IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION

PARALLEL NETWORKS, LLC	§	
	§	
Plaintiff	§	
	§	
vs.	§	**CASE NO. 6:10cv111 (Lead Case)**
	§	**PATENT CASE**
ABERCROMBIE & FITCH, ET AL.	§	
	§	
Defendants	§	

MEMORANDUM OPINION AND ORDER

This case comes before the Court for claim construction of three key terms and consideration of Defendants' motion for summary judgment of noninfringement (Docket No. 481). Having considered the parties' written submissions and oral arguments, the Court construes the three terms as stated below and **GRANTS,** in part, Defendants' motion for summary judgment.

BACKGROUND

Procedural History

Plaintiff Parallel Networks, LLC ("Parallel") originally brought this case as four separate actions, naming over 100 defendants, and alleging infringement of United States Patent No. 6,446,111 ("the '111 Patent") in all four cases. On March 1, 2011, the Court held a status conference in all four cases to discuss the most efficient manner in which to handle this case with this many defendants. Defendants suggested that three claim terms would be case dispositive for nearly all Defendants and requested an early claim construction and summary judgment hearing on these terms. Parallel acknowledged that certain terms would be important to its infringement case. Believing that

construing these three limited terms early in the case could resolve several important issues, the Court set this case for an early *Markman* and summary judgment hearing.

Technology

The '111 Patent, entitled "Method and Apparatus for Client-Server Communications Using a Limited Capability Client Over a Low Speed Communications Network," issued on September 3, 2002. The claimed invention generally relates to a method and apparatus for client-server communication. Upon a client request, the server dynamically generates an applet having selected characteristics enabled as a response to the client request and sends the applet to the client. '111 patent at 1:7–13. The applet is substantially self-sufficient and does not depend on services being available on the client for the applet to use. '111 patent at 2:59–61. Exemplary claim 1 requires:

1. A data processing system comprising a server coupled to a communications link and operable to receive a request from a client device and to collect a plurality of data items, wherein the data items comprise specific information collected as a function of the request; an executable applet dynamically generated by the server in response to the request, a constituent system associated with the applet comprising a subset of the data items, each data item in the subset used as at lease one pre-loaded value in the applet; a further constituent system associated with the applet comprising a data interface capability configured to provide a plurality of operations on the pre-loaded values, the operations comprising operations associated with the subset of the data items; and the applet operable to be transferred over the communications link to the client device.

CLAIM CONSTRUCTION

Applicable Law

"It is a 'bedrock principle' of patent law that 'the claims of a patent define the invention to which the patentee is entitled the right to exclude.'" *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (quoting *Innova/Pure Water Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115 (Fed. Cir. 2004)). In claim construction, courts examine the patent's intrinsic evidence to define the patented invention's scope. *See id.*; *C.R. Bard, Inc. v. U.S. Surgical Corp.*,

388 F.3d 858, 861 (Fed. Cir. 2004); *Bell Atl. Network Servs., Inc. v. Covad Commc'ns Group, Inc.*, 262 F.3d 1258, 1267 (Fed. Cir. 2001). This intrinsic evidence includes the claims themselves, the specification, and the prosecution history. *See Phillips*, 415 F.3d at 1314; *C.R. Bard, Inc.*, 388 F.3d at 861. Courts give claim terms their ordinary and accustomed meaning as understood by one of ordinary skill in the art at the time of the invention in the context of the entire patent. *Phillips*, 415 F.3d at 1312-13; *Alloc, Inc. v. Int'l Trade Comm'n*, 342 F.3d 1361, 1368 (Fed. Cir. 2003).

The claims themselves provide substantial guidance in determining the meaning of particular claim terms. *Phillips,* 415 F.3d at 1314. First, a term's context in the asserted claim can be very instructive. *Id.* Other asserted or unasserted claims can also aid in determining the claim's meaning because claim terms are typically used consistently throughout the patent. *Id.* Differences among the claim terms can also assist in understanding a term's meaning. *Id.* For example, when a dependent claim adds a limitation to an independent claim, it is presumed that the independent claim does not include the limitation. *Id.* at 1314-15.

