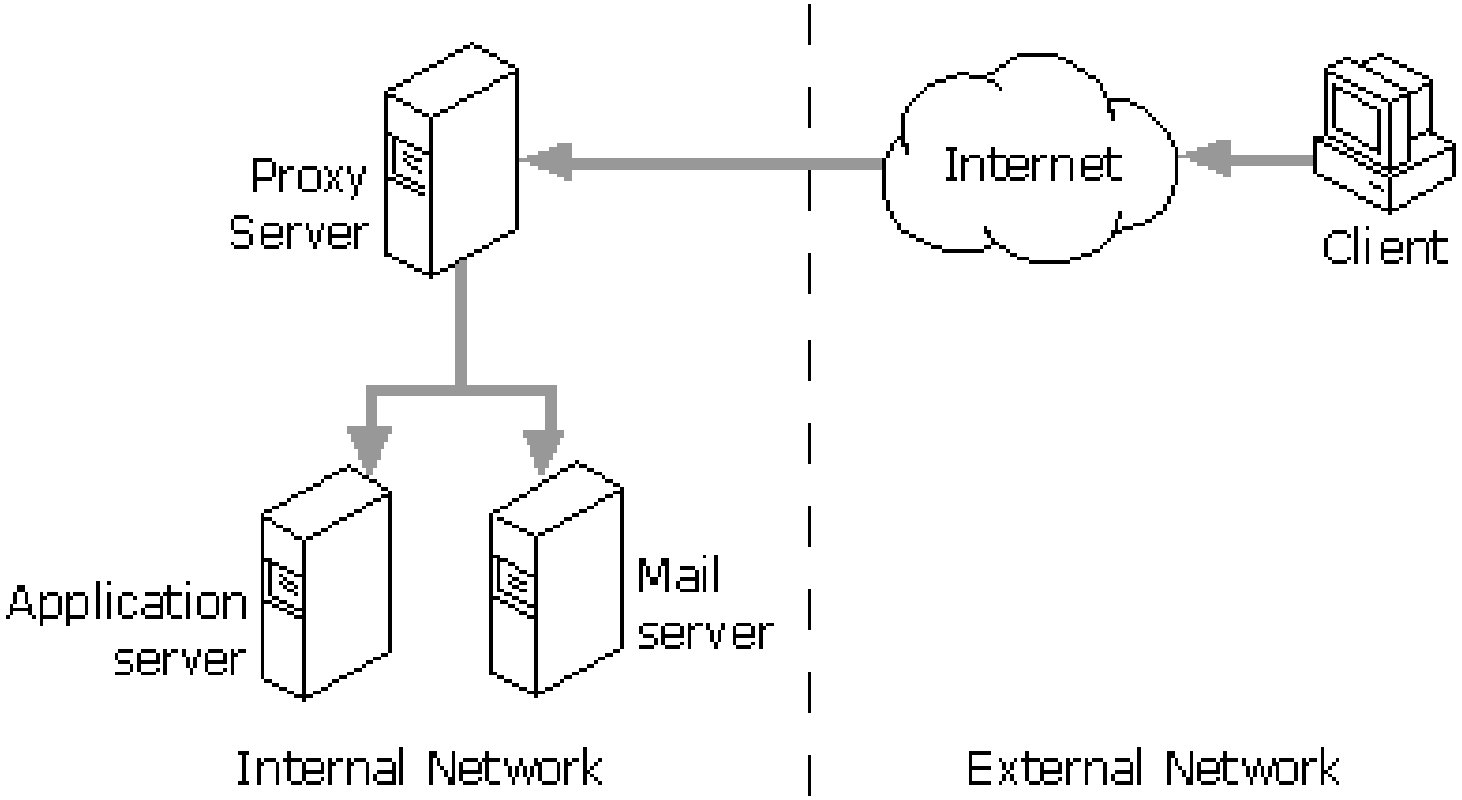
Ch 4 – Web Communication

- Domain Name Server
- HTTP
- Caching
- Internet Information Server (IIS)
- Proxy Server

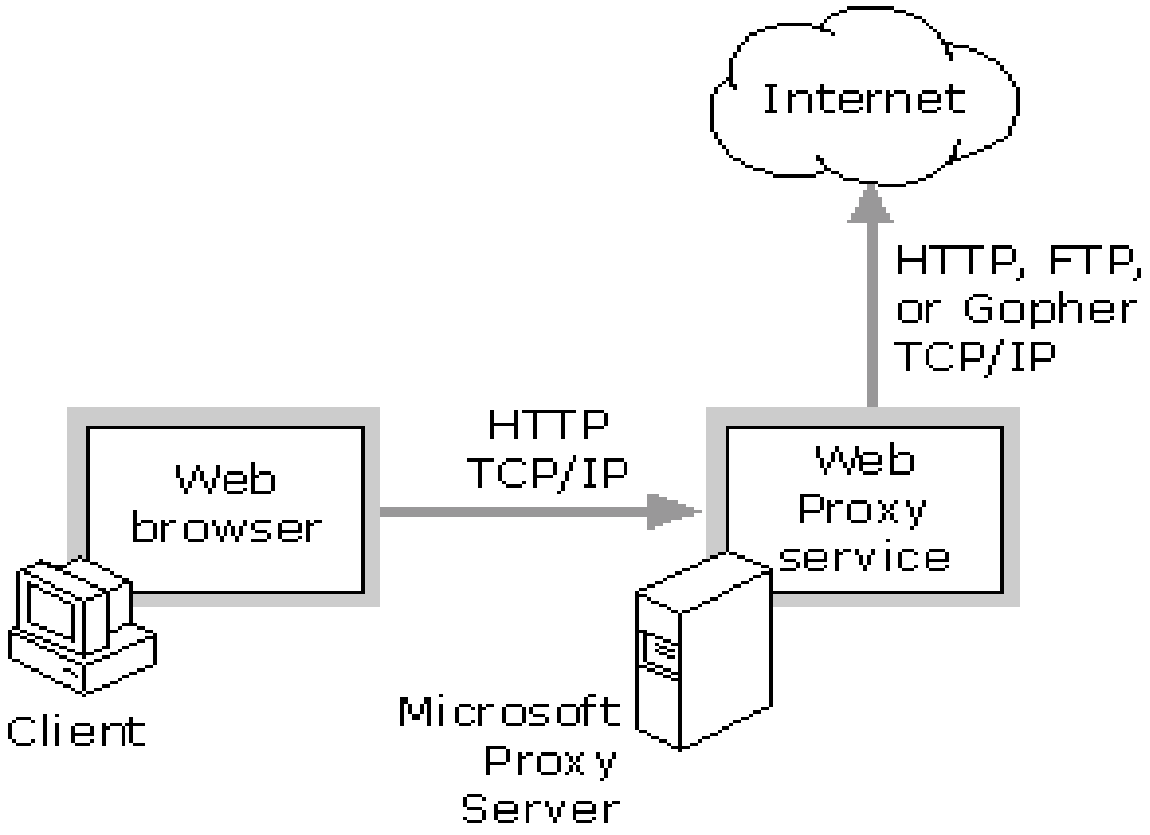
Overview of HTTP

- HTTP is a request/response protocol for client/server communication.
- The client is an application program in the user agent, to the end user (human) is connected.
- The original server is an application program which process the request from the client, and return the result to the client.
- Three types of intermediaries in the request/response chain:
  - tunnel (not discussed)
  - gateway (not discussed)
  - proxy server

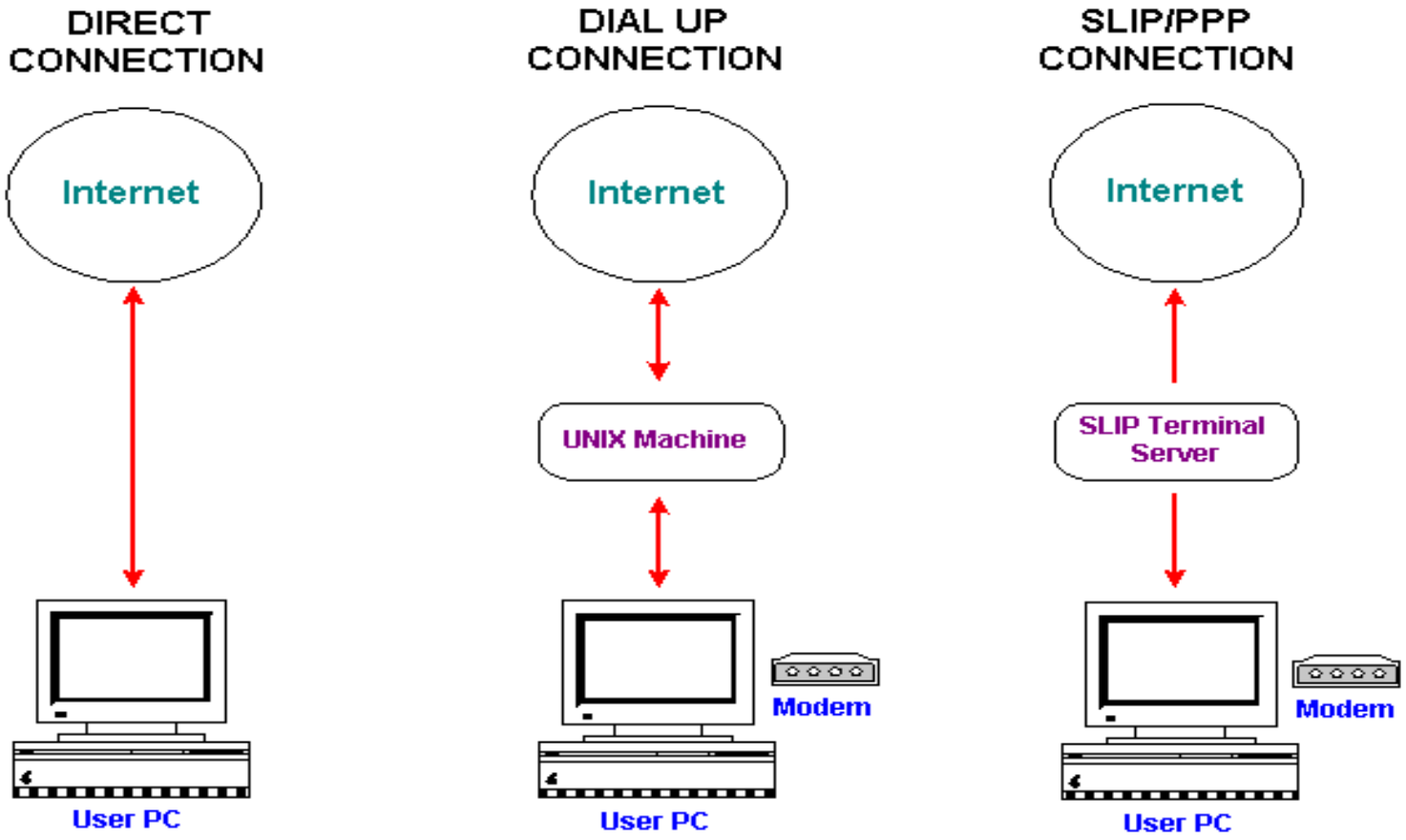
Proxy Server (inward bound)



