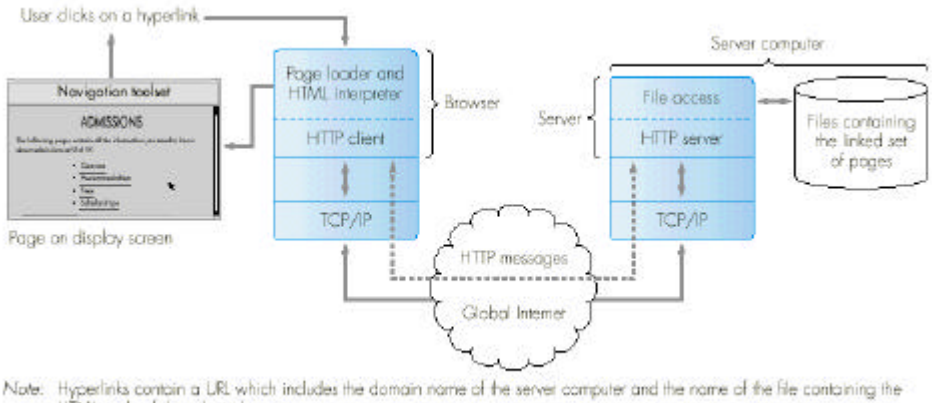
Multimédia e Comunicações

O caso da World Wide Web

conceitos e exemplos de utilização

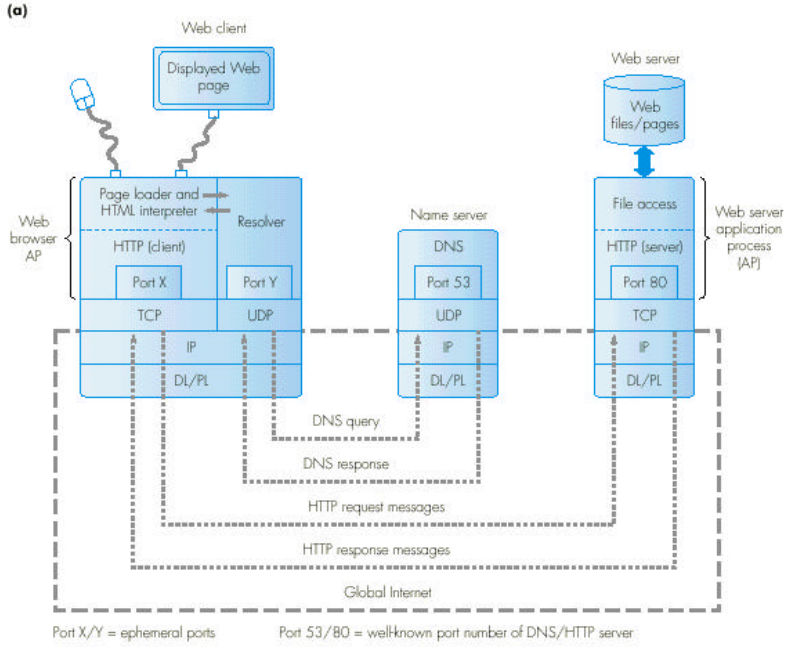
7 de Junho de 2001

(adaptado de Fred Halsall, *Multimedia Communications*, 2001 - Addison Wesley)



Note: Hyperlinks contain a URL which includes the domain name of the server computer and the name of the file containing the HTML code of the selected page

(a)



(b)

HTTP	Use
GET <file name>	Read a Web page/block of data from the named file
HEAD <file name>	Read the header only of the specified Web page
PUT <file name>	Write a Web page or block of data to the named file
POST <file name>	Append a Web page/block of data to that in the named file
DELETE <file name>	Delete the named file

(a) Example request message relating to a URL of

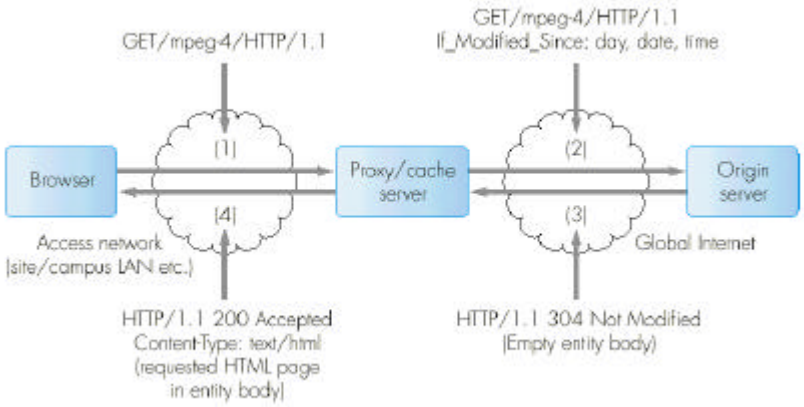
http://www.mpeg.org/mpeg-4/index.html

GET/mpeg-4/index.HTML HTTP/1.1
Connection:close
User-agent: Browser name/version number
Accept: text/html, image/gif, image/jpeg