"[C]laims 'must be read in view of the specification, of which they are a part.'" *Id.* (quoting *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 979 (Fed. Cir. 1995) (en banc)). "[T]he specification 'is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.'" *Id.* (quoting *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996)); *Teleflex, Inc. v. Ficosa N. Am. Corp.*, 299 F.3d 1313, 1325 (Fed. Cir. 2002). This is true because a patentee may define his own terms, give a claim term a different meaning than the term would otherwise possess, or disclaim or disavow the claim scope. *Phillips*, 415 F.3d at 1316. In these situations, the inventor's lexicography governs. *Id.* Also, the specification may resolve ambiguous claim terms "where the ordinary and accustomed

meaning of the words used in the claims lack sufficient clarity to permit the scope of the claim to be ascertained from the words alone." *Teleflex, Inc.*, 299 F.3d at 1325. But, "'[a]lthough the specification may aid the court in interpreting the meaning of disputed claim language, particular embodiments and examples appearing in the specification will not generally be read into the claims.'" *Comark Commc'ns, Inc. v. Harris Corp.,* 156 F.3d 1182, 1187 (Fed. Cir. 1998) (quoting *Constant v. Advanced Micro-Devices, Inc.*, 848 F.2d 1560, 1571 (Fed. Cir. 1988)); *see also Phillips*, 415 F.3d at 1323. The prosecution history is another tool to supply the proper context for claim construction because a patent applicant may also define a term in prosecuting the patent. *Home Diagnostics, Inc. v. Lifescan, Inc.*, 381 F.3d 1352, 1356 (Fed. Cir. 2004) ("As in the case of the specification, a patent applicant may define a term in prosecuting a patent.").

Although extrinsic evidence can be useful, it is "'less significant than the intrinsic record in determining the legally operative meaning of claim language.'" *Phillips*, 415 F.3d at 1317 (quoting *C.R. Bard, Inc.,* 388 F.3d at 862). Technical dictionaries and treatises may help a court understand the underlying technology and the manner in which one skilled in the art might use claim terms, but technical dictionaries and treatises may provide definitions that are too broad or may not be indicative of how the term is used in the patent. *Id.* at 1318. Similarly, expert testimony may aid a court in understanding the underlying technology and determining the particular meaning of a term in the pertinent field, but an expert's conclusory, unsupported assertions as to a term's definition is entirely unhelpful to a court. *Id.* Generally, extrinsic evidence is "less reliable than the patent and its prosecution history in determining how to read claim terms." *Id.*

"Executable applet"

Parallel's Proposed Construction	Defendants' Proposed Construction
"program code that interacts locally with a client and that can be discarded"	"a small, substantially self-sufficient application, containing the requisite functionality and the necessary data to operate, that depends on substantially no services (e.g., web browser or plug-in) on a client device."

Parallel argues the specification discloses that the applet contains computer program code that is executed on the client device. Parallel further argues the specification discloses that while the applet is executed, it interacts locally on the client device without the need to communicate with the server and the applet can be discarded after it is used (although it may also remain on the client device). Parallel contends Defendants' proposed construction improperly reads limitations into the claims, e.g., "small" and "substantially self sufficient," and Defendants' contention that the applet cannot use a "web browser or a plug-in" is explicitly contradicted by the clear disclosures in the specification.

Defendants argue the patentee expressly defined "executable applet" in the specification:

> The present invention provides a number of technical advantages. One such technical advantage is the capability to respond to client requests for information from a server using a dynamically generated, selected characteristic enabled, transient applet. The applet that is sent to the client is substantially self-sufficient and depends on substantially no services being available on the client for the applet to use. . . . [T]he use of such an applet allows the human operator of a handheld device which has limited memory and limited or no secondary storage capabilities to initiate and complete a transaction with a server without having to load and store a variety of support programs which may be required by the server.

'111 Patent at 2:55–3:5 (emphasis by Defendants). Defendants argue that the patentee emphasized the present invention's self-sufficiency as an advantage over the prior art. They argue that the

claimed "executable applet" can be executed without the prior art's paraphernalia, such as a web browser or plug-in. Defendants further contend that the executable applet must contain both the requisite functionality and necessary data to operate. *See id.* at 2:64–67, 17:14–16, Fig. 2. Finally, Defendants contend Parallel's proposed construction is supported by the specification's description of the prior art rather than the specification's description of the invention.