Proxy Server (outward bound)



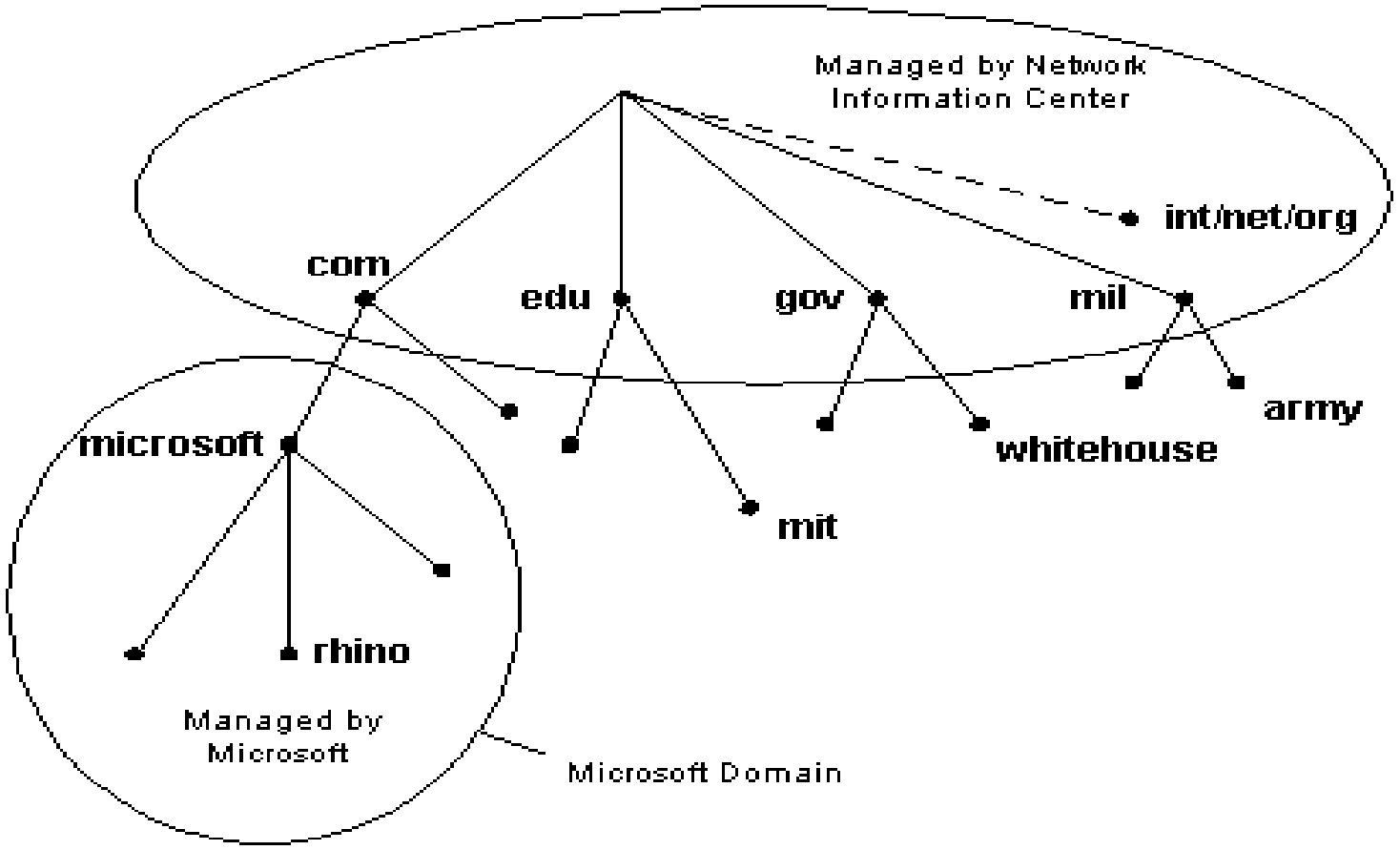
User / User Agents (client)



Domain Name System

- Each internet host must be given either an IP or a host name which may be translated into an IP, by means of a domain name system (DNS).
- A host is named following an internet standard.
- A DNS is a set of protocols and services on a TCP/IP network. It is separate from HTTP.

DNS Domains and Subdomains



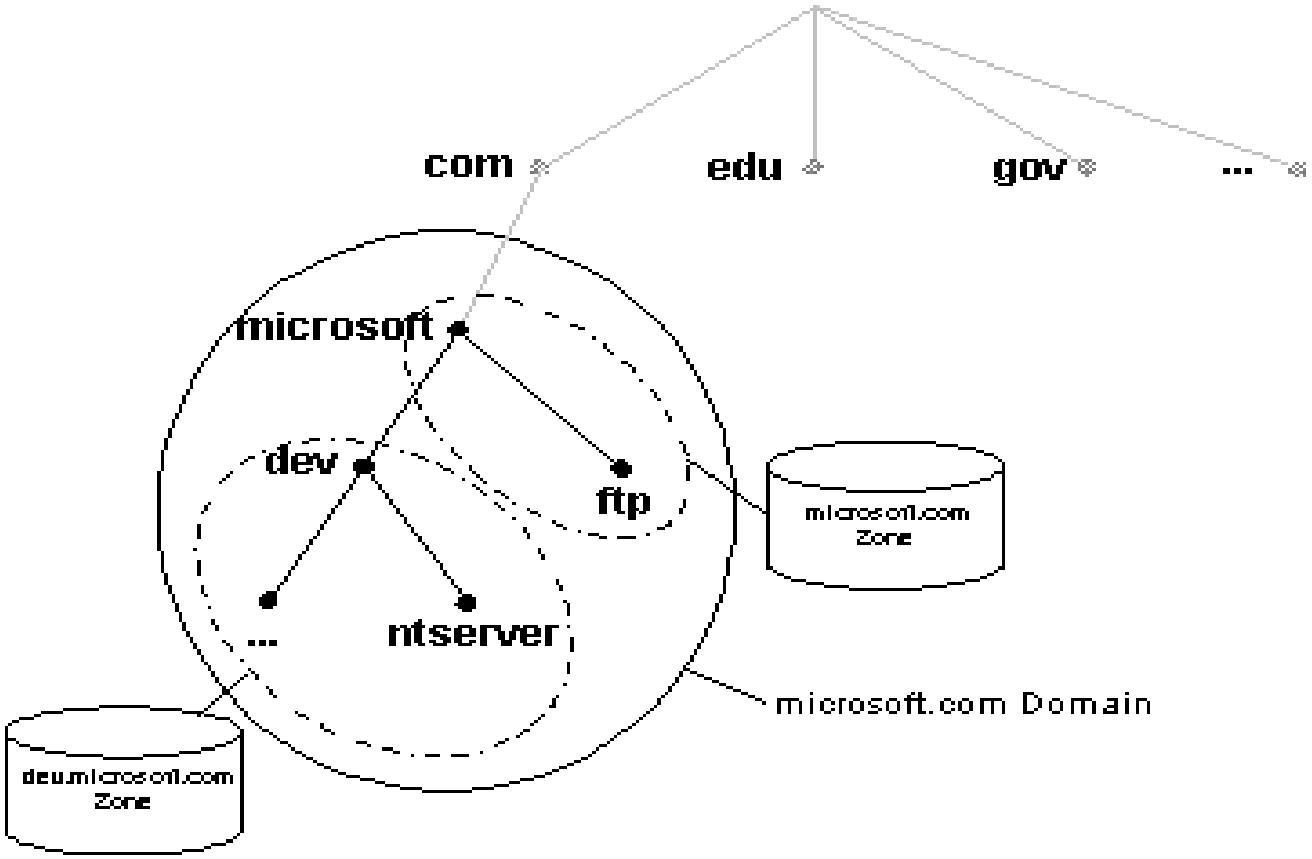
DNS Namespace

- The names in the DNS database is a logical tree structure.
- Domain is a node in the tree along with all its descendants.
- The logical tree structure is implemented by a zone system.
- Each zone is some portion of the DNS server whose database record exist and are managed in a particular zone file.

Zones

- Zone is anchored at a specific domain (node), but it does not necessarily contain all the subdomains.
- A DNS Server is a host which is responsible for one or more zones.
- Primary name server is the one gets the data for its zones from local files. It may have a number of secondary name servers, which will provide more redundancy, share the load, and offer services in remote locations.

Domain with Zones



Name<->IP Resolution

- Specific DNS Name Servers are designated as forwarders, which are allowed to communicate with other forwarders on Internet to resolve a specific DNS request.
- **`PING`** is the command which requests an IP address, given its DNS name.
- An IP address may be translated into a DNS name by a DNS name server.

HTTP History

- Version 0.9: developed at CERN in early 90s.
- Version 1.0: developed by IETF; first released in August, 1994;
- Version 1.1: developed by IETF, first released in January 1997 (latest: RFC2616)
- Many others that are relevant, i.e.
  - HTTP State Management Mechanism (cookie) (RFC2109)
  - Distributed Authoring and Versioning Protocol for WWW (RFC2291)

Components of HTTP Message

- (message type) Request / Response
- Request-line / Status-line
  - method (e.g. POST/GET), request-URI, and HTTP version
  - http version, status-code, reason-phrase
- General header
- Request-header / Response-header
- Entity header
- Message body

Topics Addressed by HTTP Headers

- Message: age, date, length
- Content: (natural) languages, encoding methods, type, multi-part message
- User-agent: email address, host name, product ID
- Server: product ID
- Transmission Management: discussed later
- Caching: discussed later; also in other chapters
- Authorization/Security: discussed in the Security chapter

Transmission Management

- Re-direction:
  - The resource requested is available as another URI instead.
  - The redirection may be taken without informing the client.
- Persistent connection (later)
- Chunked-encoding (later)

Persistent Connection

- The default option of HTTP/1.1
- The TCP/IP connection is open between the client and server for multiple transmissions
- Main benefits:
  - Reduced overhead in connection management
  - Pipelining of requests: multiple requests without waiting for each response.