(b) Example response message relating to this request:

HTTP/1.1 200 Accepted
Server: Aname
Location: www.mpeg.org
Subject: MPEG home page
LastModified: Day/month/year/time
Content-Type: text/html
Content-Length: 7684

Entity body comprising a string of 7684 NVT ASCII characters

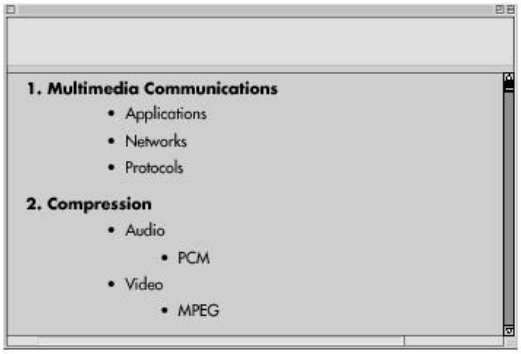


(a)

(b)

```
<HTML>
<HEAD>
<TITLE> U of W Prospectus </TITLE>
</HEAD>
<BODY
<H1> UNIVERSITY OF WONDERLAND </H1>
<BR> Welcome to the University of Wonderland, simply click on one of the
following to find out more about:
<UL><LI><A HREF = "university"> The University </A>
      <LI><A HREF = "academic"> Academic departments </A>
      <LI><A HREF = "research"> Research </A>
      <LI><A HREF = "admissions"> Admissions </A>
</UL>
</BODY>
</HTML>
```

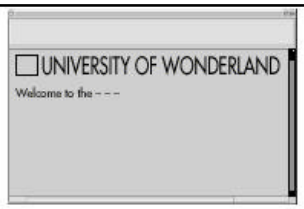
(a)



(b)

```
<HTML>
<HEAD><TITLE> Lists </TITLE></HEAD>
<BODY>
<OL><LI> Multimedia Communications
      <UL><LI> Applications
            <LI> Networks
            <LI> Protocols
      </UL>
      <LI> Compression
      <UL><LI> Audio
            <UL><LI>PCM</UL>
            <LI>Video
            <UL><LI>MPEG</UL>
</OL>
</BODY></HTML>
```

(a)




```
<BODY>
<IMG SRC = "crest.jpeg" ALIGN=LEFT><H1>UNIVERSITY OF WONDERLAND</H1>
<BR> Welcome to the – – –
    •
    •
    •
</BODY>
```

(b)

```
<BODY>
    •
    •
    •
<IMG SRC = "image.gif" ALIGN = MIDDLE> This text starts at the center of the image and
continues until it can start at the left window edge.
    •
    •
    •
</BODY>
```

(a)

(b)

```
<BODY>
    •
    •
    •
<HR>
    •
    •
    •
<HR WIDTH = 50%  ALIGN = CENTER>
    •
    •
    •
<HR SIZE = 5 WIDTH = 75%  ALIGN = RIGHT>
    •
    •
    •
</BODY>
```

(a)

Opening tag	Closing tag	Use
<TABLE>	</TABLE>	Start and end of an unboxed table
<TABLE BORDER>	</TABLE>	Start and end of a boxed table
<TR>	</TR>	Start and end of a row
<TH>	</TH>	Start and end of a heading
<TD>	</TD>	Start and end of a cell content
<CAPTION>	</CAPTION>	Start and end of a table caption

(b)

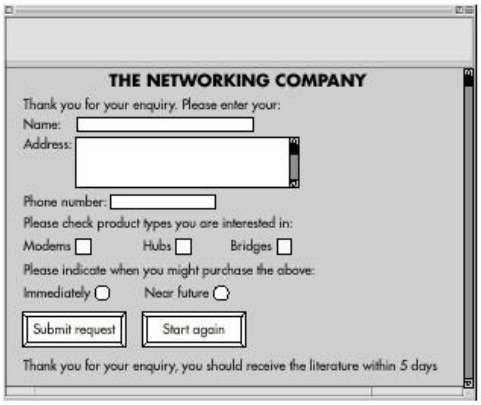
NETWORK	CO/CLS	CBR/VBR
PSTN/ISDN	co	cbr
LAN	cls	vbr
ATM	co	vbr
Internet	cls	vbr

Network operating modes

(c)

```
<HTML><HEAD><TITLE> Example table</TITLE></HEAD>
<BODY>
<TABLE BORDER  ALIGN = CENTER>
<TR><TH>NETWORK</TH><TH>CO/CLS</TH><TH>CBR/VBR</TH></TR>
<TR><TD>PSTN/ISDN</TD><TD> co </TD><TD> cbr </TD></TR>
<TR><TD>LAN</TD><TD> cls </TD><TD vbr </TD></TR>
<TR><TD>ATM</TD><TD> co </TD><TD vbr </TD></TR>
<TR><TD> Internet </TD><TD> cls </TD><TD> vbr </TD></TR>
<CAPTION> Network operating modes </CAPTION>
</TABLE>
</BODY></HTML>
```

