(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
11 October 2001 (11.10.2001)

PCT

(10) International Publication Number
WO 01/75612 A1

(51) International Patent Classification[7]: G06F 13/00

(21) International Application Number: PCT/US01/10879

(22) International Filing Date: 4 April 2001 (04.04.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/542,120 4 April 2000 (04.04.2000) US

(63) Related by continuation (CON) or continuation-in-part (CIP) to earlier application:
US 09/542,120 (CON)
Filed on 4 April 2000 (04.04.2000)

(71) Applicant *(for all designated States except US)*: **LEARNINGACTION, INC.** [US/US]; 8th Floor, 55 Summer Street, Boston, MA 02110 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **ABRAMOWITZ, Josh** [US/US]; Apartment 1517, 45 Tudor City, New York, NY (US). **AARONSON, Samual, J.** [US/US]; 4 Harvard Avenue, Brookline, MA 02446 (US).

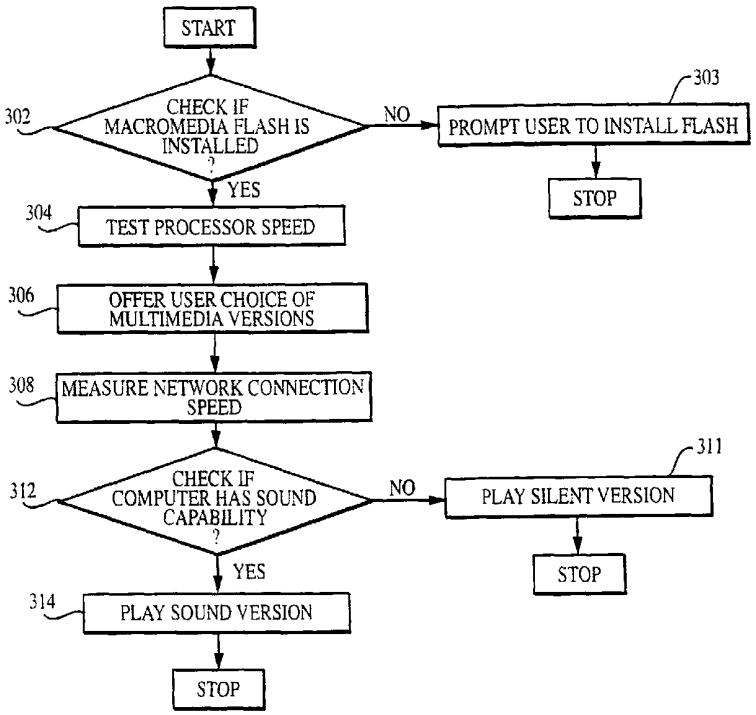
(74) Agent: **BORODACH, Samuel**; Fish & Richardson P.C., 45 Rockefeller Plaza, Suite 2800, New York, NY 10111 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: CHOOSING A MULTIMEDIA PRESENTATION



(57) Abstract: A method of presenting multimedia information on a computer includes: playing a test presentation; determining a processing speed of the computer (304) based on a playing time of the test multimedia presentation; selecting multimedia information based on the determined processing speed (306); and displaying the selected multimedia information (311,314).

WO 01/75612 A1

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

CHOOSING A MULTIMEDIA PRESENTATION

TECHNICAL FIELD

This invention relates to presenting multimedia information, and more particularly to choosing a multimedia presentation.

BACKGROUND

Server computers are often used to present information over a network, such as the Internet, an extranet or an intranet, for display on a client computer. The presented information may be a time-ordered sequence, or stream, of multimedia information, such as image frames captured from a moving object or sound amplitude signals associated with a sound or voice.

Macromedia Flash™, available from Macromedia Inc., is sometimes used to display files representing multimedia streams. The Macromedia files contain a multimedia presentation consisting of a timeline of frames. The frames may contain graphics with associated audio, similar to the frames in a movie clip. Each frame is displayed for pre-determined duration before the next frame in the timeline is displayed. As the timeline advances, the frames are displayed in sequence, creating an animated picture. Alternatively, the frames may contain commands that cause Macromedia Flash™ to perform actions that are associated with the commands.

Different multimedia streams, may have different presentation requirements. For example, a multimedia movie stream with a large number of simultaneously displayed symbols in motion that also has a lot of color effects requires a lot of computer processing power to display, while a movie with a lot of sound and image resolution may require an internet connection with a high bandwidth to download the required high resolution information.

SUMMARY

In general, one aspect of the invention relates to a method of presenting multimedia information on a computer including: playing a test presentation; determining a processing speed of the computer based on a playing time of the test

presentation; selecting multimedia information based on the determined processing speed; and displaying the selected multimedia information.

In general, another aspect of the invention relates to a multimedia presentation, stored on a computer-readable medium, including: a test presentation; a first presentation of multimedia information; a second presentation of multimedia information; and a control file for causing a processor to select and display one of the first and second presentations according to the previously described method.

In general, yet another aspect of the invention relates to a multimedia presentation including a control file for causing a processor to perform the previously described method.

Embodiments of the invention may include one or more of the following. The processing speed of the computer is determined based on whether or not a refresh directive contained within a hypertext markup language file is executed before the playing of the test presentation is completed. The hypertext markup language file includes a link to the test presentation.

The method includes determining a speed of a network prior to selecting the multimedia information and then selecting of the multimedia information is based on the determined speed of the network. The selected multimedia information is loaded over the network before it is displayed. The speed of the network is determined based on a loading time of a test presentation, which is loaded over the network. The method includes playing a test presentation that includes sound information prior to selecting the multimedia information and then determining whether a user heard the sound information. The multimedia information is selected based on whether or not the user heard the sound information. Determining whether the user heard the sound information includes: asking the user whether the sound information was heard and checking the users response.

At least one of the test presentation and the multimedia information is represented in a Macromedia Flash file, Windows Media format, Real Video format, or QuickTime format. The test presentation is a Macromedia Flash presentation for display on a web browser and the method includes determining whether a Macromedia Flash plugin is installed on the web browser, prior to selecting the

multimedia information. The multimedia information is selected based on whether the plugin is installed.

The invention chooses a multimedia information for displaying to a user based on the speed of the processor on the user's computer, the speed of the network connection, the ability of the user's computer to play sound, and so forth. It allows a multimedia developer to create presentations that have different requirements and present each user with the best presentation quality that can be supported by the users computer and network connection.

The details of one or more embodiments of the invention are set forth in the accompanying drawings and the description below. Other features, objects, and advantages of the invention will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

FIG. 1 is a block diagram of a network of computers including a server for presenting multimedia information to client computers;

FIG. 2A is a snapshot of a multimedia stream displayed on a client computer of FIG. 1;

FIG. 2B is a block diagram of the server of FIG. 1;

FIG. 3 is a flow chart of a procedure for choosing a multimedia presentation using a set of test files;

FIG. 4 shows a HTML test file for determining whether a user's computer has a Macromedia Flash plugin installed;

FIG. 5 shows a HTML test file for determining the processing speed of a user's computer;

FIG. 6 shows a web page for prompting a user to select a preferred multimedia presentation version;

FIG. 7A is a flow chart of the procedure for determining the speed of the network;

FIG. 7B is a flow chart of the procedure for determining whether or not to include sound when playing a high bandwidth presentation;

FIG. 7C is a flow chart of the procedure for determining whether or not to include sound when playing a low bandwidth presentation.

Like reference symbols in the various drawings indicate like elements;

DETAILED DESCRIPTION

Referring to Figs. 1, a server 1 presents multimedia information 2 over a network 6 to multiple users 4a-4d. The network 6 includes the Internet 3, an intranet 5, and an extranet (not shown). Alternatively, instead of accessing multimedia information over the network 6, a user may directly access the multimedia information using a web browser on a display (not shown) associated with the server 1.

Referring to Fig. 2A, a user 4a-4d may view the multimedia information 2 on a suitably equipped user computer (not shown) that has a web browser 20 by typing a uniform resource location (URL) associated with the multimedia information in an address input 23 of the web browser 20. The multimedia information 2 includes an animated image 21, text 22 (which may or may not be animated), and a sound signal (not shown). An image 24 that is not animated is also included with the multimedia information 2. The web browser requires additional software, such as a Macromedia Flash plugin, to display the multimedia presentation. Other embodiments may use other plugins or stand-alone programs, such as RealPlayer™ by RealNetworks Inc., a Windows Media Viewer™ by Microsoft Inc, and so forth to display the multimedia information.

Referring to Fig. 2B, the server 1, includes a processor 26 for executing computer programs and storage 27 associated with the processor 26. Storage 27 may be a computer memory, a hard disk, a hard disk array, a tape disk, a floppy disk or a CDROM. Server 1 also includes a multimedia server 25, which is a software program executed by the processor 26 to present multimedia information. The multimedia server 25 program may be stored within storage 27 and retrieved from storage prior to the processor 26 executing the program. The presented multimedia information is

also stored in storage 27. The multimedia information includes multimedia files 28 and test files 29 for determining whether the network 6 and a user's computer meet the presentation requirements of the multimedia files 28.

Referring to Fig. 3, a process, according to the invention, of choosing a multimedia presentation will be described. The process begins by checking 302 whether a Macromedia Flash plugin is installed on the browser 20. If the Flash plugin is not installed, the presentation prompts 303 the user 4a-4d to install Flash and then stops. Otherwise, if the Flash plugin is installed, the process tests 304 the processor speed of the user's computer to determine which presentations can be supported by the processor. The process then offers 306 the user a choice of different multimedia presentation versions that are supported by the processor of the user's computer, and waits for the user to select one of the versions. The process then measures 308 the connection speed of the network 6, to determine the kinds of multimedia presentations that are supported by the network.

The process checks 312 whether the user's computer is capable of playing sound. If the user's computer is incapable of playing sound, the process plays 311 a silent version of the multimedia presentation that is supported by the processor speed and the network connection speed of the user's computer. Otherwise, if the user's computer is capable of playing sound, the process plays 314 a version of the multimedia presentation that includes sound and is supported by the processor speed and the network connection speed of the user's computer.

Thus, the process automatically chooses a multimedia presentation based on the processor speed of the user's computer, the ability of the user's computer to play sound, and the speed of the network connection. The process can be extended to choose the multimedia presentation based on other known tests which determine, for example, the available memory on the user's computer, the available storage on the user's computer, and so on.

The process for determining whether a Flash plugin is installed on a user's computer is described in great detail in Macromedia Technology note number 14086, which is titled "How to detect the Flash 4 Player without using JavaScript" and can be retrieved over the Internet at the uniform resource location (URL):

http://www.macromedia.com/support/flash/ts/documents/scriptfree_detection.htm

The full text of technology note number 14086 is included in appendix A of this patent application.

Referring to FIG. 4, a hypertext markup language (HTML) file 40 for checking 302 (FIG. 3) whether a Flash plugin is installed on a user's computer will be briefly described. The file 40, includes a link 44 for embedding a multimedia presentation, which is contained within a Flash file 44. The flash file 44 contains a getURL action, which redirects the web browser 20 to display the next web page in the presentation sequence. However, the getURL action is only executed if the web browser 20 has a flash plugin installed. Otherwise, the flash file 44 is never played and the getURL action is never executed.

