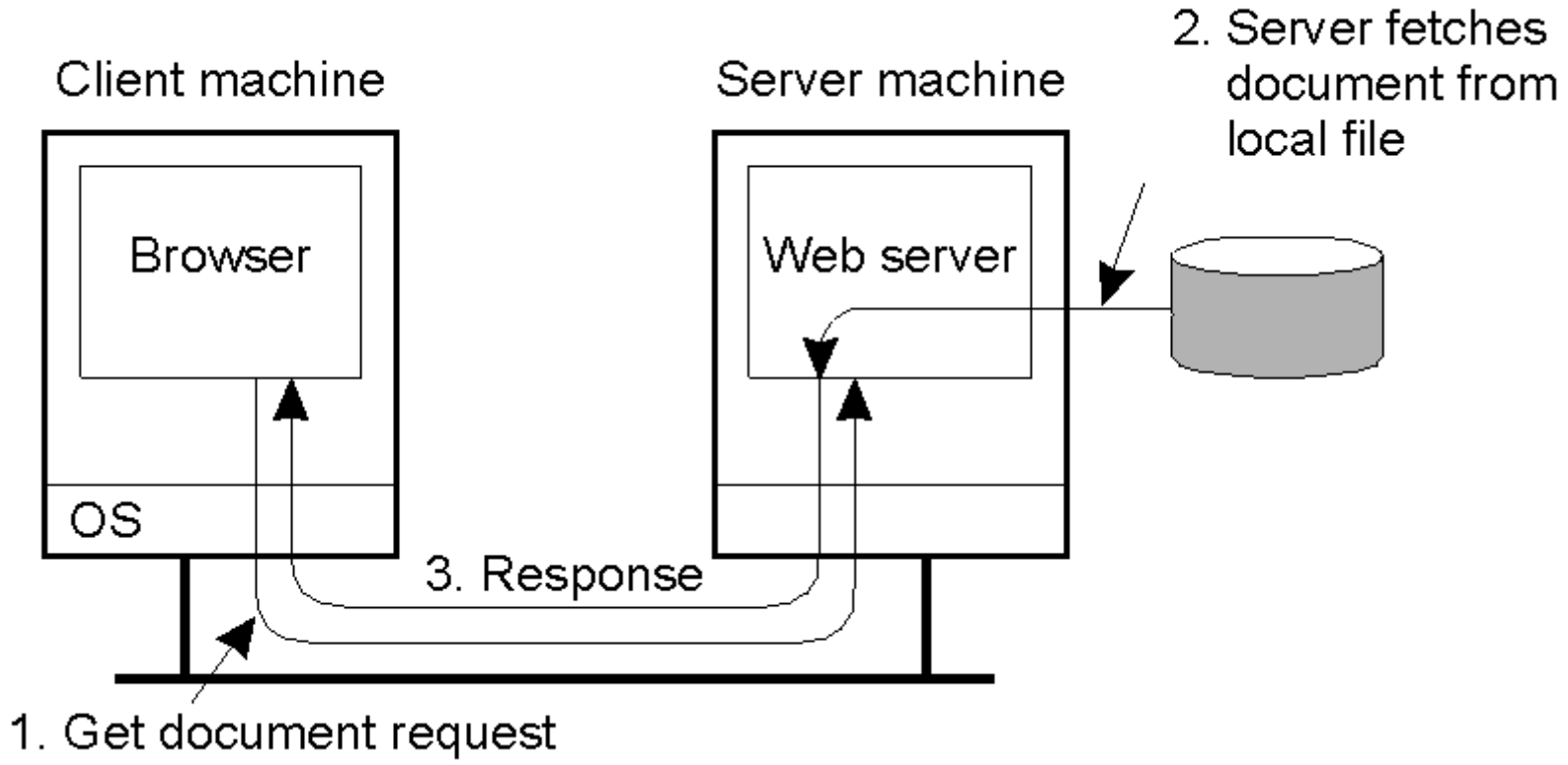
Distributed Document-Based Systems

Chapter 11

The World Wide Web



Overall organization of the Web.

Document Model (1)

```
<HTML>                                          <!- Start of HTML document     -->
<BODY>                                          <!- Start of the main body     -->
<H1>Hello World/H1>                             <!- Basic text to be displayed -->
<P>                                             <!- Start of a new paragraph   -->
<SCRIPT type = "text/javascript">               <!- identify scripting language -->
  document.writeln ("<H1>Hello World</H1>;      // Write a line of text
</SCRIPT>                                       <!- End of scripting section   -->
</P>                                            <!- End of paragraph section -->
</BODY>                                         <!- End of main body           -->
</HTML>                                         <!- End of HTML section        -->
```

A simple Web page embedding a script written in JavaScript.

Document Model (2)

(1) <!ELEMENT article (title, author+,journal)>
(2) <!ELEMENT title (#PCDATA)>
(3) <!ELEMENT author (name, affiliation?)>
(4) <!ELEMENT name (#PCDATA)>
(5) <!ELEMENT affiliation (#PCDATA)>
(6) <!ELEMENT journal (jname, volume, number?, month? pages, year)>
(7) <!ELEMENT jname (#PCDATA)>
(8) <!ELEMENT volume (#PCDATA)>
(9) <!ELEMENT number (#PCDATA)>
(10) <!ELEMENT month (#PCDATA)>
(11) <!ELEMENT pages (#PCDATA)>
(12) <!ELEMENT year (#PCDATA)>

An XML definition for referring to a journal article.

Document Model (3)

```
(1)     <?xml = version "1.0">
(2)     <!DOCTYPE article SYSTEM "article.dtd">
(3)     <article>
(4)        <title> Prudent Engineering Practice for Cryptographic Protocols</title>
(5)        <author><name>M. Abadi</name></author>
(6)        <author><name>R. Needham</name></author>
(7)        <journal>
(8)                <jname>IEEE Transactions on Software Engineering</jname>
(9)                <volume>22</volume>
(10)               <number>12</number>
(11)               <month>January</month>
(12)               <pages>6 – 15</pages>
(13)               <year>1996</year>
(14)       </journal>
(15)    </article>
```

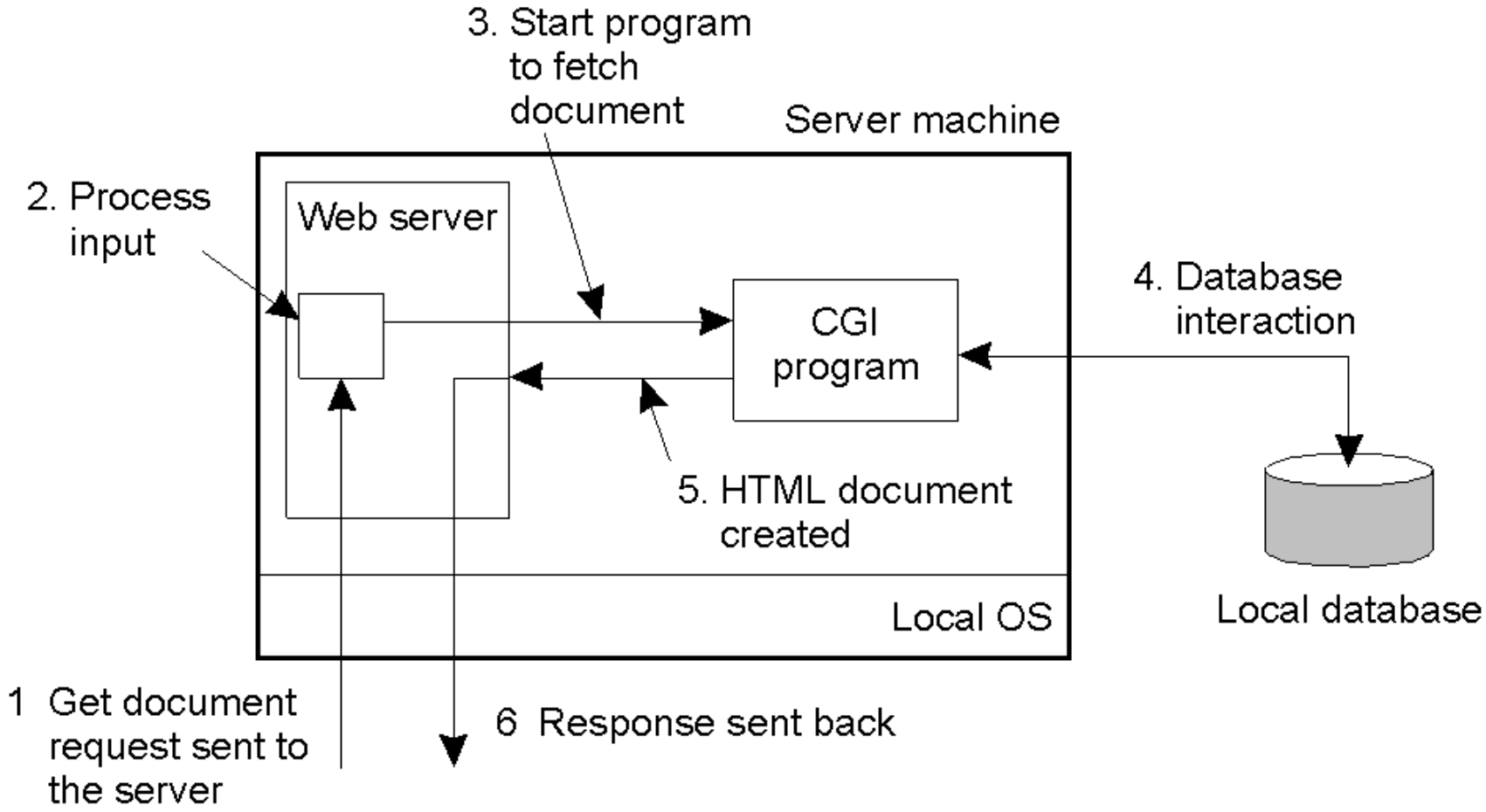
An XML document using the XML definitions from previous slide

Document Types

Type	Subtype	Description
Text	Plain	Unformatted text
	HTML	Text including HTML markup commands
	XML	Text including XML markup commands
Image	GIF	Still image in GIF format
	JPEG	Still image in JPEG format
Audio	Basic	Audio, 8-bit PCM sampled at 8000 Hz
	Tone	A specific audible tone
Video	MPEG	Movie in MPEG format
	Pointer	Representation of a pointer device for presentations
Application	Octet-stream	An uninterrupted byte sequence
	Postscript	A printable document in Postscript
	PDF	A printable document in PDF
Multipart	Mixed	Independent parts in the specified order
	Parallel	Parts must be viewed simultaneously

Six top-level MIME types and some common subtypes.

