# CLAIM CHART EXHIBIT 14
# "COHEN"

**INVALIDITY CLAIM CHART FOR U.S. PATENT NO. 5,838,906**

- **US PATENT 5,367,621 TO COHEN ET AL., ("COHEN")[1]. THE BODY OF MY REPORT PROVIDES A NARRATIVE DESCRIPTION OF THIS PRIOR ART AND SHOULD BE CONSIDERED PART OF THIS CHART.**

| Claim Text from '906 Patent | Cohen |
|---|---|
| **906-1.a**:<br>A method for running an application program in a computer network environment, comprising: | Cohen discloses an application program. *See, e.g.*, :<br><br>Cohen's disclosure is described in connection with the BookManager BUILD program. "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1, lines 24-32)<br>Cohen further discloses "[a] method, program and data processing system are disclosed, for providing a generalized link from a reference point within an organized hierarchy of a formatted text stream in an on-line book, to an arbitrary type multimedia object." (col. 2, lines 11-16)<br><br>Cohen discloses a computer network environment. *See, e.g.*, :<br><br>Cohen discloses a distributed environment, including a LAN and a client-server environment. "FIG. 4 illustrates a workstation for displaying a |

[1] For all asserted claims this reference is a 103 reference due to my understanding of the plain meaning of the limitations relating to "location" (e.g. 901-1.f and 906-1.g and 985-1.f and 985.1g) and the Court's discussion of the issue on page 17 of its August 22, 2011 Order. Thus, for these particular limitations, the reference is not anticipatory, but rather, as explained in the body of my report, this limitation would be combined with a prior art web browser like Mosaic, CERN's web browser, Viola, or MediaView. Likewise, to satisfy the HTML limitations in the '985 patent, the reference must be combined with a web browser or HTML teaching, such as Mosaic, CERN's web browser, or Viola. For both all such limitations it would have been obvious to a person of ordinary skill in the art at the time to do so as explained in the body of my report and the teachings, for example, of Tim Berners-Lee posted on the CERN website discussing the Web and relating features and pointers to other browser technologies including HyperCard, Viola and MediaView. See also Bina Ex. 7 (suggesting inlining multimedia objects). This was an obvious and natural extension of prior hypermedia functions and features and an inevitable development in the marketplace at the time of the invention and based on the state of the art.

| Claim Text from '906 Patent | Cohen |
|---|---|
| | softcopy book, in accordance with the invention. The workstation 200 includes the bus 202 which interconnects the CPU 204, DASD 206, display and keyboard adapter 208, local area network (LAN) 210, video adapter 212, audio adapter 214, mouse adapter 216, and the memory 220." (col. 8, lines 61-67) "The softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200, or alternately it can be loaded from a diskette on the DASD 206. " (col. 9, lines 27-30)<br>Cohen further discloses a distributed hypermedia document in that the client workstations retrieve and browse on-line documents that include multimedia. "The link tags described herein specify hypertext links which are created within on-line documents and between on-line documents. Using the GML described in the above referenced BookMaster publications, new tags and concepts described herein enable the creation of hypertext links within and between on-line documents. In accordance with the invention disclosed herein, those hypertext links are improved to represent and present multimedia objects in a manner which is not constrained by any mechanism which invokes the link. Hypertext links connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base." (col. 4, lines 34-47) |
| **906-1.b**:<br>providing at least one client workstation and one network server coupled to said network environment, wherein said network environment is a distributed hypermedia environment; | Cohen discloses a client workstation. *See, e.g.*, :<br><br>Cohen discloses that the browser is on a client workstation. "FIG. 4 illustrates a workstation for displaying a softcopy book, in accordance with the invention. The workstation 200 includes the bus 202 which interconnects the CPU 204, DASD 206, display and keyboard adapter 208, local area network (LAN) 210, video adapter 212, audio adapter 214, mouse adapter 216, and the memory 220." (col. 8 line 61 – col. 9 line 14). Cohen further discloses that "[i]t is within the scope of the invention that the architecture of FIG. 4 can represent a host data processing system or alternately a self-contained, portable data processor such as a laptop or |

| Claim Text from ’906 Patent | Cohen |
| --- | --- |
| | palm top personal computer." (col. 9, lines 34-38)<br><br>Cohen discloses a network server. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65)<br>Cohen further discloses that the client workstation downloads documents from a network server: "[t]he softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200… ." (col. 9, lines 27-30). "It is within the scope of the invention that the architecture of FIG. 4 can represent a host data processing system or alternately a self-contained, portable data processor such as a laptop or palm top personal computer." (col. 9, lines 34-38).<br><br>Cohen discloses a distributed hypermedia environment. *See, e.g.*, :<br><br>Cohen discloses a distributed environment, including a LAN and a client-server environment. "FIG. 4 illustrates a workstation for displaying a softcopy book, in accordance with the invention. The workstation 200 includes the bus 202 which interconnects the CPU 204, DASD 206, display and keyboard adapter 208, local area network (LAN) 210, video adapter 212, audio adapter 214, mouse adapter 216, and the memory 220." (col. 8, lines 61-67) "The softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200, or alternately it can be loaded from a diskette on the DASD 206. " (col. 9, lines 27-30)<br>Cohen further discloses a distributed hypermedia document in that the client workstations retrieve and browse on-line documents that include multimedia. "The link tags described herein specify hypertext links which are created within on-line documents and between on-line documents. Using the GML described in the above referenced BookMaster publications, new tags and concepts described herein enable the creation |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | of hypertext links within and between on-line documents. In accordance with the invention disclosed herein, those hypertext links are improved to represent and present multimedia objects in a manner which is not constrained by any mechanism which invokes the link. Hypertext links connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base." (col. 4, lines 34-47) |
| **906-1.c**:<br>executing, at said client workstation, a browser application, that parses a first distributed hypermedia document to identify text formats included in said distributed hypermedia document and for responding to predetermined text formats to initiate processing specified by said text formats; | Cohen discloses a browser application. *See, e.g.*, :<br><br>Cohen discloses "[h]ypertext links [that] connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base. Links can be thought of as similar to cross-references in a printed document. For example, while reading about a topic in an encyclopedia, the reader may come across a reference to another topic. The reader of the hardcopy book will place a finger on the page that references the topic and will turn back to the new referenced page. The reader has just created a link from one part of the hardcopy document to another. In printed documents, a reader turns to related information. In an on-line softcopy document, the BookManager program creates a link to related information, and the on-line reader can then display that information. The way a reader selects a reference for BookManager to display can be by using a pointing device such as a mouse to activate a link tag in the displayed text. A previously stored address pointer relates the link tag to the target portion of the document to which the link tag refers. The target may also be another document." (col. 4, lines 44-55)<br><br>Cohen discloses that the browser application parses a hypermedia document. *See, e.g.*, :<br>eCohen discloses that a hypermedia document's formatted text stream is parsed<br>Cohen discloses that a hypermedia document's formatted text stream is parsed to identify markup tags. "The softcopy book READ program 400 |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG. 1b are located." (col. 10, lines 29-36). See also col. 11 line 40 (describing parsing of DATA string).<br><br>Cohen discloses a hypermedia document with text formats. *See, e.g.*, :<br><br>Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>"The method begins by storing a formatted text stream in the data processing system. The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 10-26)<br>Cohen provides examples of text formats at columns 5 through 7. |
| **906-1.d:**<br>utilizing said browser to display, on said client workstation, at least a portion of a first hypermedia document received over said network from said server, | Cohen discloses that a hypermedia document is received from the server. *See, e.g.*, :<br><br>Cohen discloses that the client workstation receives hypermedia documents from a network server: "[t]he softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200… ." (col. 9, lines 27-30). "It is within the scope of the invention that the architecture of FIG. 4 can represent a host data processing system or alternately a self-contained, portable data processor such as a laptop or |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | palm top personal computer." (col. 9, lines 34-38).<br><br>Cohen discloses that the browser displays a hypermedia document. *See, e.g.*, :<br><br>Cohen's disclosure is directed to the BookManager READ browser, and describes a browser that displays documents: "a softcopy book reading program whose presentation format primarily displays on a monitor display screen." (col. 3, line 65 - col. 4, line 1)<br>Cohen further discloses: "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1 lines 24-32)<br>Cohen further describes the mechanics by which the hypermedia document is displayed: "The display buffer 238 stores the resulting picture displayed on the monitor display screen 208. Currently, the graphics 190' and the text 174' are shown in the display buffer 238 of FIG. 4." (col. 9, lines 22-26)<br>Cohen discloses that the browser displays hypermedia documents. "The resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an internal object within the softcopy book, or alternately from external files or external data bases. This enables multimedia objects such as high resolution photographic quality graphics, motion video, sound or animation to be supported, as specified by the author at the time of writing his book. The author may also provide for alternate multimedia objects to be displayed where particular specified multimedia hardware or software is not present in a user's workstation." (col. 15, lines 30-43). |
| **906-1.e**: | Cohen discloses that a hypermedia document is displayed in a browser window. |

| Claim Text from '906 Patent | Cohen |
|---|---|
| wherein the portion of said first hypermedia document is displayed within a first browser-controlled window on said client workstation, | *See, e.g.*, :<br><br>Cohen's disclosure is directed to the BookManager READ browser, and describes " a softcopy book reading program whose presentation format primarily displays on a monitor display screen." (col. 3, line 65 - col. 4, line 1)<br>Cohen further discloses: "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1 lines 24-32)<br>Cohen further describes the mechanics by which the hypermedia document is displayed: "The display buffer 238 stores the resulting picture displayed on the monitor display screen 208. Currently, the graphics 190' and the text 174' are shown in the display buffer 238 of FIG. 4." (col. 9, lines 22-26)<br>Cohen further discloses an "application window." (col. 6, lines 27-29)<br>Cohen discloses that the browser displays hypermedia documents. "The resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an internal object within the softcopy book, or alternately from external files or external data bases. This enables multimedia objects such as high resolution photographic quality graphics, motion video, sound or animation to be supported, as specified by the author at the time of writing his book. The author may also provide for alternate multimedia objects to be displayed where particular specified multimedia hardware or software is not present in a user's workstation." (col. 15, lines 30-43). |
| **906-1.f**:<br>wherein said first distributed hypermedia | Cohen discloses an embed text format at a first location in a hypermedia document. *See, e.g.*, : |