Chunked Transfer Coding

- HTTP does not have message length limitation.
- A message may be modified so that it is transferred as a series of chunks, each with its own size indicator, ending with the last chunk of size 0.
- Benefits:
  - pipelining: transfer parts of a message before the end of generation of the entire message.
  - increased throughput

Caching: Introduction

- Caching is one of the most important means to improve system performance.
- The proxy, server and client may keep the previously generated/received responses in its cache.
- The proxy may provide its response (non-first-hand) from its cache as the response to a request.
- In HTTP, caching will:
  - eliminate the need to send requests in many cases (through expiration mechanism)
  - eliminate the need to send full responses in many other cases (through validation mechanism)

Expiration Model

- Expiration time in the future, indicating that a response may be used to satisfy subsequent requests:
  - explicitly provided by the server, by means of a header field `expires`
  - In absence of the above, estimated by the client with info available in the headers, e.g. `age` and `last-modified.`

Validation

- A cache may check with the origin server to see if a cached entry is still usable.
- A validator (for the cached entry) is sent to the server for validation.
- Two forms of cache validator:
  - the URI of the resource and one of the following values in a header field:
    - If-modified-Since (a date)
    - Last-modified (a date)
  - Entity Tag and one of the following values in a header filed:
    - If-matched
    - If-not-matched

Cache-Directive

- This is a field in the general header of a HTTP message.
- This will be used by client/server to override the default cacheability.
- Examples:
  - no-store
  - expires (a date)
  - must-revalidate
  - private

Caching in IE

- A directory in the login user's profile, `Temporary Internet files` is designated as the (private) cache.
- A user may modify the caching behavior of the browser by specifying how the cache may be updated.
- In the HTML document, use the <META> tag:
  - `<META HTTP-EQUIV="REFRESH" CONTENT=2>`
  - `<META HTTP-EQUIV="EXPIRES" CONTENT="20 AUG. 1996">`

HTTP & Server File Management

- A new protocol was recently approved to provide additional commands in HTTP for remote file management from the browser.
- Examples:
  - PUT (file creation)
  - MKCOL (directory creation)
  - LOCK, UNLOCK
  - COPY, MOVE, SEARCH
- The web-folder feature in IE5 provides a drag-and-drop interface to these commands.

Improving HTTP/1.1

- Complexity
- Poor extensibility
- No application deployment model (remote program initiation, or RPC)
- Alternative technologies
- Poor scalability

HTTP Servers – Past & Present

- CERN & NCSA HTTP Servers (1990)
- CGI standard for *gateways* programs (1994)
- NCSA Server => Apache Server (1995)
- Debut of Netscape Server (1995)
- Microsoft's IIS 2.0 (1996), 3.0 (1997), 4.0 (1998)

IIS Components

- Administrative modules
- Microsoft management console (MMC) & *snap-in*
  - Index Server
  - Site Server
  - Microsoft Transaction Server
  - Microsoft Message Queue Server
  - Proxy Server
- IIS Services

IIS Administration

- Metabase: this is similar to system registry, but is used by only IIS to record the configuration of IIS.
- IIS Admin Objects: These objects are used by the IIS administrators to install, and maintain the IIS.

IIS Services

- WWW Services
  - CGI
  - ISAPI
  - HTMP
- Other services
  - Internet Mail (Simple Mail Transfer Protocol)
  - NNCP (Network News Control Protocol)
  - FTP
  - Java Virtual Machine

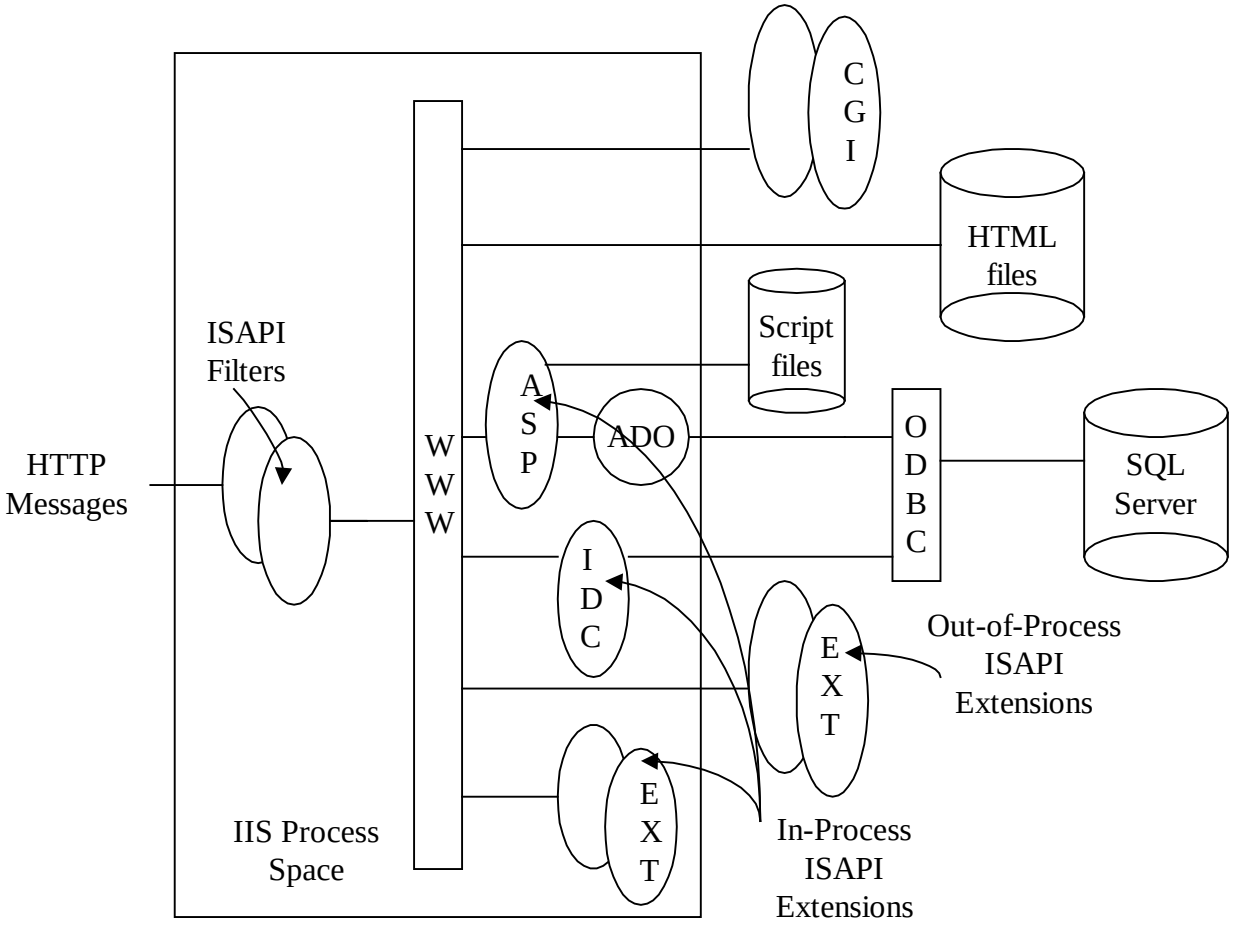
Processing a HTTP Request

1. IIS reads the raw request data.
2. IIS processes the HTTP request headers.
3. IIS maps the URL in the form of http://server/document.htm to a physical path on the machine like d:/inetpub/wwwroot/document.htm.
4. The request may include user name and password information that requires authentication by IIS.
5. If the client sends POST data with its request, the server starts to read the data in chunks.

Processing a HTTP Request (cont'd)

6. IIS and potentially a server application send HTTP headers to the client.
7. The server sends the response data to the client.
8. The server ends processing of the request. Note that the session still may be kept open.
9. The server writes data about the request to the log.
10. The server session is closed.

