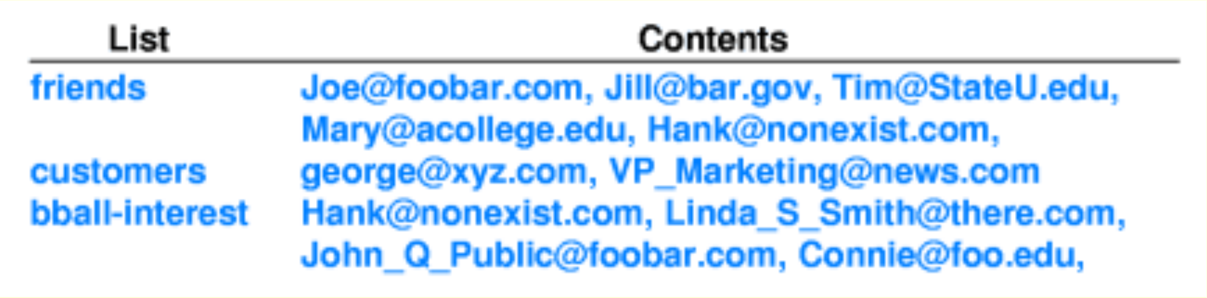

user interface on sender's computer
mail transfer program on sender's computer
recipient's mailbox
server on recipient's computer
Internet

- Server
  - Required to receive mail
  - Places message in user' s mailbox

Terminology

- Mail exploder
  - Program
  - Accepts incoming message
  - Delivers to multiple recipients
- Mailing list
  - Database
  - Used by exploder
- Mail gateway
  - Connects two mail systems

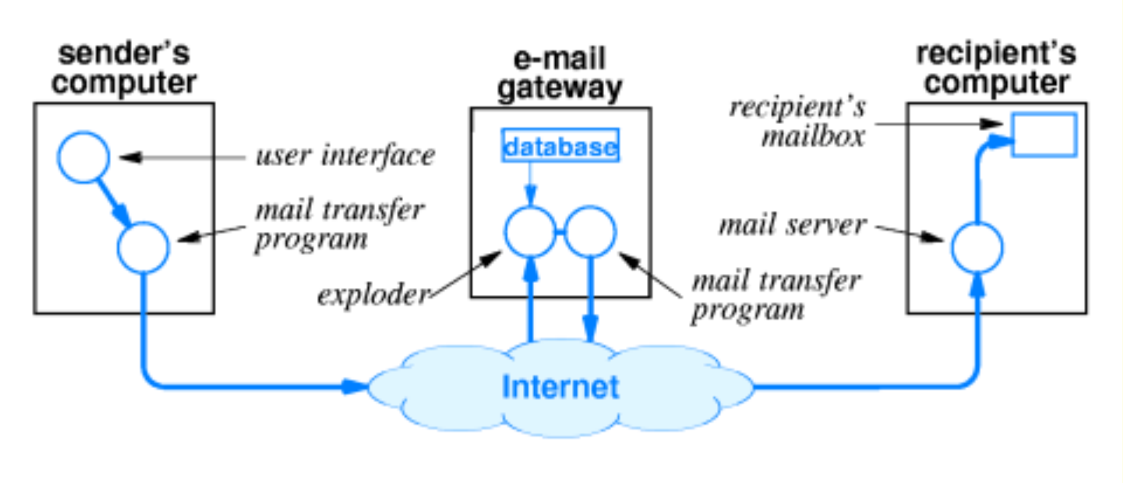
Illustration of a Mailing List



List	Contents
friends	Joe@foobar.com, Jill@bar.gov, Tim@StateU.edu, Mary@acollege.edu, Hank@nonexist.com,
customers	george@xyz.com, VP_Marketing@news.com
bball-interest	Hank@nonexist.com, Linda_S_Smith@there.com, John_Q_Public@foobar.com, Connie@foo.edu,

- Separate permissions for
  - Mailing to list
  - Adding/deleting members
    - Public – anyone can join
    - Private – access restricted by owner