| Claim Text from '906 Patent | Cohen |
| --- | --- |
| document includes an embed text format, located at a first location in said first distributed hypermedia document, that specifies the location of at least a portion of an object external to the first distributed hypermedia document, | During document parsing, the location of an LID tag is determined. This is considered a first location in the hypermedia document. Subsequent parsing and resultant action proceeds from this location. "An author-defined link is created by the book's author to establish a relationship between a source location within the softcopy text and a target location within the same text or the text in another softcopy book. The author will place a link tag in the location of the softcopy book which is the source or referencing location. Then the author will include a link description tag at the beginning of the softcopy book, which describes the information necessary to create a link from the source link tag to the target location." (col. 4, line 66 through col. 5, line 7).<br>As one example: "Turning now to FIG. 1, the softcopy book file 100 is shown which includes the link description tags 102 shown in greater detail in FIG. 1a, the book text with tags 104 shown in greater detail in FIG. 1b, the internal animation object 106 shown in greater detail in FIG. 1c, the internal audio object 108 shown in greater detail in FIG. 1d, and the internal graphics object 110 shown in greater detail in FIG. 1e. The location of the link description tags, book text and internal objects is identified in the file index 105 which stores the location offset values for each in the book file 100. In FIG. 1a, the link description tags 102 include three tags. A first tag 120 for a video object type, a second tag 140 for an audio object type, and a third tag 150 for a graphic object type. FIG. 1b shows the book text with tags 104. The softcopy book text includes a first portion 160 which is a heading denoted by :Hl. The second section 162 is a paragraph denoted by :P. The third section 164 is a multimedia hypertext link denoted by the beginning tag :L and the ending tag :eL. The link identification LID=eleph_movie for the link 164, relates the tag 164 to the link descriptor tag 120 of FIG. 1a." (col. 7, lines 31-54)<br>In Cohen, the LDESC embed text formats are kept in the prologue of the document. This was actually a feature, because it allowed an author to define the embed text format once, and then re-use that text format within |

| Claim Text from ’906 Patent | Cohen |
| --- | --- |
| | the hypermedia document using a shorter link identification tag, without needing to re-type the full LDESC text format. "The link identification LID=eleph_movie for the link 164, relates the tag 164 to the link descriptor tag 120 of FIG. 1a." (col. 7, lines 50-54).<br>Accordingly, it would have been obvious to use LDESC text formats within the document's formatted text stream such that the embed text format was at a first location in the hypermedia document.<br><br>Cohen discloses that the embed text format specifies the location of an object. *See, e.g.*, :<br><br>The LDESC tag includes an OBJECT attribute that specifies the location of an object by identifying that object. (See col. 5 lines 44-49). In addition, Cohen discloses that the formatted text stream includes object location information. "The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 18-26)<br><br>Cohen discloses an object that is external to a hypermedia document. *See, e.g.*, :<br><br>Cohen discloses that "[t]he resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an internal object within the softcopy book, or alternately from external files or external data bases." (col. 15, lines 31-36)<br>For example, for a video object, Cohen describes that "[t]he softcopy book file 100 of FIG. 1 can be stored on a magnetic diskette on the DASD 206 or it can be stored on a compact disk as a separate file from the external |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | video object 195 which would be stored as its own file on the same compact disk. Alternately, the external video object 195 can be stored in another storage medium separate from that for the softcopy book file 100. If the video object 195 of FIG. 2 were an internal object, it would be encapsulated as a part of the softcopy book file 100, in a manner similar to the incorporation of internal object 106, 108 or 110 in FIG. 1." (col. 8, lines 16-26) |
| **906-1.g**:<br>wherein said object has type information associated with it utilized by said browser to identify and locate an executable application external to the first distributed hypermedia document, and | Cohen discloses that the object has associated type information. *See, e.g.*, :<br><br>Cohen discloses the use of an object's multimedia type information. "The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information." (col. 2, lines 31-35)<br>Cohen also discloses that objects have type information indicated by the OBJTYPE attribute of the LDESC tag. "OBJTYPE=object-type, Identifies the type of information the author wants to create a link to. The object-type can be one of the following values: PROGRAM/ANIMATION/VIDEO/AUDIO/GRAPHIC/IMAGE. " (col. 5, lines 49-54)<br>For example, "In FIG. 1a, the link description tags 102 include three tags. A first tag 120 for a video object type, a second tag 130 for an audio object type, and a third tag 150 for a graphic object type." (col. 7, lines 42-45)<br>The LDESC tag also includes a DATA attribute that serves to provide type information. "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler 15 program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19) |

| Claim Text from ’906 Patent | Cohen |
| --- | --- |
| | Cohen discloses that the browser uses type information to identify and locate an executable application. *See, e.g.*, :<br><br>Cohen discloses the use of an object’s multimedia type information. “The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information.” (col. 2, lines 31-35)<br>Using the DATA attribute by way of example, Cohen discloses: "With reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 54-60) As another example, "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19)<br>Cohen discloses still other examples of executable applications that are identified and located: "The profile 300 includes the hardware types for a particular I/O function, characteristics for each hardware type, and the software drivers which enable the application programs and I/O handler programs to interact with the particular I/O hardware or software. For example, if the I/O function is audio, the user's workstation profile 300 shows that there is a stereo high fi connected through the audio adapter 214 to the workstation 200. The workstation profile 300 further describes that the stereo high fi hardware has, among its other characteristics, a frequency response of 20 to 20000 Hertz and it can handle delta mod data in FORMAT B. Still further, the workstation file 300 shows that there are |

| Claim Text from ’906 Patent | Cohen |
| --- | --- |
| | three software drivers available for producing an audio output. The first software driver is "CD__AUDIO.DVR" which is a software driver which enables an audio object stored on a compact disk player which may be connected through the video adapter 212, for example, to transfer the audio data from the audio object to the stereo high fi to produce the audio presentation. The second software driver for the audio I/O function is "TAPE__AUDIO.DVR." This audio driver enables an audio object stored on a tape drive, such as can be connected through a suitable adapter to the bus 202, to output audio data from the audio object stored thereon to the stereo high fi for the audio presentation. The third software driver for the audio I/O function is "AUDIO__DATA.DVR." This driver enables an audio object such as the internal audio object 108 in FIG. 1b, to have its audio data transferred to the stereo high fi hardware for the audio presentation." (col. 9, line 41 through col. 10, line 4)<br><br>Cohen discloses that the executable application is external to the hypermedia document. *See, e.g.*, :<br><br>Cohen discloses the use of an object’s multimedia type information. “The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information.” (col. 2, lines 31-35)<br>Using the DATA attribute by way of example, Cohen discloses: "With reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 54-60) As another example, "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler program specified in the string. Then step 426 |

| Claim Text from ’906 Patent | Cohen |
|---|---|
| | goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19) These executable applications are external to the hypermedia document.<br>Cohen discloses still other examples of executable applications that are external to the hypermedia document: "The profile 300 includes the hardware types for a particular I/O function, characteristics for each hardware type, and the software drivers which enable the application programs and I/O handler programs to interact with the particular I/O hardware or software. For example, if the I/O function is audio, the user's workstation profile 300 shows that there is a stereo high fi connected through the audio adapter 214 to the workstation 200. The workstation profile 300 further describes that the stereo high fi hardware has, among its other characteristics, a frequency response of 20 to 20000 Hertz and it can handle delta mod data in FORMAT B.<br>Still further, the workstation file 300 shows that there are three software drivers available for producing an audio output. The first software driver is "CD__AUDIO.DVR" which is a software driver which enables an audio object stored on a compact disk player which may be connected through the video adapter 212, for example, to transfer the audio data from the audio object to the stereo high fi to produce the audio presentation. The second software driver for the audio I/O function is "TAPE__AUDIO.DVR." This audio driver enables an audio object stored on a tape drive, such as can be connected through a suitable adapter to the bus 202, to output audio data from the audio object stored thereon to the stereo high fi for the audio presentation. The third software driver for the audio I/O function is "AUDIO__DATA.DVR." This driver enables an audio object such as the internal audio object 108 in FIG. 1b, to have its audio data transferred to the stereo high fi hardware for the audio presentation." (col. 9, line 41 through col. 10, line 4) |
| **906-1.h**:<br>wherein said embed text format is parsed by said | Cohen discloses that the browser parses the embed text format. *See, e.g.*, : |

| Claim Text from '906 Patent | Cohen |
|---|---|
| browser to automatically invoke said executable application to execute on said client workstation in order to display said object and enable an end-user to directly interact with said object within a display area created at said first location within the portion of said first distributed hypermedia document being displayed in said first browser-controlled window. | Cohen discloses that an embed text format is discovered by parsing an LID tag. This tag in turn points to an LDESC tag, which is also subsequently parsed.<br>"The softcopy book READ program 400 operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG. 1b are located." (col. 10, lines 29-36). See also col. 11 line 40 (describing parsing of DATA string).<br><br>Cohen discloses automatic invocation of the executable application. *See, e.g.*, :<br><br>Cohen discloses an attribute of the LDESC tag called AUTOLAUNCH that defines how to invoke or launch multimedia: either automatically the first time the page appears or in the alternative, only upon explicit selection. (col. 6 lines 13-17).<br>Cohen goes on to disclose this feature in more detail: "Then in step 414, a determination is made as to whether any link tags have a link description with the AUTOLAUNCH parameter equaling 'yes' in the corresponding link descriptor tag. Reference to the link description tag 102 in FIG. 1a will show that the first link description tag 120 has AUTOLAUNCH equal to no, the second link tag 140 has AUTOLAUNCH equal to no. However, the third link tag 150 has AUTOLAUNCH equal to yes. The link descriptor tag 150 is for a graphic object type, and refers to the internal graphics object 110 of FIG. 1e. In step 416 of FIG. 6, if an AUTOLAUNCH parameter is equal to 'yes,' then the program gets the DATA string from the link description. Reference to FIG. 1a will show that the link description tag 150 has the DATA='graph.exe \ GOCA Format C'. Then in step 418 of FIG. 6, the program outputs the data string to start the execution of the I/O handler program specified in the string. This is followed by the step 418 going to step 420. With reference to the |