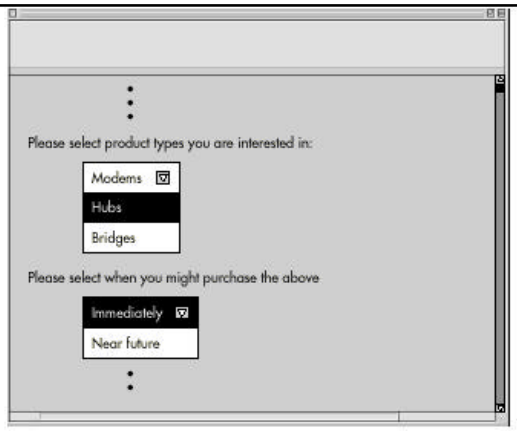
(a)



(b)

```
<HTML><HEAD><TITLE> Literature Request </TITLE></HEAD>
<BODY>
<H1> THE NETWORKING COMPANY </H1>
<FORM ACTION = "http://www.NetCo.com/cgi-bin/literature" METHOD = POST>
Thank you for your enquiry. Please enter your: <P>
Name: <INPUT NAME = "name" SIZE = 30> <P>
Address: <TEXTAREA NAME = "address" ROWS = 3 COLS = 40> </TEXTAREA>
Phone number: <INPUT NAME = "phoneno" SIZE = 20><P>
Please check product types you are interested in: <P>
Modems <INPUT TYPE = CHECKBOX NAME = "ProdType" VALUE = "modems">
Hubs <INPUT TYPE = CHECKBOX NAME = "ProdType" VALUE = "hubs">
Bridges <INPUT TYPE = CHECKBOX NAME = "ProdType" VALUE = "bridges"><P>
Please indicate when you might purchase the above: <P>
Immediately <INPUT TYPE = RADIO NAME = "PurchDate" VALUE = "now">
Near future <INPUT TYPE = RADIO NAME = "PurchDate" VALUE = "later"><P>
<INPUT TYPE = SUBMIT VALUE = "Submit Request">
<INPUT TYPE = RESET VALUE = "Start Again"><P>
</FORM></BODY></HTML>
```

(a)

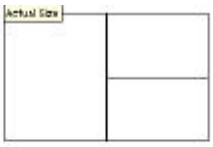


(b)

```
Please select product types you are interested in: <P>
<SELECT NAME = "ProdType" MULTIPLE>
<OPTION> "Modems"
<OPTION SELECTED> "Hubs"
<OPTION> "Bridges"
</SELECT><P>
Please select when you might purchase the above:
<SELECT NAME = "PurchDate">
<OPTION SELECTED> "Immediately"
<OPTION> "Nearfuture"
</SELECT><P>
```

(a) (i)

(ii)



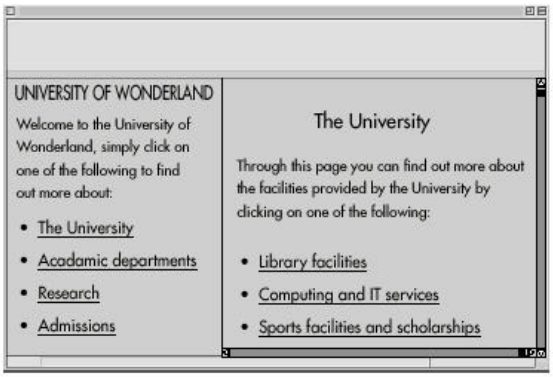
(b) (i)

```
<HTML><HEAD><TITLE>Frames1 </TITLE></HEAD>
<FRAMESET COLS = "50%, 50%">
<FRAMESET ROWS = "50%, 50%">
<FRAME SRC = "URL/local file name of the page to be displayed in the top-left frame">
<FRAME SRC = "URL/local file name of the page to be displayed in the bottom-left frame">
<FRAME SRC = "URL/local file name of the page to be displayed in the right frame">
</FRAMESET></HTML>
```

(ii)

```
<HTML><HEAD><TITLE>Frames2</TITLE></HEAD>
<FRAMESET COLS = "50%, 50%">
<FRAME SRC = "URL/local file name of the page to be displayed in the left frame">
<FRAMESET ROWS = "50%, 50%">
<FRAME SRC = "URL/local file name of the page to be displayed in the top-right frame">
<FRAME SRC = "URL/local file name of the page to be displayed in the bottom-right frame">
</FRAMESET></HTML>
```

(a)



