UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF COLUMBIA

DOW JONES & COMPANY, INC.
200 Liberty Street
New York, New York 10281

Plaintiff,

v.

ABLAISE LTD. ("Ablaise")
40 Queen Anne Street
London W1G 9EL
United Kingdom

and

GENERAL INVENTIONS INSTITUTE A, INC., ("GIIA")
Craigmuir Chambers
P.O. Box 71
Town Road
Tortola, British Virgin Islands

Defendants.

Civil Action No. 1:06CV01014

Judge James Robertson

DOW JONES REUTERS BUSINESS INTERACTIVE, LLC
200 Liberty Street
New York, New York 10281

Plaintiff,

v.

ABLAISE and GIIA

Defendants.

Civil Action No. 1:06CV01015

Judge James Robertson

ABLAISE LTD. AND GENERAL INVENTIONS INSTITUTE A, INC.'S RESPONSIVE MARKMAN BRIEF IN SUPPORT OF ITS PROPOSED CLAIM CONSTRUCTION

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES

Page

Cases

Other Authorities

INTRODUCTION

Ablaise Limited ("Ablaise") respectfully submits this Memorandum in support of its proposed construction of disputed claim terms of claim 1 of each of the patents in suit, United States Patent No. 6,295,530 ("The '530 Patent") and 6,961,737 ("The '737 Patent"). In this brief, Ablaise provides the Court with an overview of the relevant web page delivery technology, the history of the company and the invention of custom formatted web pages. As the Court will see, the '530 patent and '737 patent claim related but different ways to customize the look of a web page for someone using the Internet. Ablaise then provides the Court with the plain and ordinary meaning of the disputed patent claim terms to one of ordinary skill in the art at the time of the invention. Ablaise's definitions are supported by the claim language itself, the specification, the file history and, should the Court require it, extrinsic evidence including expert testimony.

Dow Jones & Company Inc. ("Dow") has taken a fundamentally different and incorrect approach. Dow argues that the patents-in-suit are invalid or not infringed. This result-oriented approach is not unusual, but in this case it has caused Dow to ignore many of the basic canons of claim construction. For instance, Dow repeatedly avoids any attempt to ascertain the plain and ordinary meaning of the terms actually present in the claims, and instead simply adds words or phrases describing the details of the preferred embodiment. By way of example, Dow construes the phrase "type of formatting" to mean "indexed function string." A type of formatting is essentially one way for a web page to look. The language "indexed function string" is not even an example of a type of formatting, but simply reads the minute details of the computer processing described in the preferred embodiment into the claim. Ablaise respectfully requests that the Court reject Dow's approach of rewriting the claims at the level of detail in the

specification. Instead, the disputed claim terms should be construed as they would be understood by one of ordinary skill in the art in light of the specification.

I. BACKGROUND OF THE INVENTION

In general terms, the '530 and '737 patents provide a method for specialized computers commonly referred to as "web servers" to automatically respond to how people using the World Wide Web want information on a particular web page to look. Prior to the invention, only web page programmers could dictate the look of the web pages their web servers sent out over the web. Thus, a user could retrieve a particular web page, and in rare circumstances even customize the content on that web page, but the user could not define the look and format of the content on the web page. The inventors believed it would be superior to identify what Web users wanted and to provide automatically customized web pages. They invented the technique of allowing a web server to keep track of different formats for different internet users through the use of "format identifiers" thereby allowing users to control the look of web pages without the involvement of a computer programmer.

The inventors' computerized method of delivering web pages could have many capabilities, but there are two keys to achieving the results of the claimed inventions. First, the inventors separated the information on the web page from where it appears or how it looks. The information itself is usually text or graphics and is called "content." Where the information appears or how it looks is called "format." The inventors made format independent of "content" so that the same content could be sent to different users in different formats. Second, they conceived of using a format identifier to determine the format for a particular user. The choice of one format over another is based on information that comes from an external source to the computer generating the web page.

A. Delivering Web Pages Over the Internet

The Internet is a global network of computers that allows computers to communicate with each other. The Internet is the delivery mechanism for many different methods of communication. One of the most well-known methods of communications over the Internet today is the delivery of web pages over the World-Wide Web.

The delivery of web pages is usually completed by communication between a "browser" and a "server." Users can view or "browse" web pages by using a "browser." A browser is a computer software program that is capable of sending a request for a web page. The server responds to the request by returning a file corresponding to the page requested by the browser. A browser is also capable of interpreting the file returned from the server and displaying it to a user.