Defendants' proposed construction is based on an underlying and erroneous premise that claims 1 and 17 are limited to web browsing. *Cf. id.* at 8:30–34 ("the network 14 may comprise the Internet or another suitable type of Local Area Network (LAN), Medium Area Network (MAN), Wide Area Network (WAN), intranet or any other suitable computer network well-known in the art"). Defendants' contention that "executable" means the applet can be run on the CPU without additional resources is misplaced. Further, the terms "small," "substantially self-sufficient," "substantially no services," "requisite functionality," and "necessary data" in Defendants' proposed construction are vague and possibly indefinite.

Parallel's construction that the applet can be discarded imports a limitation from the preferred embodiment that is not required by the claims. Moreover, it describes what can be done to the applet but does not describe what the applet is. Additionally, Parallel's limitation "that interacts locally with a client" is based on a reference in the specification to a capability to "interact with the user locally," *see id.* at 5:66–6:1, but the user and client are not necessarily synonymous.

The parties do not dispute that an "applet" is a program. The Court construes "executable applet" as "program code that can be used by a client device."

"[Applet] dynamically generated by the server in response to the request" (Claim 1) / "generating [an executable applet] dynamically in response to the data request" (Claim 17)

Parallel's Proposed Construction	Defendants' Proposed Construction
"Generated by the server to have particular services and data for the client based on the request"/ "Generating [an executable applet] to have particular services and data for the client based on the data request"	"Constructed at the server, by combining the requisite functionality with the necessary data, at the time of and in response to the client request" / "Constructing at the server, by combining the requisite functionality with the necessary data, at the time of and in response to the client request"

This term raises three issues: (1) what it means that the applet is "dynamically generated," (2) when is the applet generated, and (3) whether the applet is necessarily sent in a single transmission.

Parallel argues its construction is supported by the specification while Defendants improperly attempt to limit the claim language by reading into the claims other terms, such as "constructing" or "created," that are inconsistent with the specification's use of the term "generated." Parallel contends the applet does not have to be generated at or after the client request so long as the applet contains what is requested.

Defendants argue that because the claim language includes "dynamically generated," the applet must be generated at the time of and in response to the client request. Defendants contend the applet cannot exist before the server receives the request. In addition to the claim language itself, Defendants rely on the prosecution history and the patentee's statements distinguishing the Johnson prior art to support their position. Defendants contend that the applet is constructed at the time of the request by combining the necessary functionality and data. Finally, Defendants contend that the applet must be transferred to the client in one package containing both functionality and data.

Parallel's construction and arguments read out the limitation "in response to the data request." The claim language expressly requires that the applet is generated in response to the request. For the applet to be generated in response to the request, the request must come first and the applet cannot predate the request. This is consistent with other portions of the claim language. The applet includes data items comprised of "specific information collected as a function of the request." '111 patent at 17:51–52 (claim 1); *see also id.* at 18:59–60 ("collecting on the server a plurality of data items in response to the data request") (claim 17).

This interpretation is also consistent with the prosecution history. The examiner rejected the claims as obvious in light in U.S. Patent No. 5,923,885 ("Johnson"). To overcome the rejection, the patentee argued that:

> Johnson does not teach or suggest an executable applet "dynamically generated by the server in response to the request" as Johnson teaches that software applications from the server are identified and selected, not "generated by the server in response to the request." Johnson, Abstract. Also, Johnson does not teach generating "in response to the request" as Johnson only teaches selection and identification of existing software applications.
>
> In addition, Johnson teaches away from an applet "dynamically generated by the server in response to the request" because Johnson teaches selection and identification of already existing software applications. One skilled in the art would be motivated by Johnson to create software application and then make those applications available to a client, not applets dynamically generated by the server "in response to the request."

April 23, 2001 Response to Office Action at 8 (Docket No. 481–39). In distinguishing Johnson, the patentee disclaimed pre-existing applets that were merely selected, and not generated, based on the client request. Parallel's arguments on this issue directly contradict how the patentee distinguished Johnson to the examiner. In order for the applet to be "dynamically generated," it must be generated after the client request is made.