| Claim Text from ’906 Patent | Cohen |
| --- | --- |
| | graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 36-60)<br><br>Cohen discloses that the executable application displays the object. *See, e.g.*, :<br><br>Cohen discloses executable applications that display objects. For example, Cohen discloses an executable video.exe that displays objects as indicated in Figure 7a. "The video handler program whose flow diagram is shown in FIG. 7a, will now be described. In step 502, the softcopy book READ program 400 outputs the DATA string, and this step 562 corresponds to either step 418 or step 426 of the softcopy book READ program 400 of FIG. 6. In the example of activating the link tag 164 in the softcopy book text 104 of FIG. 1b, this is the link to initiate the multimedia video display of a motion picture of an African elephant family." (col. 11, lines 30-38) "In step 508 of the video handler program of FIG. 7a, it is determined whether the workstation profile includes the required video support. Since the profile 300 indicates that the support is present in the workstation 200, the flow proceeds to step 524 which outputs the DATA string 'CD Video File Format A' and other control information necessary for the playing of the video information from the compact disk player which is connected through the video adapter 212 to the workstation and the presentation of the resulting motion picture on the display 208 at the workstation. Step 524 accesses the necessary data form the external video object 195 of FIG. 2 as specified by the object name 'family__clip.vid', which is the file handle for the video object 195 on the compact disk device, and the STORE=external parameters in the link descriptor 120 of FIG. 1a. The data from the object is transferred to the software driver CD_VIDEO.DVR, specified in the workstation profile 300, and the software driver controls the presentation of the motion picture received from the video adapter 212 for display on the display 208. Step |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | 526 of FIG. 7a displays the CD video on the display 208." (col. 12, lines 1-23)<br>As another example, Cohen discloses a graph.exe executable application that displays objects as indicated in Figure 7c.<br><br>Cohen does not explicitly disclose direct interaction with an object, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br>Cohen does suggest, however, the possibility of direct interaction because it discloses a variety of multimedia objects, including those that inherently require user interaction at the first location, such as spreadsheet objects.<br>"Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base. " (col. 2, line 63 - col. 3, line 2)<br>Cohen discloses that the object is displayed on an auxiliary display device and not at a first location in the hypermedia document, i.e. the location of an LID tag determined by parsing. Therefore there is no disclosure that interaction with an object, if possible, could be at the first location in the document. |
| | |
| **906-2.a**:<br>The method of claim 1, wherein said executable application is a controllable application and further comprising the step of: interactively controlling said controllable application on said client workstation via inter-process communications between said browser and said controllable application. | Cohen discloses interactive control via inter-process communications between a browser and an application. *See, e.g.*, :<br><br>As one example, Cohen discloses inter-process communication between the browser and executable applications specified by the DATA attribute.<br>"DATA='string' Lets the author pass data to multimedia object handler programs for the first, primary element. For example, string may be parameters to create a link to an animation sequence. Values for string |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | depend on the capabilities of the user's installation." (col. 6, lines 7-12) |
| | |
| **906-3.a**:<br>The method of claim 2, wherein the communications to interactively control said controllable application continue to be exchanged between the controllable application and the browser even after the controllable application program has been launched. | Cohen does not explicitly disclose ongoing inter-process communications, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br><br>Cohen does suggest, however, the possibility of ongoing inter-process communications because it discloses multimedia objects such as spreadsheet objects that would require ongoing user interaction and therefore, ongoing inter-process communications in order to process those interactions. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base. " (col. 2, line 63 - col. 3, line 2) |
| | |
| **906-6.a**:<br>A computer program product for use in a system having at least one client workstation and one network server coupled to said network environment, wherein said network environment is a distributed hypermedia environment, the computer program product comprising: | Cohen discloses an application program in a computer network environment. *See* evidence recited for 906-1.a.<br><br>Cohen also discloses a client workstation and a network server in a distributed hypermedia environment. *See* evidence recited for 906-1.b. |
| **906-6.b**:<br>a computer usable medium having computer readable program code physically embodied therein, said computer program product further comprising: | Cohen discloses computer code physically embodied on a medium. *See, e.g.*, :<br><br>Cohen discloses that the BookManager READ program to which Cohen's disclosure is directed is stored on computer code physically embodied on a medium: "Also stored in the memory 220 is the softcopy book READ program 400 of FIG. 6, the I/O handler programs 500, 530 and 560 of |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | FIGS. 7a, 7b and 7c, respectively, and the drivers and operating system 590. The CPU 204 of FIG. 4, executes the instructions embodied in the program 400 and in the handler programs 500, 530 and 560, in accordance with the invention." (col. 9, lines 6-13.)<br>Cohen further discloses that "FIG. 6 is a flow diagram of the softcopy book READ program 400, in accordance with the invention. The program resides in a partition of the memory 220 of the workstation in FIG. 4, or alternately in the memory of a host processor in a host data processing system. " (col. 10, lines 12-16) |
| **906-6.c**:<br>computer readable program code for causing said client workstation to execute a browser application to parse a first distributed hypermedia document to identify text formats included in said distributed hypermedia document and to respond to predetermined text formats to initiate processes specified by said text formats; | Cohen discloses a browser application that parses a hypermedia document with text formats. *See* evidence recited for 906-1.c. |
| **906-6.d**:<br>computer readable program code for causing said client workstation to utilize said browser to display, on said client workstation, at least a portion of a first hypermedia document received over said network from said server, | Cohen discloses a hypermedia document received from a server and a browser that displays the hypermedia document. *See* evidence recited for 906-1.d. |
| **906-6.e**:<br>wherein the portion of said first hypermedia document is displayed within a first browser-controlled window on said client workstation, | Cohen discloses that the hypermedia document is displayed in a browser window. *See* evidence recited for 906-1.e. |
| **906-6.f**:<br>wherein said first distributed hypermedia document includes an embed text format, located at a first location in said first distributed hypermedia document, that specifies the location | Cohen discloses an embed text format at a first location in a hypermedia document; that the embed text format specifies the location of an object; and that the object is external to the hypermedia document. *See* evidence recited for 906-1.f. |

| Claim Text from '906 Patent | Cohen |
|---|---|
| of at least a portion of an object external to the first distributed hypermedia document, | |
| **906-6.g**:<br>wherein said object has type information associated with it utilized by said browser to identify and locate an executable application external to the first distributed hypermedia document, and | Cohen discloses that the object has associated type information, that the browser uses the type information to identify and locate an executable application, and that the executable application is external to the hypermedia document. *See* evidence recited for 906-1.g. |
| **906-6.h**:<br>wherein said embed text format is parsed by said browser to automatically invoke said executable application to execute on said client workstation in order to display said object and enable an end-user to directly interact with said object within a display area created at said first location within the portion of said first distributed hypermedia document being displayed in said first browser-controlled window. | Cohen discloses that the browser parses the embed text format; that the browser automatically invokes the executable application; that the executable application displays the object<br>. *See* evidence recited for 906-1.h. |
| | |
| **906-7.a**:<br>The computer program product of claim 6, wherein said executable application is a controllable application and further comprising:<br>computer readable program code for causing said client workstation to interactively control said controllable application on said client workstation via inter-process communications between said browser and said controllable application. | Cohen does not explicitly disclose ongoing inter-process communications, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br><br>Cohen does suggest, however, the possibility of ongoing inter-process communications because it discloses multimedia objects such as spreadsheet objects that would require ongoing user interaction and therefore, ongoing inter-process communications in order to process those interactions. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | base. " (col. 2, line 63 - col. 3, line 2)<br><br>*See* evidence recited for 906-2.a. |
| | |
| **906-8.a**:<br>The computer program product of claim 7, wherein the communications to interactively control said controllable application continue to be exchanged between the controllable application and the browser even after the controllable application program has been launched. | Cohen does not explicitly disclose ongoing inter-process communications, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br><br>Cohen does suggest, however, the possibility of ongoing inter-process communications because it discloses multimedia objects such as spreadsheet objects that would require ongoing user interaction and therefore, ongoing inter-process communications in order to process those interactions. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base. " (col. 2, line 63 - col. 3, line 2)<br><br>*See* evidence recited for 906-3.a. |
| | |
| **906-11.a**:<br>The method of claim 3, wherein additional instructions for controlling said controllable application reside on said network server, wherein said step of interactively controlling said controllable application includes the following substeps: | Cohen discloses additional instructions on the server. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br><br>It would have been obvious to have additional instructions on the server in |

| Claim Text from '906 Patent | Cohen |
|---|---|
| | view of Cohen's disclosure of network interoperability and database application interoperability. |
| **906-11.b**:<br>issuing, from the client workstation, one or more commands to the network server; | Cohen discloses that the client issues commands to the server *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br><br>It would have been obvious for the client to issue commands to the server in view of Cohen's disclosure of network interoperability and database application interoperability. |
| **906-11.c**:<br>executing, on the network server, one or more instructions in response to said commands; | Cohen discloses that the server executes instructions in response to client commands. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br><br>It would have been obvious for the server to execute in response to commands in view of Cohen's disclosure of network interoperability and database application interoperability. |
| **906-11.d**:<br>sending information from said network server to said client workstation in response to said executed | Cohen discloses that the server responds with information to the client. *See, e.g.*, :<br>Cohen discloses that that the client workstation can connect to other |

| Claim Text from '906 Patent | Cohen |
|---|---|
| instructions; and | computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br><br>It would have been obvious for the server to respond with information to the client in view of Cohen's disclosure of network interoperability and database application interoperability. |
| **906-11.e**:<br>processing said information at the client workstation to interactively control said controllable application. | Cohen discloses that the client uses information from the server to interactively control the application. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br><br>It would have been obvious for the client to use information from the server to control the application in view of Cohen's disclosure of network interoperability and database application interoperability. |
| | |
| **906-13.a**:<br>The computer program product of claim 8, wherein additional instructions for controlling said controllable application reside on said network server, wherein said computer readable program code for causing said client workstation to interactively control said controllable application | Cohen discloses additional instructions on the server *See* evidence recited for 906-11.a. |

| Claim Text from '906 Patent | Cohen |
|---|---|
| on said client workstation includes: | |
| **906-13.b**:<br>computer readable program code for causing said client workstation to issue from the client workstation, one or more commands to the network server; | Cohen discloses that the client issues commands to the server. *See* evidence recited for 906-11.b. |
| **906-13.c**:<br>computer readable program code for causing said network server to execute one or more instructions in response to said commands; | Cohen discloses that the server executes instructions in response to client commands. *See* evidence recited for 906-11.c. |
| **906-13.d**:<br>computer readable program code for causing said network sever to send information to said client workstation in response to said executed instructions; and | Cohen discloses that the server responds with information to the client. *See* evidence recited for 906-11.d. |
| **906-13.e**:<br>computer readable program code for causing said client workstation to process said information at the client workstation to interactively control said controllable application. | Cohen discloses that the client uses information from the server to interactively control the application. *See* evidence recited for 906-11.e. |
| | |

**INVALIDITY CLAIM CHART FOR U.S. PATENT NO. 7,599,985**

- **BASED ON US PATENT 5,367,621 TO COHEN ET AL., ("COHEN")**

| Claim Text from '985 Patent | Cohen |
|---|---|
| **985-1.a**:<br>A method for running an application program in a distributed hypermedia network environment, wherein the network environment comprises at least one client workstation and one network server coupled to the network environment, the method comprising: | Cohen discloses an application program. *See, e.g.*, :<br><br>Cohen's disclosure is described in connection with the BookManager BUILD program. "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1, lines 24-32)<br>Cohen further discloses "[a] method, program and data processing system are disclosed, for providing a generalized link from a reference point within an organized hierarchy of a formatted text stream in an on-line book, to an arbitrary type multimedia object." (col. 2, lines 11-16)<br><br>Cohen discloses a computer network environment. *See, e.g.*, :<br><br>Cohen discloses a distributed environment, including a LAN and a client-server environment. "FIG. 4 illustrates a workstation for displaying a softcopy book, in accordance with the invention. The workstation 200 includes the bus 202 which interconnects the CPU 204, DASD 206, display and keyboard adapter 208, local area network (LAN) 210, video adapter 212, audio adapter 214, mouse adapter 216, and the memory 220." (col. 8, lines 61-67) "The softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200, or alternately it can be loaded from a diskette on the DASD 206. " (col. 9, lines 27-30)<br>Cohen further discloses a distributed hypermedia document in that the |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | client workstations retrieve and browse on-line documents that include multimedia. "The link tags described herein specify hypertext links which are created within on-line documents and between on-line documents. Using the GML described in the above referenced BookMaster publications, new tags and concepts described herein enable the creation of hypertext links within and between on-line documents. In accordance with the invention disclosed herein, those hypertext links are improved to represent and present multimedia objects in a manner which is not constrained by any mechanism which invokes the link. Hypertext links connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base." (col. 4, lines 34-47)<br><br>Cohen discloses a client workstation. *See, e.g.*, :<br><br>Cohen discloses that the browser is on a client workstation. "FIG. 4 illustrates a workstation for displaying a softcopy book, in accordance with the invention. The workstation 200 includes the bus 202 which interconnects the CPU 204, DASD 206, display and keyboard adapter 208, local area network (LAN) 210, video adapter 212, audio adapter 214, mouse adapter 216, and the memory 220." (col. 8 line 61 – col. 9 line 14). Cohen further discloses that "[i]t is within the scope of the invention that the architecture of FIG. 4 can represent a host data processing system or alternately a self-contained, portable data processor such as a laptop or palm top personal computer." (col. 9, lines 34-38)<br><br>Cohen discloses a network server. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65)<br>Cohen further discloses that the client workstation downloads documents |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | from a network server: "[t]he softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200… ." (col. 9, lines 27-30). "It is within the scope of the invention that the architecture of FIG. 4 can represent a host data processing system or alternately a self-contained, portable data processor such as a laptop or palm top personal computer." (col. 9, lines 34-38).<br><br>Cohen discloses a distributed hypermedia environment. *See, e.g.*, :<br><br>Cohen discloses a distributed environment, including a LAN and a client-server environment. "FIG. 4 illustrates a workstation for displaying a softcopy book, in accordance with the invention. The workstation 200 includes the bus 202 which interconnects the CPU 204, DASD 206, display and keyboard adapter 208, local area network (LAN) 210, video adapter 212, audio adapter 214, mouse adapter 216, and the memory 220." (col. 8, lines 61-67) "The softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200, or alternately it can be loaded from a diskette on the DASD 206. " (col. 9, lines 27-30) Cohen further discloses a distributed hypermedia document in that the client workstations retrieve and browse on-line documents that include multimedia. "The link tags described herein specify hypertext links which are created within on-line documents and between on-line documents. Using the GML described in the above referenced BookMaster publications, new tags and concepts described herein enable the creation of hypertext links within and between on-line documents. In accordance with the invention disclosed herein, those hypertext links are improved to represent and present multimedia objects in a manner which is not constrained by any mechanism which invokes the link. Hypertext links connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base." (col. 4, lines 34-47) |
| **985-1.b**: | Cohen discloses a browser application. *See, e.g.*, : |