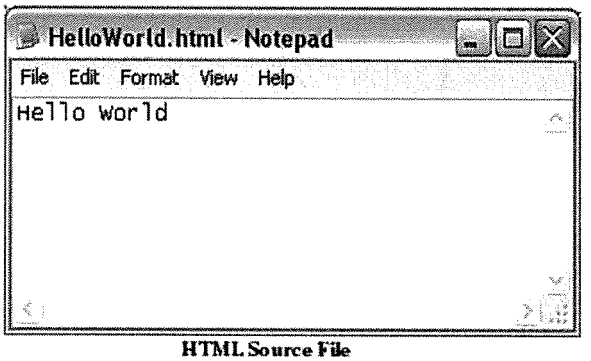
Web pages must conform to standard types of formatting languages called "markup languages." These standard markup languages consist of instructions that allow people to write content that can be specially formatted and viewed by others regardless of the type of operating system (e.g., Windows or Macintosh) or hardware that the user is running. If files returned to a web browser from a server contain these standard markup instructions, the web browser is able to parse or interpret the markup file and display content such as text and/or graphics in a particular format. Rosenbloom Decl. Exh. A, at Col. 3, ll. 18-38.

The most common markup language used in web pages is the HyperText Markup Language known by its acronym "HTML." HTML is used to format the content on a web page in a particular way. It accomplishes this by surrounding certain content with special codes referred to as "tags." Most "tags" correspond to a particular style of formatting. Tags typically come in pairs: an open tag and a close tag. For instance, if a programmer writing a webpage wished to place the phrase "Hello World" in a heading format (e.g., larger text size and bolded)

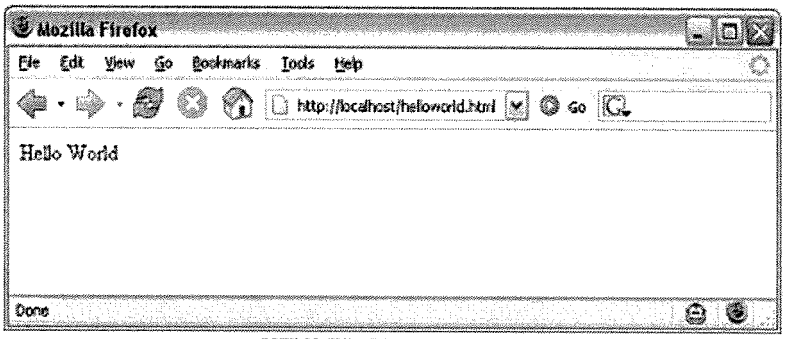
using HTML, she would write the following "<H1>Hello World</H1>" into a file called the HTML source file. See, e.g., Rosenbloom Decl. Exh. A, Fig. 8, ll 4. In this example, the "<H1>" is an open heading tag and the "</H1>" is a close heading tag. Any content between the open heading tag and the close heading tag will be displayed in a larger text size and bolded when a browser displays that particular page.

The examples visually shown on the following page illustrate how HTML tags format a web page. Example One shows content that does not include formatting HTML tags. The page on the left is the HTML source file. The page on the right shows how a browser renders that HTML file. As you can see, the content in Example One is not displayed in any special format. Example Two illustrates how that same content looks when heading tags are used in the HTML source file. Now you can see in the browser window that the format of the content has changed. The text has increased in size and the text is now bold. Likewise, Example Three shows what happens to the content when it is placed in both heading tags and center tags. This has the effect of changing the location of the content displayed at the browser. In particular, the large bold text is now centered within the browser window.