The HTML file 40 also includes a HTML refresh directive 42 that causes the web browser 20 to wait for a timeout period 42b (10 seconds) and then display a web page 42c, which prompts 303 (FIG. 3) the user to install a Flash plugin. If the getURL action of the flash file 44 is not executed, the refresh directive 42 causes the web browser 20 to display the web page 42c, prompting 303 the user to install a Flash plugin. Otherwise, the getURL action redirects the web browser 20 to display the next web page in the presentation sequence before the timeout period 42b expires, thereby preempting the refresh directive 42. Thus the HTML file 40 determines whether a Flash plugin is installed on the user's computer.

Referring to FIG. 5, a hypertext markup language (HTML) file 50 for testing 304 (FIG. 3) the speed of the processor on a user's computer will be described. The file 50 includes a link 54 for embedding a multimedia presentation, which is contained within a Flash file 54. The flash file 54 contains a multimedia presentation that takes a lot of processor time to display. The flash file may contain a large number of simultaneously displayed symbols in motion with a lot of color effects. After the multimedia presentation, the flash file 54 contains a getURL action, which directs the web browser 20 to display a web page that is associated with computers that have fast processors.

The HTML file 50 also includes a HTML refresh directive 52 that causes the web browser 20 to wait for a timeout period 52b (10 seconds) and then display a web

page 52c. The URL for the web page includes information 52e that indicates to the server 1 that the processor on the user's computer is slow. The URL for the web page also includes a session identifier 52d which the server uses to associate the information about the processor speed with the user.

If the processor on the user's computer is slow, the getURL action of the flash file 54 is not executed before the timeout period 52b expires. Consequently, the refresh directive 52 causes the web browser 20 to display the web page 52c, indicating that the processor on the user's computer is slow. Otherwise, the getURL action redirects the web browser 20 to display the web page associated with a fast processor before the timeout period 52b expires, thereby preempting the refresh directive 52. Thus the HTML file 50 tests 304 (FIG. 3) the speed of the processor on the user's computer.

Referring to Fig. 6, a HTML file 60 for offering 306 (FIG. 3) a choice of multiple presentation versions to a user will be described. The user may choose between a version that includes sound 61, a silent version 62, and a non-animated version 63 by using a mouse pointer associated with the users computer on the selected choice. For example, a user in a public place, such as a library, may choose a silent version if the sound from the presentation is likely to disturb others.

Referring to Fig. 7A, the process for measuring 308 (FIG. 3) the speed of the network connection will be described. The process begins by initiating 701 the loading of a big test multimedia presentation file. The process then waits 702 for predetermined load time and then checks 703 whether the loading of the big test file is complete. If the loading is complete, then the speed of the network 6 is high. The process continues as described for FIG. 7B below. Otherwise, if the loading is incomplete, the process unloads 704 the big test file and initiates 705 the loading of a small test file. The process waits 706 for a predetermined load time and then checks 707 whether the loading of the small test file is complete. If the loading of the small test file is complete, the process continues as described for FIG. 7C below. Otherwise, if the loading is incomplete then the network connection speed is very low, i.e. the bandwidth is very low. The process plays 708 a very low bandwidth version of the presentation with no sound and then stops.

Referring to Fig. 7B and 7C, the process for checking 312 (FIG. 3) the sound capability of the user's computer will be described. The FIGs are similar, except Fig. 7B depicts the process for a high-speed network connection while FIG. 7C depicts the process for a low-speed network connection. The process begins by initiating 731, 761 the playing of a test file that was previously loaded 701, 705 and then waiting 732, 762 for a predetermined play time. The process then asks 733, 763 the user whether the sound associated with the test file was heard by the user. If the sound was not heard, a silent version of the presentation is loaded 734, 764 and played 735, 765. Otherwise, if the sound was heard, a version of the presentation that has sound is loaded 736, 766 and played 737, 767.

In implementing the presentation using Macromedia flash, presentation files are loaded and unloaded by, respectively, executing the "load movie" and "unload movie" commands of Macromedia Flash. The status of the loading of a file can be checked using the "If frame loaded" clause of Macromedia flash. The processes described for FIGs. 7A-7C may be implemented in Macromedia Flash control file using commands contained within a sequence of multimedia frames. Alternatively, the processes may be implemented as separate standalone programs.

Other embodiments are within the scope of the following claims.

APPENDIX B

Title: CHOOSING A MULTIMEDIA PRESENTATION

Applicant: LearningAction, Inc.

Pages: 26

STREAMING MULTIMEDIA INFORMATION

TECHNICAL FIELD

This invention relates to presenting information, and more particularly to streaming multimedia information.

BACKGROUND

Server computers are often used to present information over a network, such as the Internet, an extranet or an intranet, for display on a client computer. The presented information may be a time-ordered sequence, or stream, of multimedia information, such as image frames captured from a moving object or sound amplitude signals associated with a sound or voice. The presenting of a stream of multimedia information is sometimes referred to as streaming.

Macromedia Flash™, available from Macromedia Inc., is sometimes used to display files representing multimedia streams. The Macromedia files contain a multimedia presentation consisting of a timeline of frames. The frames may contain graphics with associated audio, similar to the frames in a movie clip. Each frame is displayed for a time period associated with the timeline before the next frame in the timeline is displayed. As the timeline advances, the frames are displayed in sequence, creating an animated picture. Alternatively, the frames may contain commands that cause Macromedia Flash™ to perform actions that are associated with the commands.

An upper mutlimedia stream may be overlapped on a lower multimedia stream to obstruct the lower stream. The graphics displayed on the upper stream may contain a transparent portion to allow a user to view a corresponding portion of the lower stream through the transparent portion. The ordering of overlapped streams is determined using a numeric property of the streams known as a "level", which is specified by the creator of the presentations. A stream with a particular level overlaps all other streams that have a lower level than the particular level.

SUMMARY

In general, one aspect of the invention relates to a method of presenting multimedia information including: loading a first segment of the multimedia information; displaying the first segment; loading a second segment of the multimedia information, the loading of at least a portion of the second segment being concurrent with the displaying of the first segment; and displaying the second segment after displaying the first segment.

In general, another aspect of the invention relates to a multimedia presentation, stored on a computer-readable medium, including: a first segment of multimedia information, a second segment of multimedia information, and a control file for causing a processor to perform the previously described method.

In general, yet another aspect of the invention relates to a multimedia presentation including a control file for causing a processor to perform the previously described method.

The invention reduces the delay between the end of the playing of the first segment and the beginning of the playing of the second segment by loading at least a portion of the second segment while the first segment is playing. Loading the multimedia information in segments also reduced the total memory or hard disk storage required to display the multimedia information.

Embodiments of the invention may include one or more of the following features. At least one of the first segment and the second segment is loaded over a computer network, such as the Internet, an intranet, or an extranet. The loading of the multimedia information in segments allows the information to be loaded over an interminent network connection that might not allow all the information to be loaded all at once. At least one of the first and second segments is loaded from storage associated with a computer system, which may or may not be connected to a network. The second segment is loaded after the loading of the first segment is completed, thereby reducing the delay before the first segment can be displayed by using the entire loading capacity of the system to load the first segment that will be displayed. At least one of the first segment and the second segment is loaded into a memory associated with a computer system.

At least one of the first and second segments is represented in a Macromedia flash file or in a format selected from a group including Real Video, Quicktime, and Windows Media. The first segment is loaded into a higher level than the second segment so that multimedia information associated with the first segment obstructs multimedia information associated with the second segment for at least a portion of the time when the first segment is being displayed. This allows the second segment to be loaded while the first segment is still playing, without obstructing the first segment. The first segment includes a sequence of frames, the last frame of the sequence being transparent, thereby revealing multimedia information associated with the second segment. Thus the appearance of a seamless transition between the segments of the multimedia presentation is created.

The method further includes unloading the first segment and then loading a third segment of multimedia information subsequent to the unloading of the first segment. By unloading the first segment before loading the third segment, the computing resources, such as memory or disk space required to display the multimedia information is reduced as only two segments are concurrently resident on the computer. The method further includes, prior to displaying the second segment: loading a marker file to indicate that the first segment has been displayed. This allows the multimedia presentation to track the segments that have been viewed by different users of the multimedia presentation. The method further includes: playing the marker file; playing a timer file; checking whether the playing of the marker file is completed before the playing of the timer file is completed. Thus the multimedia presentation can determine whether the playing of the marker file has taken longer than the predetermined time it takes to play the timer file. This allows the multimedia presentation to detect a break in the data path used to load the multimedia segments.

The details of one or more embodiments of the invention are set forth in the accompanying drawings and the description below. Other features, objects, and advantages of the invention will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

FIG. 1 is a block diagram of a network of computers including server for streaming multimedia to client computers;

FIG. 2A is a snapshot of a multimedia stream displayed on a client computer of FIG. 1;

FIG. 2B is a block diagram of the server of FIG. 1;

FIG. 3 is a flow chart of a procedure for streaming multimedia, for example, using a base control file, a main control file, a play-loop file, a timer file, a mark chapter file, and a Macromedia Flash chapter file;

FIG. 4A is a high-level representation of the base control file of FIG. 3;

FIG. 4B is a high-level representation of the play-loop file of FIG. 3;

FIG. 5A is a high-level representation of a startup sequence of the main control file of FIG. 3;

FIG. 5B is a high-level representation of a main loop of the main control file of FIG. 3;

FIG. 6 is a high-level representation of the Macromedia Flash chapter file of FIG. 3.

Like reference symbols in the various drawings indicate like elements;

FIG. 7 is a flow chart of the operations of the mark chapter and the timer files of FIG. 3.

DETAILED DESCRIPTION

Referring to Figs. 1, a server 1 streams multimedia information 2 over a network 6 to multiple users 4a-4d. The network 6 includes the Internet 3, an intranet 5, and an extranet (not shown). Alternatively, instead of accessing multimedia information over the network 6, a user may directly access the multimedia information using a web browser on a display associated with the server 1.

Referring to Fig. 2A, a user 4a-4d may view the multimedia information 2 on a suitably equipped client computer that has a web browser 20 by typing a uniform

resource location (URL) associated with the multimedia information in an address input 23 of the web browser 20. The multimedia information includes an animated image 21, text 22 (which may or may not be animated), and a sound signal (not shown). An image 24 that is not animated is also included with the multimedia information 2. The web browser requires additional software, such as a Macromedia Flash plugin, to display the multimedia presentation. Other embodiments may use other plugins or stand-alone programs, such as RealPlayer™ by RealNetworks Inc., a Windows Media Viewer™ by Microsoft Inc, and so forth, to display the multimedia information 2.

Referring to Fig. 2B, the server 1, includes a processor 40 for executing computer instructions and storage 27 associated with the processor 40. Storage 27 may be a computer memory, a hard disk, a hard disk array, a tape disk, a floppy disk or a CDROM. Server 1 also includes a multimedia server 25, which is a software program executed by the processor 40 to present multimedia information. The multimedia server program may be stored in storage 27 and retrieved prior to being executed by the processor 40. The presented information is also stored in storage 27. The multimedia information contains chapter files 28, a main control file 29, a base control file 30, a timer file 31, a mark chapter file 32 and a play loop file 33.