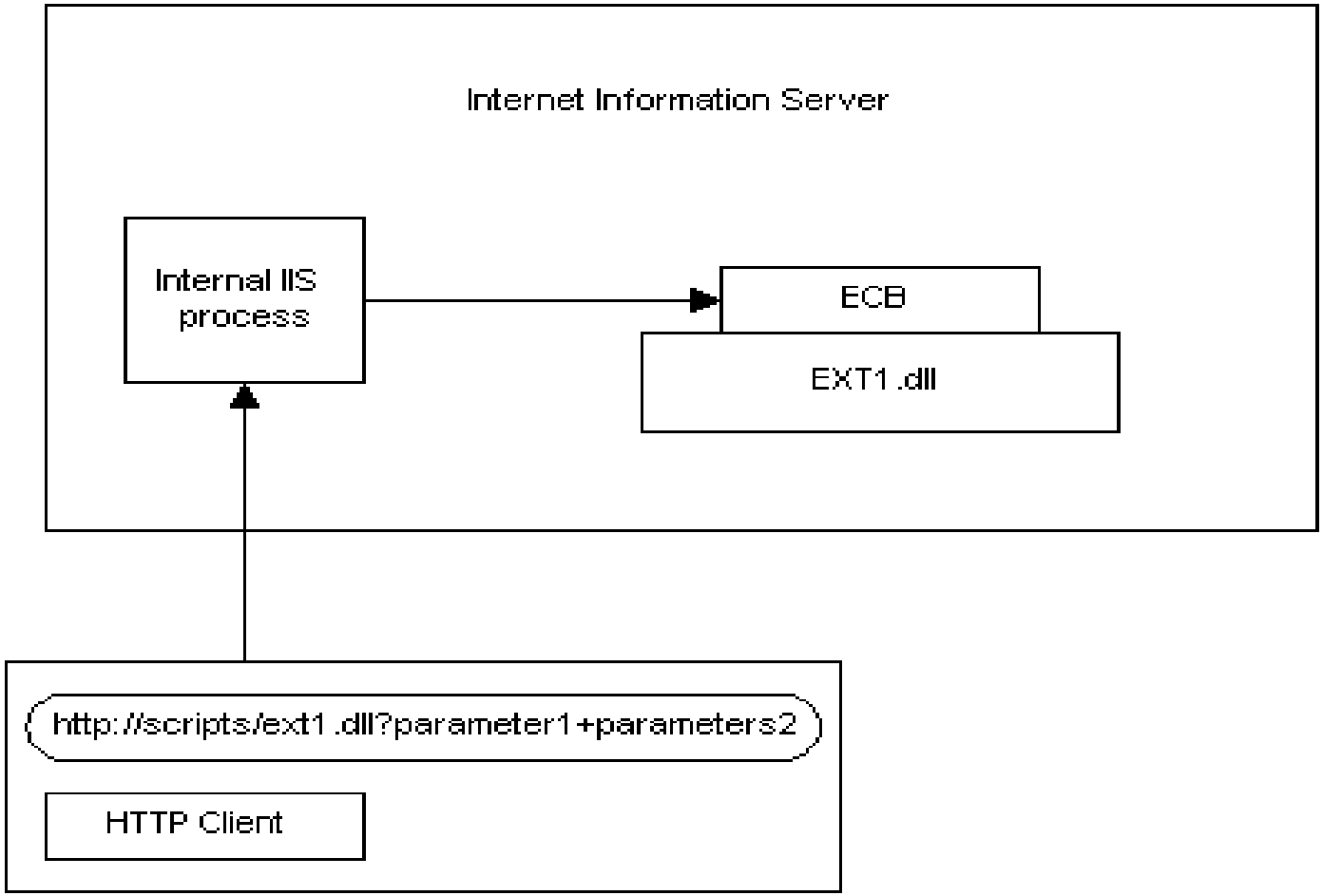
Internet Information Server (IIS)



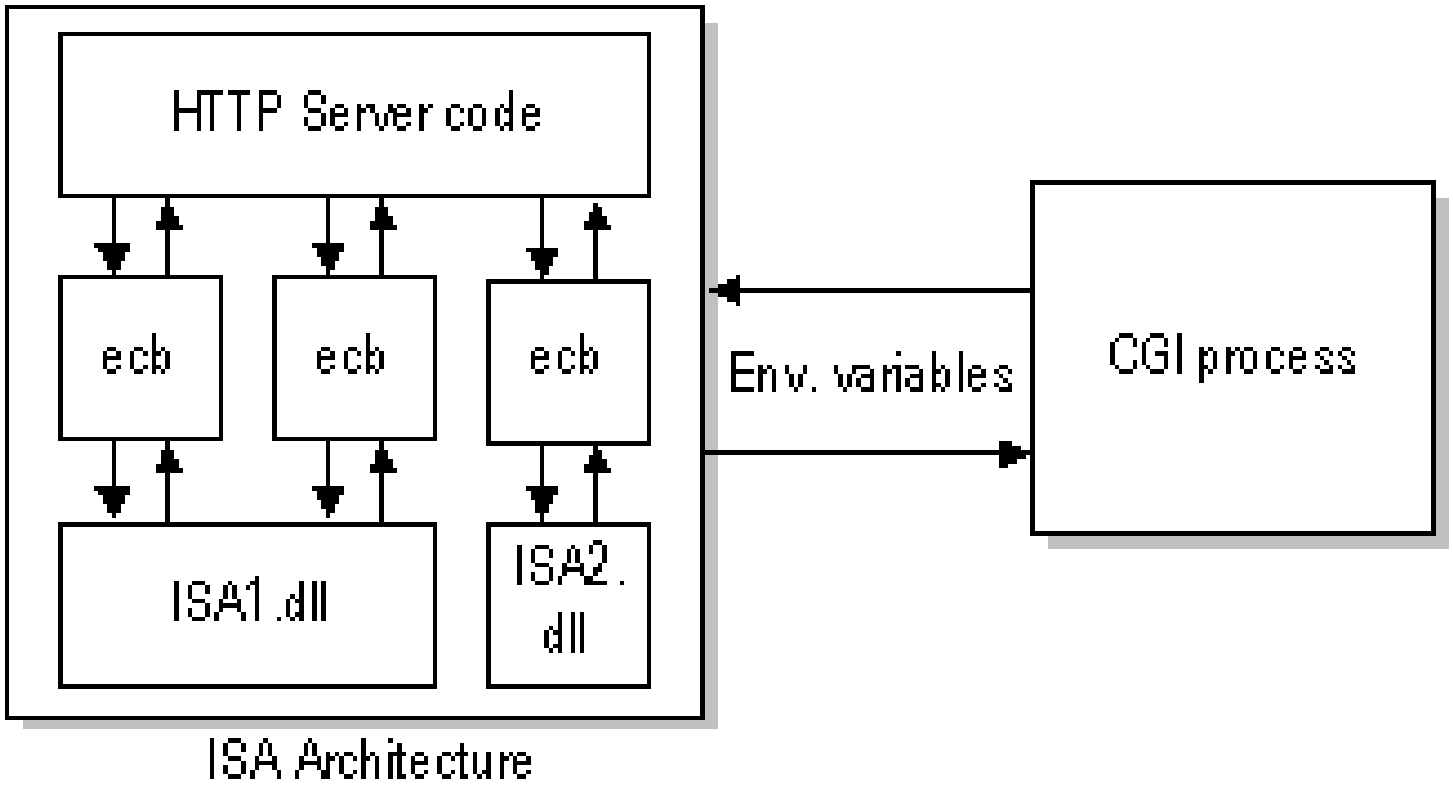
ISAPI

- ISAPI is an interface for external modules to link up to IIS.
- CGI is interfaced to IIS through a special interface (set of environmental variables).
- CGI was designed for UNIX environment.
- ISAPI Extensions & Filters are means to enhance the capability of IIS.
- ECB (External Control Block) contains a set of variables similar to environmental variables.

In-Process ISAPI Extension



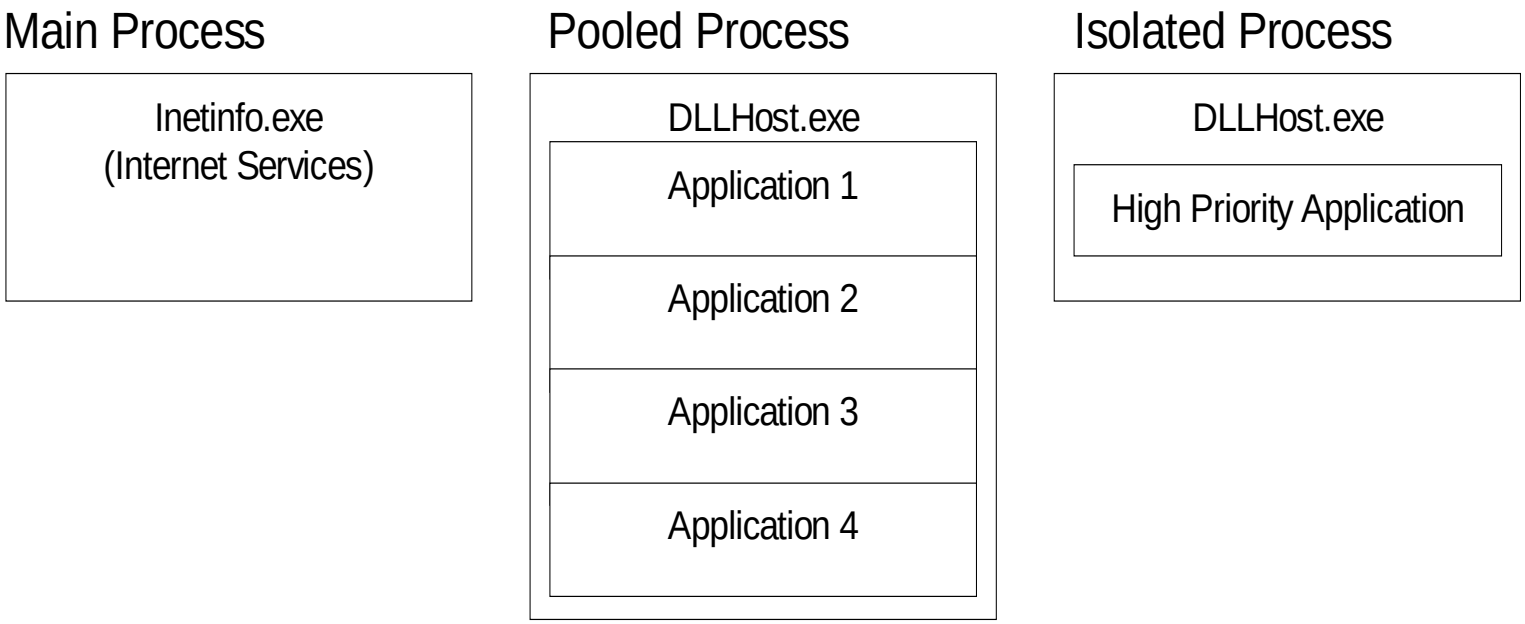
CGI & IIS


HTTP Server code
ecb
ecb
ecb
ISA1.dll
ISA2.
dll
ISA Architecture
Env. variables
CGI process

In-Process vs. Out-of-Process

- Performance
- State Information
- Stability
- Programmability
- Flexibility

IIS Process Isolation



Web Application Namespace

- A web application is a collection of HTML pages, server extensions (e.g. ASP and ISAPI) and ActiveX server components.
- These resources may be named as a set of URL's in a tree structure, the namespace.
- The root node is called the application starting point, which is either a home directory, or virtual directory.
- A virtual directory maps into a local path of the server directory.

Application Boundary

- The application scope includes all items within the directory and the sub directories below.
- The exception to the rule above is that all subdirectories that are included in another applications excluded, that is, outside the application boundary.
- Application may run inside the memory space of IIS or in a separate process.