Example One - No HTML Tags

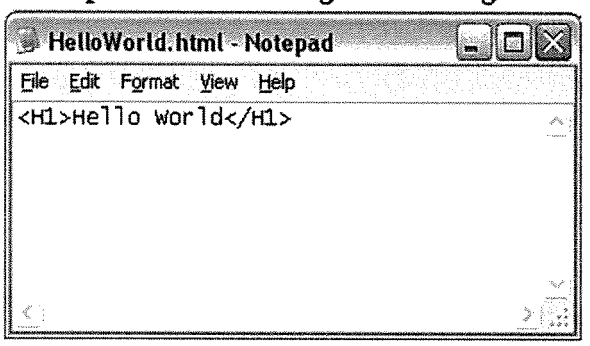


HTML Source File

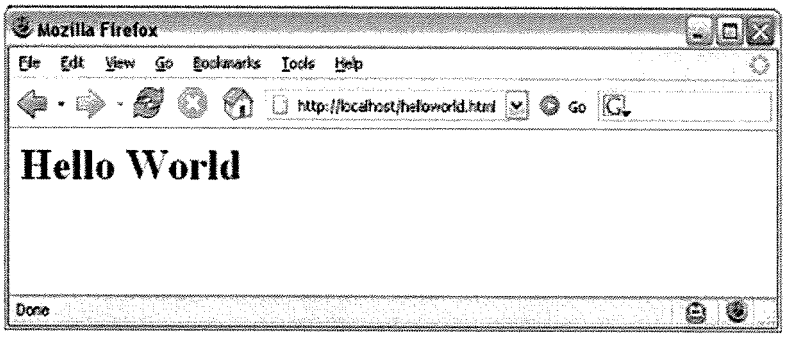


HTML File Displayed in Browser

Example Two - Heading HTML Tags

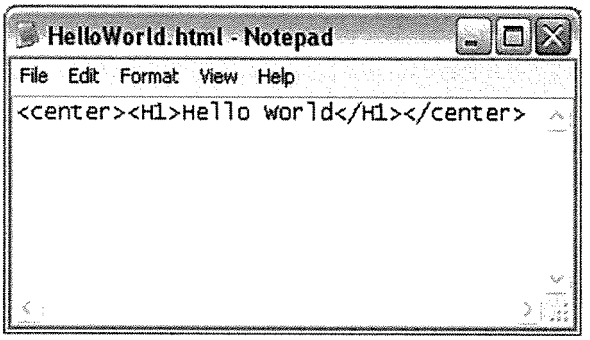


HTML Source File



HTML File Displayed in Browser

Example Three - Heading & Center HTML Tags



HTML Source File

HTML File Displayed in Browser

As you can see from these examples, a programmer can cause a browser to display the same content in different formats by altering HTML tags in the HTML source file.

B. The State of the World Wide Web Prior to the Invention

The Web has undergone a remarkable transformation over its short history. For example, the Web had only 18,957 web servers in August of 1995 and now has over 100 million web

servers. (Hicks Aff. ¶ 20.) At the time of the invention, in 1995, the Web was in its earliest stages of development. Unlike the highly dynamic and rich web pages available today, most web pages at the time of the invention consisted of "static" content and formatting defined exclusively by web developers, who wrote the HTML files by hand. Rosenbloom Decl. Exh. A at Col. 4 ll. 23-49. Users could not send a request to a web server to change either the content displayed or the layout. A user could contact the web developer who would then have to recreate the page by hand in accordance with the user's request—often abandoning the previous version of the document. Although relatively rare, some web pages during this time period were dynamically created "on the fly" with content from databases or some other external source.[1] (Hicks Aff. ¶ 19.) The more common static web pages merely consisted of preset HTML files served to requesting users. Dynamic web pages consisted of an HTML file built in response to a request. The prior art taught that a dynamic web page could be built by a computer program, such as are typically located within a CGI-BIN, which could be started by a web server that has received a request for specific request for content. Data from such a request was fed into the computer program which would then process a series of instructions that would result in lines of HTML being written until an entire HTML document was created to be delivered to the requesting browser. The result was a web page that could deliver different content in the same format. (Hicks Aff. ¶ 19.)

---

[1] Dow mischaracterizes the invention by attempting to paint the patent claims as broadly extending to any dynamic web page. Dow Br. at 9. The claim language itself clearly limits the invention to dynamically altering the format of a web page in dependence upon format preferences. Ablaise, in fact, explicitly stated this to the PTO:

> Applicant has described and claimed an arrangement whereby two different clients requesting the same content data from the same server may receive differently formatted versions of that same content data depending upon a particular format identifier received from each respective client at the server.

'737 File History Amendment dated December 16, 2003, p. 13. Thus, different formatting based on either a format identifier or format identification type data is the key. Dynamically creating web pages with the same formatting for all users is not claimed.

The prior art, however, still suffered from the lack of functionality that allowed a user of the system to control the layout or formatting of the content—dynamically generated or not—on the page. For example, while Dow relies on the prior art Fishwrap reference to show that dynamic content generation was known in 1995, Fishwrap lacked customized formatting—functionality that users explicitly were requesting. See, e.g., Rosenbloom Decl. Ex. N, at 25-26.

Customizing the format of web pages presented a substantial burden on the web developers prior to the invention. If, for example, one user needed data organized one way and another user needed a different format for the same data, the web developer would have to create two separate pages, each requiring technical skill to create. As the number of requests for specific layouts multiplied, the number of pages required would grow exponentially until the effort to facilitate the users' requests became prohibitive. Rosenbloom Decl. Exh. A at Col. 4, ll. 57-66. The invention disclosed by the '530 and '737 patents overcame these and other problems.