Each chapter file 28 represents a chapter or a segment of a multimedia presentation associated with the chapter file. The chapter files associated with a particular multimedia presentation are presented in sequence by the server 1 over the network 6 and displayed to the user 4a-4d on the web browser 20. While the web browser 20 is displaying a current chapter file, the subsequent chapter file in the multimedia stream is concurrently downloaded from the server 1 by a processing thread on the client computer. The functions of the other 29-33 multimedia files will be explained later.

Referring to Fig. 3, a process, according to the invention, of streaming multimedia information and displaying the information to a user will be described. The process begins by setting 300 the current level that will be used by the multimedia presentation to a value (30) that is larger than the sum of 3 and the number of chapter files associated with the multimedia presentation. The process also

sets 301 the current chapter being processed to an initial value of one, and initiates 302 the loading of the current chapter into the current level. The current level is a segment of memory associated with an ordering of overlapping multimedia streams. The current level may also be segment of hard disk or hard disk array, e.g., as part of a virtual memory system of a computer operating system. The process checks 303 whether the loading of the current chapter into the current level has been completed, and waits for the loading to be completed before proceeding.

When loading into the current level is completed, the process initiates 304 the playing of the chapter in the current level. Other embodiments may begin playing portions of the current chapter that are already loaded before the entire current chapter is fully loaded, to minimize delays. The process also initiates 305 the loading of the chapter subsequent to the current chapter into the level below the current level. By waiting 303 for the loading 302 of the current chapter to be completed before loading 305 the subsequent chapter, the process uses the network connection optimally to minimize the waiting time before the current chapter can be played.

The playing of the current chapter 304 and the loading of the subsequent chapter 305 are performed by separate processing threads, which are executed virtually concurrently, for example, by a multi-processing operating system or by a parallel processing computer. Thus, the process downloads a subsequent chapter while the current chapter is being viewed, thereby reducing or eliminating the waiting time between the end of the current chapter and the start of the subsequent chapter.

The process checks 306 whether the playing of the chapter in the current level has been completed and waits for the playing to be completed before proceeding. When the playing of the current chapter is completed, the process increments 307 the current chapter and decrements 308 the current level, thereby advancing to the subsequent chapter. The process then notifies 309 the server 1 that a chapter has been completely viewed and goes to step 303 to wait for the loading of the new current chapter, which was previously initiated in step 305.

The process of Fig. 3 can be implemented using a suitably configured set of Macromedia Flash files. The multimedia stream is divided into several chapter files 28 (FIG. 2B), which are configured in a particular way, and a series of control files

29-33 (FIG. 2B), which control the presentation of the chapter files. A base control file 30 initiates the presentation process, while a play-loop file 33 controls the playing of the first chapter file. A main control file 29 controls the start-up sequence of the process, and contains the loop that controls the presentation of subsequent chapter files.

Referring to Fig. 4A, the operations of the base control file 30 will be described. The base control file is the first of the multimedia presentation files to be loaded onto the user's computer 4a-4d. The base control file is loaded into level 0, and is automatically played by Macromedia flash. The base control file loads 401 the main control file 29 into level 1 and then sends 402 a play signal to the main control file. The base control file then stops.

Referring to Fig. 4B, the operations of the play loop file 33 will be described. The main control file 29 plays the first multimedia chapter file of the presentation using the play loop file 33. The play loop file 33 checks 453 to see if the loading of the first multimedia chapter file is done. If the loading is not done, the play loop file 33, sends 454 a play signal to the contents of the level where the first chapter is being loaded and waits for the loading to be completed. When the loading is completed, the play loop file 33 stops.

Referring to Figs. 5A and 5B, the operations of the main control file 29 will be described. The main control file begins by setting 501 the current level and the current chapter to 30 and 1, respectively. The main control file then initiates 502 the loading of the current chapter into the current level. A separate processing thread is executed concurrently with the main control file to perform the actual loading, for example, by a multi-processing operating system or a parallel processing computer. The main control file then loads 503 the play-loop file into level 2 and initiates 504 the playing of the play-loop file (which checks 453 the loading of the current chapter and sends 455 a play signal to the main control file when the loading is complete, as described for Fig. 4B above). The main control file then checks 505 whether a play signal has been received from the first chapter file to indicate that the first chapter file has been fully loaded, and waits for the play signal if it has not been received.

When the play signal is received, the main control file 29 unloads 506 the play loop file from level 2. The control file decrements 507 the current level and increments 507 the current chapter to indicate that it is ready to load the subsequent chapter in the multimedia stream in the level immediately below the most recently loaded chapter. The main control file initiates 508 the loading of the new current chapter in the new current level. The main control file then checks 509 whether it has received a play signal from the first chapter indicating that the first chapter has completed playing. If the play signal has not been received, the main control file waits for the play signal.

When the play signal is received, the main control file 29 initiates 510 the playing of the current level. The main control file then unloads 511 the first chapter file, and loads 511 the timer file 31 and the mark chapter file 32 into levels 2 and 3, to indicate to the server 1 that a chapter has been displayed in full. The main control file continues at step 551 (Fig. 5B).

The main control file 29 checks 551, whether it has received a play signal from the current chapter indicating that the current chapter has been fully loaded, and waits for the play signal if it has not yet received the signal. When the main control file receives the play signal, the control file decrements 552 the current level and increments 552 the current chapter to indicate that it is now ready to load the next chapter in the level below the most recently loaded chapter. The main control file then initiates 553 the loading of the new current chapter into the new current level (which is also done by a separate concurrent thread). The main control file then checks 554 whether it has received a play signal from the previous chapter indicating that the playing of the previous chapter is complete, and awaits the signal if it has not yet been received.

When the play signal is received, the main control file 29 initiates 555 the playing of the new current chapter and unloads 556 the previous chapter from the previous level. The main control file also loads the timer file 31 into level 2 and loads the mark-chapter file 32 into level 3 to indicate to the server 1 that another chapter has been displayed in full. The main control file then proceeds to step 551, and repeats the process of loading and playing chapter files in sequence as described above.

Thus the main control file 29 displays a sequence of chapters in a control loop by immediately commencing the loading of a current chapter file when the loading of a previous file is complete, thereby loading the current file while the previous file is being displayed. The control file also unloads a fully displayed chapter file before loading a new chapter file, thereby, reducing the total memory or storage required to stream the multimedia information.

Referring to Fig. 6A, the operations of a chapter file 28 will be described. Upon the initiation of the loading of a chapter file 28, the chapter file displays 601 a pre-loading sequence associated with the chapter file. The preloading sequence may depict an hourglass, indicating that the chapter file is busy preloading. Alternatively, the preloading sequence may depict a sliding scale, showing how much of the chapter file has been loaded. The pre-loading sequence then checks 602 whether the chapter has been fully loaded. If the chapter file has not been fully loaded, the chapter file updates the pre-loading sequence to reflect the state of the loading process and then checks 602 again whether the chapter has been fully loaded.

When the chapter file 28 is fully loaded, the chapter file sends 603 a play signal to the main control file to notify the main control file that the loading is complete and that a subsequent chapter file can now be loaded. Thus the process speeds up the loading of the next chapter file by only loading one chapter file at a time, thereby maximizing the available loading bandwidth. The chapter file then checks 604 whether it has received a play signal from the main control file. If the chapter file has not received a play signal, the chapter file waits for the play signal.

When the chapter file 28 receives a play signal, the chapter file displays 605 the multimedia content associated with the chapter file. The content may be a time ordered sequence of frames which when displayed in sequence depicts an animated picture. After displaying the content, the chapter file displays a transparent frame 606, which allows the user to view multimedia information associated with a chapter file at a lower level than the current level. Since the subsequent chapter file is pre-loaded into a lower level, this allows the user to immediately view the multimedia content of a subsequent chapter when the current chapter comes to an end, thereby creating the appearance of a seamless transition between the chapter files. The

current chapter file then sends 607 a play signal to the main control file to indicate that the all the content has been displayed and stops 608.

Referring to Fig. 7, the operations of the timer 31 and the mark chapter 32 files will be described. The mark chapter file 32 is loaded by the main control loop to indicate to the server 1, that a chapter has been fully viewed. The server 1 maintains a record of what chapters have been viewed by each user. This information can be used to determine whether a user has viewed all the chapter files in a multimedia presentation, which may, for example, be an online class. The timer file 31 is used to determine whether the loading of the mark chapter file 32 has taken longer than the predetermined time require to play the timer file, thereby detecting problems with the network connection to the server 1.

After the mark chapter file 32 is loaded 701 by the main control loop file into level 2, the mark chapter file plays 702. The loading 701 of the mark chapter file informs the server that the user 4a-4d has completed a chapter of the multimedia presentation. The browser may send a load request to the server 1 that contains information about the chapter that has just been played and the identity of the user that is logged onto the browser. The timer file is then loaded 703 by the main control loop file into level 3 (although other embodiments may load the timer file before or concurrently with the mark chapter file). After the timer file has begun to play 704, the timer file checks 705 whether the timer file has completed playing. The timer file takes a known length of time to finish playing. If the timer file has completed playing, the mark chapter file checks 706 whether the mark chapter file has also completed playing. If the mark chapter has also completed playing, the timer file stops. Otherwise, if the timer file completes playing before the mark chapter file, then there is a problem with the connection to the server. The timer file reports 708 the problem with the connection on the web browser 20 to the user. The timer file logs out the user from the server and stops.

If the timer file has not completed playing, the mark chapter file checks 707 whether the mark chapter file has completed playing. If the mark chapter file has not completed playing, the process proceeds to step 705. Otherwise, the mark chapter file unloads 702 the timer file from level 2 and stops. Thus the main control loop informs

the server 1 of the users progress through the multimedia presentation using the mark chapter file. The timer file is used to ensure that the server 1 responds to the mark chapter request.

In implementing the files using Macromedia flash, a person of ordinary skill would know to implement the waiting of a stop signal by pausing the playing of a file so that its processing is automatically resumed on receipt of a Macromedia "tell target" play signal. The files can be loaded and unloaded by, respectively, executing the "load movie" and "unload movie" clauses, while the status of the loading of a file can be checked using the "If frame loaded" clause of Macromedia flash.

The invention can be implemented in digital electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. Apparatus of the invention can be implemented in a computer program product tangibly embodied in a machine-readable storage device for execution by a programmable processor; and method steps of the invention can be performed by a programmable processor executing a program of instructions to perform functions of the invention by operating on input data and generating output. The invention can be implemented advantageously in one or more computer programs that are executable on a programmable system including at least one programmable processor coupled to receive data and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. Each computer program can be implemented in a high-level procedural or object oriented programming language, or in assembly or machine language if desired; and in any case, the language can be a compiled or interpreted language. Suitable processors include, by way of example, both general and special purpose microprocessors. Generally, a processor will receive instructions and data from a read-only memory and/or a random access memory. Generally, a computer will include one or more mass storage devices for storing data files; such devices include magnetic disks, such as internal hard disks and removable disks; magneto-optical disks; and optical disks. Storage devices suitable for tangibly embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory

devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CDROM disks. Any of the foregoing can be supplemented by, or incorporated in, ASICs (application-specific integrated circuits).

