UNITED STATES DISTRICT COURT
EASTERN DISTRICT OF TEXAS
TYLER DIVISION

BEDROCK COMPUTER TECHNOLOGIES LLC,	)	
	)	CASE NO. 6:09-CV-00269
Plaintiff,	)	
	)	Hon. Leonard E. Davis
v.	)	
	)	**JURY TRIAL DEMANDED**
SOFTLAYER TECHNOLOGIES, INC., CITIWARE TECHNOLOGY SOLUTIONS, LLC, GOOGLE INC., YAHOO! INC., MYSPACE INC., AMAZON.COM INC., PAYPAL INC., MATCH.COM, LLC., AOL LLC, and CME GROUP INC.,	)	
	)	
Defendants.	)	

DEFENDANTS' MOTION TO COMPEL PLAINTIFF TO COMPLY WITH PATENT RULE 3-1 AND TO EXTEND THE TIME TO SERVE INVALIDITY CONTENTIONS

TABLE OF CONTENTS

Page

I. INTRODUCTION

Plaintiff Bedrock Computer Technologies, LLC ("Bedrock") alleges that Defendants infringe its patent because they use servers that run the ***publicly available*** Linux operating system software. Each and every line of the allegedly infringing Linux source code is available to the public, including Bedrock, free of charge. Bedrock had access to the accused source code for years before filing its complaint and presumably analyzed the code beforehand consistent with its obligations under the Federal Rules. Because Bedrock has had unfettered access to all the source code it needs to comply with the Local Patent Rules, it should have provided Defendants with specific infringement contentions. Bedrock failed to do so.

Instead, Bedrock's infringement contentions consist of a single claim chart that accuses 375 versions of the Linux operating system of infringement. Each of these versions includes source code files that change and evolve throughout the 375 versions. Bedrock's omnibus claim chart falls far short of the specificity required by P.R. 3-1.

First, Bedrock's chart merely mimics the language of the claim limitations and vaguely alleges that "code contained within" or "accessed by" certain system functions meets those limitations. This gives Defendants no clue as to what particular "code contained within" or "accessed by" these functions is part of Bedrock's contentions. In fact, Bedrock maintains that still other source code may be implicated in the alleged infringement, but does not identify that code, much less say how this unidentified code meets the claim limitations. This vagueness leaves Defendants guessing about which portions of the Linux source code Bedrock contends are infringing. Given that Bedrock has had full access to the Linux source code for years, there is simply no excuse for not providing this required specific analysis.

Second, Bedrock's single claim chart accuses 375 versions of the publicly available Linux operating system, without distinguishing between versions. These versions were developed by the Linux open source community over a six-year period and include countless modifications and revisions to the source code, each of which are made available to the public. . Bedrock's shotgun infringement allegation in this single claim chart fails to account for any of these changes. Indeed, some of the 375 versions do not even contain the functions that Bedrock accuses of infringement. The Court should require Bedrock to submit a separate claim chart for each Linux version in which the lines of code alleged to be infringing have changed.

Finally, Bedrock's infringement contentions do not meet other specificity requirements contemplated by the Patent Rules. With respect to means-plus-function elements, Bedrock violates P.R. 3-1(c) in failing to identify the allegedly infringing structures corresponding to those claim elements. Bedrock also alleges that each Defendant "makes, uses, sells, offers to sell ***or*** imports (or actively induces or contributes to same)" infringing systems (emphasis added). This allegation, made in the disjunctive, covers nearly every potential infringement theory but does not explain with any particularity how Bedrock contends Defendants infringe. Finally, Bedrock improperly alleges that Defendants infringe every claim limitation both literally and under the doctrine of equivalents. Bedrock should be required to state its actual theory and should not be allowed to leave open every potential infringement theory under the sun.