| Claim Text from '985 Patent | Cohen |
| --- | --- |
| receiving, at the client workstation from the network server over the network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Cohen discloses "[h]ypertext links [that] connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base. Links can be thought of as similar to cross-references in a printed document. For example, while reading about a topic in an encyclopedia, the reader may come across a reference to another topic. The reader of the hardcopy book will place a finger on the page that references the topic and will turn back to the new referenced page. The reader has just created a link from one part of the hardcopy document to another. In printed documents, a reader turns to related information. In an on-line softcopy document, the BookManager program creates a link to related information, and the on-line reader can then display that information. The way a reader selects a reference for BookManager to display can be by using a pointing device such as a mouse to activate a link tag in the displayed text. A previously stored address pointer relates the link tag to the target portion of the document to which the link tag refers. The target may also be another document." (col. 4, lines 44-55)<br><br>Cohen discloses a file containing enabling information. *See, e.g.*, :<br><br>Cohen describes file containing enabling information in the form of a formatted text stream. "The method begins by storing a formatted text stream in the data processing system. The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 10-26)<br>Cohen provides examples of syntax for the formatted text stream in columns 5 through 7. |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | Cohen discloses that the file is received at the client workstation from the network server. *See, e.g.*, :<br><br>Cohen discloses that the client workstation receives a file from a network server: "[t]he softcopy book file 100 can be downloaded from a host through the LAN interface 210 to the workstation 200… ." (col. 9, lines 27-30). "It is within the scope of the invention that the architecture of FIG. 4 can represent a host data processing system or alternately a self-contained, portable data processor such as a laptop or palm top personal computer." (col. 9, lines 34-38).<br><br>Cohen discloses that the browser displays at least a portion of a distributed hypermedia document. *See, e.g.*, :<br><br>Cohen's disclosure is directed to the BookManager READ browser, and describes a browser that displays documents: "a softcopy book reading program whose presentation format primarily displays on a monitor display screen." (col. 3, line 65 - col. 4, line 1)<br>Cohen further discloses: "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1 lines 24-32)<br>Cohen further describes the mechanics by which the hypermedia document is displayed: "The display buffer 238 stores the resulting picture displayed on the monitor display screen 208. Currently, the graphics 190' and the text 174' are shown in the display buffer 238 of FIG. 4." (col. 9, lines 22-26)<br>Cohen discloses that the browser displays hypermedia documents. "The |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an internal object within the softcopy book, or alternately from external files or external data bases. This enables multimedia objects such as high resolution photographic quality graphics, motion video, sound or animation to be supported, as specified by the author at the time of writing his book. The author may also provide for alternate multimedia objects to be displayed where particular specified multimedia hardware or software is not present in a user's workstation." (col. 15, lines 30-43).<br><br>Cohen discloses that at least a portion of a hypermedia document is displayed in a browser-controlled window. *See, e.g.*, :<br><br>Cohen's disclosure is directed to the BookManager READ browser, and describes " a softcopy book reading program whose presentation format primarily displays on a monitor display screen." (col. 3, line 65 - col. 4, line 1)<br>Cohen further discloses: "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1 lines 24-32)<br>Cohen further describes the mechanics by which the hypermedia document is displayed: "The display buffer 238 stores the resulting picture displayed on the monitor display screen 208. Currently, the graphics 190' and the text 174' are shown in the display buffer 238 of FIG. 4." (col. 9, lines 22-26)<br>Cohen further discloses an "application window." (col. 6, lines 27-29)<br>Cohen discloses that the browser displays hypermedia documents. "The |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an internal object within the softcopy book, or alternately from external files or external data bases. This enables multimedia objects such as high resolution photographic quality graphics, motion video, sound or animation to be supported, as specified by the author at the time of writing his book. The author may also provide for alternate multimedia objects to be displayed where particular specified multimedia hardware or software is not present in a user's workstation." (col. 15, lines 30-43). |
| **985-1.c**:<br>executing the browser application on the client workstation, with the browser application: | Cohen discloses a browser application executing on the client workstation. *See, e.g.*, :<br><br>Cohen discloses "[h]ypertext links [that] connect elements in one part of an on-line document to elements in another part of the same document or in a separate on-line document or in an external file or data base. Links can be thought of as similar to cross-references in a printed document. For example, while reading about a topic in an encyclopedia, the reader may come across a reference to another topic. The reader of the hardcopy book will place a finger on the page that references the topic and will turn back to the new referenced page. The reader has just created a link from one part of the hardcopy document to another. In printed documents, a reader turns to related information. In an on-line softcopy document, the BookManager program creates a link to related information, and the on-line reader can then display that information. The way a reader selects a reference for BookManager to display can be by using a pointing device such as a mouse to activate a link tag in the displayed text. A previously stored address pointer relates the link tag to the target portion of the document to which the link tag refers. The target may also be another document." (col. 4, lines 44-55) |
| **985-1.d**:<br>responding to text formats to initiate processing | Cohen discloses responding to text formats to initiate processing specified by the text formats, i.e., parsing text formats. *See, e.g.*, : |

| Claim Text from '985 Patent | Cohen |
|---|---|
| specified by the text formats; | Cohen discloses that the text formats from its formatted text stream is parsed to identify markup tags. "The softcopy book READ program 400 operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG. 1b are located." (col. 10, lines 29-36). See also col. 11 line 40 (describing parsing of DATA string). |
| **985-1.e**:<br>displaying at least a portion of the document within the browser-controlled window; | Cohen discloses that the browser displays a hypermedia document. *See, e.g.*, :<br><br>Cohen's disclosure is directed to the BookManager READ browser, and describes a browser that displays documents: "a softcopy book reading program whose presentation format primarily displays on a monitor display screen." (col. 3, line 65 - col. 4, line 1)<br>Cohen further discloses: "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1 lines 24-32)<br>Cohen further describes the mechanics by which the hypermedia document is displayed: "The display buffer 238 stores the resulting picture displayed on the monitor display screen 208. Currently, the graphics 190' and the text 174' are shown in the display buffer 238 of FIG. 4." (col. 9, lines 22-26)<br>Cohen discloses that the browser displays hypermedia documents. "The resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | internal object within the softcopy book, or alternately from external files or external data bases. This enables multimedia objects such as high resolution photographic quality graphics, motion video, sound or animation to be supported, as specified by the author at the time of writing his book. The author may also provide for alternate multimedia objects to be displayed where particular specified multimedia hardware or software is not present in a user's workstation." (col. 15, lines 30-43).<br><br>Cohen discloses that a hypermedia document is displayed in a browser window. *See, e.g.*, :<br><br>Cohen's disclosure is directed to the BookManager READ browser, and describes " a softcopy book reading program whose presentation format primarily displays on a monitor display screen." (col. 3, line 65 - col. 4, line 1)<br>Cohen further discloses: "For example, the IBM BookManager (TM) READ program helps the user manage, search and look at on-line books. There are two complementary BookManager products, BookManager BUILD creates on-line books from files marked-up with Generalized Markup Language. The BookManager READ product can then manage, search and show the on-line books created by BookManager BUILD." (col. 1 lines 24-32)<br>Cohen further describes the mechanics by which the hypermedia document is displayed: "The display buffer 238 stores the resulting picture displayed on the monitor display screen 208. Currently, the graphics 190' and the text 174' are shown in the display buffer 238 of FIG. 4." (col. 9, lines 22-26)<br>Cohen further discloses an "application window." (col. 6, lines 27-29)<br>Cohen discloses that the browser displays hypermedia documents. "The resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | internal object within the softcopy book, or alternately from external files or external data bases. This enables multimedia objects such as high resolution photographic quality graphics, motion video, sound or animation to be supported, as specified by the author at the time of writing his book. The author may also provide for alternate multimedia objects to be displayed where particular specified multimedia hardware or software is not present in a user's workstation." (col. 15, lines 30-43). |
| **985-1.f**:<br>identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object external to the file, where the object has type information associated with it; | Cohen discloses identifying an embed text format. *See, e.g.*, :<br><br>Cohen discloses identifying the LID tag. "The softcopy book READ program 400 operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG.1b are located." (col. 10, lines 20-36)<br>Cohen further discloses identifying the LDESC tag.<br>"The link tag and the LID attribute identifies the link descriptor LDESC tag that specifies a link." (col. 6 line 69 through col. 7 line 2).<br><br>Cohen discloses that the embed text format corresponds to a first location in the hypermedia document. *See, e.g.*, :<br><br>During document parsing, the location of an LID is determined. This is considered a first location in the hypermedia document. Subsequent parsing and resultant action proceeds from this location. "An author-defined link is created by the book's author to establish a relationship between a source location within the softcopy text and a target location within the same text or the text in another softcopy book. The author will place a link tag in the location of the softcopy book which is the source or referencing location. Then the author will include a link description tag at the beginning of the softcopy book, which describes the information necessary to create a link from the source link tag to the target location." |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | (col. 4, line 66 through col. 5, line 7).<br>As one example: "Turning now to FIG. 1, the softcopy book file 100 is shown which includes the link description tags 102 shown in greater detail in FIG. 1a, the book text with tags 104 shown in greater detail in FIG. 1b, the internal animation object 106 shown in greater detail in FIG. 1c, the internal audio object 108 shown in greater detail in FIG. 1d, and the internal graphics object 110 shown in greater detail in FIG. 1e. The location of the link description tags, book text and internal objects is identified in the file index 105 which stores the location offset values for each in the book file 100. In FIG. 1a, the link description tags 102 include three tags. A first tag 120 for a video object type, a second tag 140 for an audio object type, and a third tag 150 for a graphic object type. FIG. 1b shows the book text with tags 104. The softcopy book text includes a first portion 160 which is a heading denoted by :Hl. The second section 162 is a paragraph denoted by :P. The third section 164 is a multimedia hypertext link denoted by the beginning tag :L and the ending tag :eL. The link identification LID=eleph_movie for the link 164, relates the tag 164 to the link descriptor tag 120 of FIG. 1a." (col. 7, lines 31-54)<br>In Cohen, the LDESC embed text formats are kept in the prologue of the document. This was actually a feature, because it allowed an author to define the embed text format once, and then re-use that text format within the hypermedia document using a shorter link identification tag, without needing to re-type the full LDESC text format. "The link identification LID=eleph_movie for the link 164, relates the tag 164 to the link descriptor tag 120 of FIG. 1a." (col. 7, lines 50-54).<br>Accordingly, it would have been obvious to use LDESC text formats within the document's formatted text stream such that the embed text format corresponds to a first location in a hypermedia document.<br><br>Cohen discloses that the embed text format specifies the location of an object. *See, e.g.*, : |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | The LDESC tag includes an OBJECT attribute that specifies the location of an object by identifying that object. (See col. 5 lines 44-49). In addition, Cohen discloses that the formatted text stream includes object location information. "The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 18-26)<br><br>Cohen discloses that the object is external to the file containing enabling information. *See, e.g.*, :<br><br>Cohen discloses that "[t]he resulting invention provides a generalized link from a reference point within an organized hierarchy of text in a softcopy on-line book, to an arbitrary multimedia object which may be represented by data from an internal object within the softcopy book, or alternately from external files or external data bases." (col. 15, lines 31-36)<br>For example, for a video object, Cohen describes that "[t]he softcopy book file 100 of FIG. 1 can be stored on a magnetic diskette on the DASD 206 or it can be stored on a compact disk as a separate file from the external video object 195 which would be stored as its own file on the same compact disk. Alternately, the external video object 195 can be stored in another storage medium separate from that for the softcopy book file 100. If the video object 195 of FIG. 2 were an internal object, it would be encapsulated as a part of the softcopy book file 100, in a manner similar to the incorporation of internal object 106, 108 or 110 in FIG. 1." (col. 8, lines 16-26)<br><br>Cohen discloses that the object has associated type information. *See, e.g.*, : |