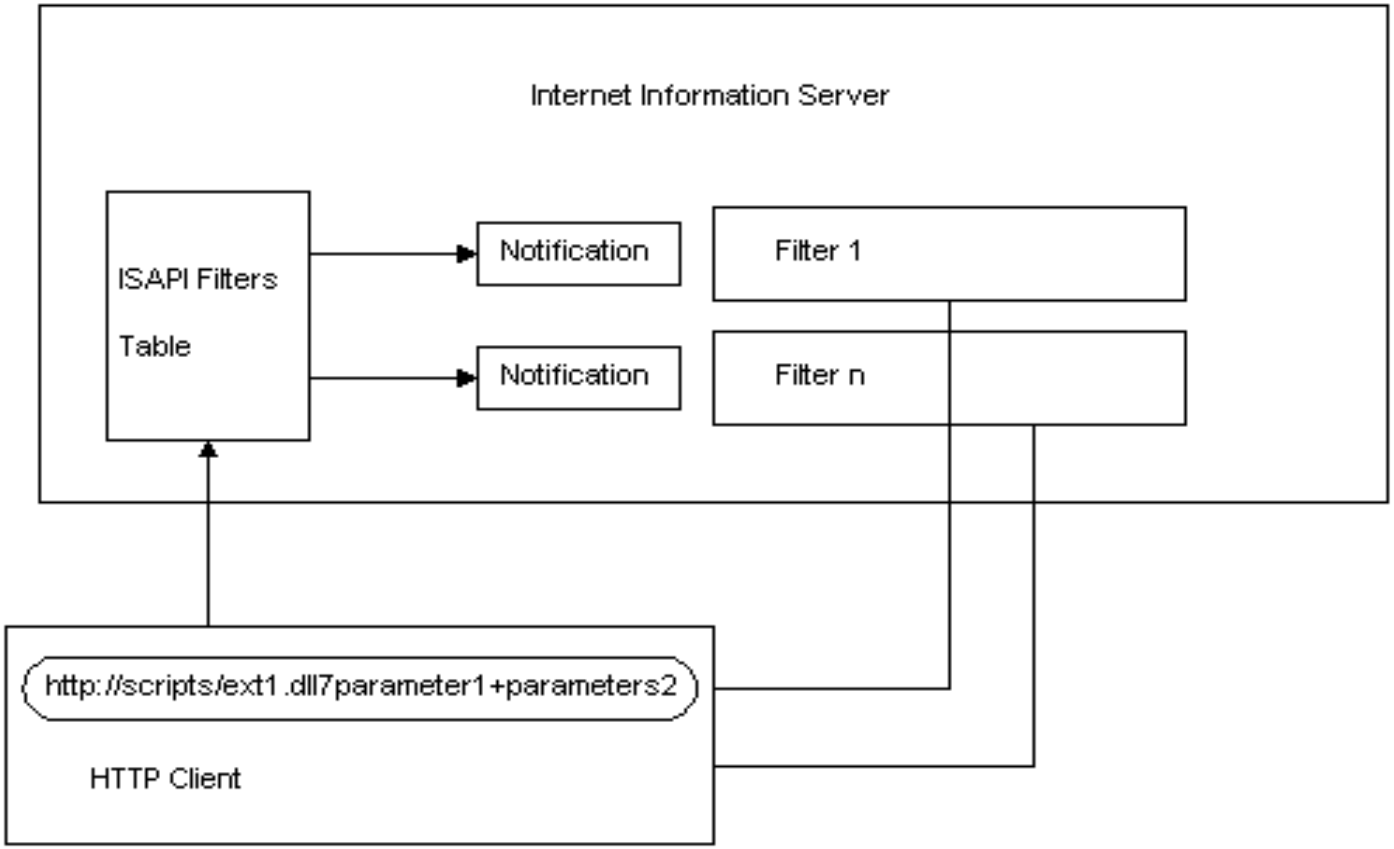
Web Application in an Isolated Process

- The application is written as an DLL
- An instance of COM object WAM (Web Application Manager) is created by IIS (inetinfo.exe), to run as a separate process.
- This WAM loads the DLL, and other relevant programs, such as asp.dll is loaded into the process of WAM.

ISAPI Filter Overview

- At every step of IIS processing (slides 30-31), the IIS might be interrupted if there are event handlers registered for that event.
- An ISAPI filter is an event handler, which may be a user-supplied DLL.
- Filters are able to examine/modify incoming and outgoing streams of data.
- An ISAPI filter is always loaded as long as the IIS is up.

ISAPI Filter Architecture



Why Filters?

- Customer authentication
- Compression
- Encryption
- Logging
- Traffic Analysis or other request analyses.

Overview of Proxy Server

- A proxy server (proxy) routes requests and responses between the Internet and computers on the internal network.
- Users on Internet and the internal network sees a combination of the proxy and origin server behavior
- Aside from authentication and security, users will get the same response whether the connection was direct, or through a proxy server.

Why is a Proxy not Combined in the Web Server?

- Enhanced security
- Ease of administration
- Modularization of development
- Marketing

History of Proxy Server

- At CERN, gateway was the term used to refer to devices that packet forwarding and occasionally, protocol conversion.
- It was renamed to Web Proxy Server to distinguish them from (information) gateways that are called CGI applications now.
- Proxy acts on behalf of clients, and CGI applications act on behalf of the server.

Performance Improvement Methods

- Use multiple proxies and/or origin servers.
  - Replication of contents, especially in geographically different sites.
  - Distribution of workload (contents in a shared database)
  - Distribution of cached contents among multiple proxies
- Caching: automatic, pervasive, and adaptive

Load Distribution

- Distribute load by contents does not work
- Distribute load randomly (load balancing)
  - Round-robin DNS
  - Re-direction (almost non-transparent)
  - Cache array routing protocol (CARP) (discussed later)

Round-Robin-based Load Balancing

- Proxy maps a single hostname to a multiple different physical server machines, with different IP addresses.
- A local DNS server in the proxy is set up so that IP addresses are given out as primary IP for the requested hostname on round-robin basis
- Problem: it may be a problem for multi-layer proxies.

Replication

- Mirror sites:
  - non transparent to users
- Replication (caching) of objects in geographically different sites.

Content Delivery System Akamai Tech Inc

- Akamai's services enable Web site owners to deliver their content with high-performance and reliability.
- Akamai runs a global server network of 2,000 servers, ensuring content is always close to users.
- Visitors to Akamai’s Web sites need no modification of browser software or plug-ins.

Akamai's Network of 2000 Servers

Network Operation Room

Figure 1: "Internet Content Delivery Without FreeFlow"

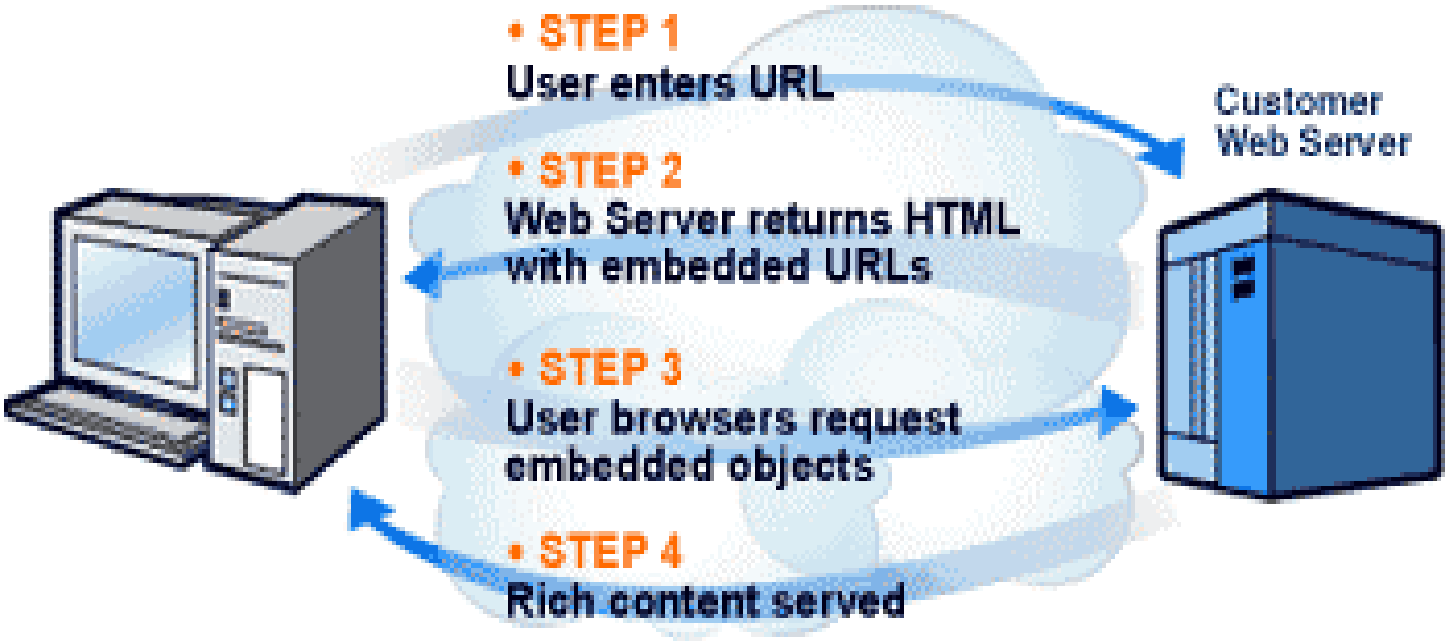


Figure 2: "Internet Content Delivery With FreeFlow"