If this was a typical software patent case, Defendants' request for specific contentions might be premature at this stage because typically Plaintiffs accuse an application or product, the source code of which is more often than not proprietary and not publicly available. That is not the case here, where it is the publicly available code itself that is accused. Bedrock must have reviewed and identified the specific publicly available code that it believes amounts to infringement in order to meet its Rule 11 obligations prior to filing its complaint. Bedrock must provide contentions that identify this specific code; it has all of the information it needs to formulate specific infringement contentions. Until Bedrock supplements its contentions, Defendants will continue to be prejudiced as they attempt to develop their defenses, including developing adequate invalidity contentions and non-infringement contentions. Indeed, Bedrock's contentions as to how the Linux operating system software infringes its claims are critical to determining the scope and substance of Defendants' invalidity contentions. Therefore, Defendants request that this Court extend the time for Defendants to serve invalidity contentions until the latter of (i) 45 days after the date Bedrock serves infringement contentions compliant with Rule 3-1, or (ii) the date set forth in the Docket Control Order.

II. FACTUAL BACKGROUND

A. The Accused Linux Source Code Is Publicly Available

Bedrock has accused the Linux software operating on Defendants' servers of infringement. Linux, unlike most commercial software, is "open source." "Open source" means that the public has access to and can review, comment on, and even modify this Linux code. Indeed, the public is encouraged to do all of these things because there is no one company that is solely responsible for the development of the Linux software. Rather, each version of Linux (known as a "kernel") reflects the input of thousands of open source community members – academics, industry professionals, and other computer programmers from around the world. In addition to writing the code, the Linux community debates and analyzes Linux on publicly accessible websites and online forums, in print, and at trade shows and conferences.

The Linux kernel source code consists of an extensive folder and file structure. There are thousands of files in the Linux kernel, each written in the C programming language. Each of these program files consists of numerous "functions" that consist of lines of code that instruct the computer on what to do. These functions may in turn “call” other functions, in either the same file or another file within the Linux kernel. When another function is “called,” the lines of code within the “called” function are run. Because functions within a particular C file can call functions in any other file in Linux, all of these lines of code within the thousands of files can interact with each other, and instruct the computer to run any function in any file. In addition to functions, the files further refer to data structures. A data structure is a way in which all of these functions and lines of code can organize and use data.

The source code for every Linux version is stored online and made available freely to the public at numerous locations, including at http://www.linux.org. This website explains: “Linux is a free Unix-type operating system originally created by Linus Torvalds with the assistance of developers around the world. Developed under the GNU General Public License, the source code for Linux is freely available to everyone.”[1] Because the Linux source code is publicly available and free, Bedrock had complete, constant, unfettered access to it long before filing the complaint. This is true not only of the current version of the Linux kernel, but also of all prior versions.[2] Thus Bedrock does not need any discovery from Defendants to reveal how the accused Linux code works.

B. Bedrock’s Infringement Contentions Fail To Identify The Allegedly Infringing Linux Source Code With Specificity

Bedrock filed its complaint for patent infringement against Defendants on June 16, 2009. The complaint made only bare allegations of infringement based on U.S.P. 5,893,120 and did not identify ***any*** product or service of any of the Defendants as allegedly infringing the patent-in-

[1] *See* http://www.linux.org.

[2] *See* The Public Linux Archive, available at http://www.kernel.org/pub/ (housing for the public every version of the Linux kernel).

suit. Bedrock's complaint left Defendants completely in the dark as to the products or services that were at issue in this case, and several Defendants sought relief from the Court.[3]

On October 9, 2009, Bedrock served its infringement contentions on Defendants.[4] The contentions were accompanied by a single claim chart for each Defendant, which alleged infringement of all eight claims in the '120 patent. The single claim chart purported to apply to 375 different versions of Linux that were developed over a six year period.[5]