| Claim Text from ’985 Patent | Cohen |
|---|---|
| | Cohen discloses the use of an object’s multimedia type information. “The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information.” (col. 2, lines 31-35)<br>Cohen also discloses that objects have type information indicated by the OBJTYPE attribute of the LDESC tag. "OBJTYPE=object-type, Identifies the type of information the author wants to create a link to. The object-type can be one of the following values: PROGRAM/ANIMATION/VIDEO/AUDIO/GRAPHIC/IMAGE. " (col. 5, lines 49-54)<br>For example, "In FIG. 1a, the link description tags 102 include three tags. A first tag 120 for a video object type, a second tag 130 for an audio object type, and a third tag 150 for a graphic object type." (col. 7, lines 42-45)<br>The LDESC tag also includes a DATA attribute that serves to provide type information. "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler 15 program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19) |
| **985-1.g**:<br>utilizing the type information to identify and locate an executable application external to the file; and | Cohen discloses that the browser uses type information to identify and locate an executable application. *See, e.g.*, :<br><br>Cohen discloses the use of an object’s multimedia type information. “The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information.” (col. 2, lines 31-35)<br>Using the DATA attribute by way of example, Cohen discloses: "With |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 54-60) As another example, "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19)<br>Cohen discloses still other examples of executable applications that are identified and located: "The profile 300 includes the hardware types for a particular I/O function, characteristics for each hardware type, and the software drivers which enable the application programs and I/O handler programs to interact with the particular I/O hardware or software. For example, if the I/O function is audio, the user's workstation profile 300 shows that there is a stereo high fi connected through the audio adapter 214 to the workstation 200. The workstation profile 300 further describes that the stereo high fi hardware has, among its other characteristics, a frequency response of 20 to 20000 Hertz and it can handle delta mod data in FORMAT B. Still further, the workstation file 300 shows that there are three software drivers available for producing an audio output. The first software driver is "CD__AUDIO.DVR" which is a software driver which enables an audio object stored on a compact disk player which may be connected through the video adapter 212, for example, to transfer the audio data from the audio object to the stereo high fi to produce the audio presentation. The second software driver for the audio I/O function is "TAPE__AUDIO.DVR." This audio driver enables an audio object stored on a tape drive, such as can be connected through a suitable adapter to the bus 202, to output audio data from the audio object stored thereon to the stereo high fi for the audio presentation. The third software driver for the |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | audio I/O function is "AUDIO__DATA.DVR." This driver enables an audio object such as the internal audio object 108 in FIG. 1b, to have its audio data transferred to the stereo high fi hardware for the audio presentation." (col. 9, line 41 through col. 10, line 4)<br><br>Cohen discloses that the executable application is external to the file containing enabling information. *See, e.g.*, :<br><br>Cohen discloses the use of an object’s multimedia type information. “The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information.” (col. 2, lines 31-35)<br>Using the DATA attribute by way of example, Cohen discloses: "With reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 54-60) As another example, "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19) These executable applications are external to the file containing enabling information.<br>Cohen discloses still other examples of executable applications that are external to the file containing enabling information: "The profile 300 includes the hardware types for a particular I/O function, characteristics for each hardware type, and the software drivers which enable the application programs and I/O handler programs to interact with the particular I/O hardware or software. For example, if the I/O function is |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | audio, the user's workstation profile 300 shows that there is a stereo high fi connected through the audio adapter 214 to the workstation 200. The workstation profile 300 further describes that the stereo high fi hardware has, among its other characteristics, a frequency response of 20 to 20000 Hertz and it can handle delta mod data in FORMAT B.<br>Still further, the workstation file 300 shows that there are three software drivers available for producing an audio output. The first software driver is "CD__AUDIO.DVR" which is a software driver which enables an audio object stored on a compact disk player which may be connected through the video adapter 212, for example, to transfer the audio data from the audio object to the stereo high fi to produce the audio presentation. The second software driver for the audio I/O function is "TAPE__AUDIO.DVR." This audio driver enables an audio object stored on a tape drive, such as can be connected through a suitable adapter to the bus 202, to output audio data from the audio object stored thereon to the stereo high fi for the audio presentation. The third software driver for the audio I/O function is "AUDIO__DATA.DVR." This driver enables an audio object such as the internal audio object 108 in FIG. 1b, to have its audio data transferred to the stereo high fi hardware for the audio presentation." (col. 9, line 41 through col. 10, line 4) |
| **985-1.h**:<br>automatically invoking the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window. | Cohen discloses that the browser parses the embed text format. *See, e.g.*, :<br><br>Cohen discloses that an embed text format is discovered by parsing an LID tag. This tag in turn points to an LDESC tag, which is also subsequently parsed.<br>"The softcopy book READ program 400 operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG. 1b are located." (col. 10, lines 29-36). See also col. 11 line 40 (describing parsing of DATA string). |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | Cohen discloses automatic invocation of the executable application. *See, e.g.*, :<br><br>Cohen discloses an attribute of the LDESC tag called AUTOLAUNCH that defines how to invoke or launch multimedia: either automatically the first time the page appears or in the alternative, only upon explicit selection. (col. 6 lines 13-17).<br>Cohen goes on to disclose this feature in more detail: "Then in step 414, a determination is made as to whether any link tags have a link description with the AUTOLAUNCH parameter equaling 'yes' in the corresponding link descriptor tag. Reference to the link description tag 102 in FIG. 1a will show that the first link description tag 120 has AUTOLAUNCH equal to no, the second link tag 140 has AUTOLAUNCH equal to no. However, the third link tag 150 has AUTOLAUNCH equal to yes. The link descriptor tag 150 is for a graphic object type, and refers to the internal graphics object 110 of FIG. 1e. In step 416 of FIG. 6, if an AUTOLAUNCH parameter is equal to 'yes,' then the program gets the DATA string from the link description. Reference to FIG. 1a will show that the link description tag 150 has the DATA='graph.exe \ GOCA Format C'. Then in step 418 of FIG. 6, the program outputs the data string to start the execution of the I/O handler program specified in the string. This is followed by the step 418 going to step 420. With reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 36-60)<br><br>Cohen discloses that the executable application displays the object. *See, e.g.*, :<br><br>Cohen discloses executable applications that display objects. For example, Cohen discloses an executable video.exe that displays objects as indicated in Figure 7a. "The video handler program whose flow diagram |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | is shown in FIG. 7a, will now be described. In step 502, the softcopy book READ program 400 outputs the DATA string, and this step 562 corresponds to either step 418 or step 426 of the softcopy book READ program 400 of FIG. 6. In the example of activating the link tag 164 in the softcopy book text 104 of FIG. 1b, this is the link to initiate the multimedia video display of a motion picture of an African elephant family." (col. 11, lines 30-38) "In step 508 of the video handler program of FIG. 7a, it is determined whether the workstation profile includes the required video support. Since the profile 300 indicates that the support is present in the workstation 200, the flow proceeds to step 524 which outputs the DATA string 'CD Video File Format A' and other control information necessary for the playing of the video information from the compact disk player which is connected through the video adapter 212 to the workstation and the presentation of the resulting motion picture on the display 208 at the workstation. Step 524 accesses the necessary data form the external video object 195 of FIG. 2 as specified by the object name 'family__clip.vid', which is the file handle for the video object 195 on the compact disk device, and the STORE=external parameters in the link descriptor 120 of FIG. 1a. The data from the object is transferred to the software driver CD_VIDEO.DVR, specified in the workstation profile 300, and the software driver controls the presentation of the motion picture received from the video adapter 212 for display on the display 208. Step 526 of FIG. 7a displays the CD video on the display 208." (col. 12, lines 1-23)<br>As another example, Cohen discloses a graph.exe executable application that displays objects as indicated in Figure 7c.<br><br>Cohen does not explicitly disclose direct interaction with an object, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br>Cohen does suggest, however, the possibility of direct interaction because it discloses a variety of multimedia objects, including those that inherently |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | require user interaction, such as spreadsheet objects. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2, line 63 - col. 3, line 2)<br><br>Cohen does not explicitly disclose direct interaction with an object, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br>Cohen does suggest, however, the possibility of direct interaction because it discloses a variety of multimedia objects, including those that inherently require user interaction at the first location, such as spreadsheet objects. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base. " (col. 2, line 63 - col. 3, line 2)<br>Cohen discloses that the object is displayed on an auxiliary display device and not at a first location in the hypermedia document, i.e. the location of an LID tag determined by parsing. Therefore there is no disclosure that interaction with an object, if possible, could be at the first location in the document. |
| | |
| **985-2.a**:<br>The method of claim 1 where: the information to enable comprises text formats. | Cohen discloses that the enabling information in the file is text formats. *See, e.g.*, :<br><br>Cohen discloses a formatted text stream in accordance with a generalized |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | markup language.<br>The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>"The method begins by storing a formatted text stream in the data processing system. The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 10-26)<br>Cohen provides examples of text formats at columns 5 through 7. |
| | |
| **985-3.a**:<br>The method of claim 2 where the text formats are HTML tags. | Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>"The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>The use of HTML tags was not disclosed by Cohen but it would have been obvious to one having ordinary skill in the art to use HTML instead of GML. |
| | |
| **985-4.a**:<br>The method of claim 1 where the information contained in the file received comprises at least one embed text format. | Cohen discloses that the enabling information in the file includes an embed text format. *See, e.g.*, :<br><br>Cohen discloses link description tags (LDESC) and link identification tags (LID).<br>The format for LDESC is set forth in column 5 line 12 through column 6 line 64. The LID is set forth at column 7 lines 3 through 30. |
| | |