A number of embodiments of the invention have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the invention. For example, the multimedia chapter files need not be represented in Macromedia Flash files. They may be represented in Real Video, Real Audio, MPEG, Quick-time, and Windows Media formats, to mention a few. The control file need not contain actual commands. The commands may be written into a computer program that performs a similar function based on a reduced control file that only contains a listing of the chapter files to be streamed. Indeed, the multi-media files do not have to be loaded over a network. They may be loaded from storage, such as a CDROM device, a tape device, a hard disk, or a disk drive array, associated with the user's 4a-4d computer. Accordingly, other embodiments are within the scope of the following claims.

APPENDIX A

Title: CHOOSING A MULTIMEDIA PRESENTATION

Applicant: LearningAction, Inc.

Pages: 3

FLASH
technotes

How to detect the Flash 4 Player without using JavaScript

Product: Flash	Platform: All	Versions: all	ID: 14086

Introduction

To display Flash content, a browser must have the Flash Player installed. Many Operating syste and browsers ship with the Flash Player pre-installed. (See the Flash White Paper for more information.) However, there may be users visiting a Web site with Flash content who don't have the Flash Player. Flash developers must decide among the many options for providing content to all users and creating a smooth flow for those who need to download the Flash Player. Many suc strategies are outlined in the article "Using Flash Player on your Web site" located in the Macromedia Flash Support Center.

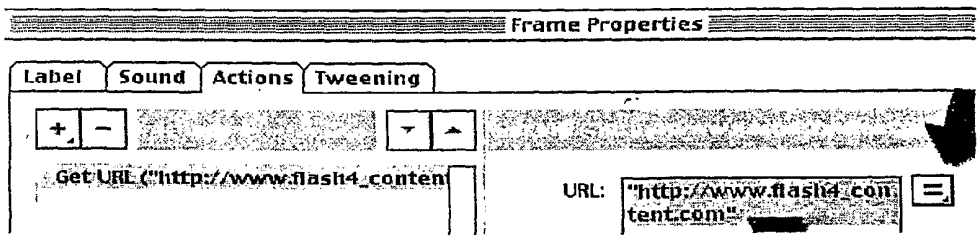
One of the most popular strategies for Flash Player detection is to use JavaScript to detect if the user's browser has the Flash Player installed. Using Flash's Publish Settings and choosing the appropriate template will insert this code for you. However, this technique does not necessarily work on every browser and platform (please refer to What Browsers Support Detection of the Fla Player? Tech Note 12853). This TechNote outlines a technique for detecting the Flash Player th does not require any type of JavaScript; the detection is done using a Flash movie itself.

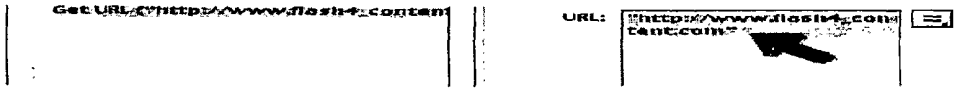
How to create no script detection for the Flash 4 Player

This approach to Flash Player detection relies on two key elements; a Flash plug-in "sniffer" that uses the GetURL action which only the Flash 4 Player will respond to, and an HTML page with a META refresh tag. The index page of the site will use a Flash movie to detect for version 4 of the Flash Player and direct those browsers to the Flash 4 content. If the user hits the index page eith without the Flash Player or with version 2 or 3 of the player, the META refresh tag will automatic send them to a non-Flash page which will prompt them to download version 4 of the Flash Player This scheme works without using any JavaScript; making it both simpler and more reliable.

Creating a Flash 4 sniffer:

1. Create a new Flash movie.
2. In Modify > Movie set the movie's dimensions to 18px wide and 18px high. This is the smallest a Flash movie can be. Since this movie's only function is to check for the presence of the Flash Player we want to keep it as small as possible. This SWF will be in the index.html page for the site. If the URL a browser is searching for is to a directory such as (http:www.my_site.com), not a particular html file (http://www.my_site.com/sniffer.html), it will launch the file in the root directory named "index.html."
3. Make the movie two frames long, select frame 2, choose Modify > Frame. In the Frame Properties dialog box attach a Get URL action. In the URL field put the URL to the HTML page that will make up the real Flash site. Most importantly: put the URL in quotes and set the Get URL field to expression.





4 Add another action to this list - Stop. Select OK.

5 Using the Flash 4 Publish feature create an HTML file using the Flash 4 only (default) template. This will be the index.html page for this Flash site.

Creating a META refresh tag in a HTML document:

1 Open the index.html document in an editor such as Macromedia's Dreamweaver.

2 In the <HEAD> tag of the HTML document create a META refresh tag. This is a tag that tells the browser to refresh to another URL after a set amount of time. The tag syntax, <META HTTP-EQUIV="Refresh" CONTENT="10; URL=alternative.html">, has two parts that need to be set. The first is CONTENT which determines how many seconds until the browser refreshes, in this case it is set to 10 seconds. The second part is the URL that it will refresh to, in this case it is set to refresh to the HTML document alternative.html. Only the file name is written because the file resides in the same directory as index.html. This is a relative path or it could also be to an absolute path. An absolute path is the entire path to a document. An example would be http://www.my_site.com/alternative.html.

3 The next step is to make the page that the refresh tag directs the browser to. This is the place that tells the viewer that the browser needs to have the Flash 4 Player in order to view the Flash content on this site.

4 This page will have a link to the Macromedia Flash Player download center: http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Version=ShockwaveFlash. The Macromedia 'Get Flash' image for linking to the download center can be downloaded from http://www.macromedia.com/support/shockwave/info/linking/. This page could also have a link to an HTML alternative for the Flash site in addition to the link to the Flash plug-in download center.

Note: On browsers that use the plug-in (Netscape and Internet Explorer for Macintosh), the user may be given a dialog box telling them that they don't have the appropriate plug-in and asking if they would like to view the plug-in directory. If they choose "View plug-in directory," they will be routed to the Macromedia Flash Player Download Center. If they choose "Cancel," the META tag will refresh the page to the alternative.html page. There are many different versions of Web browsers. Each one behave differently and can behave differently based on user settings as well The best advice is to test early and test often on every possible browser/platform combination.

Additional information

For additional information about Flash player detection please refer to Using JavaScript to Detec for the Flash Player or search the TechNotes using the keyword 'detection'.

Keywords: Flash Player, detection, plug-in, browser, ActiveX control, script-free, no script,	
Last updated:	December 7, 1999
Author:	Jeff den Broeder
Area:	Tips & Tricks

What is claimed is:

1. A method of presenting multimedia information comprising:

loading a first segment of the multimedia information;

displaying the first segment;

loading a second segment of the multimedia information, the loading of at least a portion of the second segment being concurrent with the displaying of the first segment; and

displaying the second segment after displaying the first segment.

2. The method of claim 1, wherein at least one of the first segment and the second segment is loaded over a computer network.

3. The method of claim 1, wherein at least one of the first segment and second segment is represented in a Macromedia flash file.

4. The method of claim 1, wherein the first segment is loaded into a higher level than the second segment so that multimedia information associated with the first segment obstructs multimedia information associated with the second segment for at least a portion of the time when the first segment is being displayed.

5. The method of claim 1, wherein the first segment displayed includes a sequence of frames, the last frame of the sequence being transparent, thereby revealing multimedia information associated with the second segment.

6. The method of claim 1, further comprising:

unloading the first segment;

loading a third segment of multimedia information subsequent to said unloading of the first segment.

7. The method of claim 1, wherein at least one of the first and the second segment is represent in a format selected from a group including Real Video, Quicktime, and Windows Media.

8. The method of claim 2, wherein the computer network is selected from a group including the Internet, an intranet, and an extranet.

9. The method of claim 1, wherein at least one of the first segment and the second segment is loaded from a storage associated with a computer system.

10. The method of claim 1, wherein the second segment is loaded after the loading of the first segment is completed.

11. The method of claim 1, wherein at least one of the first segment and the second segment is loaded into a memory associated with a computer system.

12. The method of claim 1, further comprising prior to displaying the second segment:

loading a marker file to indicated that the first segment has been displayed.

13. The method of claim 12, further comprising:

playing the marker file;

playing a timer file;

checking whether the playing of the marker file is completed before the playing of the timer file is completed.

14. A multimedia presentation, stored in a computer-readable medium, comprising:

a first segment of multimedia information;

a second segment of multimedia information; and

a control file for causing a processor to:

load the first segment of the multimedia information;

display the first segment;

load the second segment of the multimedia information, the loading of at least a portion of the second segment being concurrent with the displaying of the first segment; and

display the second segment after displaying the first segment.

15. The presentation of claim 14, wherein at least one of the the first segment and the second segment is loaded over a computer network.

16. The presentation of claim 14, wherein at least one of the control file, the first segment and second segment is represented in a Macromedia flash file.

17. The presentation of claim 14, wherein the first segment is loaded into a higher level than the second segment so that multimedia information associated with the first segment obstructs multimedia information associated with the second segment for at least a portion of the time when the first segment is being displayed.

18. The presentation of claim 17, wherein the first segment includes a sequence of frames, the last frame of the sequence being transparent, thereby revealing multimedia information associated with the second segment.

19. The presentation of claim 14, wherein the control file further causes the processor to:

unload the first segment;

load a third segment of multimedia information subsequent to said unloading of the first segment.

20. The presentation of claim 14, wherein the control file includes a series of commands for performing the loading and displaying the first segment and the second segment.

21. The presentation of claim 14, wherein the control file comprises a list of the first file and the second file, the processor being associated with a computer program configured to perform the loading and displaying of the first and second segment in response to the list.

22. The presentation of claim 14, wherein at least one of the first segment and the second segment is loaded from a storage associated with the processor.

23. The presentation of claim 22, wherein the storage is selected from a group including a CDROM device, a tape device, a hard disk, and a hard disk array.

24. The presentation of claim 22, wherein at least one of the first segment and the second segment is loaded into a memory associated with the processor.

25. The presentation of claim 14, further comprising a base file for causing the processor to load the control file.

26. The presentation of claim 14, further comprising:

a marker file, wherein the control file further causes the processor after

displaying the first segment to:

load the marker file to indicate that the first segment has been displayed.

27. The presentation of claim 14, further comprising:

a timer file, wherein the control file further causes the processor after to:

play the marker file;

play the timer file;

check whether the playing of the marker file is completed before the playing of the timer file is completed.

28. A multimedia presentation controller, stored in a computer-readable medium. comprising:

a control file for causing a processor to:

load a first segment of multimedia information;

display the first segment;

load a second segment of multimedia information, the loading of at least a portion of the second segment being concurrent with the displaying of the first segment; and

display the second segment after displaying the first segment.

What is claimed is:

1. A method of presenting multimedia information on a computer, comprising:

playing a test presentation;

determining a processing speed of the computer based on a playing time of the test presentation;

selecting multimedia information based on the determined processing speed; and

displaying the selected multimedia information.

2. The method of claim 1, wherein the processing speed of the computer is determined based on whether or not a refresh directive contained within a hypertext markup language file is executed before the playing of the test presentation is completed.