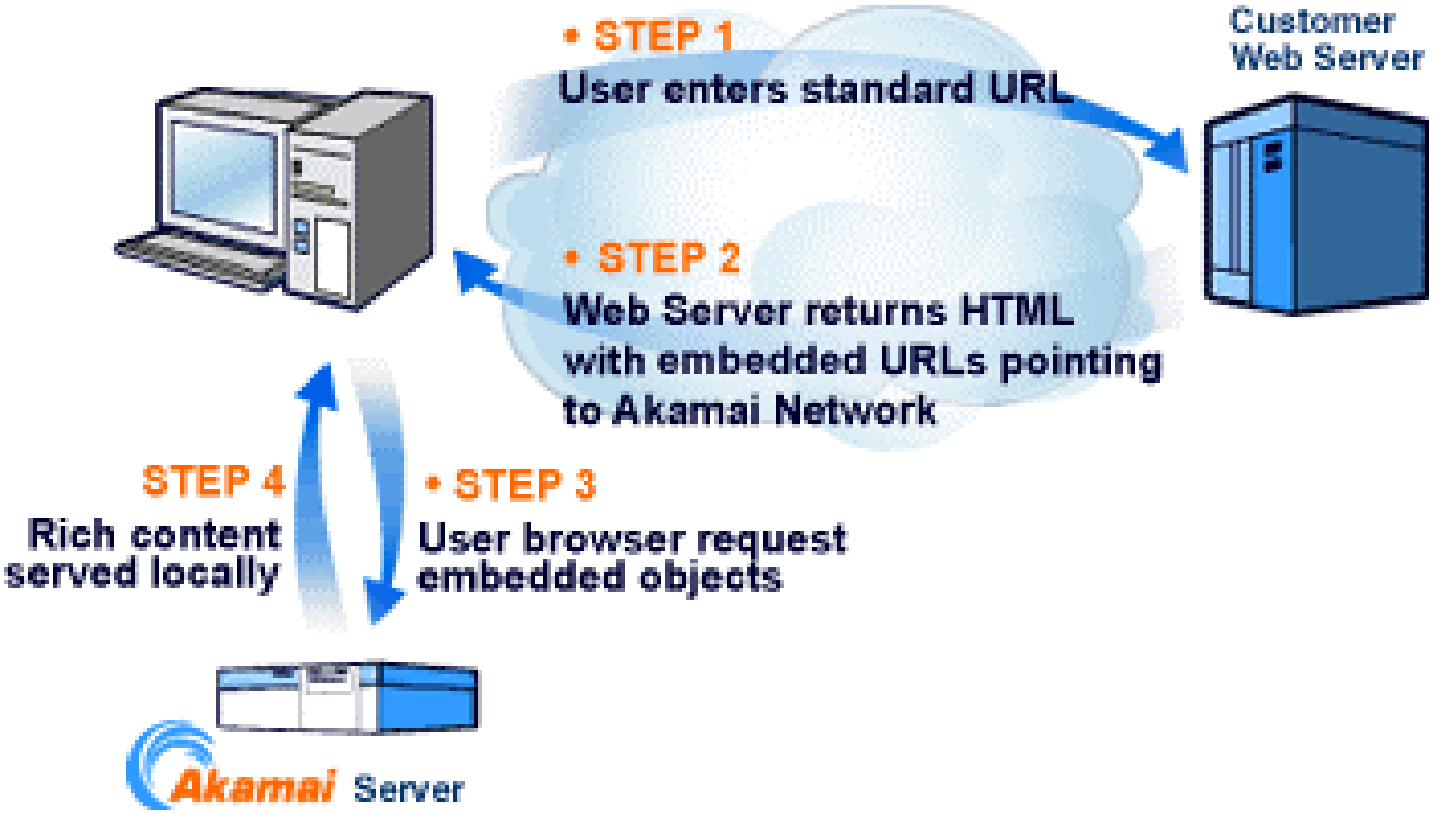
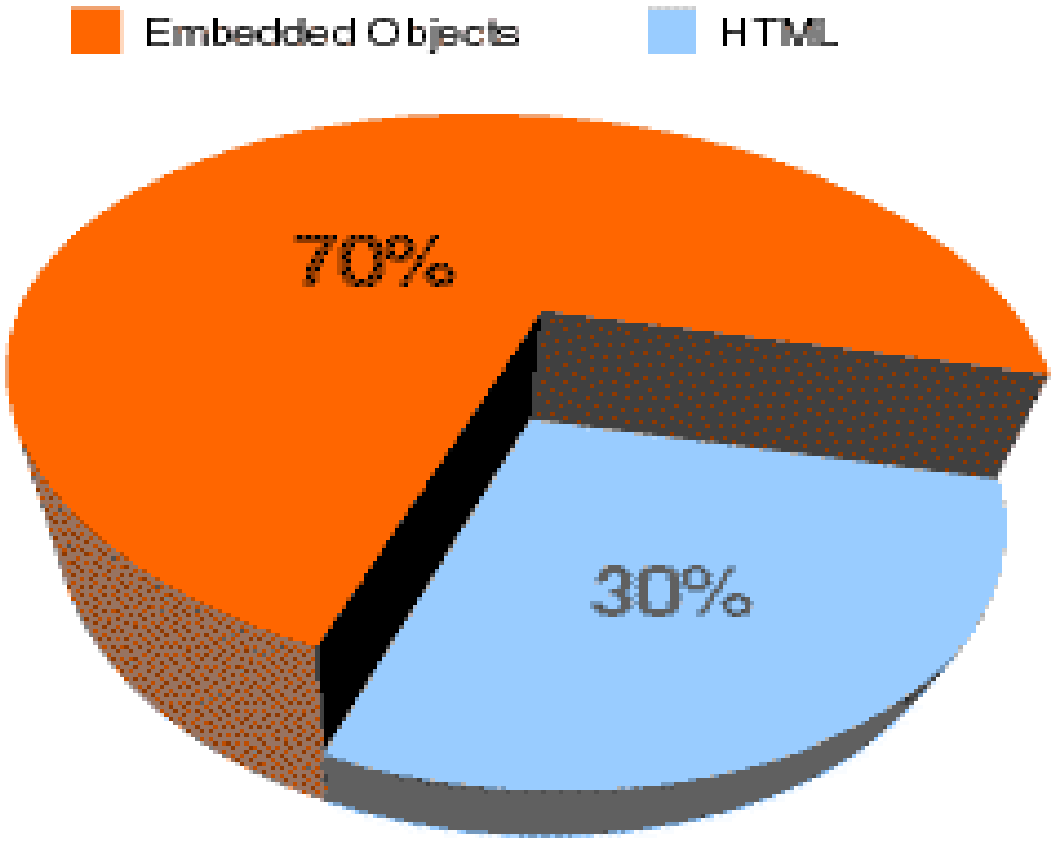
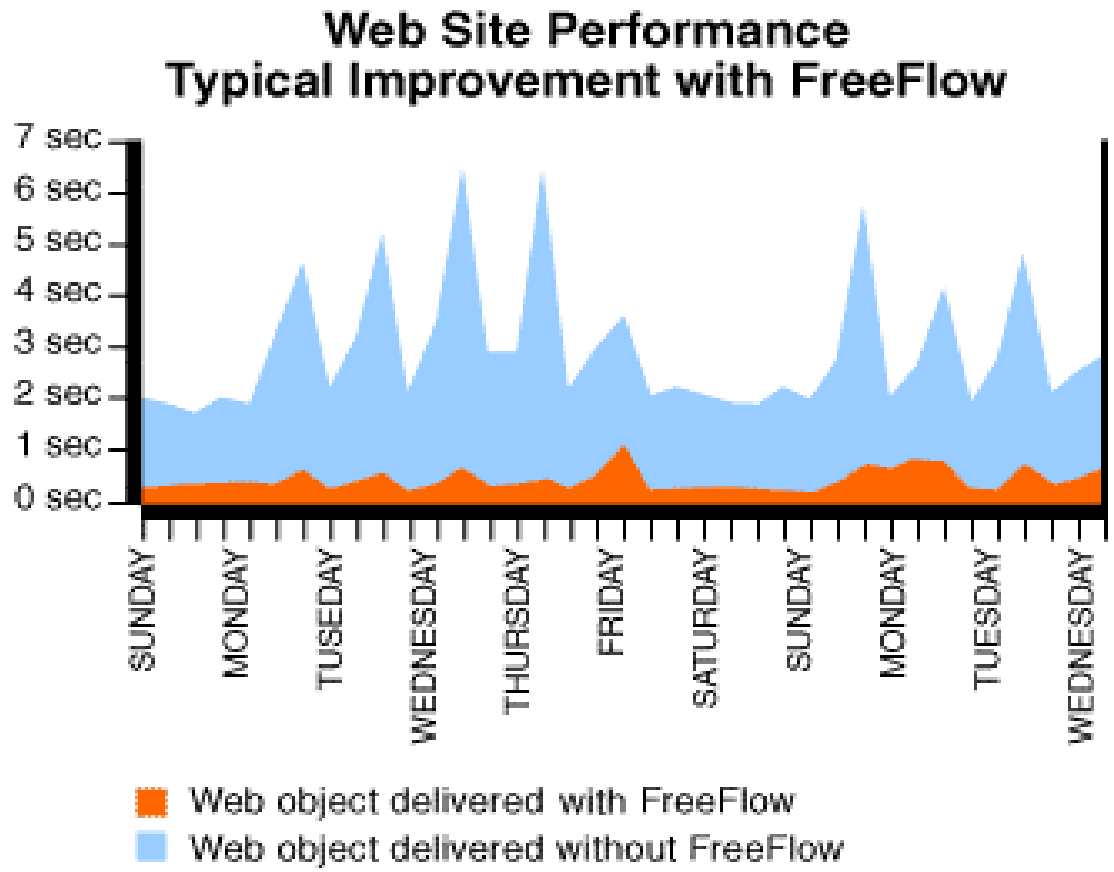


Figure 3: "A Typical Web Page Composition"