| Claim Text from '985 Patent | Cohen |
|---|---|
| **985-5.a**:<br>The method of claim 1 where the step of identifying an embed text format comprises: parsing the received file to identify text formats included in the received file. | Cohen discloses that the embed text format is identified by parsing the file containing enabling information. *See, e.g.*, :<br><br>Cohen discloses that an embed text format is discovered by parsing an LID tag. This tag in turn points to an LDESC tag, which is also subsequently parsed.<br>"The softcopy book READ program 400 operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG. 1b are located." (col. 10, lines 29-36). See also col. 11 line 40 (describing parsing of DATA string). |
| | |
| **985-6.a**:<br>The method of claim 5 where the parsing is by a parser in the browser. | Cohen discloses that the parser is in the browser *See, e.g.*, :<br><br>Cohen discloses that the browser includes the parser. "The softcopy book READ program 400 operates on the book text and its tags in the page buffer 236 and constructs the memory image of the picture to be displayed, which is stored in the display buffer 238 of the memory 220. In step 412, the link tags are located in the softcopy book text. In particular, the link tags 164, 168 and 172 in the book text of FIG. 1b are located." (col. 10, lines 29-36). |
| | |
| **985-7.a**:<br>The method of claim 1 where the processing specified by the text formats is specified directly. | Cohen discloses that the text formats directly specify the processing. *See, e.g.*, :<br><br>Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>"The method begins by storing a formatted text stream in the data |

| Claim Text from ’985 Patent | Cohen |
| --- | --- |
| | processing system. The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 10-26)<br>Cohen provides examples of text formats at columns 5 through 7.<br>These text formats directly specify processing. By way of example only, "LDESC is the link tag which identifies the information that the author wishes to create a link to that follows the DOCDESC tag in the prologue of the softcopy document, and must have a DOCID attribute that points to the DOCDESC tag. The link tag: L and its matching end tag: eL enclose a word or phrase in the body of the document that the author wants to create a link from. The LID attribute refers to one or more LDESC document link tags." (col. 7, lines 22-30)<br>Also, "FIG. 1b shows the book text with tags 104. The softcopy book text includes a first portion 160 which is a heading denoted by :Hl. The second section 162 is a paragraph denoted by :P. The third section 164 is a multimedia hypertext link denoted by the beginning tag :L and the ending tag :eL. The link identification LID=eleph_movie for the link 164, relates the tag 164 to the link descriptor tag 120 of FIG. 1a. Continuing if FIG. 1b, the portion 166 is a paragraph, as denoted by the tag :P. The portion 168 is another hypertext multimedia link, as denoted by the begin tag :L and the end tag :eL. The link identifier LID=elph_sound for the link 168, relates it to the link descriptor tag 140 in FIG. 1a. In FIG. 1b, the portion 170 is a paragraph as denoted by the tag :P. The portion 172 is another hypertext multimedia link, as denoted by the begin tag :L and the corresponding end tag :eL. The link identification LID=pop_graphic for the link 172 relates it to the link descriptor tag 150 in FIG. 1a." (col. 7, lines 46-66) |
| | |

| Claim Text from '985 Patent | Cohen |
|---|---|
| **985-8.a**:<br>The method of claim 1 where the correspondence is implied by the order of the text format in a set of all of the text formats. | Cohen discloses that the correspondence is implied by the order of text formats. *See, e.g.*, :<br><br>On Cohen, the correspondence of displayed objects was implied by the order of link identification ("LID") tags. By way of example, Cohen discloses in col. 7 lines 46-60 on example of how tags correspond to locations within the hypermedia document. (See also Figure 1b)<br>In Cohen, the LDESC embed text formats are kept in the prologue of the document. This was actually a feature, because it allowed an author to define the embed text format once, and then re-use that text format within the hypermedia document using a shorter link identification tag, without needing to re-type the full LDESC text format. "The link identification LID=eleph_movie for the link 164, relates the tag 164 to the link descriptor tag 120 of FIG. 1a." (col. 7, lines 50-54).<br>Accordingly, it would have been obvious to use LDESC text formats within the document's formatted text stream such that the correspondence would be implied by the order of the LDESC text format. |
| | |
| **985-9.a**:<br>The method of claim 1 where the embed text format specifies the location of at least a portion of an object directly. | The embed text format, an instance of an LID tag, does not directly specify the location of an object. Rather, it references an LDESC tag which directly specifies the location of an object.<br><br>The LDESC tag includes an OBJECT attribute that specifies the location of an object directly by identifying that object. (*See* col. 5 lines 44-49). In addition, Cohen discloses that the formatted text stream includes an object's direct location information. "The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 18-26) |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | |
| | |
| **985-10.a**:<br>The method of claim 1 where having type information associated is by including type information in the embed text format. | Cohen discloses that the type information is in the embed text format. *See, e.g.*, :<br><br>Cohen discloses that the embed text format, which is discovered by parsing an LID tag, contains type information by implication. This ensues from the LID tag reference to a corresponding LDESC tag stored in the document prologue. There, type is indicated by the OBJTYPE attribute of the LDESC tag. "OBJTYPE=object-type, Identifies the type of information the author wants to create a link to. The object-type can be one of the following values: PROGRAM/ANIMATION/VIDEO/AUDIO/GRAPHIC/IMAGE. " (col. 5, lines 49-54)<br>By way of example, "In FIG. 1a, the link description tags 102 include three tags. A first tag 120 for a video object type, a second tag 130 for an audio object type, and a third tag 150 for a graphic object type." (col. 7, lines 42-45)<br>The LDESC tag also includes a DATA attribute in the embed text format that serves to provide type information. "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler 15 program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. 7a." (col. 11, lines 12-19) |
| | |
| **985-11.a**:<br>The method of claim 1 where automatically invoking does not require interactive action by the user. | Cohen discloses that automatic invocation does not require interactive action by the user. *See, e.g.*, :<br><br>Cohen discloses an attribute of the LDESC tag called AUTOLAUNCH that defines how to invoke or launch multimedia: either automatically the first time the page appears, or in the alternative only upon explicit |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | selection. (col. 6 lines 13-17).<br>Cohen goes on to disclose this feature in more detail: "Then in step 414, a determination is made as to whether any link tags have a link description with the AUTOLAUNCH parameter equaling 'yes' in the corresponding link descriptor tag. Reference to the link description tag 102 in FIG. 1a will show that the first link description tag 120 has AUTOLAUNCH equal to no, the second link tag 140 has AUTOLAUNCH equal to no. However, the third link tag 150 has AUTOLAUNCH equal to yes. The link descriptor tag 150 is for a graphic object type, and refers to the internal graphics object 110 of FIG. 1e. In step 416 of FIG. 6, if an AUTOLAUNCH parameter is equal to 'yes,' then the program gets the DATA string from the link description. Reference to FIG. 1a will show that the link description tag 150 has the DATA='graph.exe \ GOCA Format C'. Then in step 418 of FIG. 6, the program outputs the data string to start the execution of the I/O handler program specified in the string. This is followed by the step 418 going to step 420. With reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 36-60) |
| | |
| **985-16.a**:<br>One or more computer readable media encoded with software comprising computer executable instructions, for use in a distributed hypermedia network environment, wherein the network environment comprises at least one client workstation and one network server coupled to the network environment, and when the software is executed operable to: | Cohen discloses computer code physically embodied on a medium. *See, e.g.*, :<br><br>Cohen discloses that the BookManager READ program to which Cohen's disclosure is directed is stored on computer code physically embodied on a medium: "Also stored in the memory 220 is the softcopy book READ program 400 of FIG. 6, the I/O handler programs 500, 530 and 560 of FIGS. 7a, 7b and 7c, respectively, and the drivers and operating system 590. The CPU 204 of FIG. 4, executes the instructions embodied in the program 400 and in the handler programs 500, 530 and 560, in accordance with the invention." (col. 9, lines 6-13.)<br>Cohen further discloses that "FIG. 6 is a flow diagram of the softcopy |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | book READ program 400, in accordance with the invention. The program resides in a partition of the memory 220 of the workstation in FIG. 4, or alternately in the memory of a host processor in a host data processing system. " (col. 10, lines 12-16)<br><br>Cohen discloses a client workstation and a network server in a distributed hypermedia environment. *See* evidence recited for 985-1.a. |
| **985-16.b**:<br>receive, at the client workstation from the network server over the network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Cohen discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. *See* evidence recited for 985-1.b. |
| **985-16.c**:<br>cause the client workstation to utilize the browser to: | Cohen discloses a browser application executing on the client workstation. *See* evidence recited for 985-1.c. |
| **985-16.d**:<br>respond to text formats to initiate processing specified by the text formats; | Cohen discloses parsing text formats. *See* evidence recited for 985-1.d. |
| **985-16.e**:<br>display at least a portion of the document within the browser-controlled window; | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e. |
| **985-16.f**:<br>identify an embed text format corresponding to a first location in the document, the embed text format specifying the location of at least a portion of an object external to the file, with the object having type information associated with it; | Cohen discloses identifying an embed text format; that the embed text format corresponds to a first location in a hypermedia document; that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information; and that the object has associated type information. *See* evidence recited for 985-1.f. |
| **985-16.g**:<br>utilize the type information to identify and locate an executable application external to the file; and | Cohen discloses using type information to identify and locate an executable application external to the file. *See* evidence recited for 985-1.g. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| **985-16.h**:<br>automatically invoke the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window. | Cohen discloses automatically invoking the executable application; that the executable application displays the object. *See* evidence recited for 985-1.h. |
| | |
| **985-17.a**:<br>The computer readable media of claim 16 where: the information to enable comprises text formats. | Cohen discloses that the enabling information in the file is text formats. *See* evidence recited for 985-2.a. |
| | |
| **985-18.a**:<br>The computer readable media of claim 17 where: the text formats are HTML tags. | Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>"The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>The use of HTML tags was not disclosed by Cohen but it would have been obvious to one having ordinary skill in the art to use HTML instead of GML.<br><br>*See* evidence recited for 985-3.a. |
| | |
| **985-19.a**:<br>The computer readable media of claim 16 where: the information contained in the file received comprises at least one embed text format. | Cohen discloses that the enabling information in the file includes an embed text format. *See* evidence recited for 985-4.a. |
| | |
| **985-20.a**:<br>A method of serving digital information in a | Cohen discloses digital information. *See, e.g.*, : |