Architectural Overview (1)



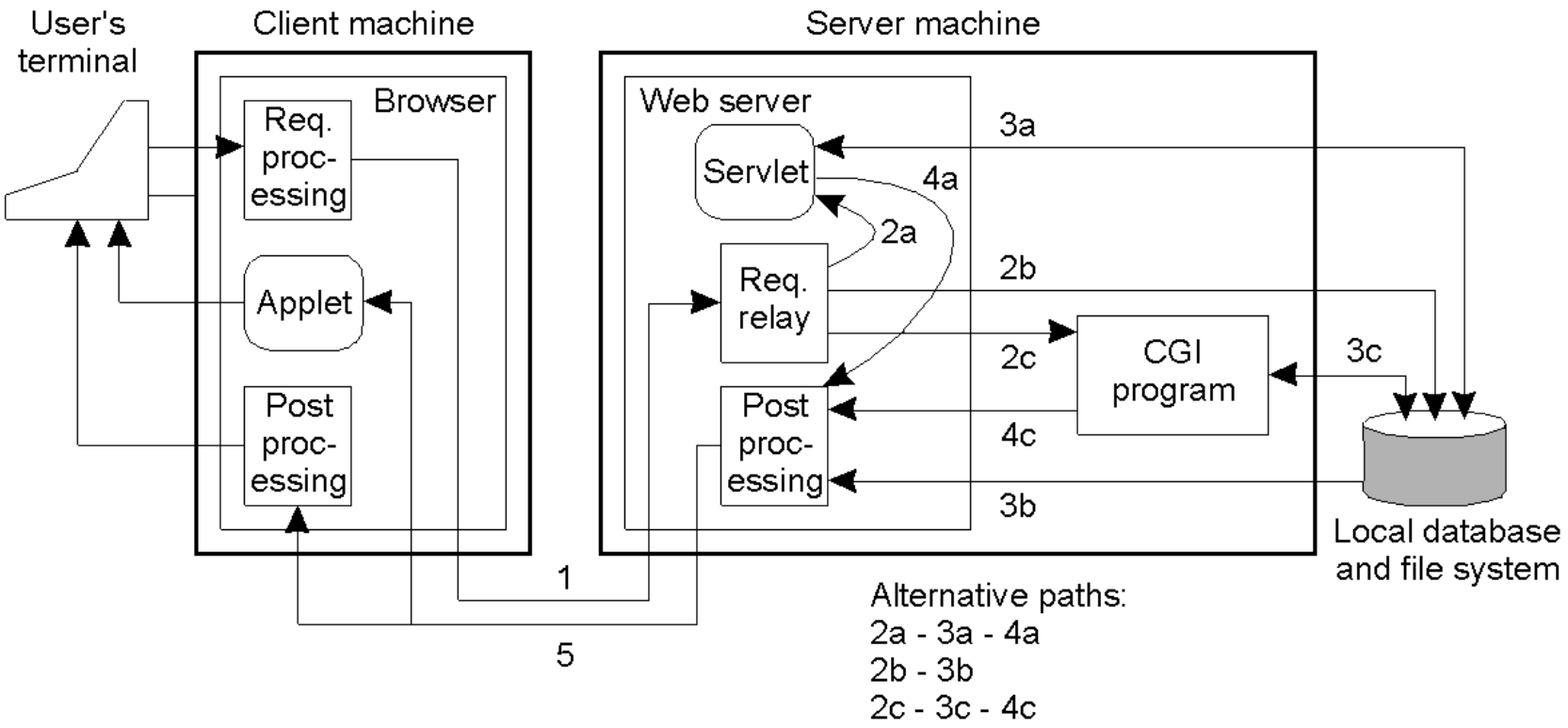
The principle of using server-side CGI programs.

Architectural Overview (2)

```
(1)     <HTML>
(2)     <BODY>
(3)     <P>The current content of <pre>/data/file.txt</PRE>is:</P>
(4)     <P>
(5)     <SERVER type = "text/javascript");
(6)         clientFile = new File("/data/file.txt");
(7)         if(clientFile.open("r")){
(8)                 while (!clientFile.eof())
(9)                     document.writeln(clientFile.readln());
(10)                clientFile.close();
(11)        }
(12)    </SERVER>
(13)    </P>
(14)    <P>Thank you for visiting this site.</P>
(15)    </BODY>
(16)    </HTML>
```

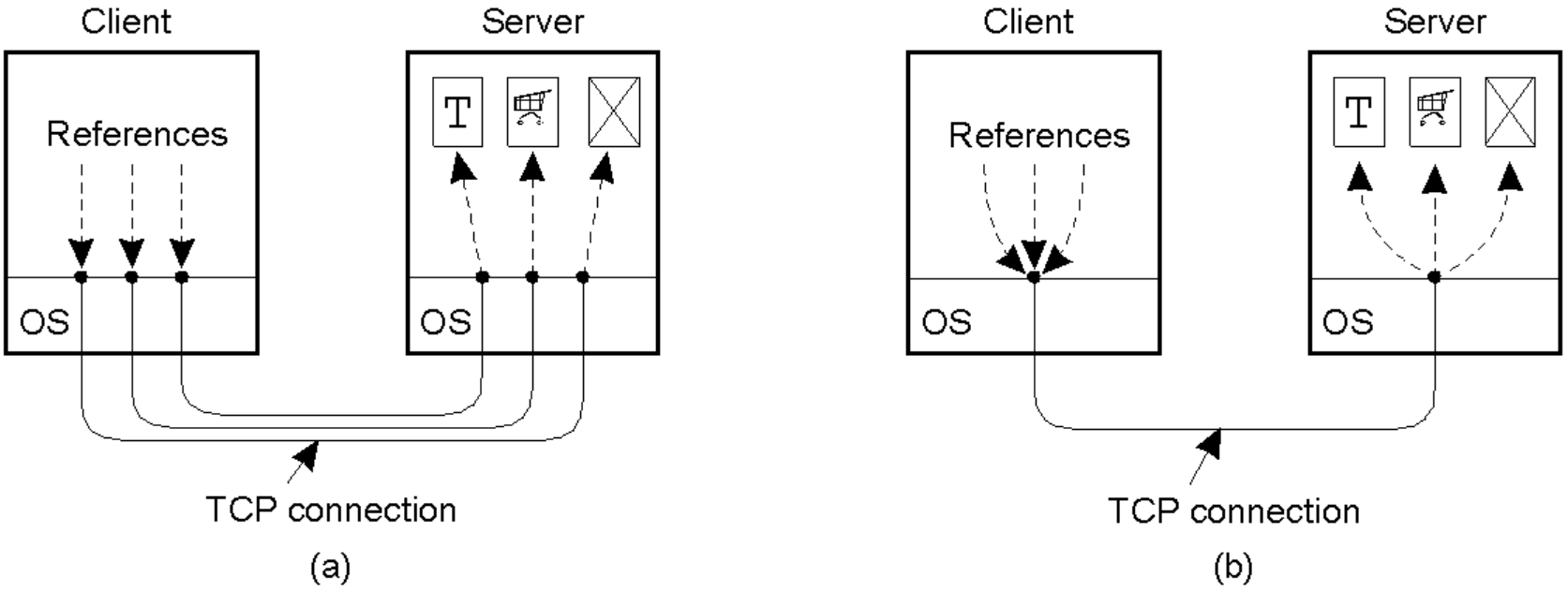
An HTML document containing a JavaScript to be executed by the server

Architectural Overview (3)



Architectural details of a client and server in the Web.

HTTP Connections



a) Using nonpersistent connections.