Illustration of a Mail Gateway



- Can connect two
  - Heterogeneous systems
  - Internet to non-Internet

Automated Mailing Lists

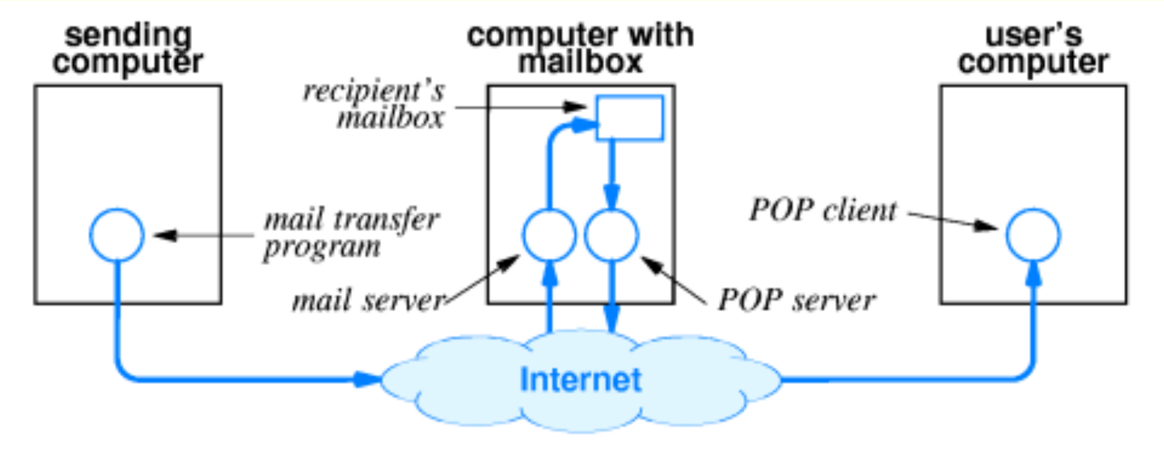
- Automated program to handle routine chores of maintaining mailing list: list manager
- Used in conjunction with exploder
- Example expected command:

  add *mailbox* to *list*

Computers Without Mail Servers

- Typically
  - Small, personal computer or mobile device
  - Not continuously connected to Internet
- To receive e-mail, user must
  - Establish mailbox on large computer
  - Access mailbox as necessary
- Post Office Protocol (POP) or IMAP used

POP



- Current version named POP3

IMAP

- Internet Message Access Protocol (IMAP4 rev1 1996)
- A more recent protocol for accessing email messages
- Leaves mail on server
- Built in security features

Summary

- Domain Name System
  - Maps name to IP address
  - Uses on-line servers
  - Uses caching for efficiency
- Three e-mail transfer protocols
  - SMTP
  - POP3
  - IMAP

Remote Login

- Provide interactive access to computer from remote site
- Standard protocol is TELNET
- Secure version, SSH

TELNET

- Text-oriented interface
- User
  - Invokes client
  - Specifies remote computer
- Client
  - Forms TCP connection to server
  - Passes keystrokes over connection
  - Displays output on screen

File Transfer

- Complete file copy
- Major protocol is File Transfer Protocol (FTP)
  - Uses TCP
  - Supports binary or text transfers
  - Large set of commands
  - Until 1995 was major source of packets in Internet

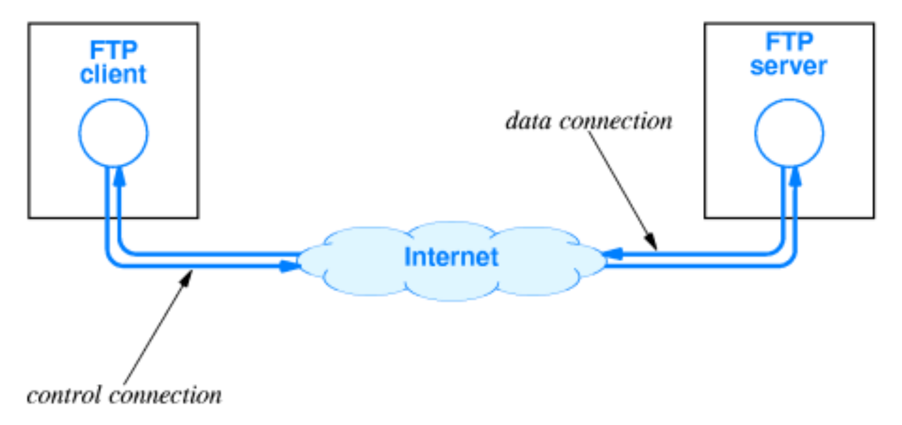
FTP Paradigm

- Command-line interface
- User
  - Forms TCP connection to server (called control connection)
  - Logs in
  - Enters commands to list directories, transfer files
- Server
  - Established new TCP connection for each transfer

Transfer Modes

FTP has two basic transfer modes: one used for text files and the other for all non-text files. Although binary mode produces an exact copy of the bits, the resulting copy may be meaningless because does not convert values to the local representations.