Parallel argues that the applet may be generated by using templates or existing code created before the client request is made and therefore an applet can exist before the request is received from the client. *See id.* at 12:15–20. The specification describes various ways to generate the applet. "The applet 26 may be generated by combining various predefined units together based on the data in the data storage system 32. The applet may also be generated by using templates which are customized based on the data to be included in the data storage system 32." *Id.* at 12:11–16. The Court's construction does not exclude combining pre-existing pieces of code to build the applet in response to the client request. However, the specification and prosecution history are clear that the claimed invention does not encompass applets that exist as completed applets before the client request is made.

Also supporting Defendants' construction, the specification describes that the applet's construction has data manipulation functionality combined with the data:

> FIG. 2 is a block diagram showing further details of the system of FIG. 1. As discussed previously, because of the unique construction of the applet, the applet 26 comprises both a data manipulation system 30 and a data storage system 32 which are each constituent systems associated with the applet 26. The data manipulation system 30 may also comprise a portion of the applet 26, similarly, the data storage system 32 may also comprise a portion of the applet 26. The data manipulation system 30 includes all capabilities required for the operator of the client 12 to utilize data in the data storage system 32. The data manipulation system 30 enables the applet 26 to handle the various characteristics associated with the data included in the applet 26 items 28 retrieved in response to the request from the client 12 and to handle client specific characteristics. In particular, the data manipulation system 30 of the applet 26 will provide the client 12 with whatever suitable data interface is required to access and utilize the data in the data storage system 32.

'111 patent at 11:7–29.

> The data manipulation system 30 may include specific functionality required by the data in the data storage system 32. By combining the functionality in the data manipulation system 30 with the data of the data storage system 32 the design

> methodology behind the applet 26 can be shifted from the traditional focus on writing a generic program one time and using that program with a variety of different data types, to writing a program specifically for particular data.

Id. at 11:54–61. Because the applet combines the functionality with data, the constituent operations and data systems are transferred to the client in a single transmission rather than separate transmissions of each part. The Summary of the Invention expressly states that the functionality and data are sent in a single transmission:

> A further advantage is that the total amount of data transferred between the client and the server over a communications link is decreased. By transmitting the appropriate data and associated data handling capabilities as a group, the client may be required to communicate over a low-speed communications link a greatly reduced number of times or, in some cases, only once. Yet another advantage is found in the avoidance of transmitting duplicate data by using updateable elements and a loader within the applet to load only new data desired by the client from the server, instead of completely retransmitting all the information in order to include small changes.

Id. at 3:11–22. The Detailed Description of the Invention furthers this understanding:

> The total number of accesses made over the communications link 16 from the client 12 is also decreased by including the data requested by the client 12 in the applet 26 as part of a single transmission. Stated another way, instead of downloading generic program functionality to the client 12 and then having the program use the communication link 16 in order to access required data, the needed functionality and the required data are bundled together in the applet 26 and may be transferred to the client 12 as part of the single transmission.

Id. at 12:2–11. The specification is clear that one advantage of the claimed invention is that the data and functionality are sent together to the client to reduce the number of transmissions. This is especially useful when communicating over low-speed communications links. This does not however exclude additional communications between the server and client once the applet is in place. *See id.* at 14:57–62.

Accordingly, the Court adopts Defendants' proposed construction and construes "[applet]

dynamically generated by the server in response to the request" (claim 1) as "constructed at the server, by combining the requisite functionality with the necessary data, at the time of and response to the client request" and "generating [an executable applet] dynamically in response to the data request" (claim 17) as "constructing at the server, by combining the requisite functionality with the necessary data, at the time of and in response to the client request."

"Data interface capability"

Parallel's Proposed Construction	Defendants' Proposed Construction
"program code that allows the client to access and use data"	"data manipulation functionality that operates without a web browser or plug-ins"

Parallel argues its construction is supported by the specification and Defendants improperly narrow the claim language by reference to a web browser or a plug-in in contradiction to the specification.

Defendants contend the specification makes clear the claimed invention is narrower than the plain claim language. Defendants argue the patentee distinguished his invention's data manipulation functionality from the operations of web browsers or plug-ins both in the specification and during prosecution.

As previously discussed, Defendants' construction for this term is based on an erroneous assumption that the claims are limited to a web-based environment. The specification does not support Defendants' narrow construction. The specification only indicates a permissive use that excludes web browsers or plug-ins:

> The teachings of the present invention allow a server computer to provide data and applications of various types to clients without the use of plug-ins. The data and information may be packaged together in an applet with associated functionality for the data and information. The data and information may be represented using

> pre-loaded, non-updateable elements or through a selected combination of non-updateable elements and updateable elements in the applet based on a prediction made by the server regarding which method will result in lower total amounts of information being transferred over a low-speed link coupling the server to the client. Thus, more efficient use of client and communications link resources over traditional methods is provided.