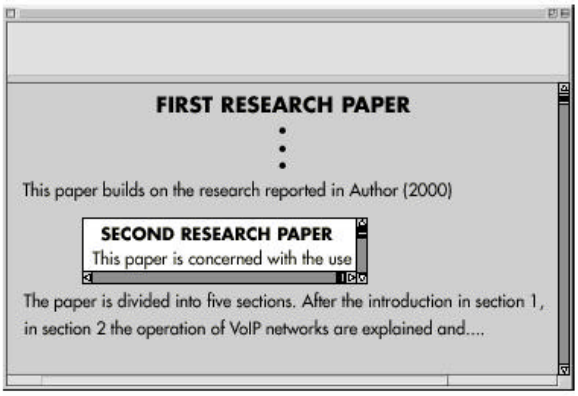
(b)

```
<HTML><HEAD><TITLE> Frames2 </TITLE></HEAD>
<FRAMESET COLS = "35%, 65%">
<FRAME SRC = "http://www.UoW.edu">
<FRAME SRC = "university" NAME = "right">
</FRAMESET></HTML>
```

(c)

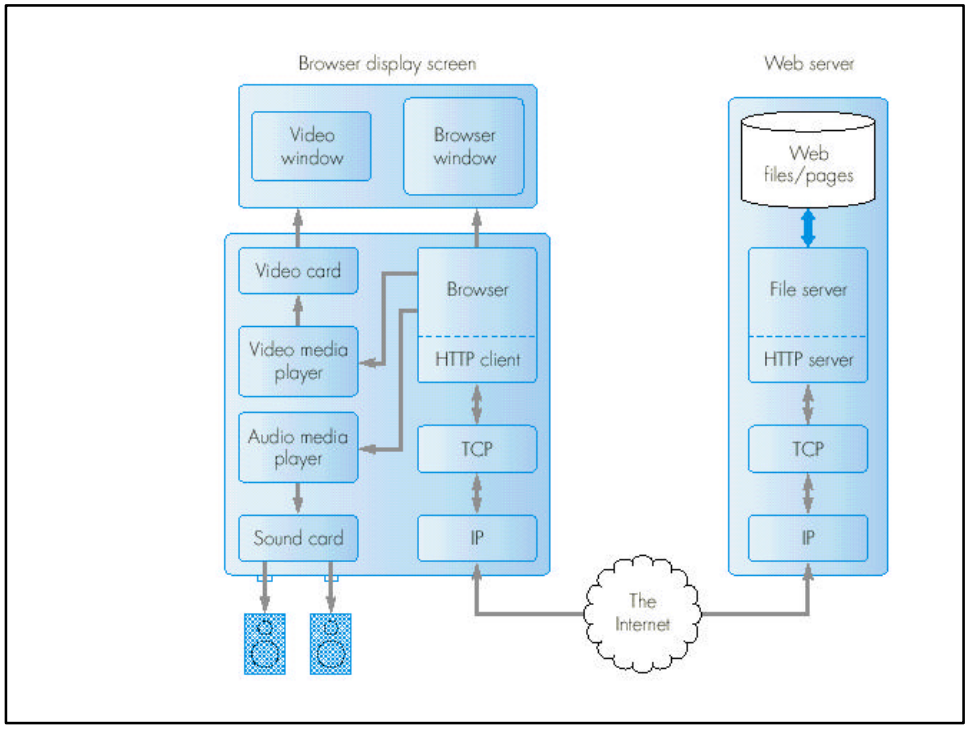
```
    •
    •
    •
<UL><LI><A HREF = "university" TARGET = "right"> The University </A>
    •
    •
    •
    <LI><A HREF = "admissions" TARGET = "right"> Admissions </A>
    •
    •
    •
```

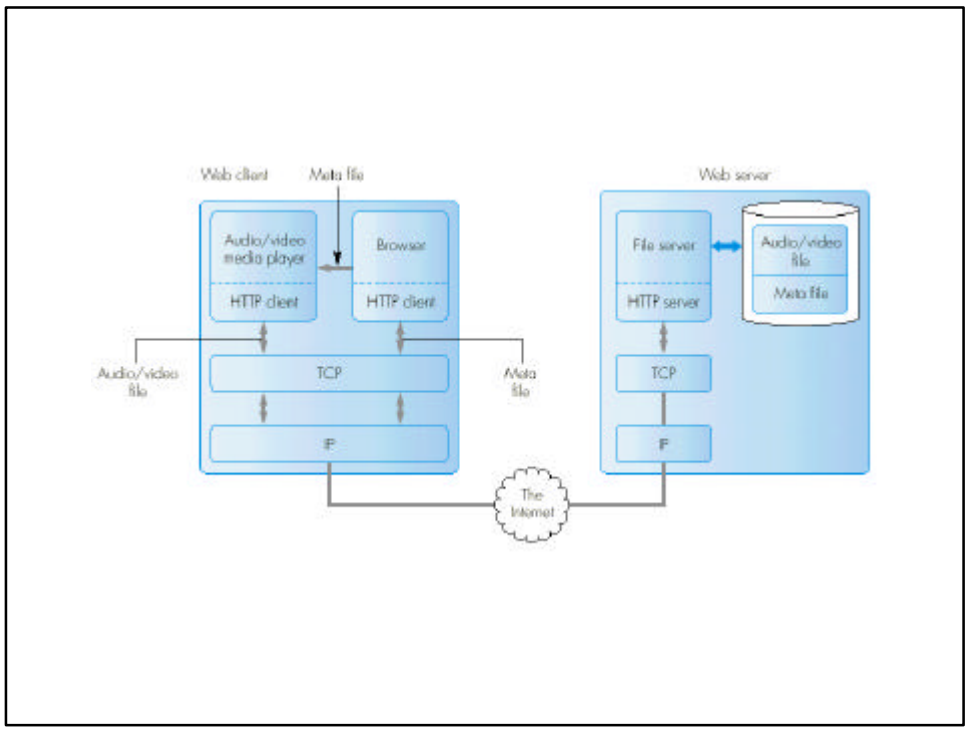
(a)