System Performance



Main Functions of MS Proxy Server 2.0

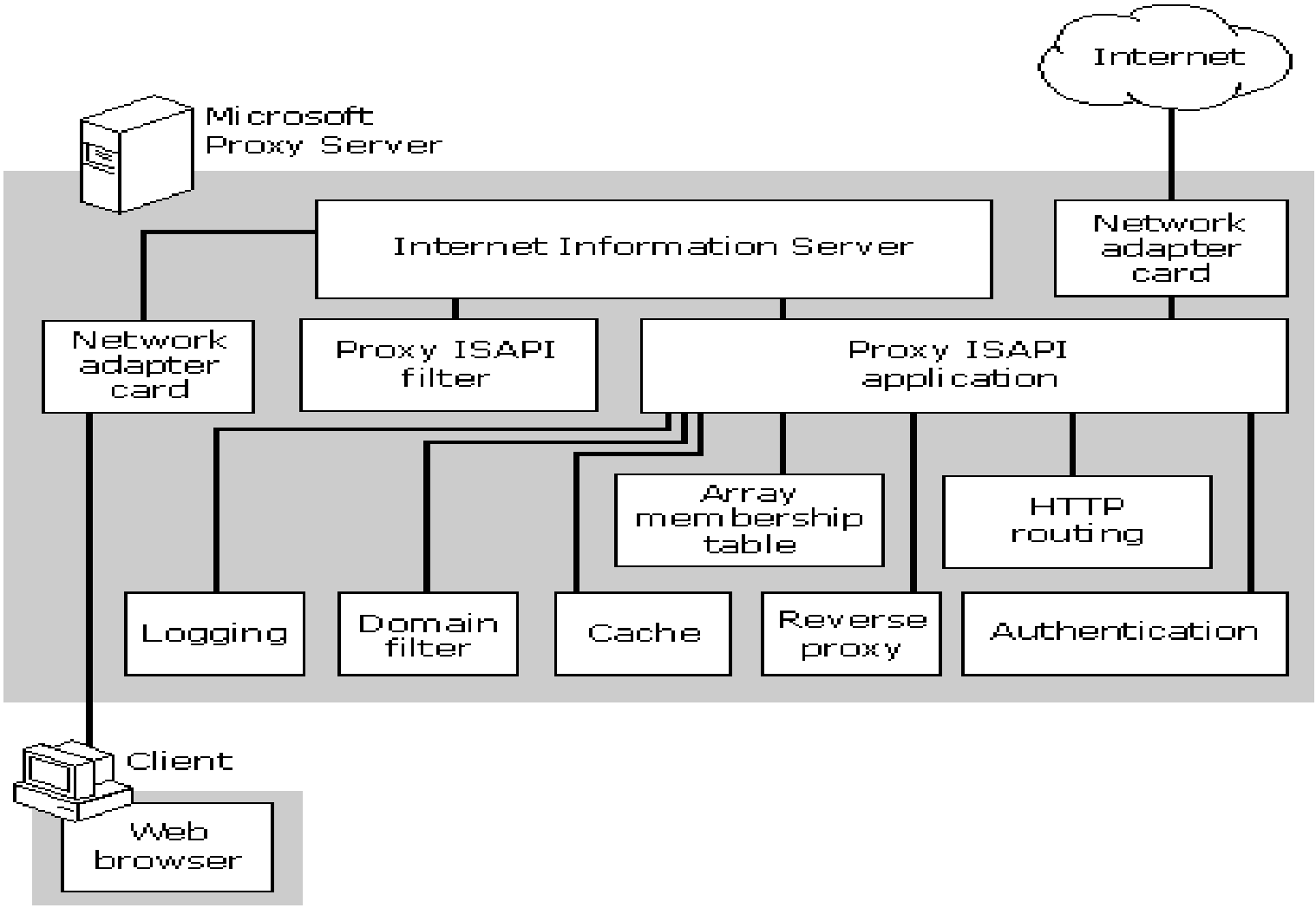
- Extend Internet applications to computers on the internal network
- Improve performance and access for Internet-based services on the internal network
- Provide secure access between the internal network and the internet.

Major Services

MS Proxy Server 2.0 offers a choice of the following services:

- Web Proxy Service
- WinSock Proxy Service
- SOCKS (discussed later)

Web Proxy Server Architecture



Web Proxy (Topics)

- (Outward bound) Traffic Re-Direction
- (Outward bound) Caching
  - Passive Caching
  - Active Caching
  - Distributed Caching
  - Cache Array Routing Method
- (Inward bound) Reverse Proxy

Traffic Re-direction

1. Proxy ISAPI intercepts every request from client computers to IIS.
2. If the request is a proxy request (with a URL complete with protocol and domain name) the name of the Proxy ISAPI application application to the URL, so that the request will be forwarded there for processing.
3. If the request is not a proxy request, the Proxy ISAPI allows ordinary Web publishing to occur between the IIS and client.

Traffic Re-Direction (cont'd)

4. Authenticates the client, and verifies domain name restriction (domain filtering)
5. Looks for "fresh" objects in the cache and returns objects from there
6. Gets the objects from the Internet, sends them to the client, and adds them to the cache if appropriate.

Passive Caching

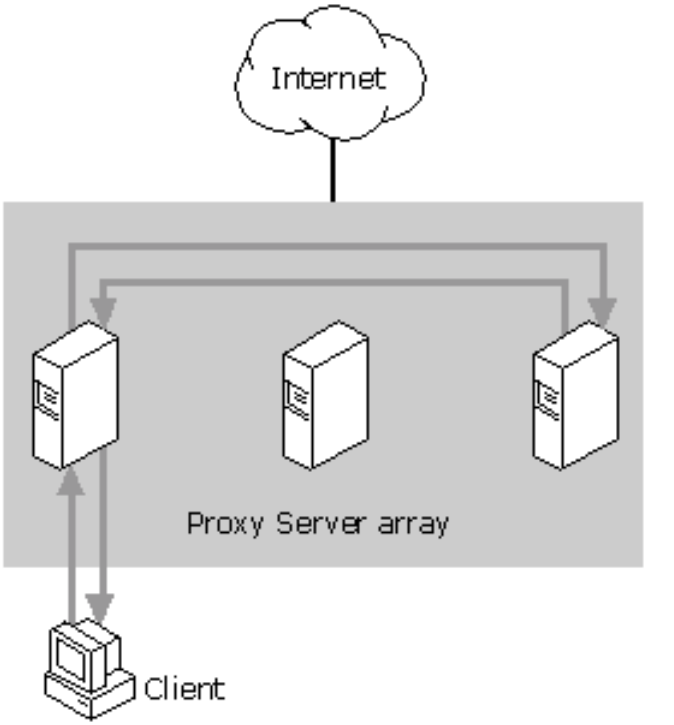
- Usual load distribution and replication methods
- HTTP caching mechanisms and cache directives
- What should not be cached: security sensitive stuff, e.g. keywords and encrypted messages.
- Inlined objects?

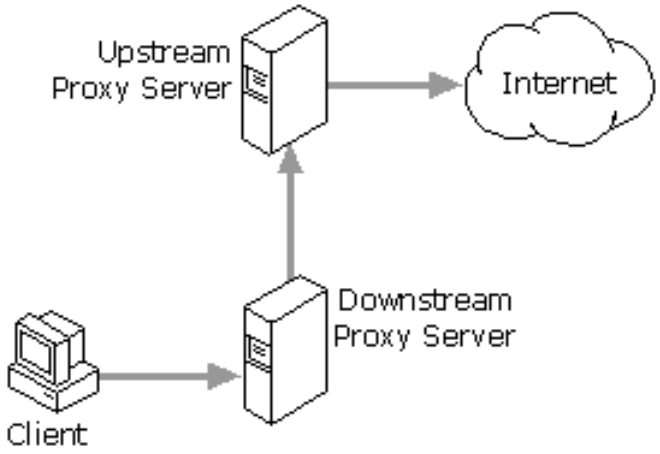
Active Caching

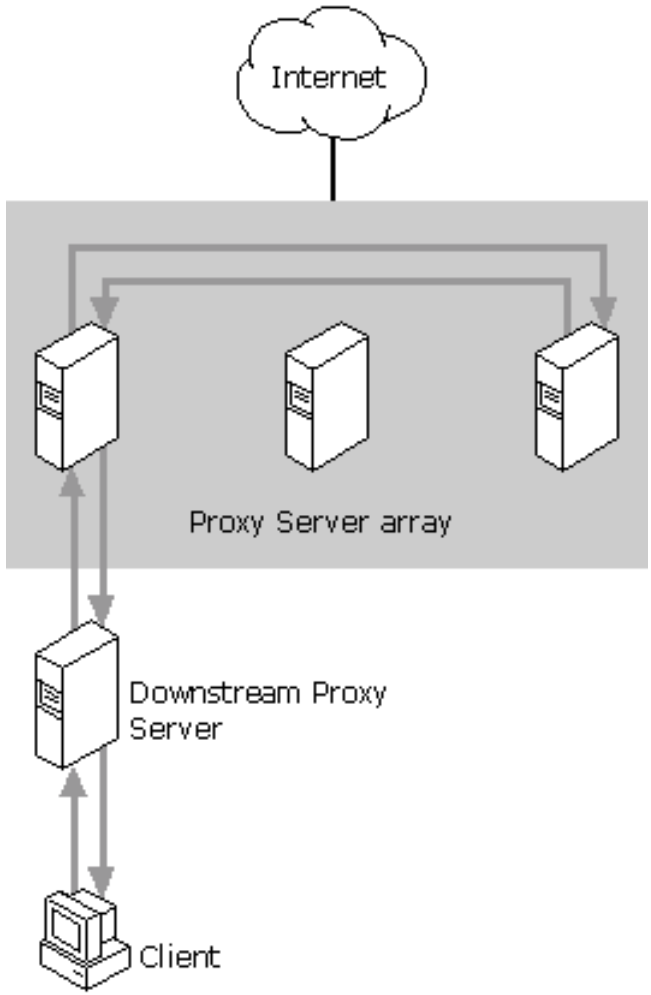
- Active caching usually means pre-fetching from the origin server.
- Popularity: Proxy maintains some statistics on the popularity of some object (e.g. hit ratio)
- Expiry-date: longer date is more valuable – check the validity before it expires
- Server load: more aggressive pre-fetching when the system load is low.