Id. at 17:12–25.

Nor does the prosecution history support Defendants' position. In distinguishing Johnson, the patentee referred to the user interface of a browser rather than the data interface of the executable applet: "Johnson does not teach or suggest an 'applet dynamically generated by the server [. . .]' because Johnson merely involves . . . transmitting the user interface of a browser." Oct. 16, 2001 Resp. to Office Action at 8 (Docket No. 481–41); Nov. 1, 2001, Adv. Action at 2 (Docket No. 481-42). The patentee's statement was not a disclaimer of the use of web browsers or plug-ins. Defendants also argue the patentee added the claim language "a data interface capability" to gain allowance over Johnson. However, the patentee actually amended the entire "further constituent system" limitation, adding to the claim language "a further constituent system associated with the applet comprising a data interface capability configured to provide a plurality of operations on the pre-loaded values, the operations comprising operations associated with the subset of the data items." Nov. 16, 2001 Amend. at 8 (Docket No. 481-43). Accordingly, Defendants' proposed construction is not supported by the specification or prosecution history.

Parallel's proposed construction is consistent with the specification as a whole, and the Court construes "data interface capability" as "program code that allows the client to access and use data." *See* '111 patent at 11:21–24.

SUMMARY JUDGMENT

Background

Defendants move for summary judgment on two issues. First, Defendants contend they do not infringe because the alleged "executable applet[s]," with their "data interface capability," are not substantially self-sufficient and depend on a web browser or plug-in to operate. As the Court is not adopting Defendants' proposed claim constructions that would import this limitation, Defendants' first summary judgment issue is moot. Second, Defendants contend they do not infringe because the alleged "executable applet[s]" do not combine the requisite data and associated functionality at the server and are not sent to the client in a single transmission. Having adopted Defendants' construction for the "dynamically generated" term, the Court will examine whether Defendants are entitled to summary judgment.

Applicable Law

To prove literal infringement, the patentee must show that the accused device contains every limitation in the asserted claims. *Cybor Corp. v. FAS Techs., Inc.,* 138 F.3d 1448, 1467 (Fed. Cir. 1998) (en banc). Summary judgment is appropriate following a claim construction when there is no genuine issue of material fact remaining. FED. R. CIV. P. 56; *Ormco Corp. v. Align Tech., Inc.*, 498 F.3d 1307, 1312 (Fed. Cir. 2007).

Analysis

At the time this motion was filed, this issue affected 99 Defendants (the "Moving Defendants"). Parallel has only accused the Moving Defendants of literal infringement. Parallel does not dispute how the Moving Defendants' systems operate. *See* Parallel's Reply in Support of Its Opening Claim Construction Brief and Response to Defendants' Motion for Summary Judgment

at 28 (Docket No. 526) ("[D]efendants' experts identify three "load sequences" for the transmission of the applets in the accused instrumentalities, which Parallel will accept for purposes of this summary judgment response."). Defendants' experts identify three ways in which the Moving Defendants' systems function. Defendants' Resp. Claim Const. Brief and Mtn. for Summ. J. Exs. A1 at 40, 41, 42 (Docket No. 481-2); A2 at 42, 43, 44 (Docket No. 481-3); A3 at 44, 45, 46 (Docket No. 481-4). In the first case, the HTML response to the client request includes the necessary data and a link to the functional code. In the second case, both the data and functional code are contained as a link in the HTML response. In the third case, the data is contained in the functional code as a link. Thus, in all three instances, at least one of the functional code or necessary data is contained in the HTML response as only a link.

The Court has construed "[applet] dynamically generated by the server in response to the request" (claim 1) as constructed at the server, by combining the requisite functionality with the necessary data, at the time of and response to the client request" and "generating [an executable applet] dynamically in response to the data request" (claim 17) as "constructing at the server, by combining the requisite functionality with the necessary data, at the time of and in response to the client request." The issue on summary judgment is whether the Moving Defendants' accused instrumentalities combine the data and functionality at the server.