3. The method of claim 2, wherein the hypertext markup language file includes a link to the test presentation.

4. The method of claim 1, further comprising, prior to selecting the multimedia information:

determining a speed of a network, the selecting of the multimedia information being based on the determined speed of the network.

5. The method of claim 1, further comprising, prior to displaying the selected multimedia information:

loading the multimedia information over the network.

6. The method of claim 4, wherein the speed of the network is determined by:

loading a test presentation over the network;

determining the speed of the network based on a loading time of the test presentation.

7. The method of claim 1, wherein at least one of the test presentation and the multimedia information is represented in a Macromedia Flash file.

8. The method of claim 1, wherein at least one of the test presentation and the multimedia information is represented in at least one of Windows Media, Real Video and QuickTime.

9. The method of claim 1, further comprising, prior to selecting the multimedia information:

playing a test presentation that includes sound information;

determining whether the sound information was heard by a user, the selecting of the multimedia information being based on whether or not the sound information was heard by the user.

10. The method of claim 9, wherein the determining whether the sound information was heard includes:

asking the user whether the sound information was heard;

checking the users response to the asking.

11. The method of claim 1, wherein the test presentation is a Macromedia Flash presentation for display on a web browser, the method further comprising, prior to selecting the multimedia information:

determining whether a Macromedia Flash plugin is installed on the web browser, the selecting of the multimedia information being based on whether the plugin is installed.

12. A multimedia presentation, stored in a computer-readable medium, comprising:

a test presentation;

a first presentation of multimedia information;

a second presentation of multimedia information; and

a control file for causing a processor to:

play the test presentation;

determine a processing speed of the processor based on a playing time of the test presentation;

select between the first presentation and the second presentation of multimedia information based on the determined processing speed; and

display the selected multimedia information.

13. The presentation of claim 12, wherein the processing speed of the computer is determined based on whether or not a refresh directive contained within a hypertext markup language file is executed before the playing of the test presentation is completed.

14. The presentation of claim 13, wherein the hypertext markup language file includes a link to the test presentation.

15. The presentation of claim 12, further comprising, prior to selecting the multimedia information:

determining a speed of a network, the selecting of the multimedia information being based on the determined speed of the network.

16. The presentation of claim 12, further comprising, prior to displaying the selected multimedia information:

loading the multimedia information over the network.

17. The presentation of claim 15, wherein the speed of the network is determined by:

loading a test presentation over the network;

determining the speed of the network based on a loading time of the test presentation.

18. The presentation of claim 12, wherein at least one of the test presentation and the multimedia information is represented in a Macromedia Flash file.

19. The method of claim 12, wherein at least one of the test presentation and the multimedia information is represented in at least one of Windows Media, Real Video and QuickTime.

20. The presentation of claim 12, further comprising, prior to selecting the multimedia information:

playing a test presentation that includes sound information;

determining whether the sound information was heard by a user, the selecting of the multimedia information being based on whether or not the sound information was heard by the user.

21. The presentation of claim 20, wherein the determining whether the sound information was heard includes:

asking the user whether the sound information was heard;

checking the users response to the asking.

22. The presentation of claim 12, wherein the test presentation is a Macromedia Flash presentation for display on a web browser, the method further comprising, prior to selecting the multimedia information:

determining whether a Macromedia Flash plugin is installed on the web browser, the selecting of the multimedia information being based on whether the plugin is installed.

23. A multimedia presentation controller, stored in a computer-readable medium, comprising:

a control file for causing a processor to:

play a test presentation;

determine a processing speed of the processor based on a playing time of the test presentation;

select between a first presentation and a second presentation of multimedia information based on the determined processing speed; and

display the selected multimedia information.

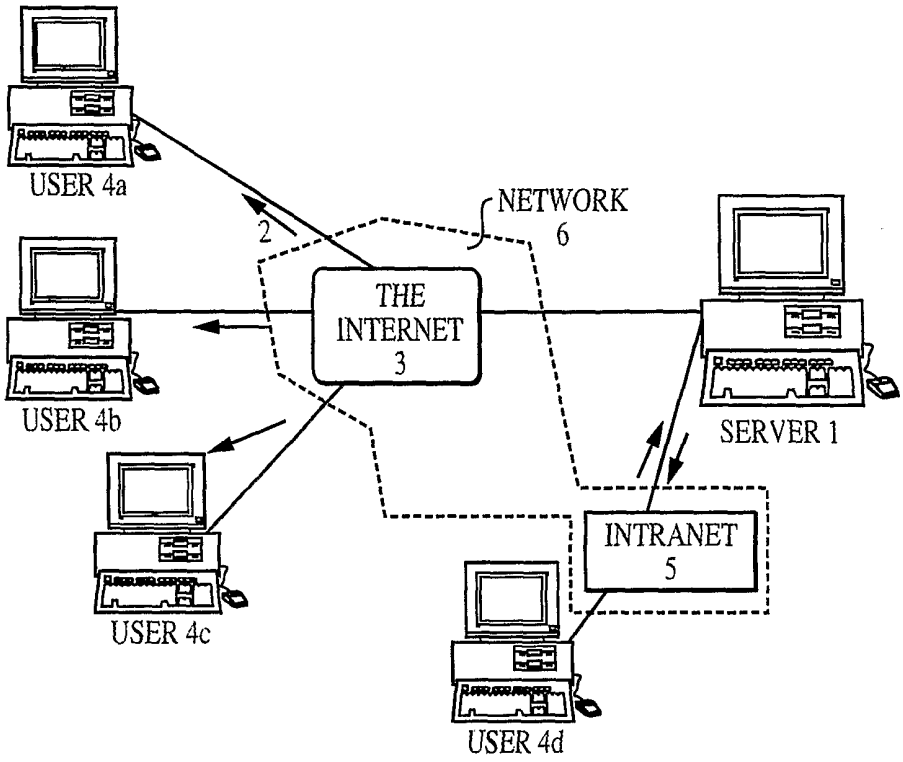

USER 4a
2
NETWORK
6
THE
INTERNET
3
USER 4b
SERVER 1
INTRANET
5
USER 4c
USER 4d

FIG. 1

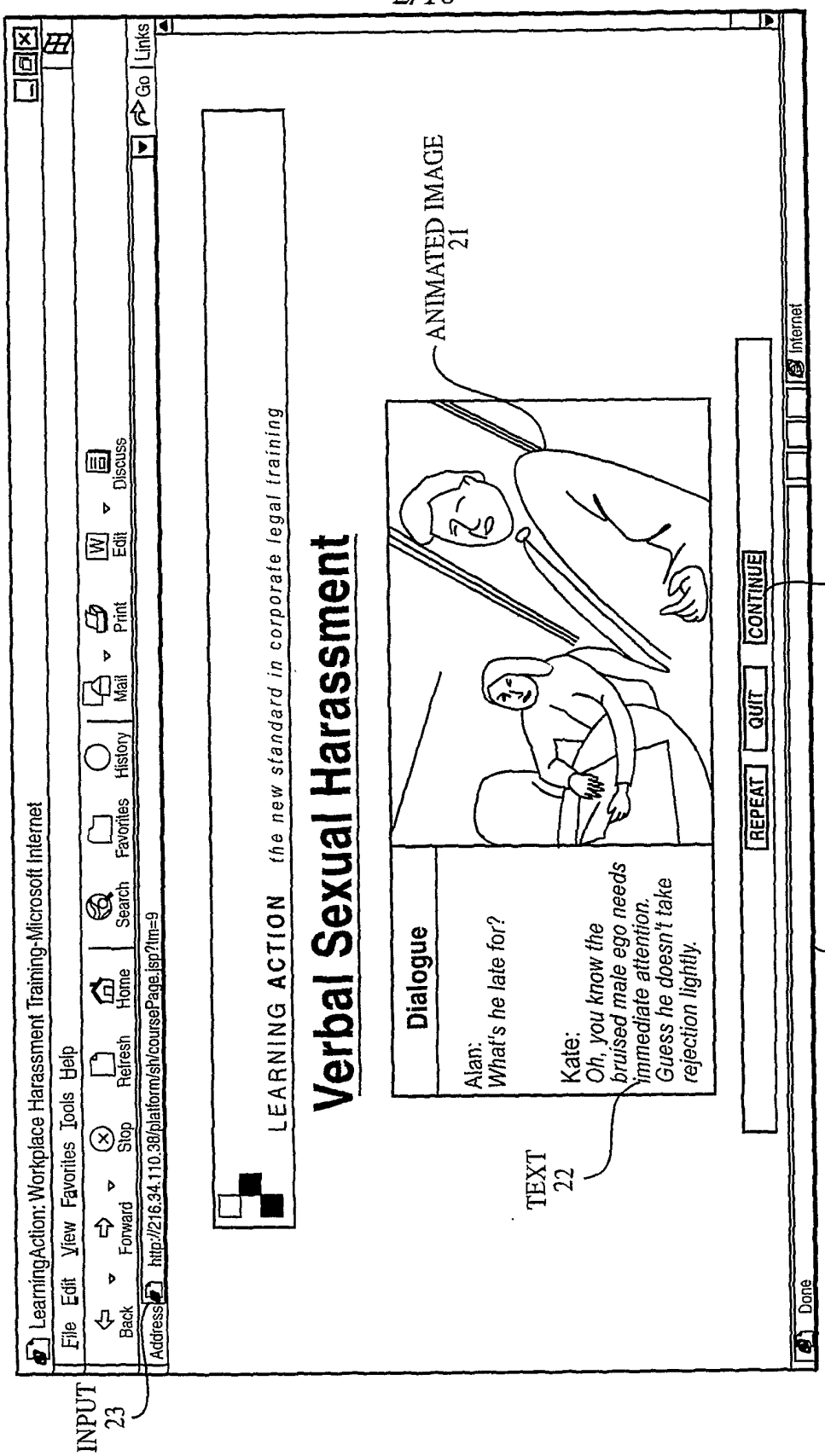

LearningAction: Workplace Harassment Training-Microsoft Internet
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss
Address http://216.34.110.38/platform/sh/coursePage.jsp?lm=9
Links
Go
LEARNING ACTION the new standard in corporate legal training
Verbal Sexual Harassment
Dialogue
Alan:
What's he late for?
Kate:
Oh, you know the bruised male ego needs immediate attention. Guess he doesn't take rejection lightly.
ANIMATED IMAGE
21
TEXT
22
REPEAT
QUIT
CONTINUE
IMAGE
24
Done
Internet
INPUT
23
20