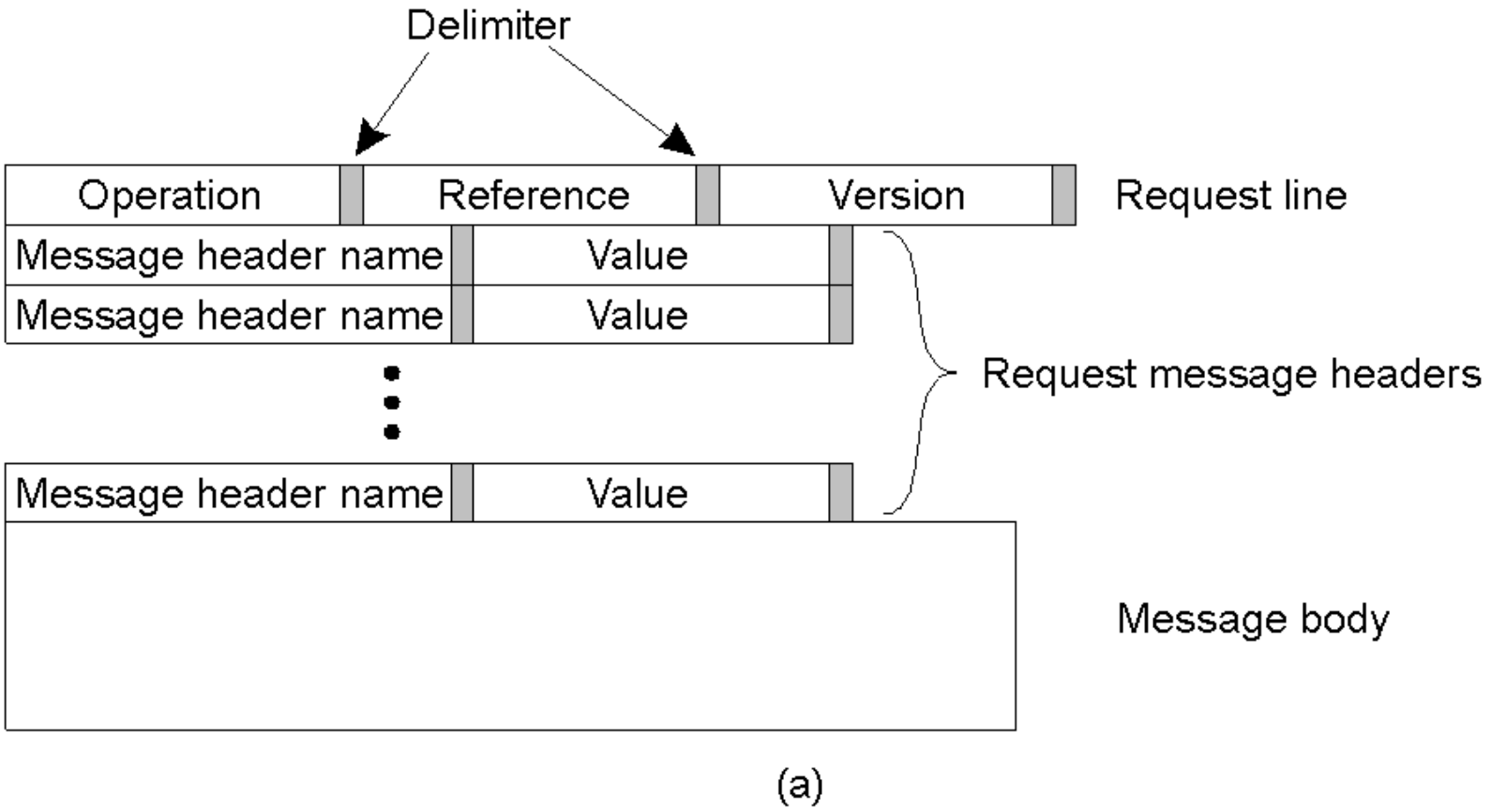
b) Using persistent connections

HTTP Methods

Operation	Description
Head	Request to return the header of a document
Get	Request to return a document to the client
Put	Request to store a document
Post	Provide data that is to be added to a document (collection)
Delete	Request to delete a document

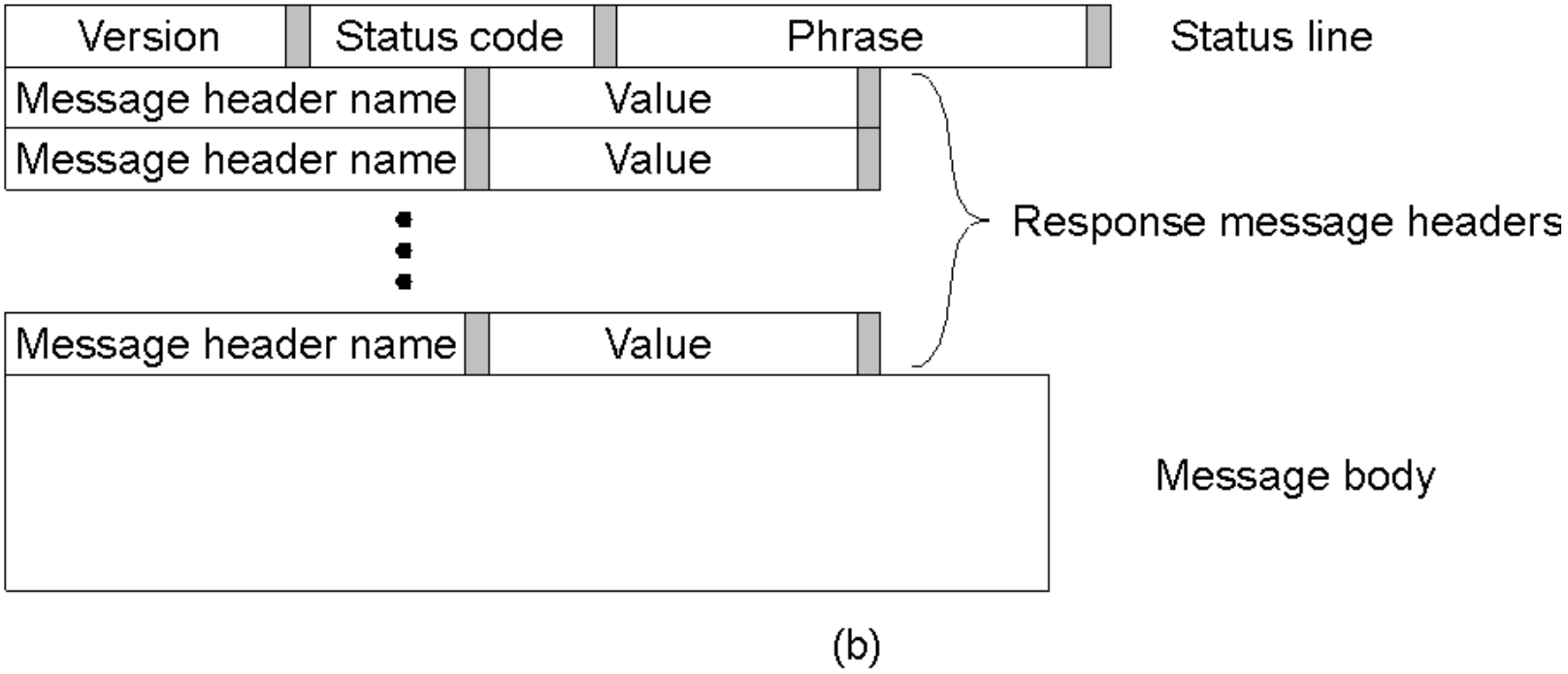
Operations supported by HTTP.

HTTP Messages (1)



HTTP request message

HTTP Messages (2)



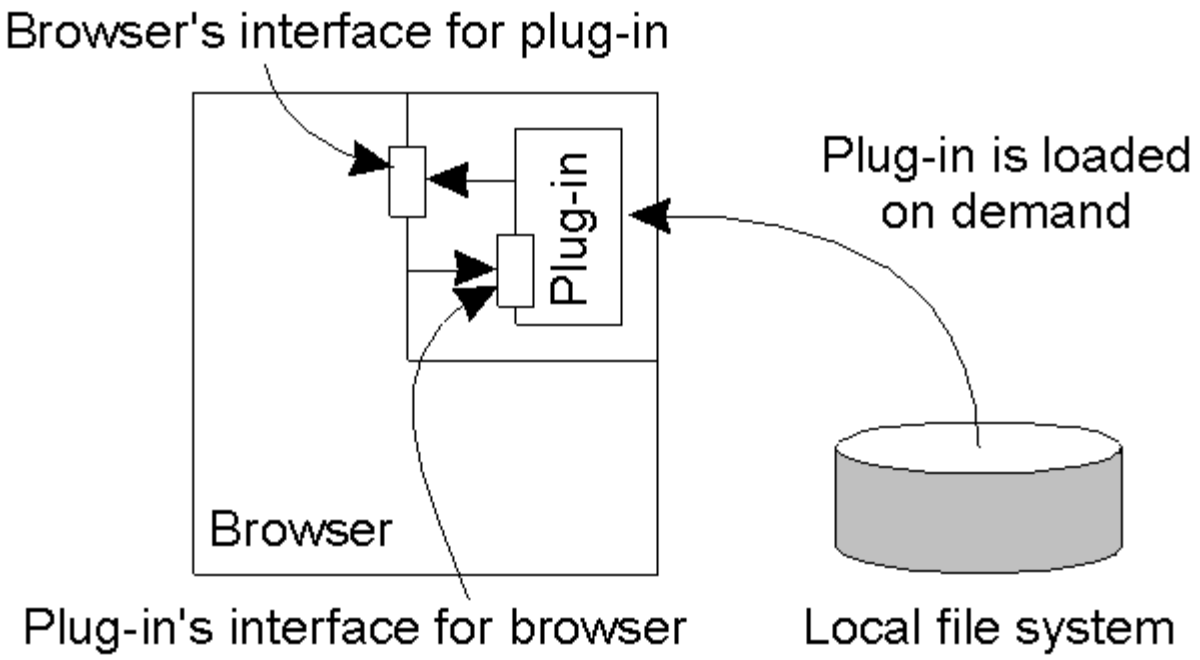
HTTP response message.

HTTP Messages (3)

Some HTTP message headers.