Parallel contends the accused instrumentalities' data and functionality are associated with the applets because an applet can be constructed with the functionality and data sent as separate files to be assembled and executed on the client device. Parallel also disputes whether Defendants' single transmission limitation is a proper claim limitation.

The present invention's applet combines both the functional code and data. The claim language requires both the data and functional code exist as operative components of the applet. The claim language describes the applet as transferred over the communications link to the client device and having two constituent systems: data and data interface capability (functionality). Therefore, the claim language indicates that in the HTML context, the data and functional code are associated by actually being listed within the HTML code and not merely provided in a link.

The specification also indicates that the applet contains an executable program and particular data: "the applet 26 does not merely contain an executable program as with typical applets. In contrast, the applet 26 also comprises particular services and data for the client 12 based on the request." '111 patent at 9:39–43. "Another way of looking at the applet 26 is to consider applet 26 as comprising a program, which acts as the data manipulation system 30, with a particular payload of data occupying the data storage system 32." *Id.* at 12:52–55; *see, e.g., id.* at 11:9–12; 17:12–16.

> When the applet 26 is generated, the applet 26 also comprises substantially all of the data manipulation capabilities for reviewing, manipulating, and utilizing the data items 28 represented by pre-loaded elements 36 in the applet 26. Since the applet 26 comprises substantially all of the data items 28 appropriate to respond to the request, repeated transfers of information over the communications link 16 may be avoided. Stated another way, the pre-loaded elements 36 may leech off of the loading of the data manipulation capabilities to provide data and functionality to the applet 26.

Id. at 15:36–46.

Not only does the patent teach that the functionality and data are combined in the applet, the patent teaches away from the necessity of multiple transmissions to communicate the applet to the client as would be required by including either the data or functionality in an HTML link. "By transmitting the appropriate data and associated data handling capabilities as a group, the client may be required to communicate over a low-speed communications link a greatly reduced number of

times or, in some cases, only once." *Id.* at 3:13–17.

> The total number of accesses made over the communications link 16 from the client 12 is also decreased by including the data requested by the client 12 in the applet 26 as part of a single transmission. Stated another way, instead of downloading generic program functionality to the client 12 and then having the program use the communication link 16 in order to access required data, the needed functionality and the required data are bundled together in the applet 26 and may be transferred to the client 12 as part of the single transmission.

Id. at 12:2-11; *see also id.* at 8:4-8.

The Moving Defendants' accused instrumentalities do not operate in the manner claimed by the '111 patent. Moreover, they function in the precise way taught away from by the patent. By including either the data or functionality in a link rather than actual code to communicate the data or functionality, multiple transmissions between the server and client are required to transmit the necessary aspects of the applet.

Parallel has accused the Moving Defendants of literal infringement. By including a link, rather than the actual code, to either the data or functionality of the accused applet, the Moving Defendants do not satisfy the "[applet] dynamically generated by the server in response to the request" limitation of Claim 1 or the "generating [an executable applet] dynamically in response to the data request" of Claim 17. Accordingly, the Moving Defendants are entitled to summary judgment of non-infringement.

CONCLUSION

For the reasons given, the Court construes the terms as stated and **GRANTS** Defendants' motion for summary judgment in part.

As of the filing of Defendants' claim construction brief and summary judgment motion, 112 Defendants remained in the case. The summary judgment motion on the "dynamically generated"

issue has resolved this case as to 99 of the 112 Defendants. The Court notes that in many patent cases before it involving multiple defendants, it is frequently faced with motions for severance and transfer to many different districts. Had the Court taken that approach in this case, Parallel and Defendants would be litigating this patent all over the country in many districts at great additional expense to all parties and the judiciary.

The Court commends the parties in this case for working together to identify issues common to nearly all Defendants and moving the case to resolution of these important issues in a timely and economic manner. By doing so, this case was resolved in a manner of months—as opposed to years—for the vast majority of Defendants. By all Defendants remaining in one case in one District, the Court was able to resolve the controversy in the most judicially economic manner sparing many other courts from repetitive work, and at the same time saving the parties very significant sums of money in attorneys fees.

So ORDERED and SIGNED this 12th day of August, 2011.

LEONARD DAVIS
UNITED STATES DISTRICT JUDGE