Illustration of TCP Connections During an FTP File Transfer



- Two TCP connections used

FTP Commands

- Over 50 different commands
- Both for local and remote side
- Only a small subset used today
- Examples: ascii, binary, cd, get, rmdir
- Today you often have a graphical user interface

TFTP

- Second file transfer service in TCP/IP: Trivial File Transfer Protocol (TFTP)
- Uses UDP instead of TCP
- Only supports file transfer
- Useful for bootstrapping a hardware device that has no disk for system software

Network File System

- File transfer not needed for all data transfers
- File access service allows remote clients to copy or change small parts of file
- One file access mechanism used with TCP/IP is Network File System (NSF)
  - Allows client to copy or change pieces of file
  - Allows shared file access
  - Integrated into computer’s file system

Summary

- Remote login
  - Remote, interactive use
  - Protocol is TELNET
- File transfer
  - Copy of entire file
  - Protocol is FTP

Extra material

RPC and Middleware

Middleware

- ♦Tools to help programmers
- ♦Makes client-server programming
  - Easier
  - Faster
- ♦Makes resulting software
  - Less error-prone
  - More reliable

Middleware Approach

- Allow programmer to work with familiar language constructs
- Provide tools to help programmer
  - Special translators
  - Libraries
- Automatically generate code for
  - Network communication
  - Connection management

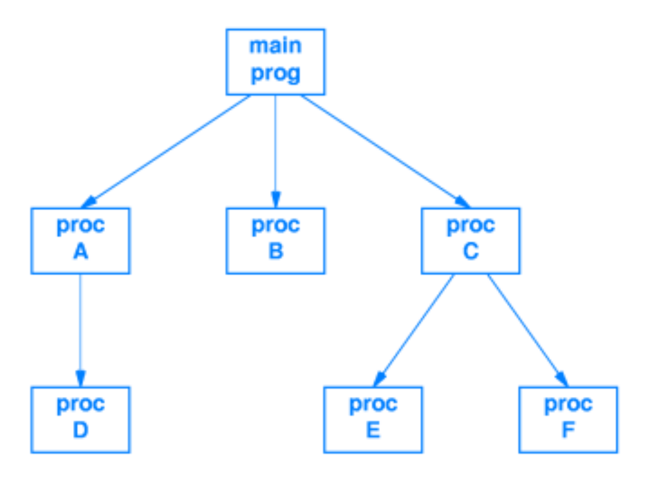
Remote Procedure Call

- Uses standard procedure call paradigm
- Divides program along procedure call boundaries
  - Main program and procedures for user interaction in client side
  - Other procedures in server side

Reason for Remote Procedure Call

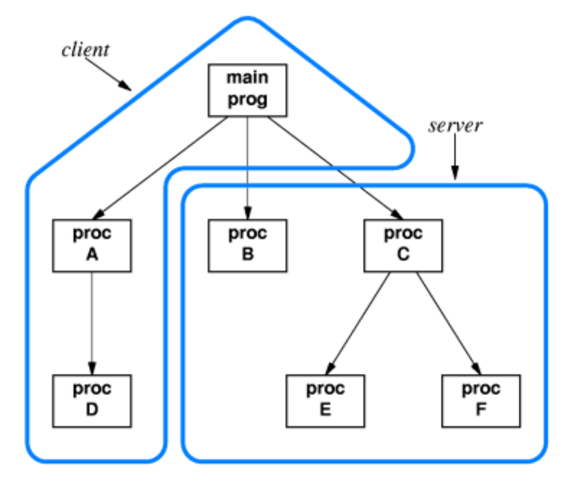
If a programmer follows the same procedure call paradigm used to build conventional programs when building client and server software, the programmer will find the task easier and will make fewer mistakes