Bedrock's chart referred to Defendants' alleged use of certain functions and data structures in the "route.c" module of Linux, but did not go beyond the names of those code components to identify which particular portions or lines of code are involved. Rather, Bedrock purported to explain its contentions by merely reciting the claim language for each element. As explained in more detail below, this is insufficient because these functions contain significant amounts of code that may or may not be relevant to Bedrock's contentions, and these functions call other functions that may or may not be relevant to Bedrock's contentions. Defendants have no way of knowing what specific code within the identified functions and what specific functions called by the unidentified code are alleged to infringe. Making matters even more difficult, Bedrock notes that its chart may not identify "all of the code necessary to satisfy the claim limitations at issue."[6]

Bedrock's infringement contentions were similarly vague with regard to its theories of direct and indirect infringement. Bedrock broadly alleged that each Defendant "makes, uses,

[3] Defendants AOL, MySpace, Amazon.com, CME Group and Yahoo! filed a motion to dismiss for failure to state a claim because Bedrock failed to provide any facts that support its conclusory allegations of patent infringement. (Dkt. No. 71.) In its opposition, Bedrock indicated that the requisite specificity would be forthcoming in its infringement contentions. (Dkt. No. 84 at 2.)

[4] An example of Bedrock's Infringement Contentions is attached as Exhibit A. The contentions are the same for each Defendant, with only the name of the Defendant changed.

[5] Specifically, Bedrock accused 45 versions of Linux, which include numerous sub-versions. Including these sub-versions, there are approximately 375 accused versions of Linux.

sells, offers to sell or imports (or actively induces or contributes to same) computer equipment configured with or utilizing software based on [the accused] Linux versions."[7] Moreover, for each limitation, Bedrock asserted that Defendants' infringement was literal but nevertheless contended that "[i]n the event this limitation is construed so as to be not literally present in the Accused Instrumentalities, Bedrock would alternatively contend that the Accused Instrumentalities meet the recited limitation under the doctrine of equivalents . . ." without providing any analysis regarding the doctrine of equivalents.[8]

C. The Parties' Meet And Confer Process

On October 20, 2009, Defendants AOL, MySpace, Amazon.com, and CME Group sent a letter to Bedrock identifying deficiencies in its infringement contentions. In response, Bedrock took the position that it had complied with the Patent Rules and served an uncalled-for 800-page version of its infringement contentions, consisting of a series of identical copies of its original single claim chart for each accused version of the Linux kernel, which purportedly addressed the request that Bedrock treat the multiple accused versions of Linux individually in separate claim charts.[9] Defendants Google and Match.com and later Yahoo! sent similar letters to Bedrock and received similar responses.[10,11] Wishing to avoid unnecessary motion practice, Defendants held teleconferences with Bedrock to discuss this dispute on November 3, 2009 and December 3, 2009. Defendants requested that Bedrock address the issues raised in Defendants' letters, but

[6] Ex. A to Bedrock's Infringement Contentions at 1.

[7] *Id.*

[8] Bedrock's Infringement Contentions at 2.

[9] Ex. B (correspondence between counsel for AOL, MySpace, Amazon.com, and CME Group and counsel for Bedrock).

[10] Ex. C (correspondence between counsel for Google and Match.com and counsel for Bedrock). In this correspondence, Bedrock confirmed it was accusing Google's and Match.com's servers that run Linux of infringement, and not any specific Google or Match.com products.

[11] Ex. D (correspondence between counsel for Yahoo! and counsel for Bedrock).

Bedrock refused. Given the impasse, Defendants were left with no choice but to seek the Court's assistance.

III. ARGUMENT

A. Patent Rule 3-1 Requires Plaintiffs To Identify Specifically Where Each Claim Element Of Each Asserted Claim Is Located

The Patent Rules exist to "further the goal of full, timely discovery and provide all parties with adequate notice and information with which to litigate their cases . . . ." *Comp. Acceleration Corp. v. Microsoft Corp.*, 503 F. Supp. 2d 819, 822 (E.D. Tex. 2007). Accordingly, "[t]he Patent Rules demonstrate high expectations as to plaintiffs' preparedness before bringing suit, requiring plaintiffs to disclose their preliminary infringement contentions before discovery has even begun." *ConnecTel, LLC v. Cisco Sys., Inc.*, 391 F. Supp. 2d 526, 527 (E.D. Tex. 2005) (quoting *Amer. Video Graphics, L.P. v. Electronic Arts, Inc.*, 359 F. Supp. 2d 558, 560 (E.D. Tex. 2005). Plaintiffs "are expected to rigorously analyze all publicly available information before bringing suit and [] explain with great detail their theories of infringement." *Id.* at 528.