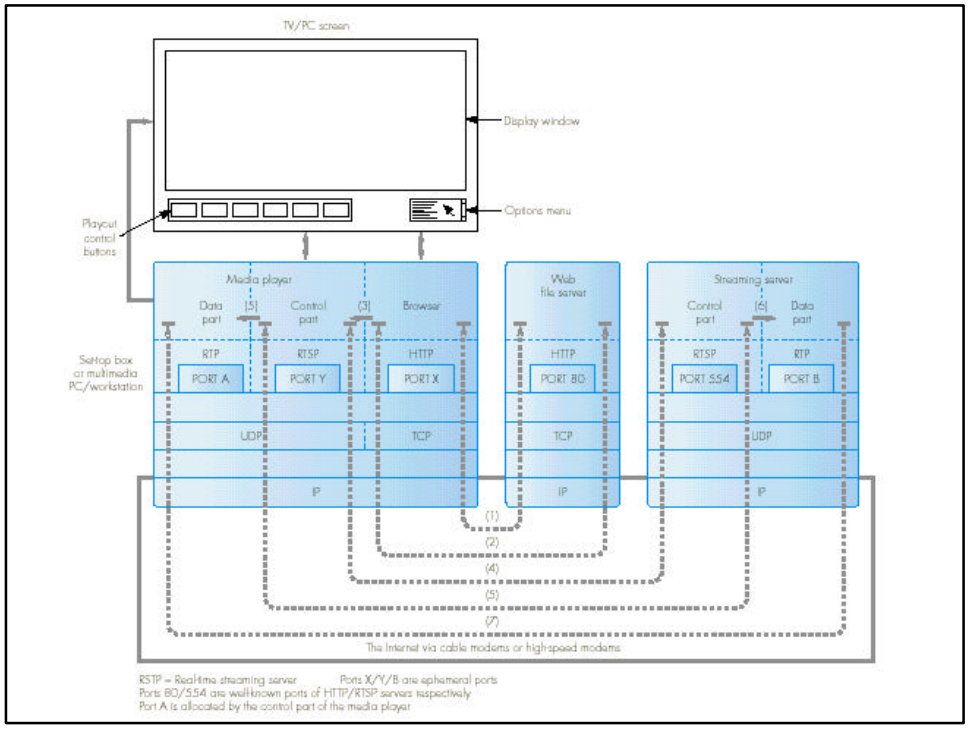


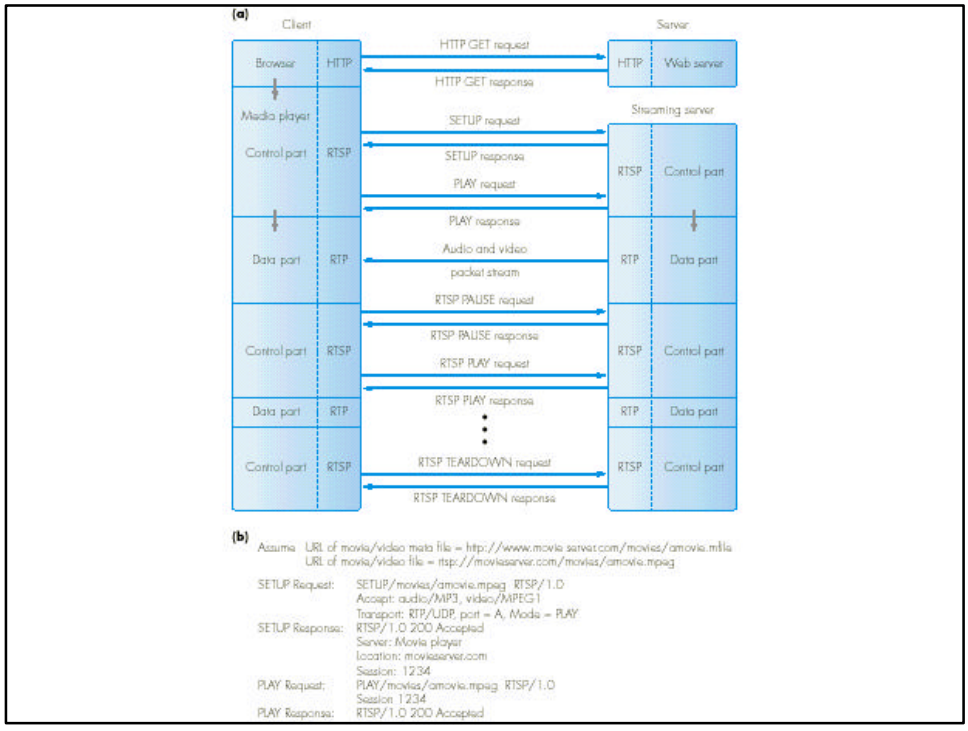
(b)

```
<HTML><HEAD><TITLE> In-line frames </TITLE></HEAD>
<BODY>
<H1> FIRST RESEARCH PAPER </H1><P>
    •
    •
    •
This paper builds on the research reported in Author (2000) <P>
<IFRAME SRC = " URL of page Author (2000)" WIDTH = "600" HEIGHT = "200"
      FRAMEBORDER = "1"></IFRAME><P>
The paper is divided into five sections. After the introduction in section 1, in section 2 the
operation of VoIP networks are explained and...
    •
    •
    •
```