Illustration of Conventional Procedure Call Graph



- ♦Arrow denotes procedure call

Procedure Call Graph Divided Into Client and Server

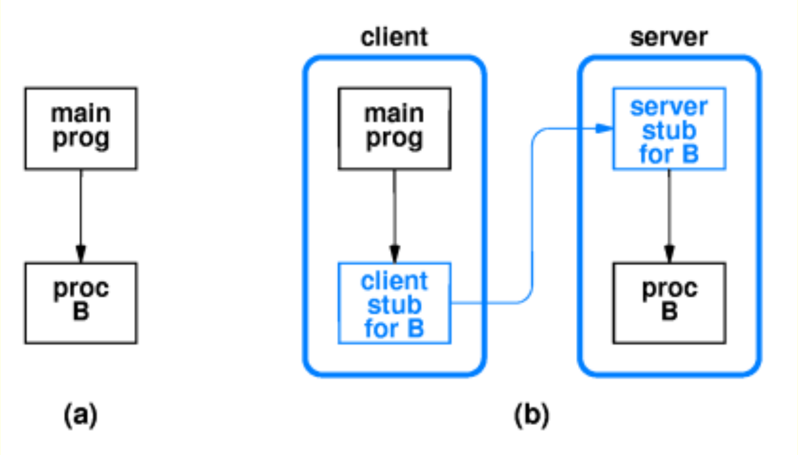


- Division occurs on call boundary
- Main program in client piece

Communication Stubs

- Inserted to enable remote "call"
- Automatically generated
- Use original call interface
- Allow calling and called procedure to remain unchanged

Illustration of Client and Server Stubs



- Original call in (a)
- Same interface with stubs in (b)

Creating Stubs

- Programmer writes
  - Code for a program
  - Specification of procedure interfaces using Interface Definition Language (IDL)
- Middleware generates
  - Client and server stub code
  - Necessary socket calls
  - Data translation

Problem with different call types

- Two ways to send arguments in a procedure call:
  - Call-by-reference
  - Call-by-value
- Call-by-reference is tricky to handle since the data is residing in local memory. RPC therefore mostly uses call-by-value.

Data Representation

- Network can connect heterogeneous computers
- Two computers may use different
  - Integer representations
  - Character codes
  - Floating point representations
- Translation required

Possible Data Translation Schemes

- Use receiver’ s representation
  - Sender translates all outgoing data
- Use sender’ s representation
  - Receiver translates all incoming data
- Use external representation (popular)
  - Sender translates to external form before sending
  - Receiver translates from external form after reception
  - Standard exist, ASN.1

ASN.1 Abstract Syntax Notation One

- ASN.1 is a standard and flexible notation that describes data structures for representing, encoding, transmitting, and decoding data
- It provides a set of formal rules for describing the structure of objects that are independent of machine-specific encoding techniques and is a precise, formal notation that removes ambiguities

Middleware Technologies That Use Remote Procedure Call

- ONC RPC
  - Open Network Computing
  - IETF standard
  - Popular in Unix world
- DCE RPC
  - Distributed Computing Environment
  - Open Group Standard

Middleware Technologies That Use Remote Procedure Call (continued)

- MSRPC
  - Microsoft
  - Variant of DCE RPC

Object-Oriented Middleware

- Designed for use with object-oriented programming languages
- Same general scheme as RPC
  - Interface Definition Language
  - Tool to build stubs
  - Libraries to handle network communication
- Uses method invocation instead of procedure call

Middleware Technologies That Use Remote Object Invocation

- CORBA
  - Common Object Request Broker Architecture
  - Well known object-oriented middleware
- MSRPC2
  - Microsoft
  - Also called Object RPC (ORPC)

Middleware Technologies That Use Remote Object Invocation

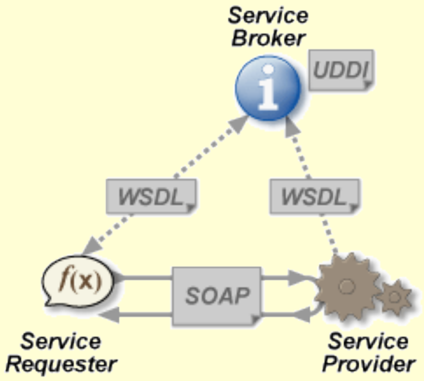
- COM / DCOM
  - Also from Microsoft
  - Component Object Model (COM)
    - Used on single computer
    - Provides mechanism for inter-object references
  - Distributed Component Object Model
    - Used across multiple computers
    - Includes communication stubs

Middleware Technologies That Use Remote Object Invocation

- RMI
  - From Sun Microsystems
  - Methods of remote Java objects can be invoked from other Java virtual machines on different hosts
- Jini
  - Also from Sun; based on Java
  - Now taken over by Apache (Project River)
  - Provides an environment for creating dynamically networked components, applications, and services that scale

Web Services

- The most common technology used today for remote processing or access of information is Web Services
- A set of several protocols handling different aspects of the total problem



Summary

- Middleware
  - Tools to help build client and server
  - Automates routine tasks
  - Two popular paradigms
    - Remote procedure call
    - Object invocation
  - Generates communication stubs