FIG. 2A

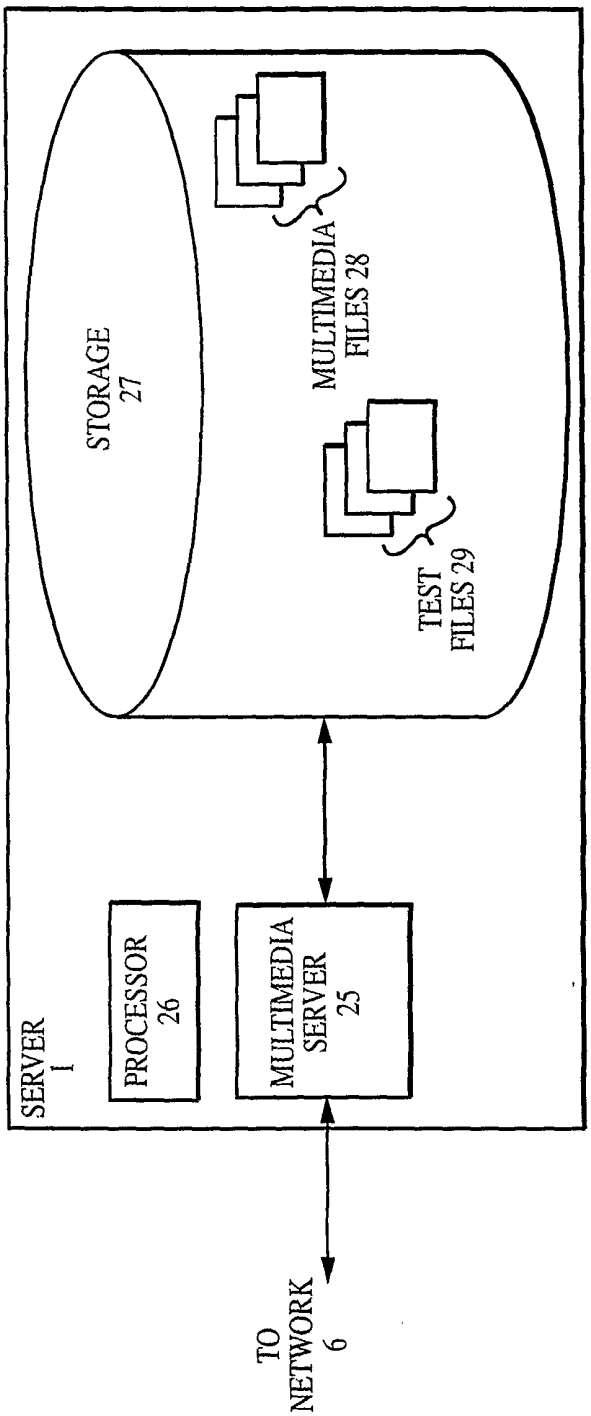

SERVER
1
PROCESSOR
26
MULTIMEDIA
SERVER
25
STORAGE
27
MULTIMEDIA
FILES 28
TEST
FILES 29
TO
NETWORK
6

FIG. 2B

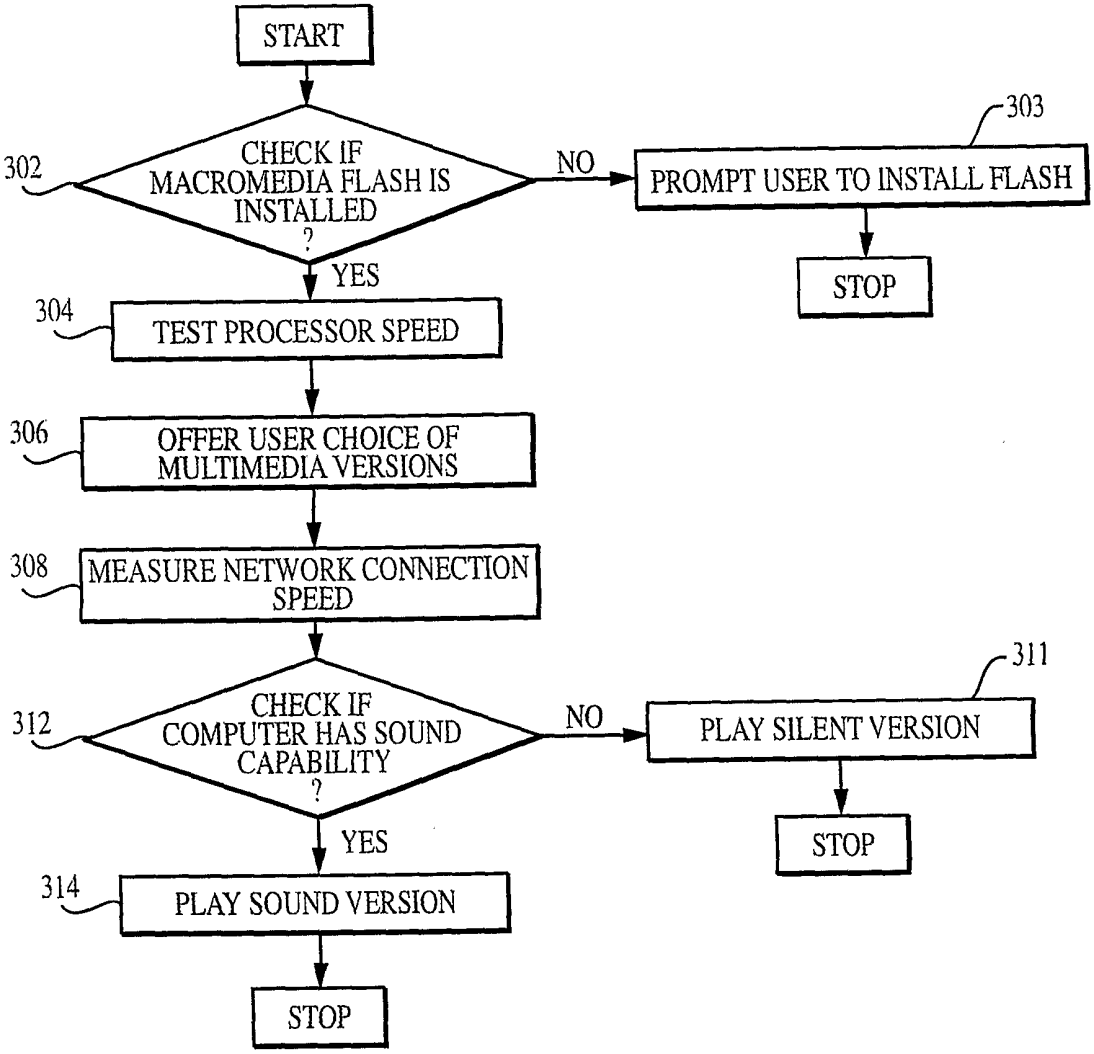

START
302
CHECK IF MACROMEDIA FLASH IS INSTALLED ?
NO
303
PROMPT USER TO INSTALL FLASH
STOP
YES
304
TEST PROCESSOR SPEED
306
OFFER USER CHOICE OF MULTIMEDIA VERSIONS
308
MEASURE NETWORK CONNECTION SPEED
312
CHECK IF COMPUTER HAS SOUND CAPABILITY ?
NO
311
PLAY SILENT VERSION
STOP
YES
314
PLAY SOUND VERSION
STOP

FIG. 3

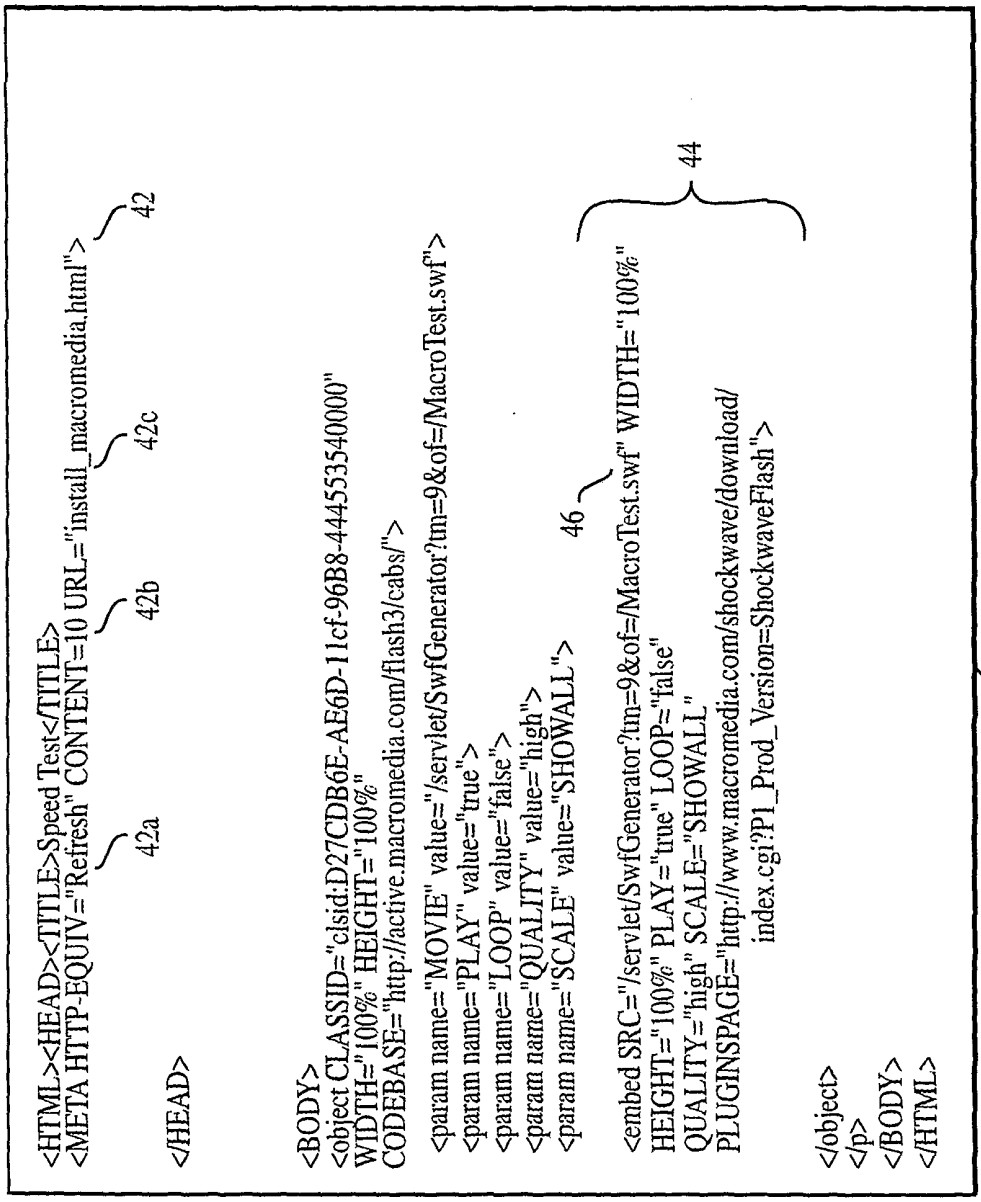

<HTML><HEAD><TITLE>Speed Test</TITLE>
<META HTTP-EQUIV="Refresh" CONTENT=10 URL="install_macromedia.html">
42a
42b
42c
42
</HEAD>
<BODY>
<object CLASSID="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
WIDTH="100%" HEIGHT="100%"
CODEBASE="http://active.macromedia.com/flash3/cabs/">
<param name="MOVIE" value="/servlet/SwfGenerator?tm=9&of=/MacroTest.swf">
<param name="PLAY" value="true">
<param name="LOOP" value="false">
<param name="QUALITY" value="high">
<param name="SCALE" value="SHOWALL">
46
<embed SRC="/servlet/SwfGenerator?tm=9&of=/MacroTest.swf" WIDTH="100%"
HEIGHT="100%" PLAY="true" LOOP="false"
QUALITY="high" SCALE="SHOWALL"
PLUGINSPAGE="http://www.macromedia.com/shockwave/download/
index.cgi?P1_Prod_Version=ShockwaveFlash">
44
</object>
</p>
</BODY>
</HTML>
40