Header	Source	Contents
Accept	Client	The type of documents the client can handle
Accept-Charset	Client	The character sets are acceptable for the client
Accept-Encoding	Client	The document encodings the client can handle
Accept-Language	Client	The natural language the client can handle
Authorization	Client	A list of the client's credentials
WWW-Authenticate	Server	Security challenge the client should respond to
Date	Both	Date and time the message was sent
ETag	Server	The tags associated with the returned document
Expires	Server	The time how long the response remains valid
From	Client	The client's e-mail address
Host	Client	The TCP address of the document's server
If-Match	Client	The tags the document should have
If-None-Match	Client	The tags the document should not have
If-Modified-Since	Client	Tells the server to return a document only if it has been modified since the specified time
If-Unmodified-Since	Client	Tells the server to return a document only if it has not been modified since the specified time
Last-Modified	Server	The time the returned document was last modified
Location	Server	A document reference to which the client should redirect its request
Referer	Client	Refers to client's most recently requested document
Upgrade	Both	The application protocol the sender wants to switch to
Warning	Both	Information about the status of the data in the message

Clients (1)



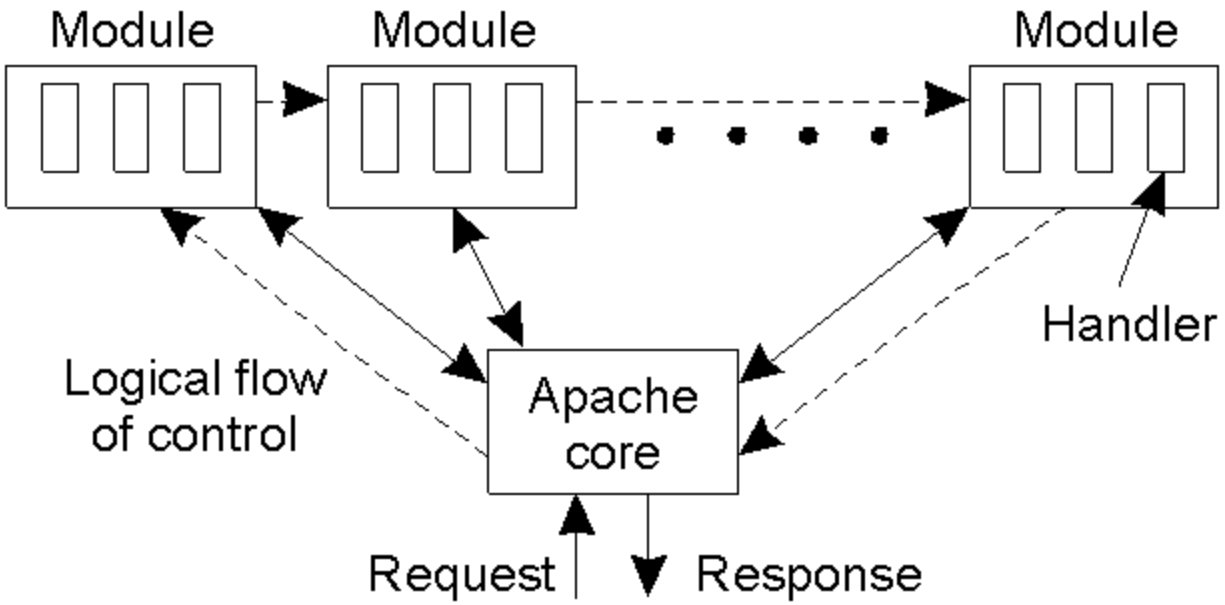
Using a plug-in in a Web browser.

Clients (2)



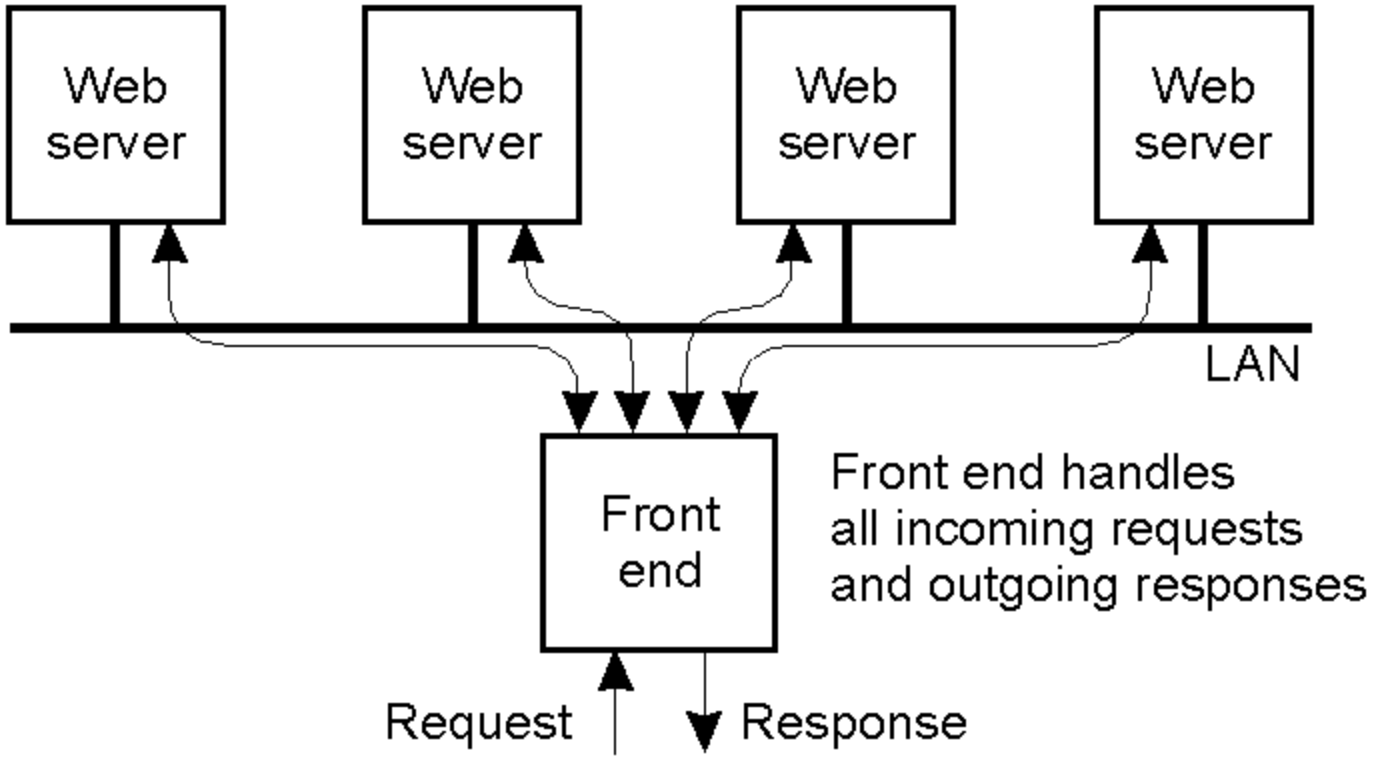
Using a Web proxy when the browser does not speak FTP.

Servers



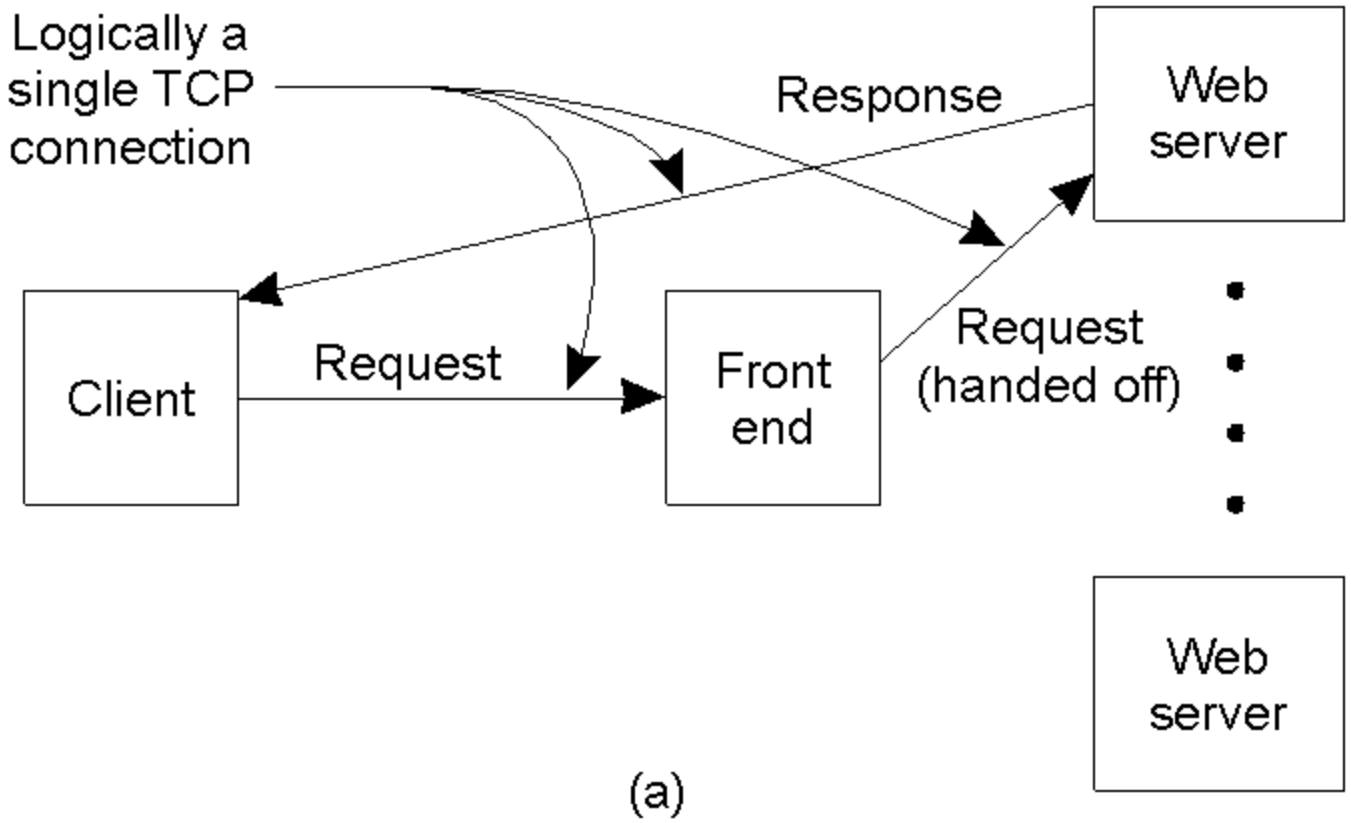
General organization of the Apache Web server.