C. The History of Ablaise and The Invention of Custom Formatted Web Pages

Inventors Andrew Ritchie and Jon Bradshaw are both individuals with extensive experience in computer network applications. (Ritchie Aff. ¶¶ 1, 5.) They developed a flexible method for customizing, formatting, and delivering web pages in order to grow a web development business. That commercially successful invention is the basis of the '530 and '737 patents.

During the spring of 1994, Mr. Ritchie was approached by someone interested in creating an online version of a large U.K.-based retail catalog called Freemans. (Ritchie Aff. ¶ 2.) Mr. Ritchie realized that creating a web page for each product in the catalog was unmanageable, because it would require the creation of nearly 10,000 static web pages. (Id. ¶ 3.) After considering several solutions, Mr. Ritchie decided that he could use a web server to create each of the catalog pages "on the fly" each time a user requested the page. (Id. ¶ 4.) More importantly,

this decision led Mr. Ritchie to the realization that he could also "utilize identification information sent from a browsing device to customize the web page created by the server to fit the needs and preferences of the users requesting the information." (Id.)

Mr. Ritchie called upon his friend, Jon Bradshaw, to help develop this new technology as part of a new business that they founded called Point4 Consulting Ltd. (Id. ¶¶ 5-6.) During the Winter of 1994-1995, Mr. Ritchie and Mr. Bradshaw built an internet application "that allowed a server to create web pages on-the-fly in response to identifiers that identified the content and/or format preferred by the user." (Id. ¶ 6.) Mr. Ritchie and Mr. Bradshaw built and sold many web-based applications to many large organizations in the United Kingdom that utilized some form of this "on-the-fly" web page creation technology. (Id. ¶¶ 7-8.)

Ritchie and Bradshaw always retained ownership of their patents, but along with the other shareholders, sold Point4 to Nettec PLC. (Id. ¶ 9.) They licensed their technology to Nettec for 3 years. (Id.) Ritchie, Bradshaw and most of the other Point4 shareholders also formed Ablaise in 2002 to license the patents to other companies. (Id. ¶ 10) Since then, several large companies have agreed to take a license on the patents, including Dell, Citigroup, and E*Trade Financial. (Id. ¶ 12.)

In addition to commercializing their technology, Mr. Ritchie and Mr. Bradshaw also decided to protect their intellectual property and apply for a patent. The inventors disclosed their method in detail in a patent application in the U.K. on May 15, 1995 including the required "best mode" of practicing the various aspects of method. The inventors filed in the U.S. on May 15, 1996 and were eventually granted two U.S. patents related to dynamically generating web pages in a preferred format, the '737 patent and the '530 patent.

The method the inventors disclosed to the patent office could generate web pages "on-the-fly" taking into account *both* content preferences and format preferences. In other words, they were not limited to serving static web pages in response to a request, but could deliver different content, and could format that content differently, depending upon information in the request and/or stored information about a user making the request relating to "a specified page format." Rosenbloom Decl. Exh. A at Col. 5, ll. 7-20. The system accomplishes this task by feeding format, user, and/or content preferences into a computer program. This computer program then executes a series of instructions that select between various content and formatting tags and places the selected data within a particular portion of a markup file to be delivered to a user's browsing device. Id. at Col. 8, ll. 34-44, Col 14, ll. 41-46. In effect, the system is capable of executing different instructions to select and write different formatting tags to the HTML document in response to different formatting preferences of a user. Id. at Col. 15, ll. 31-34, 54-56, 63-67 - Col. 16, ll. 1-7. The result is a more flexible web page capable of delivering either the same or different content paired with potentially different specified formats for the location of presentation the content. Id. at Col. 7, ll. 23-37.

The Summary of the Invention describes a method for serving viewable data "in accordance with a specified page format" wherein "HTML output instructions are generated 'on-the-fly' in response to requests." Id. at Col. 5, l. 12, 22-24 (emphasis added). As the Summary further explains, "any type of data" may be served, Id. at Col. 5 l. 32. The invention, however, allows the system to intelligently select specific formatting markup instructions for a markup document to control the look of a web page in dependence upon a user's preference for the overall layout of the page. Id. at Col. 5 l. 38-42; Id. at Col. 8, ll. 34-44.