FIG. 4

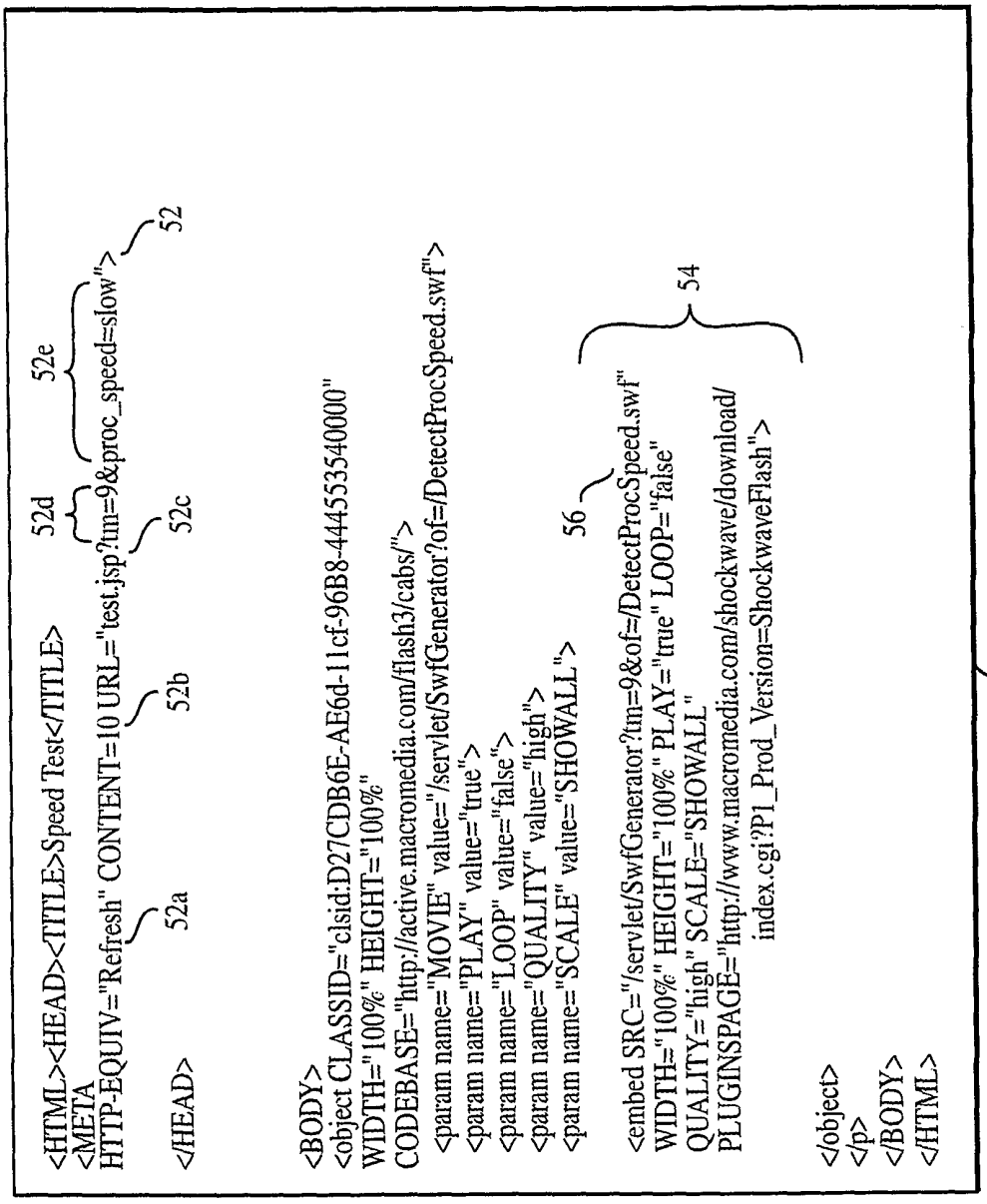

<HTML><HEAD><TITLE>Speed Test</TITLE>
<META
HTTP-EQUIV="Refresh" CONTENT=10 URL="test.jsp?tm=9&proc_speed=slow">
</HEAD>
<BODY>
<object CLASSID="clsid:D27CDB6E-AE6d-11cf-96B8-444553540000"
WIDTH="100%" HEIGHT="100%"
CODEBASE="http://active.macromedia.com/flash3/cabs/">
<param name="MOVIE" value="/servlet/SwfGenerator?of=/DetectProcSpeed.swf">
<param name="PLAY" value="true">
<param name="LOOP" value="false">
<param name="QUALITY" value="high">
<param name="SCALE" value="SHOWALL">
<embed SRC="/servlet/SwfGenerator?tm=9&of=/DetectProcSpeed.swf"
WIDTH="100%" HEIGHT="100%" PLAY="true" LOOP="false"
QUALITY="high" SCALE="SHOWALL"
PLUGINSPAGE="http://www.macromedia.com/shockwave/download/
index.cgi?P1_Prod_Version=ShockwaveFlash">
</object>
</p>
</BODY>
</HTML>
52a
52b
52c
52d
52e
52
54
56
50

FIG. 5

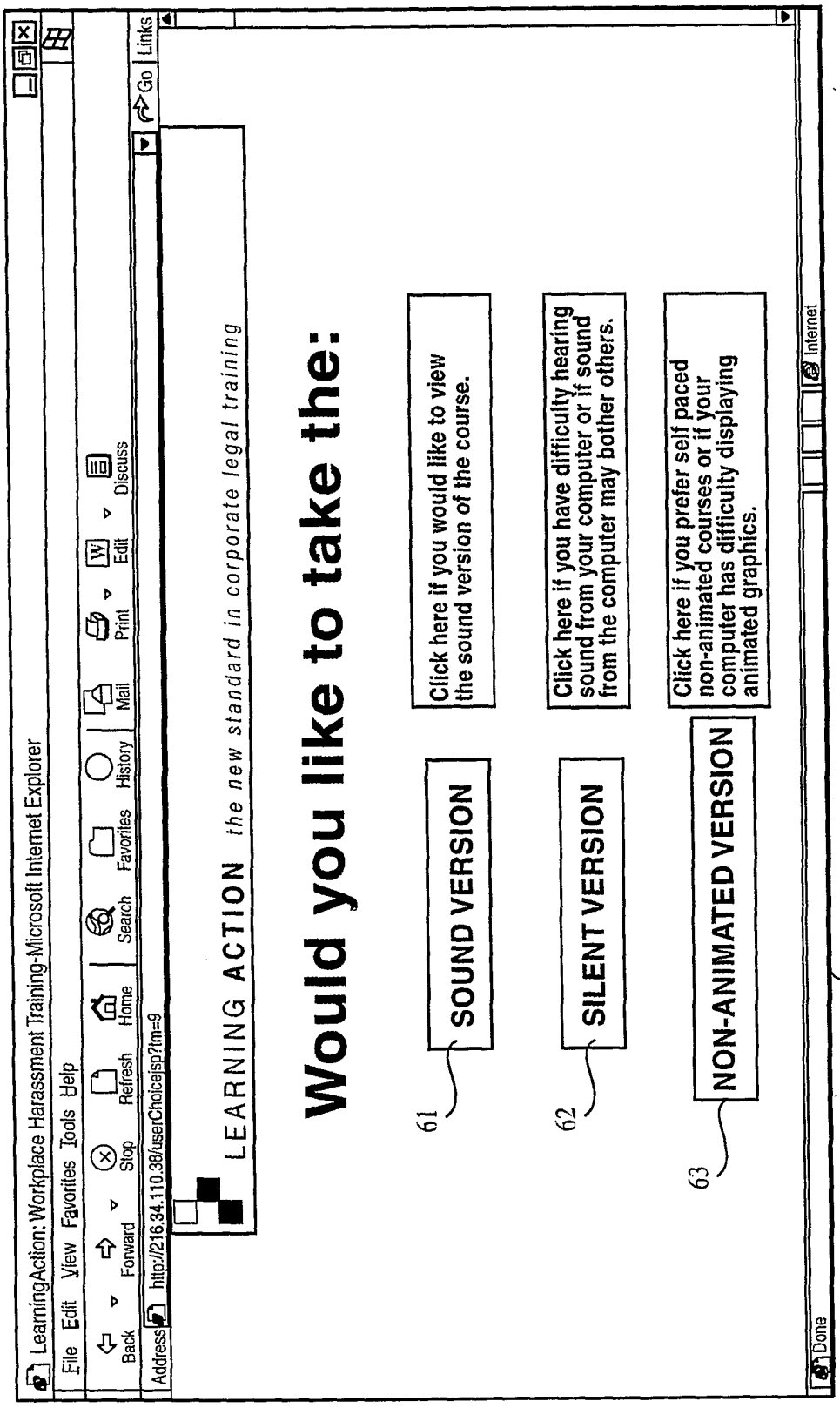

LearningAction: Workplace Harassment Training-Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss
Address http://216.34.110.38/userChoiceisp?tm=9
Go Links
LEARNING ACTION the new standard in corporate legal training
Would you like to take the:
SOUND VERSION
Click here if you would like to view the sound version of the course.
SILENT VERSION
Click here if you have difficulty hearing sound from your computer or if sound from the computer may bother others.
NON-ANIMATED VERSION
Click here if you prefer self paced non-animated courses or if your computer has difficulty displaying animated graphics.
Done
Internet
60
61
62
63