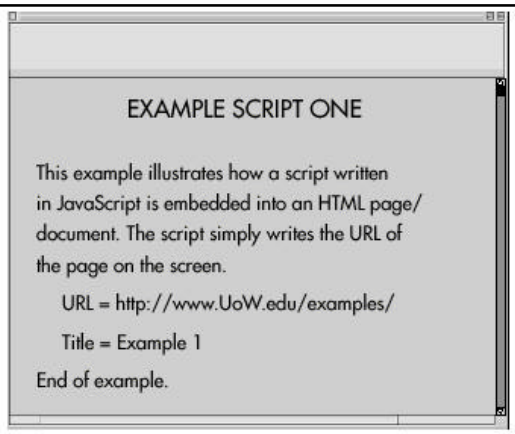

Browser display screen
Web server
Video window
Browser window
Web files/pages
Video card
Browser
File server
Video media player
HTTP client
HTTP server
Audio media player
TCP
TCP
Sound card
IP
IP
The Internet


Web client
Meta file
Web server
Audio/video media player
Browser
File server
Audio/video file
HTTP client
HTTP client
HTTP server
Meta file
Audio/video file
TCP
Meta file
TCP
IP
IP
The Internet


TV/PC screen
Display window
Options menu
Playout control buttons
Media player
Data part
(5)
Control part
(3)
Browser
Web file server
Streaming server
Control part
(6)
Data part
Set-top box or multimedia PC/workstation
RTP
RTSP
HTTP
PORT A
PORT Y
PORT X
HTTP
PORT 80
RTSP
PORT 554
RTP
PORT B
UDP
TCP
TCP
UDP
IP
IP
IP
(1)
(2)
(4)
(5)
(7)
The Internet via cable modems or high-speed modems
RSTP = RealTime streaming server
Ports X/Y/B are ephemeral ports
Ports 80/554 are well-known ports of HTTP/RTSP servers respectively
Port A is allocated by the control part of the media player


(a)
Client
Server
Browser
HTTP
HTTP GET request
HTTP GET response
HTTP
Web server
Media player
Control part
RTSP
SETUP request
SETUP response
PLAY request
PLAY response
Streaming server
RTSP
Control part
Data part
RTP
Audio and video packet stream
RTP
Data part
RTSP PAUSE request
RTSP PAUSE response
RTSP PLAY request
RTSP PLAY response
RTSP TEARDOWN request
RTSP TEARDOWN response
(b)
Assume URL of movie/video meta file = http://www.movie server.com/movies/amovie.mlfile
URL of movie/video file = rtsp://movieserver.com/movies/amovie.mpeg
SETUP Request: SETUP/movies/amovie.mpeg RTSP/1.0
Accept: audio/MP3, video/MPEG1
Transport: RTP/UDP, port = A, Mode = PLAY
SETUP Response: RTSP/1.0 200 Accepted
Server: Movie player
location: movieserver.com
Session: 1234
PLAY Request: PLAY/movies/amovie.mpeg RTSP/1.0
Session 1234
PLAY Response: RTSP/1.0 200 Accepted

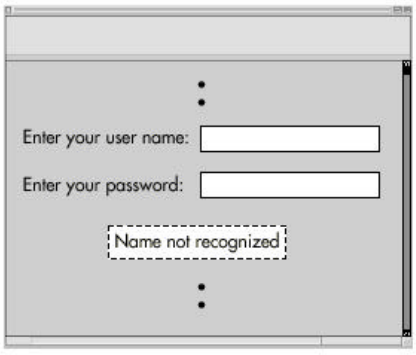
(a)



(b)