The specification describes two ways to identify format preferences: (1) from data in the request for a page; and (2) from a user database. For instance, in describing one embodiment the specification states, "[t]he URL will include an element identifying the data required [and] an element identifying the type of formatting required." Id. at Col. 14 ll. 3-5. In that case, the format preference comes with the request. Later, the specification describes how all data is processed in the preferred embodiment to generate pages on the fly including formatting tags, text data and graphics data where the formatting preferences are stored on the system. See, e.g., Id. at Col. 15, ll. 6-34. In essence, the system is able to "adjust[] the way in which the data is actually formatted" by "adjust[ing] the relationship between indexes and strings" that reference computer instructions, that when executed, select and write formatting tags within the delivered markup file, Id. at Col. 15, ll. 31-34, on the basis of "information read from the user database." Id. at Col. 16, ll. 8-14. The claimed inventions, therefore, include the ability to generate web pages on the fly in accordance with a user preferred format derived from the request or received from a component of the system such as a user database.

The patents-in-suit claim the invention in different ways. In general, the '530 patent claims receiving a request for a web page from a browser including data identifying the preferred format, such that the requested content may be provided in at least two different formats to that browser depending upon identification data in the request. The '737 patent claims storing user preferences in a database such that, in response to a request for a web page, preferred content may be returned and a preferred format applied to that content depending upon the request and the user making the request.

II. LEGAL STANDARDS OF PATENT CLAIM CONSTRUCTION

Claim construction is the process by which this Court discerns the meaning of a party's claims, and thus what it owns. In its opening brief Dow provided the Court with some of the

basic canons of claim construction and Ablaise will not burden the Court with repetition. However in its discussion of the law, and correspondingly throughout the brief, Dow downplayed the most basic canon: that the claim language defines the invention and the claim terms are to be accorded their ordinary and customary meaning to those of skill in the art. Dow ignored altogether the canon prohibiting courts from reading limitations from the specification into the claims. Finally, Dow overstated the role of the file history. The law on these three points is important to the Court's determination of claim construction in this case.

A. Focus on the Claim Language and Accord Terms Their Plain and Ordinary Meaning to One of Ordinary Skill in the Art.

The words of a claim "are generally given their ordinary and customary meaning." Phillips v. AWH Corp., 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (citing Vitronics Corp. v. Conceptronic, Inc., 90 F.3d 1576, 1582 (Fed. Cir. 1996)). "The ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application." Phillips, 415 F.3d at 1312. "In some cases, the ordinary meaning of claim language as understood by a person of skill in the art may be readily apparent even to lay judges, and claim construction in such cases involves little more than the application of the widely accepted meaning of commonly understood words." Id. at 1314. Even in more complicated cases, the inquiry is an objective one, in which the court looks to those sources available to the public that show what a person of skill in the art would have understood disputed claim language to mean. Id. "Those sources include the words of the claims themselves, the remainder of the specification, the prosecution history, and extrinsic evidence concerning relevant scientific principles, the meaning of technical terms, and the state of the art." Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc., 381 F.3d 1111, 1116 (Fed. Cir. 2004)). Extrinsic evidence can

include expert testimony on the background of the technology at issue, how a claimed invention works and the knowledge and understanding of persons skilled in the art. Id. at 1318; Pfizer, Inc. v. Teva Pharms. USA, Inc., 429 F.3d 1364, 1374 (Fed. Cir. 2005) (expert testimony supported conclusion on understanding of persons in the art).

Throughout the process of determining the ordinary and customary meaning, the focus must remain on the claims. According to the Federal Circuit, "[q]uite apart from the written description and the prosecution history, the claims themselves provide substantial guidance as to the meaning of particular claim terms." 415 F.3d at 1314. Indeed, proper claim construction requires interpretation of the entire claim in context, not a single element in isolation. ACTV, Inc. v. Walt Disney Co., 346 F.3d 1082, 1090 (Fed. Cir. 2003). "For example, the presence of a dependent claim that adds a particular limitation gives rise to a presumption that the limitation in question is not present in the independent claim." Phillips, 415 F.3d at 1314-15. Similarly, general descriptive terms will ordinarily be given their full meaning, and modifiers will not be added to broad terms standing alone. Johnson Worldwide Assoc., Inc. v. Zebco, Corp., 175 F.3d 985, 989 (Fed. Cir. 1999).

B. It is Improper to Read Limitations From the Specification Into the Claims.

While the person of ordinary skill in the art is deemed to read the claim term in the context of the claims as well as the specification, Ferguson Beauregard/Logic v. Mega Systems, 350 F.3d 1327, 1338 (Fed. Cir. 2003), in examining the specification, the Court must not at any time import limitations from the specification into the claims. CollegeNet, Inc. v. ApplyYourself, Inc., 418 F.3d 1225, 1231 (Fed. Cir. 2005). As the Federal Circuit has explained,

> [C]laims are infringed, not specifications. . . . If everything in the specification were required to be read into the claims, or if structural claims were to be limited to devices operated precisely

> as a specification-described embodiment is operated, there would be no need for claims . . . It is the claims that measure the invention.