Server Clusters (1)



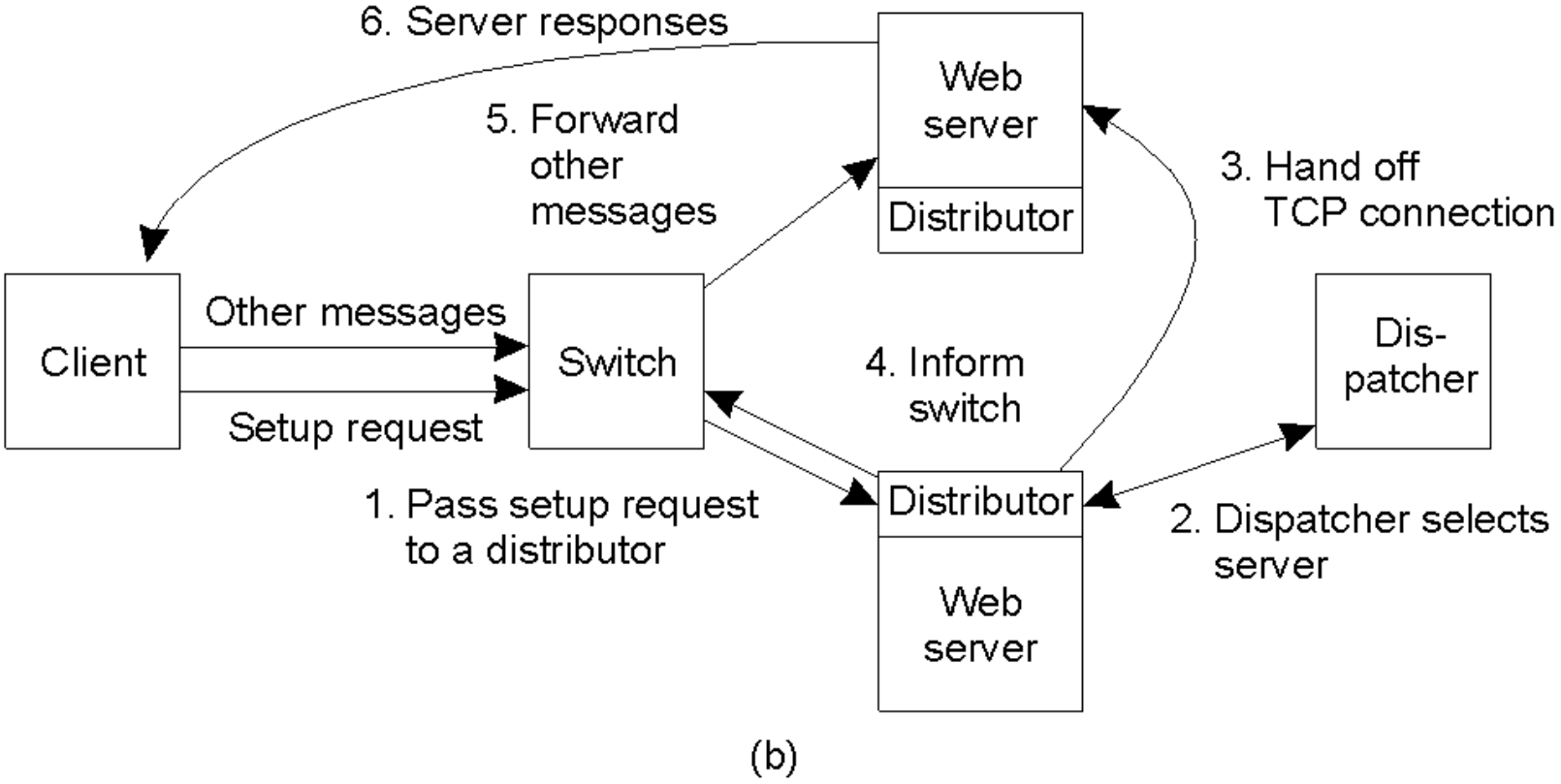
The principle of using a cluster of workstations to implement a Web service.

Server Clusters (2)



(a) The principle of TCP handoff.

Server Clusters (3)



(b) A scalable content-aware cluster of Web servers.

Uniform Resource Locators (1)

Scheme	Host name	Pathname
http ://	www.cs.vu.nl	/home/steen/mbox

(a)

Scheme	Host name	Port	Pathname
http ://	www.cs.vu.nl :	80	/home/steen/mbox

(b)

Scheme	Host name	Port	Pathname
http ://	130.37.24.11 :	80	/home/steen/mbox

(c)

Often-used structures for URLs.

a) Using only a DNS name.

b) Combining a DNS name with a port number.

c) combining an IP address with a port number.

Uniform Resource Locators (2)

Name	Used for	Example
http	HTTP	http://www.cs.vu.nl:80/globe
ftp	FTP	ftp://ftp.cs.vu.nl/pup/minx/README
file	Local file	file:/edu/book/work/chp/11/11
data	Inline data	data:text/plain;charset=iso-8859-7,%e1%e2%e3
telnet	Remote login	telnet://flits.cs.vu.nl
tel	Telephone	tel:+31201234567
modem	Modem	modem:+31201234567;type=v32

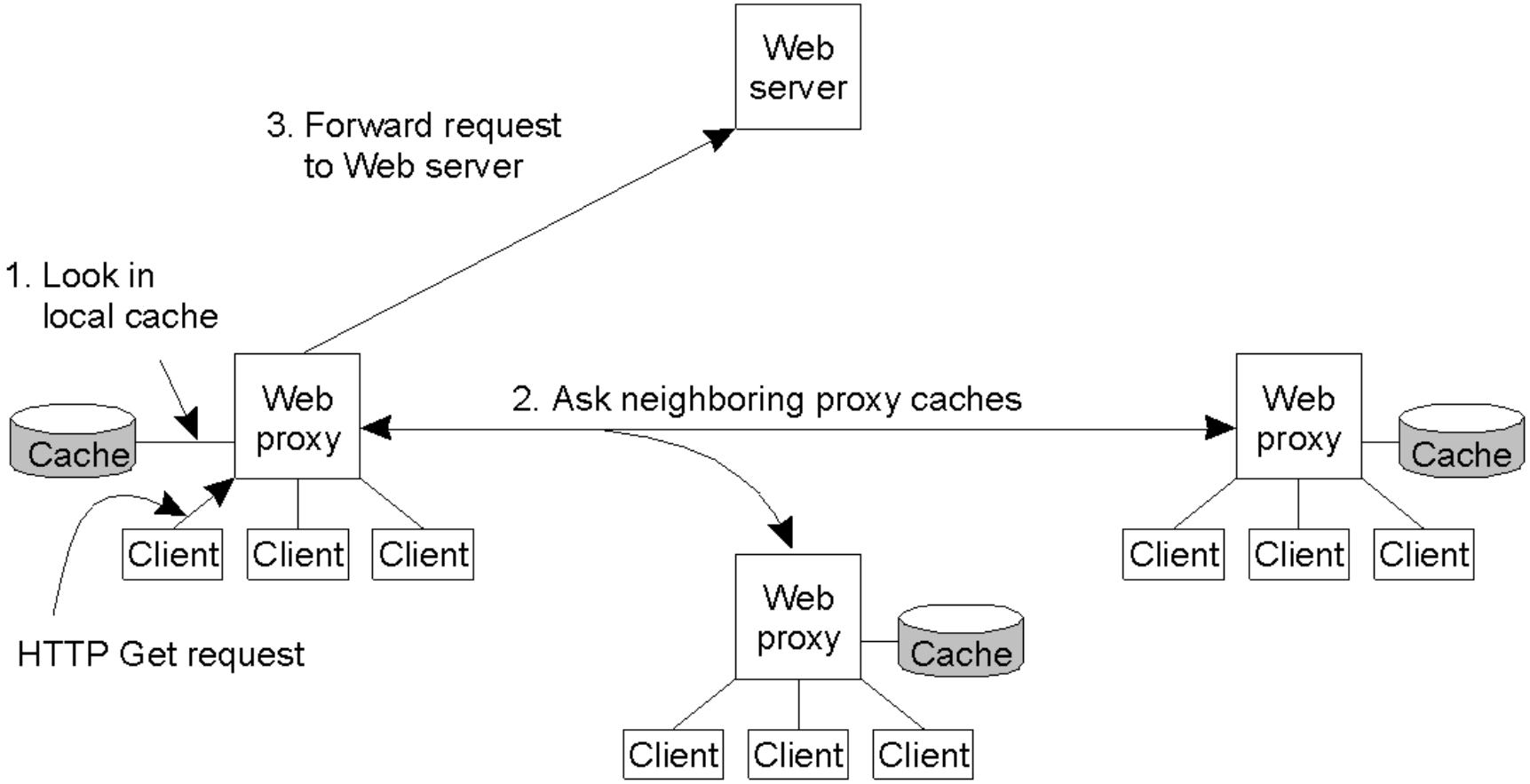
Examples of URLs.