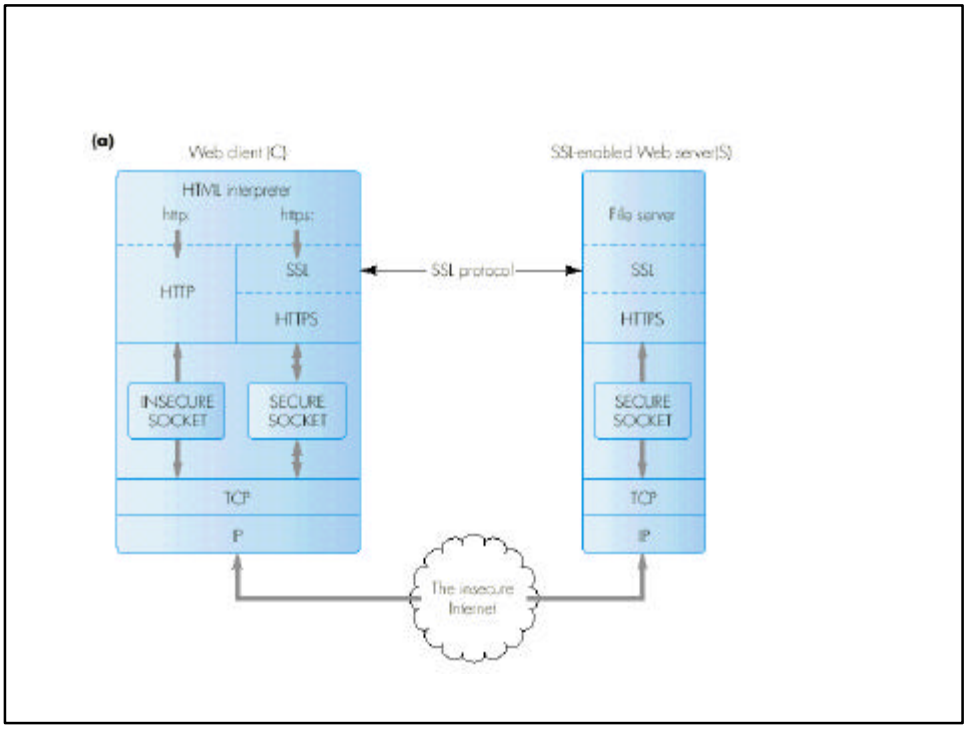
```
<HTML><TITLE> Example 1 </TITLE>
<BODY>
<H1> EXAMPLE SCRIPT ONE </H1>
This example illustrates how a script written in JavaScript is embedded
into an HTML page/document. The script simply writes the URL of the page
on the screen: <P>
<SCRIPT LANGUAGE = "JavaScript">
document.write ("URL =", document.URL) <P>
document.write ("Title =", document.title) </SCRIPT>
<P> End of example.
</BODY></HTML>
```

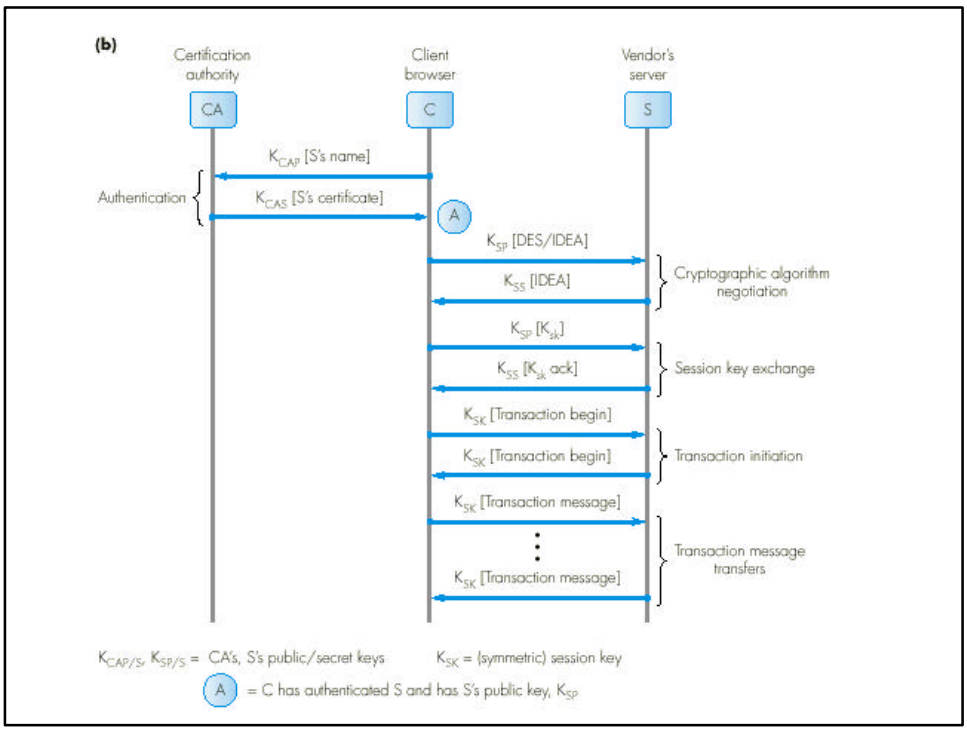
(a)