SRI Int'l, Inc. v. Matsushita Elec. Corp., 775 F.2d 1107, 1121 (Fed. Cir. 1985) (en banc).

The foundation of this rule is the basic understanding that the law of patenting is intended to be reasonable. In other words, "[t]he law does not require the impossible. Hence, it does not require that an applicant describe in his specification every conceivable and possible future embodiment of his invention." Id. Thus, "[e]ven when the specification describes only a single embodiment, the claims of the patent will not be read restrictively unless the patentee has demonstrated a clear intention to limit the claim scope using words or expressions of manifest exclusion or restriction." Liebel-Flarsheim Co. v. Medrad, Inc., 358 F.3d 898, 906 (Fed. Cir. 2004).

C. The Prosecution History Does Not Carry the Same Weight as the Specification.

As with the specification, the prosecution history limits the ordinary meaning of a claim term only where the patentee excludes certain subject matter "with reasonable clarity and deliberateness." N. Telecom Ltd. v. Samsung Elec. Co., 215 F.3d 1281, 1294-95 (Fed. Cir. 2000). In other words, the applicant must use "words or expressions of manifest exclusion or restriction during administrative proceedings before the Patent and Trademark Office" for the prosecution history to demonstrate an applicant's intention to deviate from the ordinary and accustomed meaning of the claim terms. Teleflex, Inc. v. Ficosa N. Am. Corp., 299 F.3d 1313, 1326 (Fed. Cir. 2002); see also Rambus, Inc. v. Infineon Techs. AG, 318 F.3d 1081, 1095 (Fed. Cir. 2003). In addition, "because the prosecution history represents an ongoing negotiation between the PTO and the applicant, rather than the final product of that negotiations, it often

lacks the clarity of the specification and thus is far less useful for claim construction purposes." Phillips, 415 F.3d at 1317.

III. CONSTRUCTION OF CLAIM TERMS IN CLAIM 1 OF THE '737 PATENT

In claim 1 of the '737 patent, a user operating a browsing device has requested a web page. In practical terms, the method of serving the requested web page back to the user encompasses (1) identifying the user; (2) looking up stored information about the user's preferences for content and/or formatting; (3) receiving format identifiers from the user's request, or from storage, identifying the user's preferred type of formatting; and (4) generating a customized web page in the form of viewable data. Ablaise's proposed constructions provide a logical and correct reading of the claim to accomplish these objectives.[2]

A. "Displayed at a browsing device"

Dow does not seek construction of the term "browsing device," but only of the word "displayed." This term does not require construction. "Displayed at a browsing device" is commonly understood language and has a plain meaning even to lay persons. It means displayed at a browsing device.

More importantly, "displayed at a browsing device" is part of a larger phrase that describes "formatting data" served by the server, not a step of displaying the data. In other words, it is not necessary for Dow to "display at a browsing device" in order to infringe the claim. The steps of the claim take place entirely on the server side. The preamble of claim only places limitations on the type of viewable data that is served to a browser.

The server-centric nature of the claim is apparent from the language of the claim as a whole. Claim 1 recites

[2] For the Court's convenience, a chart of the disputed terms and the claim constructions proposed by Dow and Ablaise is provided as Exhibit A to the declaration of Trevor Foster.

> A method of serving pages of viewable data to browsing devices connected to a network, wherein a page of said viewable data comprises content data defining text and/or graphics and formatting data which specifies locations of said text and/or graphics within a page, and said viewable data is displayed at a browsing device such that locations of said text and/or graphics depend on said formatting data, said method comprising:

Rosenbloom Decl. Exh. A at Col. 19, l. 65 – Col. 20, l. 5 (emphasis added). The language before the word "wherein" sets forth the scope of the claim. It is a method of serving pages of viewable data to browsing devices. It is not a method of displaying viewable data at a browsing device. The language after the word "wherein" describes the viewable data that is served, explaining that the viewable data is made up of content data and of formatting data that specifies locations for the content data. The word "location" describes how the content data will look when displayed at a browsing device. It is simply a description of the type of viewable data served by the method of this claim.

The remaining claim language further supports this understanding of the claim. All of the recited steps in the "method for serving pages of viewable data" are performed on the server side, not the browser side. The first step is "identifying requests from browsing devices" and the last step is to "generate viewable data" which is served to a browsing device. There is no step relating to how a browser renders the viewable data or what hardware is used to display the data. That is not part of a "method for serving pages."