| Claim Text from '985 Patent | Cohen |
|---|---|
| computer network environment having a network server coupled the network environment, and where the network environment is a distributed hypermedia environment, the method comprising: | Cohen discloses that the information transmitted between the client and the server is digital information. Specifically, Cohen discloses a formatted text stream, which would be transmitted as digital information. "The method begins by storing a formatted text stream in the data processing system. The formatted text stream includes a link description which contains multimedia type information, object location information and multimedia control information for a target multimedia object. The formatted text stream further includes a link tag associated with the link description, which identifies a source location in the formatted text stream from which a link is established to the target multimedia object." (col. 2, lines 17-26)<br>One example of digital information disclosed in Cohen is transmission of digital audio objects. "FIG. 3b shows an example of how the I/O data 184 in the internal audio object 108 of FIG. 1d, can be generated from a sampled analog sound signal 184'. The example shown in FIG. 3b makes use of simple pulse height modulation to characterize each sampled analog amplitude in a signal 184' as a one out of 16 value, represented as a four binary bit expression in the audio data 184. A digitized value of each sample of the sound signal can then be stored as the audio data 184 in the internal audio object 108 of FIG. 1d." (col. 8, lines 36-45)<br>Cohen discloses other types of digital information as well. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2, line 2 through col. 3 line 2)<br>Cohen also discloses networks that transmit digital information, such as a local area network. (col. 8, line 65)<br><br>Cohen discloses a network server in a distributed hypermedia environment. *See* |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | evidence recited for 985-1.a. |
| **985-20.b**:<br>communicating via the network server with at least one client workstation over said network in order to cause said client workstation to: | Cohen discloses a client workstation. *See* evidence recited for 985-1.a.<br><br>Cohen discloses communicating via network server in order to cause the client workstation to act. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br>It would have been obvious to communicate via the network server in order to cause the client workstation to act in view of Cohen's disclosure of network interoperability and database application interoperability. |
| **985-20.c**:<br>receive, over said network environment from said server, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Cohen discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. *See* evidence recited for 985-1.b. |
| **985-20.d**:<br>execute, at said client workstation, a browser application, with the browser application: | Cohen discloses a browser application executing on the client workstation. *See* evidence recited for 985-1.c. |
| **985-20.e**:<br>responding to text formats to initiate processing specified by the text formats; | Cohen discloses parsing text formats. *See* evidence recited for 985-1.d. |
| **985-20.f**:<br>displaying, on said client workstation, at least a portion of the document within the browser-controlled window; | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| **985-20.g**:<br>identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object external to the file, where the object has type information associated with it; | Cohen discloses identifying an embed text format; that the embed text format corresponds to a first location in a hypermedia document; that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information; and that the object has associated type information. *See* evidence recited for 985-1.f. |
| **985-20.h**:<br>utilizing the type information to identify and locate an executable application external to the file; and | Cohen discloses using type information to identify and locate an executable application external to the file.<br><br>*See* evidence recited for 985-1.g. |
| **985-20.i**:<br>automatically invoking the executable application, in response to the identifying of the embed text format, to execute on the client workstation in order to display the object and enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window. | Cohen discloses automatically invoking the executable application; that the executable application displays the object. *See* evidence recited for 985-1.h. |
| | |
| **985-21.a**:<br>The method of claim 20 where: the information to enable comprises text formats. | Cohen discloses that the enabling information in the file is text formats. *See* evidence recited for 985-2.a. |
| | |
| **985-22.a**:<br>The method of claim 21 where: the text formats are HTML tags. | Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>"The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>The use of HTML tags was not disclosed by Cohen but it would have been |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | obvious to one having ordinary skill in the art to use HTML instead of GML.<br><br>*See* evidence recited for 985-3.a. |
| | |
| **985-23.a**:<br>The method of claim 20 where: the information contained in the file received comprises at least one embed text format. | Cohen discloses that the enabling information in the file includes an embed text format. *See* evidence recited for 985-4.a. |
| | |
| **985-24.a**:<br>A method for running an executable application in a computer network environment, wherein said network environment has at least one client workstation and one network server coupled to a network environment, the method comprising: | Cohen discloses a client workstation and a network server in a network environment. *See* evidence recited for 985-1.a.<br><br>Cohen discloses an executable application. *See* evidence recited for 985-1.g. |
| **985-24.b**:<br>enabling an end-user to directly interact with an object by utilizing said executable application to interactively process said object while the object is being displayed within a display area created at a first location within a portion of a hypermedia document being displayed in a browser-controlled window, | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e.<br><br>Cohen discloses an object external to a file containing enabling information. *See* evidence recited for 985-1.f.<br><br>Cohen does not explicitly disclose direct interaction with an object, except for the obvious interaction of starting, pausing and stopping the presentation of multimedia objects.<br>Cohen does suggest, however, the possibility of direct interaction because it discloses a variety of multimedia objects, including those that inherently require user interaction, such as spreadsheet objects. "Many different kinds of multimedia objects can be linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | engineering diagram using a computer aided design data base. " (col. 2, line 63 - col. 3, line 2)<br><br>Cohen discloses that the object is displayed on an auxiliary display device and not at a first location in the hypermedia document, i.e. the location of an LID tag determined by parsing.<br><br>*See* evidence recited for 985-1.h. |
| **985-24.c**:<br>wherein said network environment is a distributed hypermedia environment, | Cohen discloses a client workstation and a network server in a distributed hypermedia environment. *See* evidence recited for 985-1.a. |
| **985-24.d**:<br>wherein said client workstation receives, over said network environment from said server, at least one file containing information to enable said browser application to display, on said client workstation, at least said portion of said distributed hypermedia document within said browser-controlled window, | Cohen discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. *See* evidence recited for 985-1.b. |
| **985-24.e**:<br>wherein said executable application is external to said file, | Cohen discloses an executable application external to the file. *See* evidence recited for 985-1.g. |
| **985-24.f**:<br>wherein said client workstation executes the browser application, with the browser application responding to text formats to initiate processing specified by the text formats, | Cohen discloses a browser application executing on the client workstation. *See* evidence recited for 985-1.c.<br><br>Cohen discloses parsing text formats. *See* evidence recited for 985-1.d. |
| **985-24.g**:<br>wherein at least said portion of the document is displayed within the browser-controlled window, | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e. |
| **985-24.h**:<br>wherein an embed text format which corresponds to said first location in the document is identified | Cohen discloses identifying an embed text format and that the embed text format corresponds to a first location in a hypermedia document. *See* evidence recited for 985-1.f. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| by the browser, | |
| **985-24.i**:<br>wherein the embed text format specifies the location of at least a portion of said object external to the file, | Cohen discloses that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information. *See* evidence recited for 985-1.f. |
| **985-24.j**:<br>wherein the object has type information associated with it, | Cohen discloses that the object has associated type information. *See* evidence recited for 985-1.f. |
| **985-24.k**:<br>wherein the type information is utilized by the browser to identify and locate said executable application, and | Cohen discloses using type information to identify and locate an executable application external to the file. *See* evidence recited for 985-1.g. |
| **985-24.l**:<br>wherein the executable application is automatically invoked by the browser, in response to the identifying of the embed text format. | Cohen discloses automatically invoking the executable application. *See* evidence recited for 985-1.h. |
| | |
| **985-25.a**:<br>The method of claim 24 where: the information to enable comprises text formats. | Cohen discloses that the enabling information in the file is text formats. *See* evidence recited for 985-2.a. |
| | |
| **985-26.a**:<br>The method of claim 25 where: the text formats are HTML tags. | Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>"The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>The use of HTML tags was not disclosed by Cohen but it would have been obvious to one having ordinary skill in the art to use HTML instead of GML.<br><br>*See* evidence recited for 985-3.a. |
| | |
| **985-27.a**: | Cohen discloses that the enabling information in the file includes an embed text |

| Claim Text from '985 Patent | Cohen |
|---|---|
| The method of claim 24 where: the information contained in the file received comprises at least one embed text format. | format. *See* evidence recited for 985-4.a. |
| | |
| **985-28.a**:<br>One or more computer readable media encoded with software comprising an executable application for use in a system having at least one client workstation and one network server coupled to a network environment, operable to: | Cohen discloses computer code physically embodied on a medium. *See* evidence recited for 985-16.a.<br><br>Cohen discloses a client workstation and a network server in a network environment. *See* evidence recited for 985-1.a.<br><br>Cohen discloses an executable application. *See* evidence recited for 985-1.g. |
| **985-28.b**:<br>cause the client workstation to display an object and enable an end-user to directly interact with said object while the object is being displayed within a display area created at a first location within a portion of a hypermedia document being displayed in a browser-controlled window, | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e.<br><br>Cohen discloses an object external to a file containing enabling information. *See* evidence recited for 985-1.f.<br><br>Cohen discloses that the object is displayed on an auxiliary display device and not at a first location in the hypermedia document, i.e. the location of an LID tag determined by parsing.<br><br>*See* evidence recited for 985-1.h. |
| **985-28.c**:<br>wherein said network environment is a distributed hypermedia environment, | Cohen discloses a client workstation and a network server in a distributed hypermedia environment. *See* evidence recited for 985-1.a. |
| **985-28.d**:<br>wherein said client workstation receives, over said network environment from said server, at least one file containing information to enable said browser application to display, on said client workstation, at least said portion of said distributed hypermedia document within said browser-controlled window, | Cohen discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. *See* evidence recited for 985-1.b. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| **985-28.e**:<br>wherein said executable application is external to said file, | Cohen discloses an executable application external to the file. *See* evidence recited for 985-1.g. |
| **985-28.f**:<br>wherein said client workstation executes said browser application, with the browser application responding to text formats to initiate processing specified by the text formats, | Cohen discloses a browser application executing on the client workstation. *See* evidence recited for 985-1.c.<br><br>Cohen discloses parsing text formats. *See* evidence recited for 985-1.d. |
| **985-28.g**:<br>wherein at least said portion of the document is displayed within the browser-controlled window, | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e. |
| **985-28.h**:<br>wherein an embed text format which corresponds to said first location in the document is identified by the browser, | Cohen discloses identifying an embed text format and that the embed text format corresponds to a first location in a hypermedia document. *See* evidence recited for 985-1.f. |
| **985-28.i**:<br>wherein the embed text format specifies the location of at least a portion of said object external to the file, | Cohen discloses that the embed text format specifies the location of at least a portion of an object external to the file containing enabling information. *See* evidence recited for 985-1.f. |
| **985-28.j**:<br>wherein the object has type information associated with it, | Cohen discloses that the object has associated type information. *See* evidence recited for 985-1.f. |
| **985-28.k**:<br>wherein the type information is utilized by the browser to identify and locate said executable application, and | Cohen discloses using type information to identify and locate an executable application external to the file. *See* evidence recited for 985-1.g. |
| **985-28.l**:<br>wherein the executable application is automatically invoked by the browser, in response to the identifying of the embed text format. | Cohen discloses automatically invoking the executable application. *See* evidence recited for 985-1.h. |
| | |
| **985-36.a**: | Cohen discloses an application program in a distributed hypermedia environment |