FIG. 6

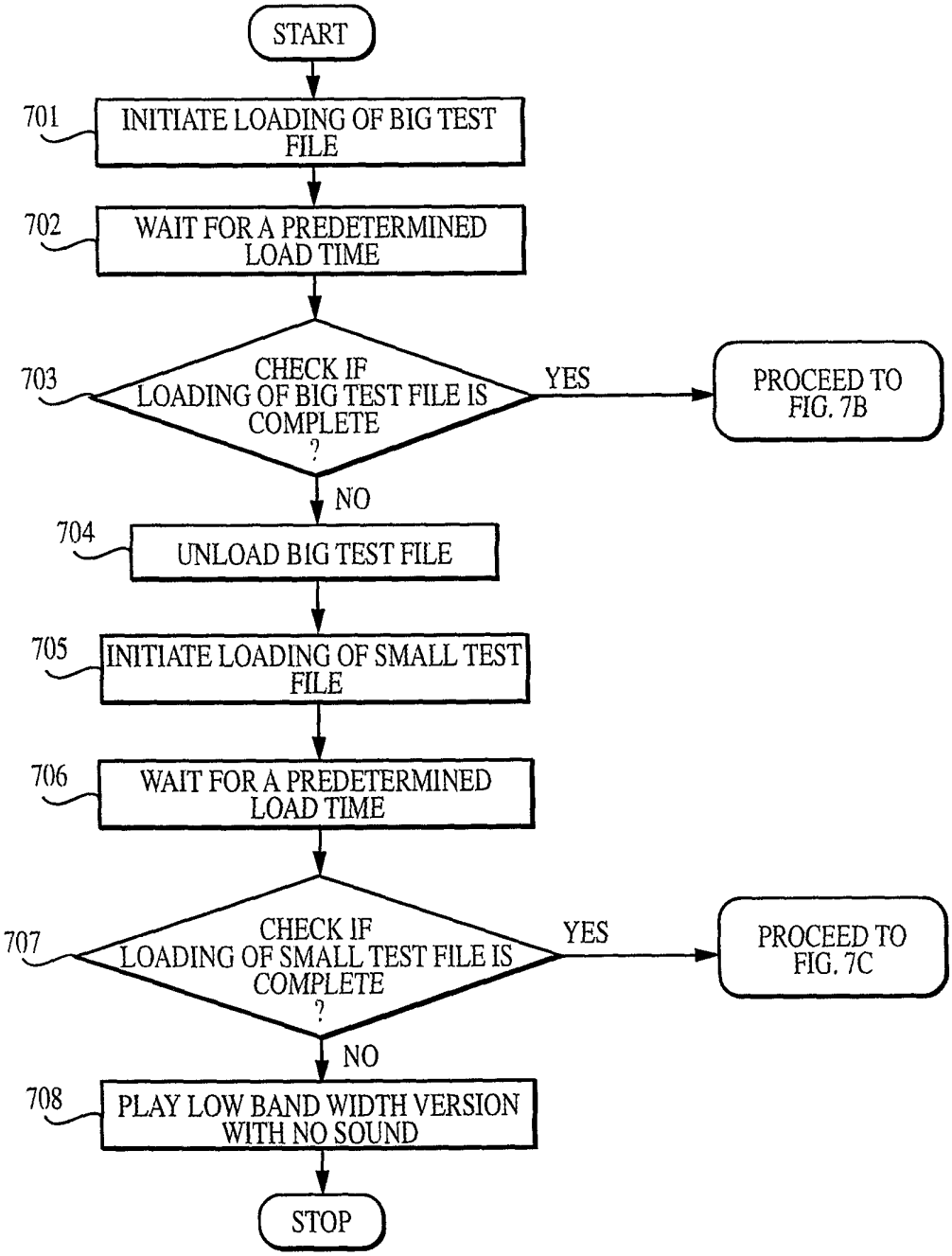

START
701
INITIATE LOADING OF BIG TEST FILE
702
WAIT FOR A PREDETERMINED LOAD TIME
703
CHECK IF LOADING OF BIG TEST FILE IS COMPLETE ?
YES
PROCEED TO FIG. 7B
NO
704
UNLOAD BIG TEST FILE
705
INITIATE LOADING OF SMALL TEST FILE
706
WAIT FOR A PREDETERMINED LOAD TIME
707
CHECK IF LOADING OF SMALL TEST FILE IS COMPLETE ?
YES
PROCEED TO FIG. 7C
NO
708
PLAY LOW BAND WIDTH VERSION WITH NO SOUND
STOP

FIG. 7A

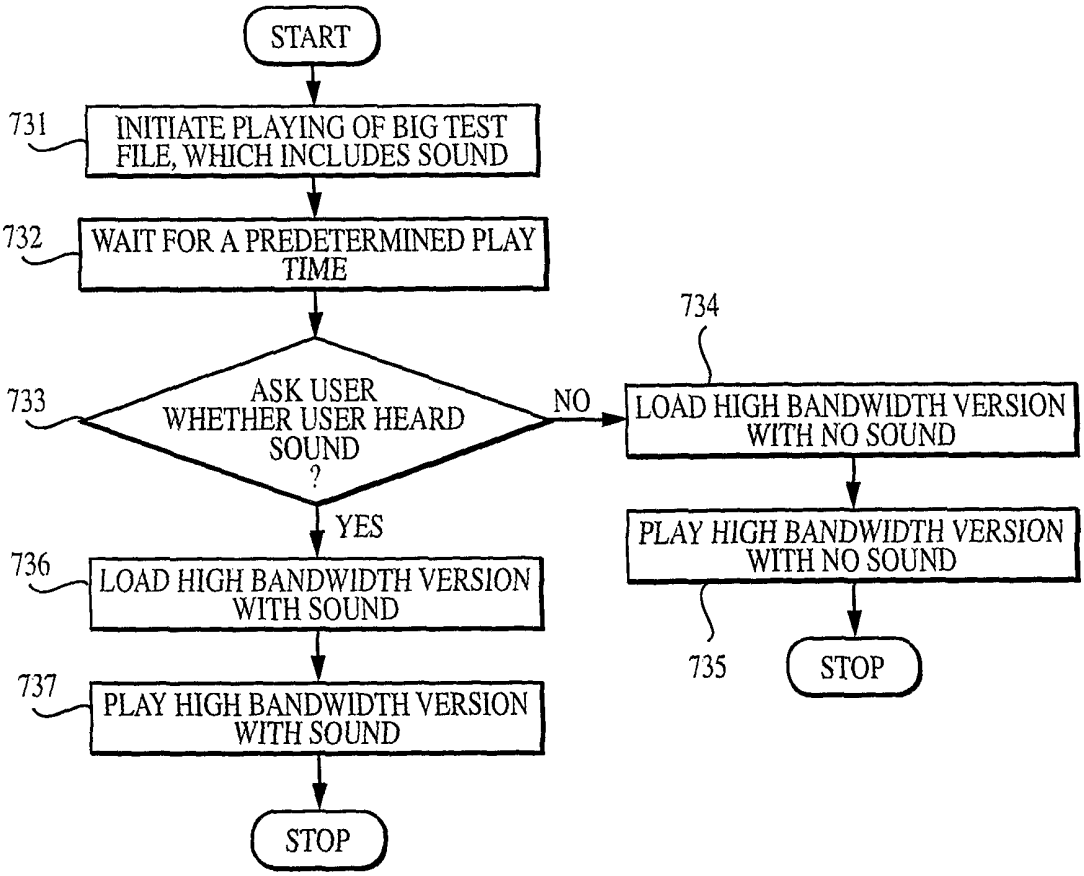

START
731
INITIATE PLAYING OF BIG TEST FILE, WHICH INCLUDES SOUND
732
WAIT FOR A PREDETERMINED PLAY TIME
733
ASK USER WHETHER USER HEARD SOUND ?
NO
734
LOAD HIGH BANDWIDTH VERSION WITH NO SOUND
PLAY HIGH BANDWIDTH VERSION WITH NO SOUND
735
STOP
YES
736
LOAD HIGH BANDWIDTH VERSION WITH SOUND
737
PLAY HIGH BANDWIDTH VERSION WITH SOUND
STOP

FIG. 7B

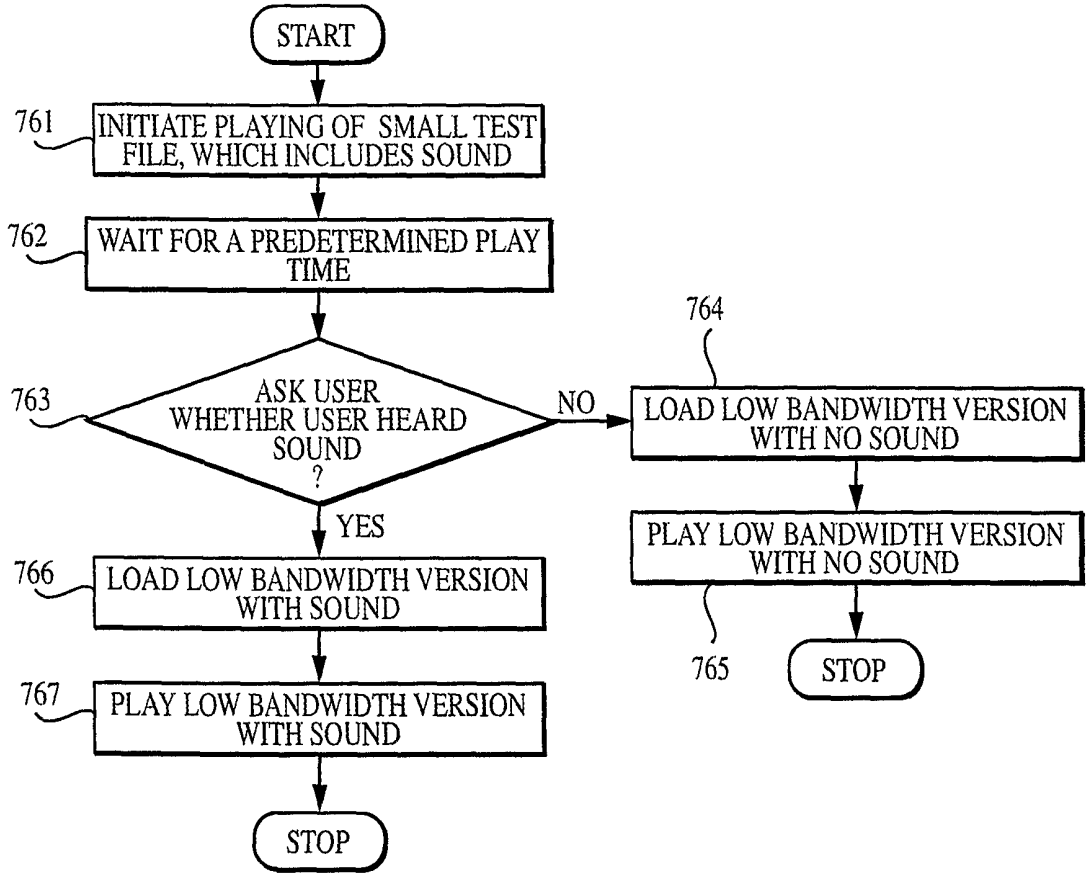

START
761
INITIATE PLAYING OF SMALL TEST FILE, WHICH INCLUDES SOUND
762
WAIT FOR A PREDETERMINED PLAY TIME
763
ASK USER WHETHER USER HEARD SOUND ?
NO
764
LOAD LOW BANDWIDTH VERSION WITH NO SOUND
PLAY LOW BANDWIDTH VERSION WITH NO SOUND
765
STOP
YES
766
LOAD LOW BANDWIDTH VERSION WITH SOUND
767
PLAY LOW BANDWIDTH VERSION WITH SOUND
STOP

FIG. 7C

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US01/10879

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 13/00
US CL :345/335

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 345/327, 329, 333, 335, 339, 978, 302; 707/501; 709/231-235

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

APS
search terms: multimedia, speed, rate, display

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A,P	US 6,105,042 A (AGANOVIC et al) 15 August 2000, figures 4-7, column 7, line 44 to column 10, line 7.	1-23
A	US 5,956,729 A (GOETZ et al) 21 September 1999, column 4, line 65 to column 5, line 47, column 10, lines 30-48, column 11, lines 48-54.	1-23
A	US 5,822,537 A (KATSEFF et al) 13 October 1998, column 14, line 56 to column 16, line 57.	1-23

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 06 JUNE 2001	Date of mailing of the international search report 09 JUL 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer Peggy Harwood CRESCELLE DELA TORRE Telephone No. (703) 308-3800

Form PCT/ISA/210 (second sheet) (July 1998) ★