Uniform Resource Names

"urn"	Name space	Name of resource
urn :	ietf :	rfc:2648

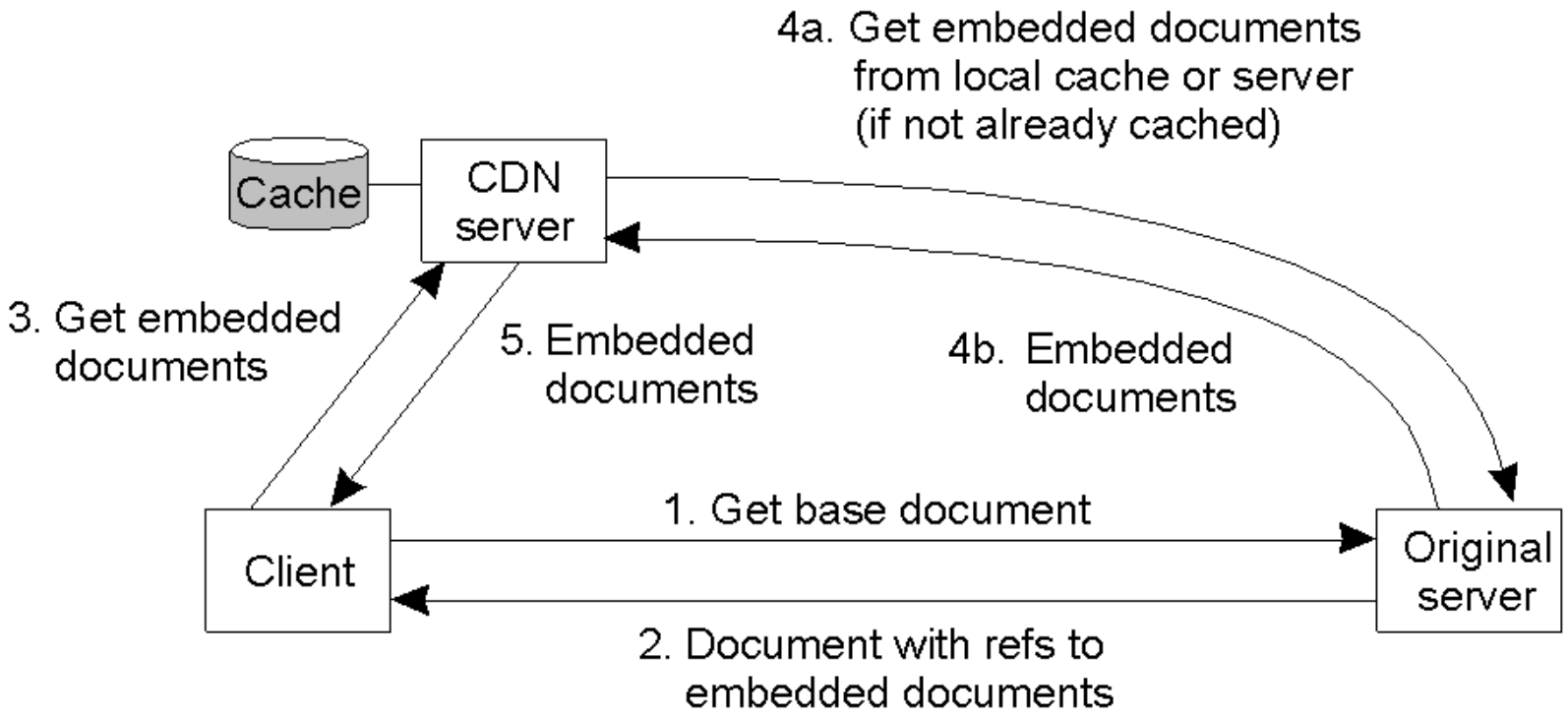
The general structure of a URN

Web Proxy Caching



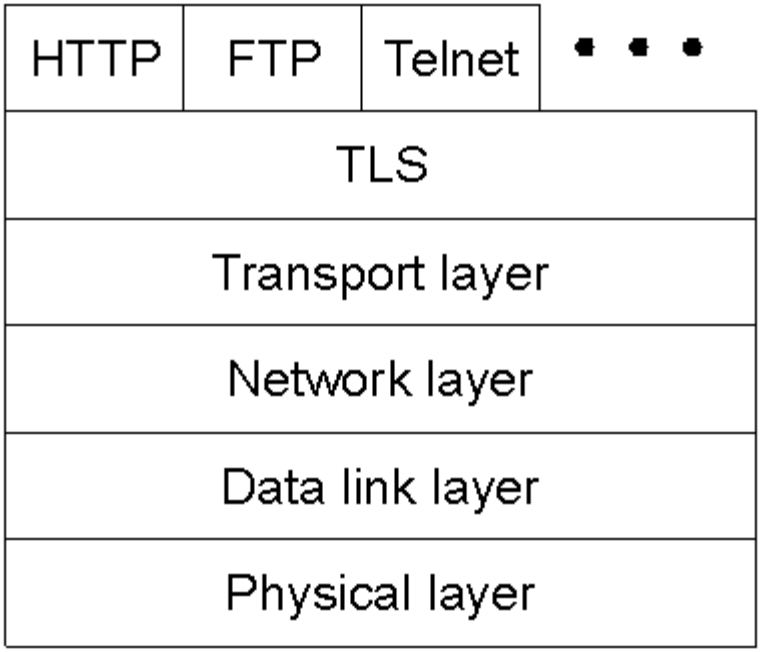
The principle of cooperative caching

Server Replication



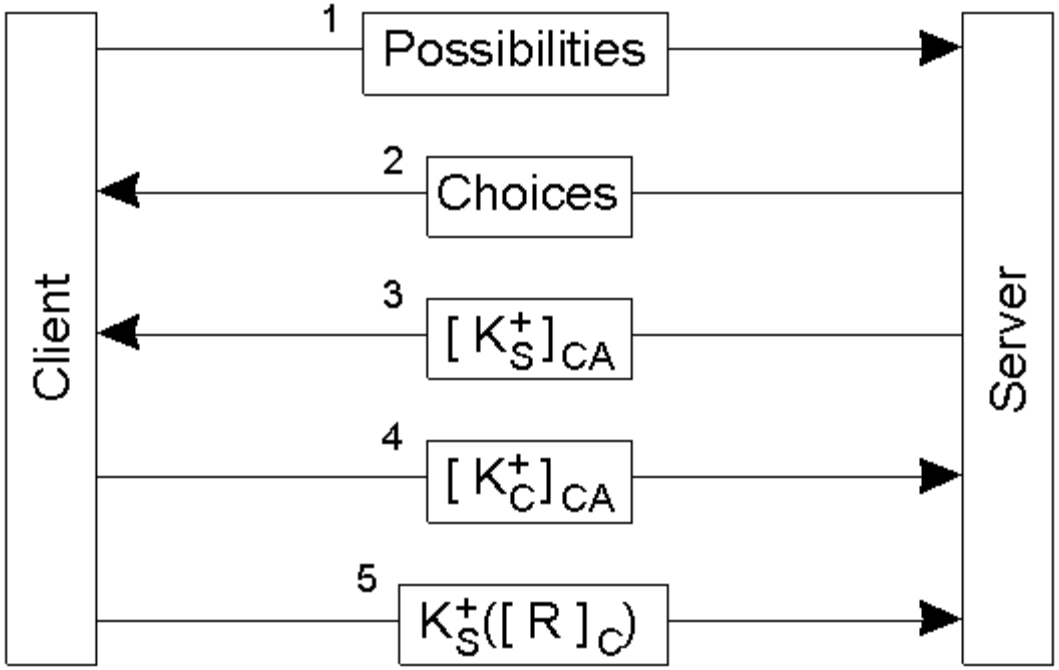
The principle working of the Akami CDN.

Security (1)



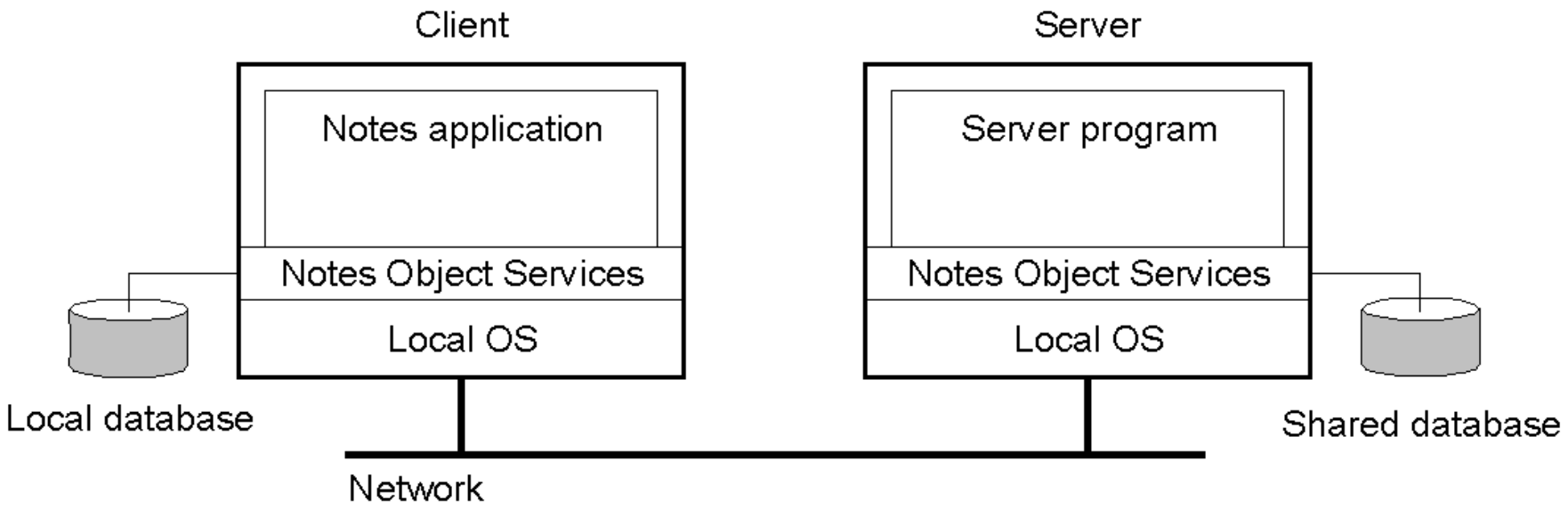
The position of TLS in the Internet protocol stack.