Distributed Caching

- Architecture of multiple proxies:
  - Proxy array: parallel proxies
  - Proxy chain: upstream/downstream proxies
  - Both
- Advantages:
  - Enhanced caching performance
  - Fault tolerance


Internet
Proxy Server array
Client


Upstream
Proxy Server
Internet
Downstream
Proxy Server
Client


Internet
Proxy Server array
Downstream Proxy
Server
Client

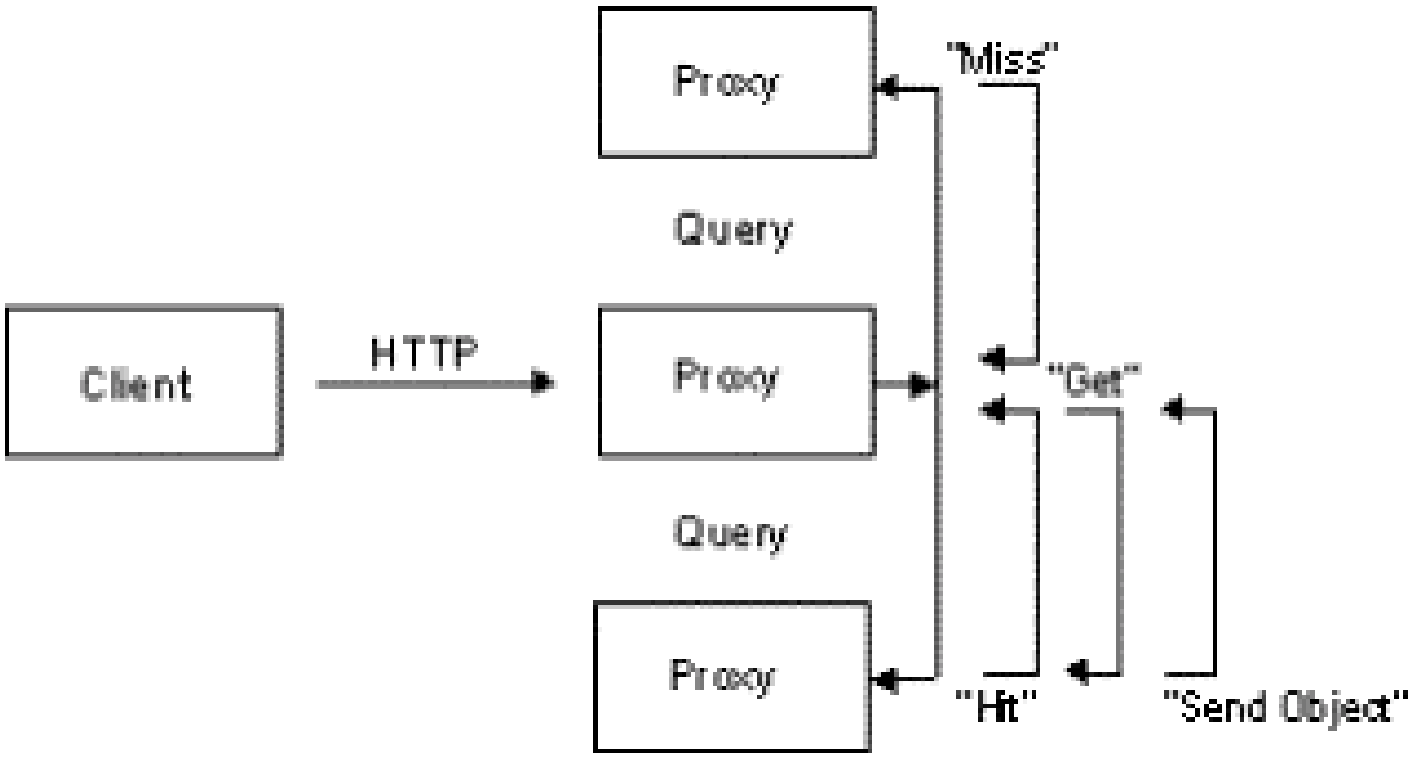
Internet Cache Protocol (ICP)

- ICP is an IETF standard introduced in 1995 to allow proxies to “query” neighbouring proxies to locate the requested cache.
- The query is usually the request-URI as found in the request-line of the HTTP message.
- If all queries fail to find the cached object, the proxy will request the object from the Internet (for outward bound traffic), or the origin server (for inward bound traffic).

How ICP Works

- Query (HTTP message) is sent to the default proxy
- If the object is not found, the default proxy sends queries to other servers
- No-hit messages are returned from the servers which fail to find the object
- A hit message confirms presence of the object, if currently cached.
- If a hit message is received, a copy of cached object is transferred to the client default proxy server.

Example


Proxy
"Miss"
Query
Client
HTTP
Proxy
"Get"
Query
Proxy
"Hit"
"Send Object"

Pitfalls of ICP

- Extra traffic generated
- “Negative” scalability: more proxies beget more network traffic
- Redundant contents gradually proliferating over the whole array of proxies

Cache Array Routing Protocol

- For each proxy, compute a hash value, say *a*
- For each cached object, computer a hash value, say *b*
- The proxy has the highest combined value will keep the object in the cache.
- Load factor of each proxy could be also taken into account (how?)

Properties of CARP

- Assume the hash function is perfectly random
- Each object is equally likely to be cached in any proxy.
- When a new proxy is added, only 1/*n*, *n* being the (new) total of proxies, cached objects need to be re-distributed to the new proxy. The same is true, when an existing proxy is withdrawn from service.

How CARP Works

- Compute the hash values of proxies in the array based on the array membership table.
  - By the client computer
  - Through a designated proxy
- Compute the hash value of the URL
- Locate the proxy which may possibly contain the object, via CARP
- If the object is not found, a request is made to Internet.

Forward vs. Reverse Proxy

- Forward (outward bound) proxy serves the community of clients, while reverse proxy (inward bound) serves one or more origin servers.
- All client computers send requests to a forward proxy, while requests to a specific origin server will be serviced by its reverse proxy.
- Forward and reverse proxy software may run on a single computer.

CARP Example: 4 proxies and 4 objects for caching

		www.microsoft.com	www.yahoo.com	www.msn.com	www.ibm.com
Proxy	Hash	19	14	5	2
Jericho1	13	5	6	10	4
Jericho2	8	9	2	7	5
Jericho3	5	7	4	3	10
Jericho4	28	4	7	8	1

CARP Example (cont'd): One more proxy is now added, which causes re-distribution of cached objects.

		www.microsoft.com	www.yahoo.com	www.msn.com	www.ibm.com
Proxy	Hash	19	14	5	2
Jericho1	13	5	6	10	4
Jericho2	8	9	2	7	5
Jericho3	5	7	4	3	10
Jericho4	28	4	7	8	1
Jericho5	14	2	9	4	6

Why Reverse Proxy?

- Security
- Caching
- Load balancing
- Virtual multihosting

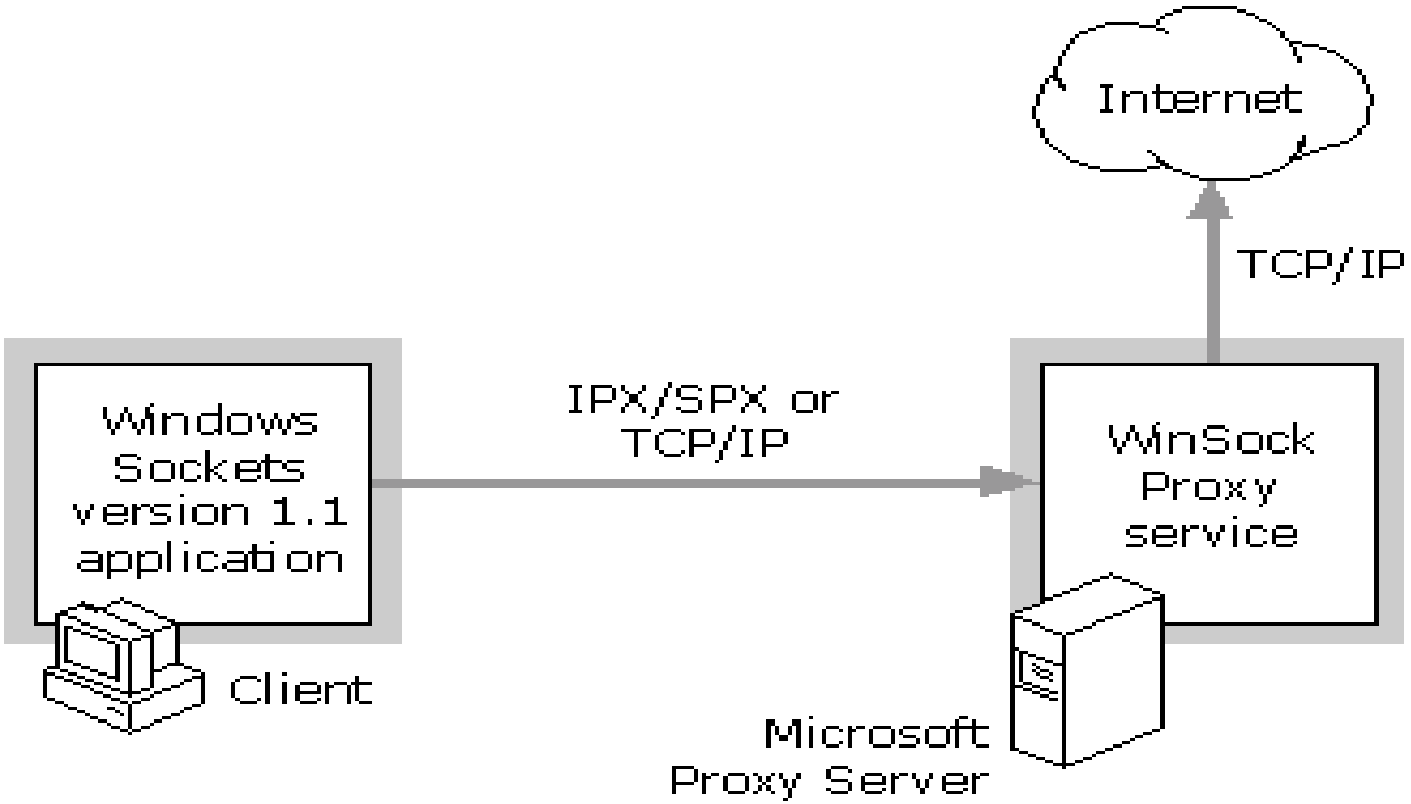
WinSock Proxy

- WinSock proxy allows a Windows sockets application running on internal network client to work as though it were directly connected to the Internet.
- Telnet, RealAudio, mail and news are some Windows socket applications.
- The WinSock proxy acts as the intermediary between the application and Internet.

Functions of WinSock Proxy

- Create virtual connections between internal applications and Internet.
- Handle “data pumping” between the two actual communication channels set up for a virtual connection.
- Acts as a TCP/IP gateway if the internal network runs a different protocol (e.g. IPX/SPX for NetWare).

WinSock Proxy Virtual Connection



How WinSock Proxy Works

- On the client computer, a Client Winsock Proxy DLL is installed to intercept all Windows Socket API calls from the application.
- The Client WinSock Proxy DLL may do one of the following:
  - Completely process the request
  - Pass the request to the actual Windows Socket DLL on the client (after possibly making changes to the request)
  - Pass control information to WinSock Proxy service on the Proxy Server computer.