| Claim Text from '985 Patent | Cohen |
|---|---|
| A method for running an application program in a distributed hypermedia network environment, wherein the distributed hypermedia network environment comprises at least one client workstation and one remote network server coupled to the distributed hypermedia network environment, the method comprising: | comprising at least client workstation and network server. *See* evidence recited for 985-1.a. |
| **985-36.b**:<br>receiving, at the client workstation from the network server over the distributed hypermedia network environment, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Cohen discloses a browser application; a file containing enabling information; that the file is received at the client workstation from the network server; that the browser displays at least a portion of a distributed hypermedia document; and that at least a portion of a hypermedia document is displayed in a browser-controlled window. *See* evidence recited for 985-1.b. |
| **985-36.c**:<br>executing the browser application on the client workstation, with the browser application: | Cohen discloses a browser application executing on the client workstation. *See* evidence recited for 985-1.c. |
| **985-36.d**:<br>responding to text formats to initiate processing specified by the text formats; | Cohen discloses parsing text formats. *See* evidence recited for 985-1.d. |
| **985-36.e**:<br>displaying at least a portion of the document within the browser-controlled window; | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e. |
| **985-36.f**:<br>identifying an embed text format which corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object; | Cohen discloses an object. *See, e.g.*, :<br><br>Cohen discloses multimedia objects.<br>For example, Cohen discloses "[a] method, program and data processing system are disclosed, for providing a generalized link from a reference point within an organized hierarchy of a formatted text stream in an on-line book, to an arbitrary type multimedia object." (col. 2, lines 11-16)<br>Further, Cohen states "[m]any different kinds of multimedia objects can be |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | linked into a softcopy book. Multimedia objects such as high resolution, photographic quality graphics, motion video, or sound can be supported by the invention. In addition, other functions which can be included in an expanded definition of multimedia can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2, line 2 through col. 3 line 2)<br>The LDESC tag includes an OBJECT attribute (which identifies an object) and an OBJTYPE attribute (which identifies the type of information for that object). (col. 5, lines 44-54)<br><br>Cohen discloses identifying an embed text format; that the embed text format corresponds to a first location in the hypermedia document; and that the embed text format specifies the location of an object. *See* evidence recited for 985-1.f. |
| **985-36.g**:<br>identifying and locating an executable application associated with the object; and | Cohen discloses that the browser identifies and locates an executable application associated with the object. *See, e.g.*,<br><br>Cohen discloses the use of an object's multimedia type information. "The method then continues by storing a multimedia handler program in the data processing system, the handler program controlling operations of a multimedia output device characterized by the multimedia type information." (col. 2, lines 31-35)<br>Using the DATA attribute by way of example, Cohen discloses: "With reference to the graphic object type link descriptor 150 of FIG. 1a, the string 'graph.exe \ GOCA Format C' is output by the softcopy book READ program 400 to begin execution of the specified I/O handler program, namely graph.exe, whose flow diagram is shown in FIG. 7c." (col. 10, lines 54-60) As another example, "The data for the link descriptor tag 120 is 'video.exe CD Video File Format A'. Step 424 of the flow diagram of FIG. 6 gets that DATA string and outputs it in step 426 to start the execution of the I/O handler program specified in the string. Then step 426 goes to step 420. The string 'video.exe CD Video File Format A' specifies the I/O handler program video.exe, whose flow diagram is shown in FIG. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | 7a." (col. 11, lines 12-19)<br>Cohen discloses still other examples of executable applications that are identified and located: "The profile 300 includes the hardware types for a particular I/O function, characteristics for each hardware type, and the software drivers which enable the application programs and I/O handler programs to interact with the particular I/O hardware or software. For example, if the I/O function is audio, the user's workstation profile 300 shows that there is a stereo high fi connected through the audio adapter 214 to the workstation 200. The workstation profile 300 further describes that the stereo high fi hardware has, among its other characteristics, a frequency response of 20 to 20000 Hertz and it can handle delta mod data in FORMAT B. Still further, the workstation file 300 shows that there are three software drivers available for producing an audio output. The first software driver is "CD__AUDIO.DVR" which is a software driver which enables an audio object stored on a compact disk player which may be connected through the video adapter 212, for example, to transfer the audio data from the audio object to the stereo high fi to produce the audio presentation. The second software driver for the audio I/O function is "TAPE__AUDIO.DVR." This audio driver enables an audio object stored on a tape drive, such as can be connected through a suitable adapter to the bus 202, to output audio data from the audio object stored thereon to the stereo high fi for the audio presentation. The third software driver for the audio I/O function is "AUDIO__DATA.DVR." This driver enables an audio object such as the internal audio object 108 in FIG. 1b, to have its audio data transferred to the stereo high fi hardware for the audio presentation." (col. 9, line 41 through col. 10, line 4) |
| **985-36.h**:<br>automatically invoking the executable application, in response to the identifying of the embed text format, in order to enable an end-user to directly interact with the object, while the object is being displayed within a display area created at the first | Cohen discloses identifying an embed text format. *See* evidence recited in 985-1.f.<br><br>Cohen discloses automatic invocation of the executable application and that the executable application displays the object. *See* evidence recited in 985-1.h. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| location within the portion of the hypermedia document being displayed in the browser-controlled window, | Cohen discloses that a hypermedia document is displayed in a browser window. *See, e.g.*, evidence recited for 985-1.e. |
| **985-36.i**:<br>wherein the executable application is part of a distributed application, and | Cohen discloses a distributed application. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br>It would have been obvious for the browser disclosed in Cohen to interoperate with a distributed application in view of Cohen's disclosure of network interoperability and database application interoperability.<br><br>Cohen discloses that the executable application is part of a distributed application. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br>It would have been obvious for the browser disclosed in Cohen to interoperate with an executable application that is part of a distributed application in view of Cohen's disclosure of network interoperability and database application interoperability. |
| **985-36.j**: | Cohen discloses that the distributed application executes at least partially on a |

| Claim Text from '985 Patent | Cohen |
|---|---|
| wherein at least a portion of the distributed application is for execution on a remote network server coupled to the distributed hypermedia network environment. | network server. *See, e.g.*, :<br><br>Cohen discloses that that the client workstation can connect to other computers using a local area network. (col. 8 line 65) Cohen further discloses that its browser could interoperate with database applications. "In addition, other functions which can be included in an expanded definition of multimedia, can also be presented, such as a spread sheet, or an engineering diagram using a computer aided design data base." (col. 2 line 66 through col. 3 line 2)<br>It would have been obvious for the browser disclosed in Cohen to interoperate with a distributed application that executes at least partially on the network server in view of Cohen's disclosure of network interoperability and database application interoperability. |
| | |
| **985-37.a**:<br>The method of claim 36 where: the information to enable comprises text formats. | Cohen discloses that the enabling information in the file is text formats. *See* evidence recited for 985-2.a. |
| | |
| **985-38.a**:<br>The method of claim 37 where: the text formats are HTML tags. | Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>"The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized Markup Language (GML)… ." (col. 4 lines 20-23.)<br>The use of HTML tags was not disclosed by Cohen but it would have been obvious to one having ordinary skill in the art to use HTML instead of GML.<br><br>*See* evidence recited for 985-3.a. |
| | |
| **985-39.a**:<br>The method of claim 36 where: the information contained in the file received comprises at least one embed text format. | Cohen discloses that the enabling information in the file includes an embed text format. *See* evidence recited for 985-4.a. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | |
| **985-40.a**:<br>A method of serving digital information in a computer network environment having a network server coupled to said computer network environment, and where the network environment is a distributed hypermedia network environment, the method comprising: | Cohen discloses digital information. *See* evidence recited for 985-20.a.<br><br>Cohen discloses a network server in a distributed hypermedia environment. *See* evidence recited for 985-1.a. |
| **985-40.b**:<br>communicating via the network server with at least one remote client workstation over said computer network environment in order to cause said client workstation to: | Cohen discloses a client workstation. *See* evidence recited for 985-1.a.<br><br>Cohen discloses communicating via network server in order to cause the client workstation to act. *See* evidence recited for 985-20.b. |
| **985-40.c**:<br>receive, over said computer network environment from the network server, at least one file containing information to enable a browser application to display at least a portion of a distributed hypermedia document within a browser-controlled window; | Cohen discloses a browser application; a file containing enabling information received from a server; that the browser displays at least a portion of a distributed hypermedia document; and that the display is in a browser-controlled window. *See* evidence recited for 985-1.b. |
| **985-40.d**:<br>execute, at said client workstation, a browser application, with the browser application: | Cohen discloses a browser application executing on the client workstation. *See* evidence recited for 985-1.c. |
| **985-40.e**:<br>responding to text formats to initiate processing specified by the text formats; | Cohen discloses parsing text formats. *See* evidence recited for 985-1.d. |
| **985-40.f**:<br>displaying, on said client workstation, at least a portion of the document within the browser-controlled window; | Cohen discloses displaying at least a portion of the document within the browser-controlled window. *See* evidence recited for 985-1.e. |
| **985-40.g**:<br>identifying an embed text format which | Cohen discloses an object. *See* evidence recited for 985-36.f. |

| Claim Text from '985 Patent | Cohen |
|---|---|
| corresponds to a first location in the document, where the embed text format specifies the location of at least a portion of an object; | Cohen discloses identifying an embed text format; that the embed text format corresponds to a first location in the hypermedia document; and that the embed text format specifies the location of an object. *See* evidence recited for 985-1.f. |
| **985-40.h**:<br>identifying and locating an executable application associated with the object; and | Cohen discloses that the browser identifies and locates an executable application associated with the object. *See* evidence recited for 985-36.g. |
| **985-40.i**:<br>automatically invoking the executable application, in response to the identifying of the embed text format, in order to enable an end-user to directly interact with the object while the object is being displayed within a display area created at the first location within the portion of the hypermedia document being displayed in the browser-controlled window, | Cohen discloses identifying an embed text format. *See* evidence recited in 985-1.f.<br><br>Cohen discloses automatic invocation of the executable application and that the executable application displays the object. *See* evidence recited in 985-1.h.<br><br>Cohen discloses that a hypermedia document is displayed in a browser window. *See, e.g.*, evidence recited for 985-1.e. |
| **985-40.j**:<br>wherein the executable application is part of a distributed application, and | Cohen discloses that the executable application is part of a distributed application. *See* evidence recited in 985-36.i. |
| **985-40.k**:<br>wherein at least a portion of the distributed application is for execution on the network server. | Cohen discloses that the distributed application executes at least partially on a network server. *See* evidence recited for 985-36.j. |
| | |
| **985-41.a**:<br>The method of claim 40 where: the information to enable comprises text formats. | Cohen discloses that the enabling information in the file is text formats. *See* evidence recited for 985-2.a. |
| | |
| **985-42.a**:<br>The method of claim 41 where: the text formats are HTML tags. | Cohen discloses a formatted text stream in accordance with a generalized markup language.<br>"The BookManager BUILD and BookManager READ program products use on-line, softcopy books which are formatted using the Generalized |

| Claim Text from '985 Patent | Cohen |
|---|---|
| | Markup Language (GML)… ." (col. 4 lines 20-23.)<br>The use of HTML tags was not disclosed by Cohen but it would have been obvious to one having ordinary skill in the art to use HTML instead of GML.<br><br>*See* evidence recited for 985-3.a. |
| | |
| **985-43.a**:<br>The method of claim 40 where: the information contained in the file received comprises at least one embed text format. | Cohen discloses that the enabling information in the file includes an embed text format. *See* evidence recited for 985-4.a. |
| | |