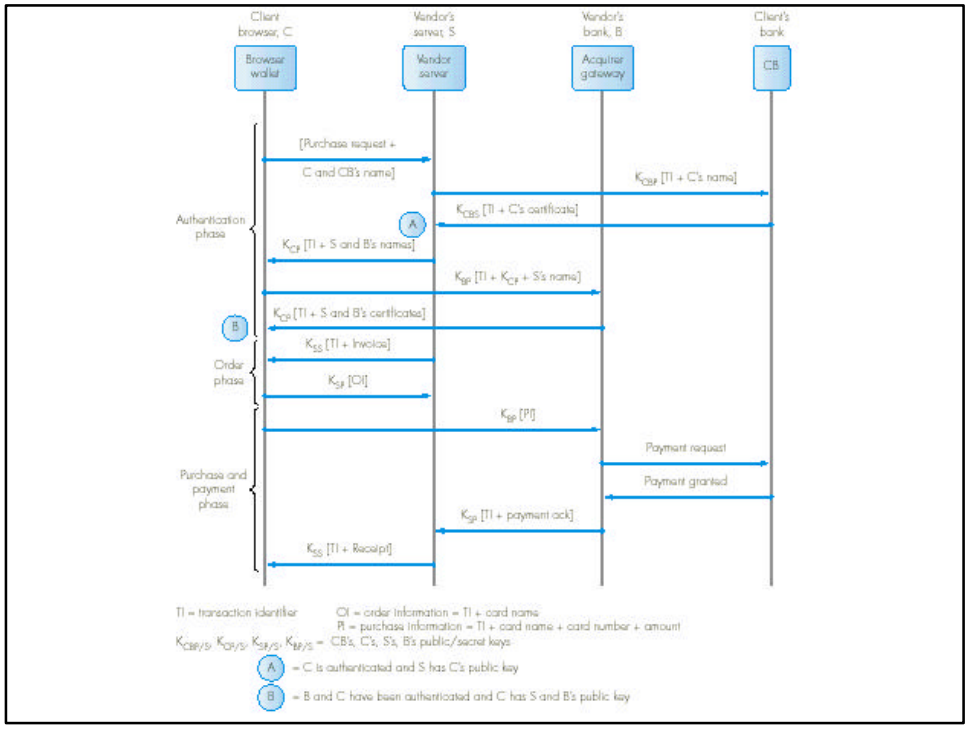


(b)

```
•
•
•
<FORM>
Enter your user name: <INPUT NAME = "username" onBlur = "checkname (username.value)">
Enter your password: <INPUT NAME = "password" onBlur = "checkpassword (password.value)">
</FORM>
•
•
•
<SCRIPT>
function checkname (aname)
{if(aname! = "myname") window.alert ("Name not recognized")}
function check password (apassword)
{if (apassword! = "mypassword") window.alert ("Wrong password")}
</SCRIPT>
```


(a)
Web client (C)
SSL-enabled Web server (S)
HTML interpreter
http:
https:
HTTP
SSL
HTTPS
SSL protocol
File server
SSL
HTTPS
INSECURE SOCKET
SECURE SOCKET
SECURE SOCKET
TCP
IP
TCP
IP
The insecure Internet


(b)
Certification authority
Client browser
Vendor's server
CA
C
S
K_{CAP} [S's name]
Authentication
K_{CAS} [S's certificate]
A
K_{SP} [DES/IDEA]
K_{SS} [IDEA]
Cryptographic algorithm negotiation
K_{SP} [K_{sk}]
K_{SS} [K_{sk} ack]
Session key exchange
K_{SK} [Transaction begin]
K_{SK} [Transaction begin]
Transaction initiation
K_{SK} [Transaction message]
K_{SK} [Transaction message]
Transaction message transfers
$K_{CAP/S}$, $K_{SP/S}$ = CA's, S's public/secret keys
K_{SK} = (symmetric) session key
A = C has authenticated S and has S's public key, K_{SP}


Client browser, C
Vendor's server, S
Vendor's bank, B
Client's bank
Browser wallet
Vendor server
Acquirer gateway
CB
[Purchase request + C and CB's name]
K_{CBB} [TI + C's name]
K_{CBS} [TI + C's certificate]
A
Authentication phase
K_{CS} [TI + S and B's names]
K_{SB} [TI + K_{CS} + S's name]
K_{CB} [TI + S and B's certificates]
B
K_{SS} [TI + Invoice]
Order phase
K_{SB} [OI]
K_{BB} [PI]
Payment request
Payment granted
Purchase and payment phase
K_{SB} [TI + payment ack]
K_{SS} [TI + Receipt]
TI = transaction identifier
OI = order information = TI + card name
PI = purchase information = TI + card name + card number + amount
$K_{CBB/S}$, $K_{CB/S}$, $K_{SB/S}$, $K_{BB/S}$ = CB's, C's, S's, B's public/secret keys
A = C is authenticated and S has C's public key
B = B and C have been authenticated and C has S and B's public key