Security (2)



TLS with mutual authentication.

Lotus Notes



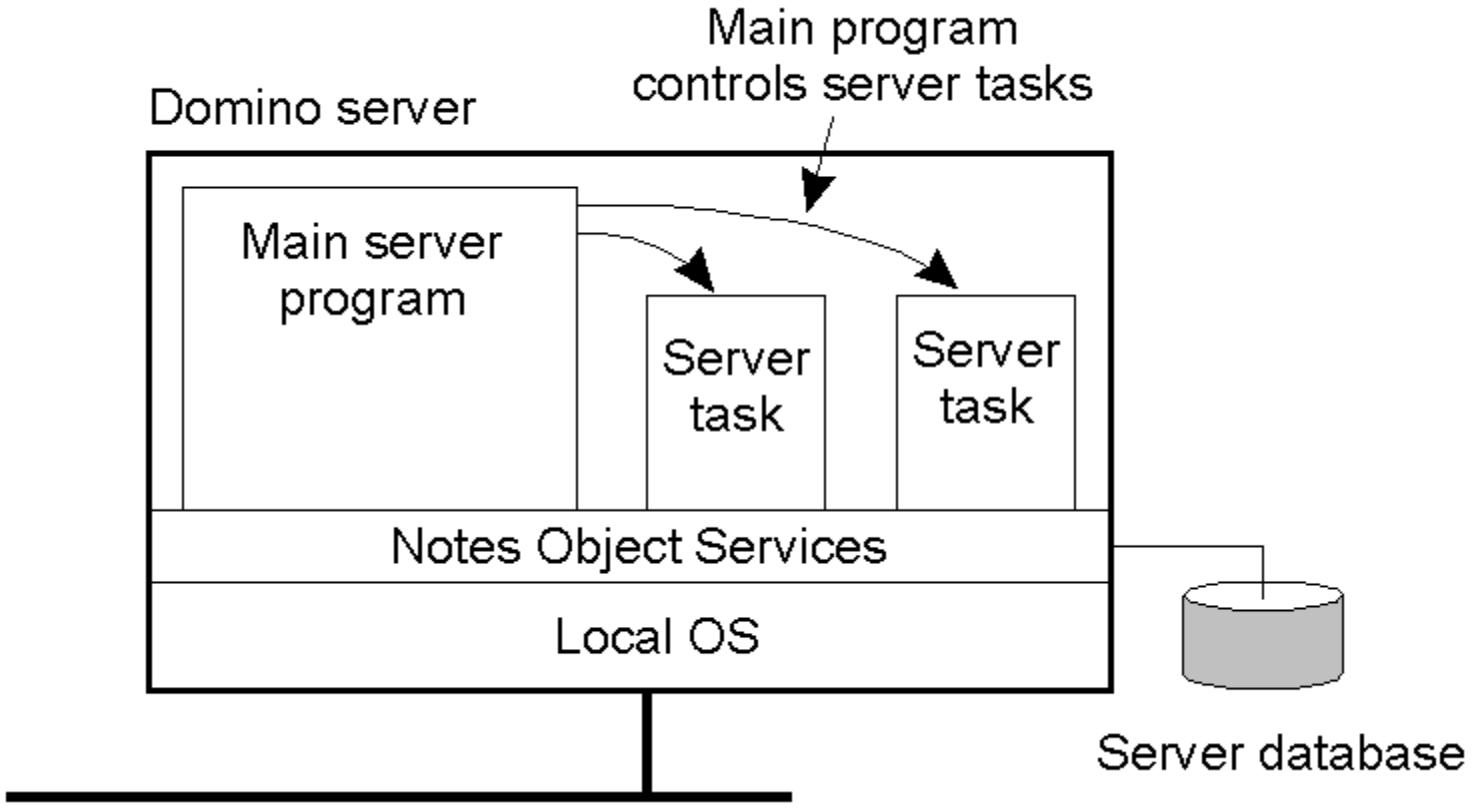
The general organization of a Lotus Notes system.

Document Model

Note type	Category	Description
Document	Data	A user-oriented document such as a Web page
Form	Design	Structure for creating, editing, and viewing a document
Field	Design	Defines a field shared between a form and subforms
View	Design	Structure for displaying a collection of documents
ACL	Administration	Contains an access control list for the database
ReplFormula	Administration	Describes the replication of the database

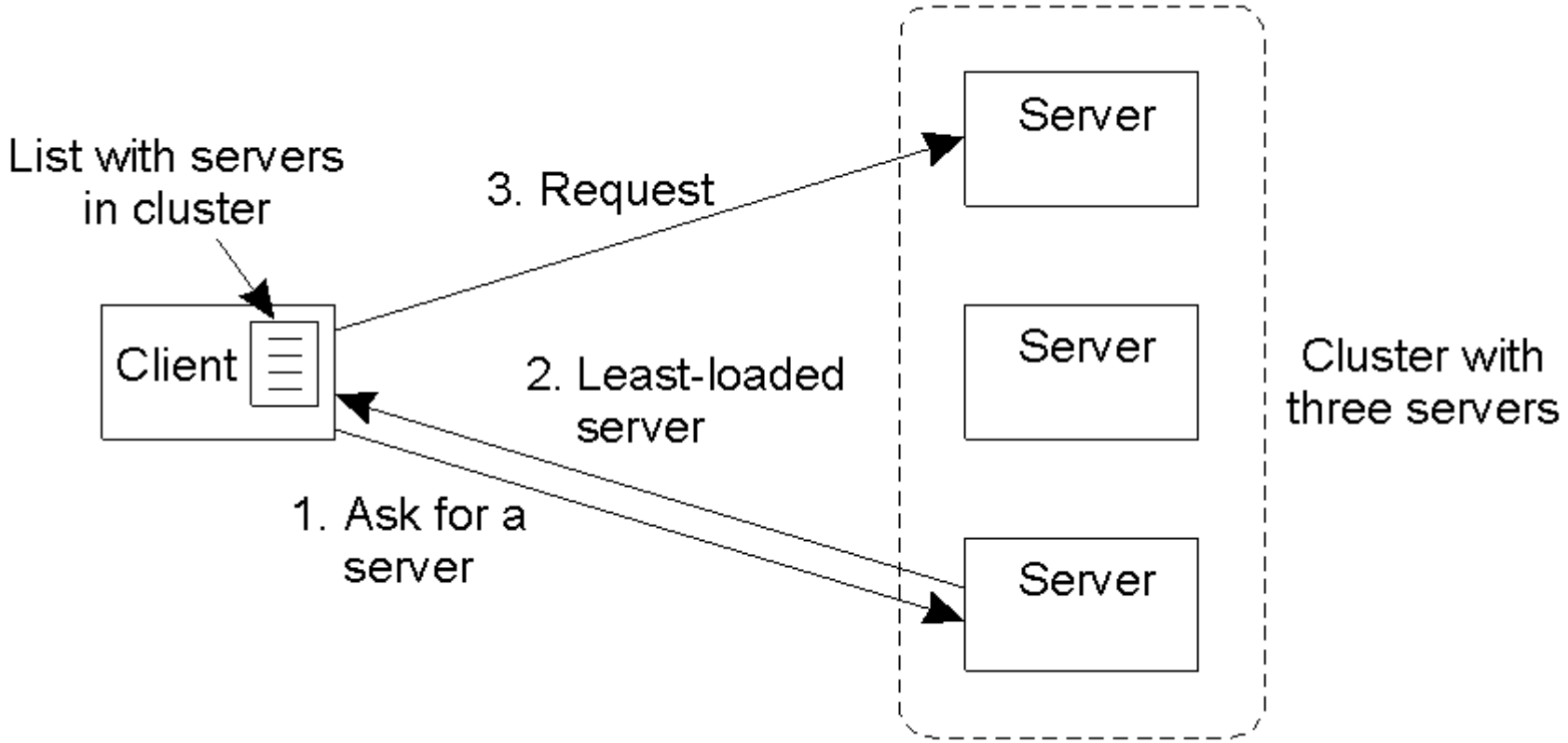
Examples of different types of notes.

Processes (1)



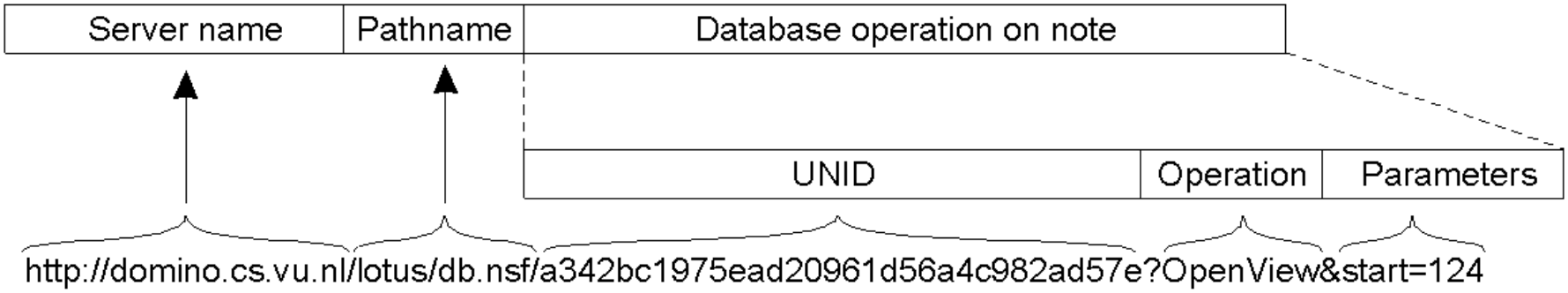
The general organization of a Domino server.