The specification confirms that the browser itself, or the method of display, is not important, and not a part of the invention of a method for serving pages. The entire specification describes serving signals to browsing devices. For instance, the Title is "Serving Signals." The Abstract begins "[o]utput signals are served from a service device to a plurality of browsing devices." The Field of the Invention states, "[t]he present invention relates to serving signals to

browsing clients." Rosenbloom Decl. Exh. A at Col. 1, ll. 12-13. It is readily apparent to one of skill in the art reading the specification that the invention relates entirely to what is done on the server side to deliver a customized web page. (Hicks Aff. ¶ 21.)

Furthermore, the phrase "displayed at a browsing device" in the preamble does not limit the scope of the invention. Rather, the fact that the served pages may be displayed at a browsing device merely states a purpose or intended use for the invention, which means that the preamble phrase is not a limitation of the claim. Catalina Mktg. Int'l v. Coolsavings.com, Inc., 289 F.3d 801, 808 (Fed. Cir. 2002) . Finally, if the claim language is to be construed regardless of the fact that it is not limiting, Dow's construction is incorrect. Dow seeks to add limitations to the claim with its construction "visually represented *on the screen of* a browsing device." The language "visually represented" is nothing more than the common meaning of "displayed." Dow then adds "on the screen" of a browsing device. A browsing device is usually associated with something that permits display, such as a computer monitor, projector or other form of display. The claim language is "displayed at a browsing device," not on the screen of a browsing device. The specification confirms that "a browser is an application capable of interpreting and displaying documents received in HTML in such a way that the information is displayed to the user in a form compatible with the user's available equipment." Rosenbloom Decl. Exh. A at Col. 3, ll. 20-24. Dow's attempt to limit the claim to one type of "available equipment" should be rejected.[3]

[3] Also worth noting is that viewable data may be displayed. Obviously that does not mean that the formatting data can itself can be seen. As the claim itself states, the formatting data "specifies locations of said text and/graphics data." What a user sees is the content data in a specified location.

B. "Storing executable functions"

There is no dispute that "**storing executable functions**" means "**storing at least two executable functions.**" The term **"function,"** however, has a plain and ordinary meaning to someone of ordinary skill in the art at the time of the invention which is **"an identifiable unit of computer instructions."** The specification supports the ordinary meaning. No other limitations should be read into this phrase because no other limitations are part of the ordinary meaning, and there is no clear and unmistakable statement elsewhere in the intrinsic record that would limit the term.

The specification clearly and repeatedly explains any identifiable unit of instructions constitutes a function. For instance, the specification states "a format function of this type may be considered as the smallest unit of instructions for producing a portion of HTML code." Id. at Col. 12, ll. 46-48 (emphasis added). These functions must be identified by the computer program. See, e.g., Id. at Col. 13 l. 42-44 (identifying functions by "index reference"). Functions also have broad applications within computer programs. For example, the specification discloses functions which write HTML, see, e.g., Id. at Col. 12 l. 46-48 or read from the database or content store, see, e.g., Id. at Col. 15, ll. 54-58. Moreover, a person of ordinary skill in the art would understand that a computer program can typically identify functions by such things as a name, memory address, index or other mechanism depending upon the circumstances. (Hicks Aff. ¶ 22.)

Dow attempts to add limitations to an otherwise simple definition without any support from the claim language itself. In fact, Dow defines the claim language *with* the claim language only to then add additional unsupported limitations. Dow defines "storing executable functions" as "storing a universal set of all available functions, where each function has a name and consists of a set of function steps (i.e., instructions), each of which, when executed, creates a portion of

code (e.g., HTML code)." (Dow Br. 19). In doing so, Dow is asking this court to write in three limitations that: (1) storing requires the storing of a universal set of all available functions, (2) functions need to be named, and (3) functions must include multiple function steps, and write HTML code. These limitations are not supported by either the intrinsic or extrinsic evidence and should be rejected.

1. Storing does not mean storing a universal set of all available functions.

Dow improperly imports from the description of the preferred embodiment that "the system as a whole includes a universal family set of all the available functions." Rosenbloom Decl. Exh. A at Col. 12, ll. 49-52. It is perfectly reasonable to interpret this statement to mean simply that the system may store a set of functions that is larger than the set the system uses in a given context. The statement does not say the system must store every known function. In any event, this description of the preferred embodiment should not be read into the claim. Teleflex, Inc., 299 F.3d at 1326. Nor did the inventors indicate that this was an "essential feature" as Dow asserts. (Dow Br. 20). There is simply no support for Dow's contention.