Processes (2)



Request handling in a cluster of Domino servers.

Naming



A Notes URL for accessing a database.

Identifiers

Identifier	Scope	Description
Universal ID	World	Globally unique identifier assigned to each note
Originator ID	World	Identifier for a note, but includes history information
Database ID	Server	Time-dependent identifier for a database
Note ID	Database	Identifier of a note relative to a database instance
Replica ID	World	Timestamp used to identify the same copies of a database

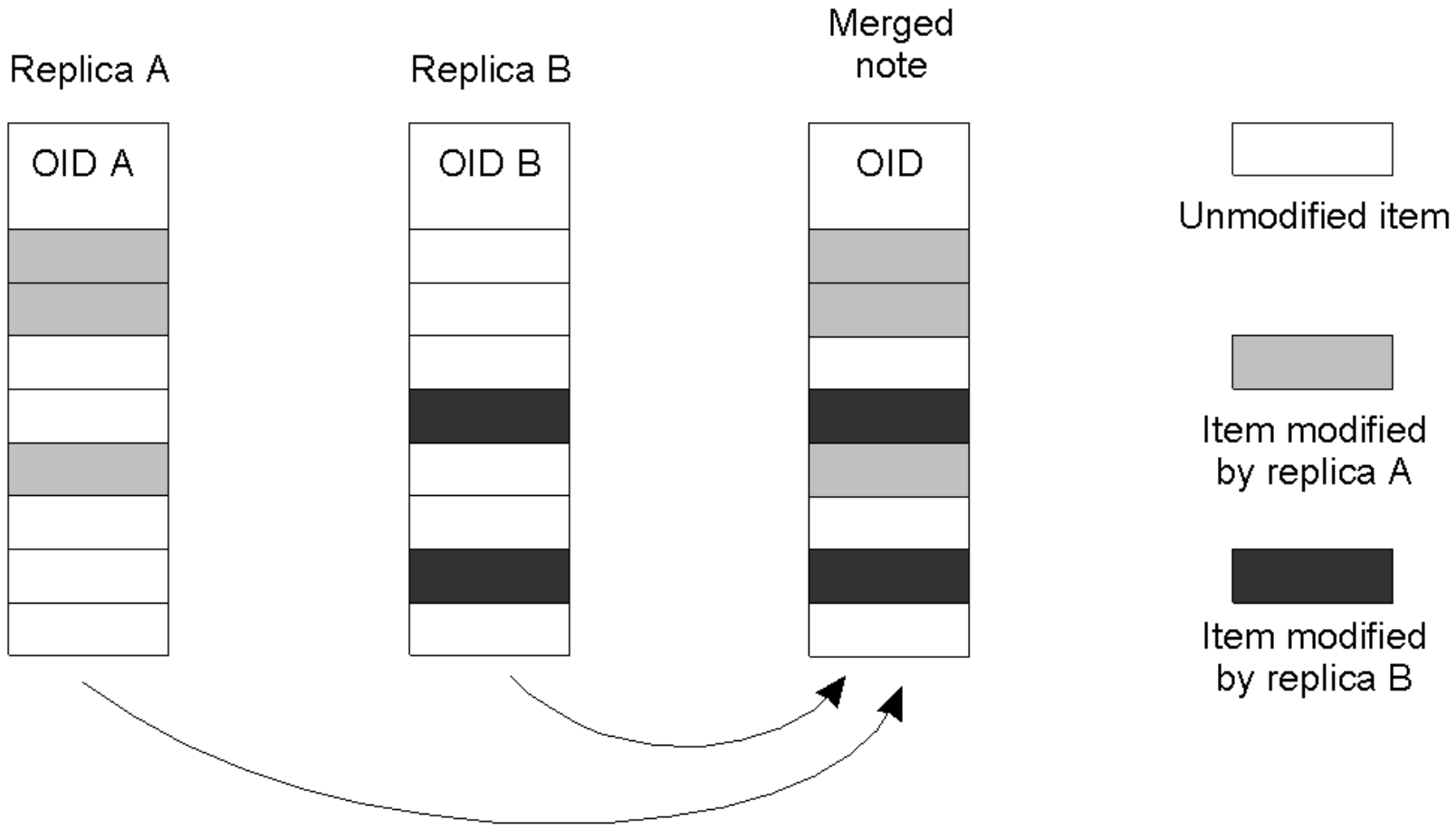
Some major identifiers in Notes.

Replication

Scheme	Description
Pull-push	A replicator task pulls updates in from a target server, and pushes its own updates to that target as well
Pull-pull	A replicator task pulls in updates from a target server, and responds to update fetch requests from that target
Push-only	A replicator task only pushes its own updates to a target server, but does not pull in any updates from the target
Pull-only	A replicator only pulls in updates from a target server, but does not push any of its own updates to that target

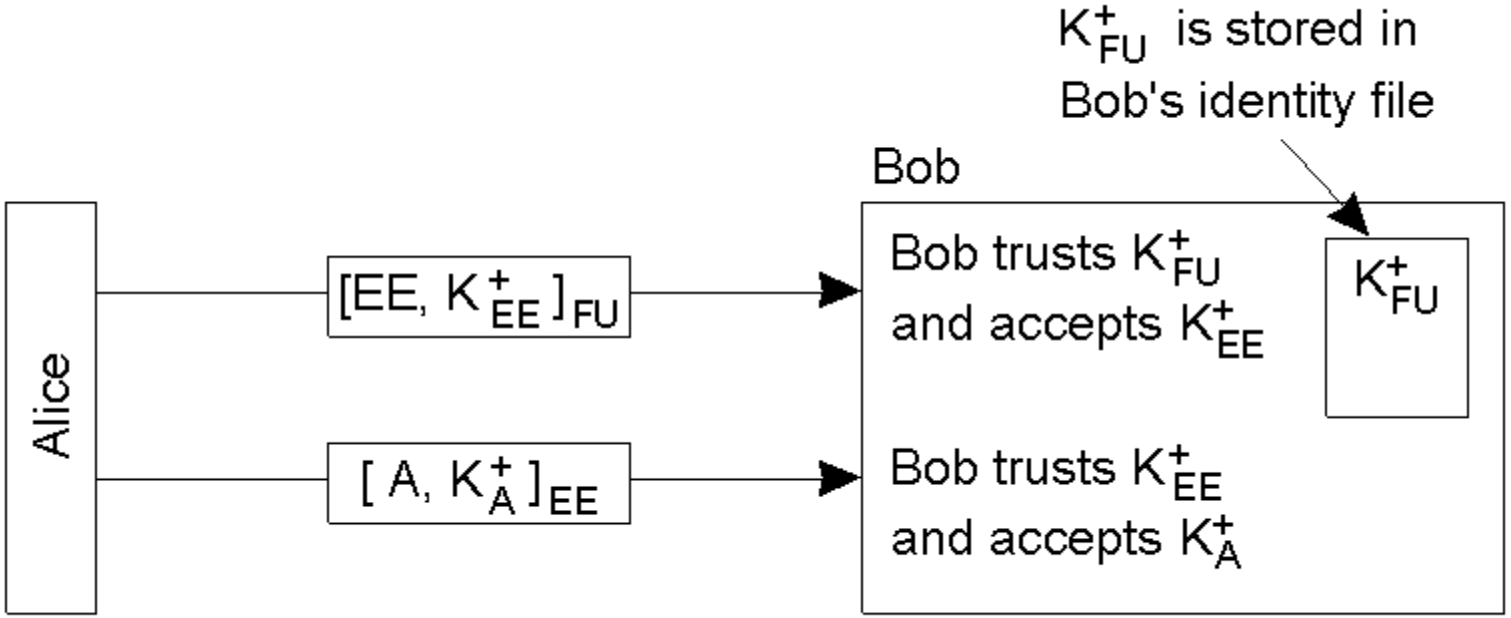
Replication schemes in Notes.

Conflict Resolution



Safely merging two documents with conflicting OIDs.

Authentication: Validating Certificates



Public-key validation in Notes

Access Control

Part	Description
Servers	ACLs specifying access rights for servers and ports
Workstations	Lists specifying execution rights for scripts and such
Databases	ACLs specifying permissions for different types of users
Files	ACLs used for controlling access by Web clients
Design notes	ACLs to control the presentation and such of documents
Documents	ACLs to control read and and write access to documents

Parts in Notes subject to access control.

Comparison of Web & Lotus Notes

Issue	WWW	Notes
Basic model	Marked-up text	List of text items (note)
Extensions	Multimedia, scripts	Multimedia, scripts
Storage model	File oriented	Database oriented
Network comm.	HTTP	RPC, E-mail
Interprocess comm.	Operating sys. dependent	Notes Object Services (NOS)
Client process	Browser, Editor	Browser, Design editor
Client extensions	Plug-ins	In basic client system
Server process	Comparable to file server	Comparable to database server
Server extensions	Servlets, CGI programs	Server tasks
Server clusters	Transparent	Nontransparent
Naming	URNs, URLs	URLs, identifiers
Synchronization	Mainly local	Mainly local
Caching	Advanced	Not documented
Replication	Mirroring, CDNs	Lazy
Fault tolerance	Reliable comm. & clusters	Clusters
Recovery	No explicit support	Single server
Authentication	Mainly TLS	Certificate validation
Access control	Server dependent	Extensive ACLs