Moreover, there is intrinsic evidence that a system would not necessarily include every possible function. The specification explains: "[a]s the system develops, new functions may be added to the family set and it is expected that the HTML standard will be enhanced, thereby requiring additional functions to be created." Rosenbloom Decl. Exh. A at Col. 12, ll. 52-54. The inventors therefore acknowledged that the "universal family set of all available functions" did not, in fact, contain every possible function to generate viewable data. The Court should not import a limitation from the specification, particularly where the specification makes it clear that the "family set" of functions stored in the system can receive additions over time. Id.

2. Functions need not be named, merely identified.

One way a function can be identified is a name, but the claim language does not limit itself to identification of functions only by name. The specification discusses functions being identifiable, with no mention of named functions. Nevertheless, Dow asserts that "the specification implies that each particular function within the universal set is associated with a unique identifier [e.g., each function has a name] because a list, by definition, is a series of names." (Dow Br. 22.) Dow provides no justification for why an identifiable function need be identified by only a name. The specification teaches differently. For example, in one embodiment functions are referenced not by name, but, instead "a function string *index* is identified . . . and at step 1204 the indexed function string is read from the string list store 1103." An index could be, for example, an internal memory address by the computer. (Hicks Aff. ¶ 22.) What matters is not the precise mode of identification, [e.g., a name] but that the function can be identified by the system. In fact, a portion of the specification relied upon by Dow describes a formatting function as "the smallest unit of instructions for producing a portion of HTML code." Rosenbloom Decl. Exh. A at Col. 12, ll. 46-48. See also (Dow Br. 20). This definition lacks any reference to the formatting functions having a name.

Given the lack of support for its definition in the intrinsic record, Dow improperly relies on a series of extrinsic dictionary definitions to support its limitation. See, e.g., Phillips, 415 F.3d at 1322, Ferguson Beauregard/Logic v. Mega Systems, 350 F.3d 1327, 1338 (Fed. Cir. 2003). However, Dow's own cited dictionaries from the time of the invention do not require functions to be named, only that they be callable by the program.[4] Dow then invites legal error

[4] For example, in the only two definitions that reflect the meaning of "function" at the time of the invention, Dow fails to demonstrate why functions necessitate a "name." Microsoft's dictionary defines a function merely as "the purpose of or the action carried out by a program or routine, a general term for a subroutine, in some languages a subroutine that returns a single value." Computer Dictionary, Microsoft Press, 179 (2d ed. 1994). Because the

by relying on several recent internet dictionaries for the proposition that "a list, by definition, is a series of names." (Dow Br. 22, n.10). Dow provides no analysis explaining why there could not be a list of anything beyond names, particularly in light of the specification's disclosure of lists of functions, Rosenbloom Decl. Exh. A at Col. 13, l. 44), strings, Id. at l. 48, ASCII text, Id. at Col. 3. l. 65, or indexes, Rosenbloom Decl. Exh. A at Col. 13, l. 20. Dow's artificial limitation to named functions should be rejected.

3. A function need not when executed, create a portion of code.

Dow asserts that each function consists of "one or more function steps (i.e. instructions), which, when executed, create a portion of a web page." (Dow Br. 20). It is true that, "formatting functions . . . generate small portions of HTML code," Rosenbloom Decl. Exh. A at Col. 13, l. 3 (emphasis added), and that function steps "may be considered as the smallest portion of a function that results in a write to the HTML buffer." Id. at Col. 18, ll. 49-51 (emphasis added); see also (Dow Br. 21). However, Dow's assertion that *all* functions must create a portion of HTML code is incorrect. For instance, some functions may simply retrieve data. The specification teaches that, "[e]xecution of a function read from the string list may result in HTML tags being written directly to the output HTML buffer 1102. Alternatively, execution of these functions may result in a call being made to the text database 1104 or to the graphics database. Rosenbloom Decl. Ex. A at Col. 15, ll. 54-58 (emphasis added). Thus, Dow's definition is directly contrary to the specification and is too limiting.

---

definition of function does not include a name, Dow instead relies on "procedure," a term only found within the definition's "see also" section. Br. at 22 n.11 This is a blatant mischaracterization of the definition, particularly in light of the definition failing to even include the word "procedure." Dow's citation to the IBM Dictionary of Computer is also suspect. Instead of relying on the dictionary's broad definition of "function" as a "subroutine that returns the value of a single variable and that usually has a single exit" they rely upon a subsequent definition specific to the C and Fortran programming languages—a limitation that is not supported by the claim language or the specification. IBM Dictionary of Computing, 1994.