To comply with the Patent Rules, plaintiffs are required to "set forth particular theories of infringement with sufficient specificity . . . beyond that which is provided by the mere language of the patent [claims] themselves." *Id.* at 527-28 (quotations omitted). P.R. 3-1 requires plaintiffs to serve "[a] chart identifying specifically where each element of each asserted claim is found within each Accused Instrumentality . . . ." P.R. 3-1(c). Plaintiffs must do more than rely on "vague, conclusory language or simply mimicking the language of the claims." *ConnecTel LLC*, 391 F. Supp. 2d at 528 Not only must the plaintiff show specifically where the elements are found in claim charts, it must also explain "*how* [the] accused infringing products read on the asserted claim language." *Id.* (emphasis added). When more than one product is accused, the plaintiff must provide a separate claim chart for each product, "tak[ing] into account the differences that exist among each Accused Product." *Linex Techs., Inc. v. Belkin Int'l, Inc.*, 628 F. Supp. 2d 703, 706 (E.D. Tex. 2008).

When software is accused of infringement and the plaintiff has access to the source code, there must be "specific references to the source code" in the claim charts. *Amer. Video Graphics*, 359 F. Supp. 2d at 561 (ordering plaintiff to add "specific references to the source code" within 30 days of the accused product's source code being deposited in escrow). Importantly, this Court has also ruled that merely identifying functions in source code is not enough. *Michael S. Sutton Ltd. v. Nokia Corp.*, No. 6:07-cv-00203-LED, at *5 (E.D. Tex. Feb. 13, 2009) (Docket No. 59) (holding that plaintiff's identification of "[t]he Editor_HandleMessagePackingEvents function" and the "EMS_Packer function" implemented in "emspacker.c" for certain steps in the claims did not "specifically identify where these steps are found in the source code"); *see also Diagnostic Sys. Corp. v. Symantec Corp.*, 2009 WL 1607717, *3-*6 (C.D. Cal. June 5, 2009) (granting motion to compel and stating that "after a plaintiff[] has had a reasonable opportunity to review the source code for the defendant's accused software product, the patentee's time for trolling the proverbial waters for a theory of infringement comes to an end, and the patentee must fish or cut bait with respect to its specific theory of infringement by providing PICs to the defendant that clearly identify and explain how the source code for the accused product infringes upon specific claims").

B. Bedrock's Infringement Contentions Do Not Provide The Specificity Required By Patent Rule 3-1

1. Bedrock's Infringement Contentions Do Not "Identify Specifically" The Source Code Accused Of Infringement

P.R. 3-1 requires a "chart identifying ***specifically where each element of each asserted claim is found within each Accused Instrumentality.***" (Emphasis added). Bedrock refers to certain functions or data structures – or vaguely to code "contained within" or "accessed by" those functions or data structures – and then merely "mimic[s] the claim language of the patents-at-issue" rather than explaining how the code within the function or structure meets the corresponding claim limitation. *ConnecTel, LLC*, 391 F. Supp. 2d at 527. Bedrock's references are not sufficient because the functions or data structures Bedrock identifies contain hundreds of lines of code and call dozens of other functions (which in turn contain hundreds, perhaps thousands, more lines of code). The allegedly infringing code could therefore be in any of the lines of code in the functions Bedrock refers to, or in any line of code in the functions called by those functions. For the Court's convenience, Defendants have attached as Exhibit E a chart detailing element-by-element the deficiencies in Bedrock's infringement contentions.

By way of example, for claim element 1(c), Bedrock's contention mimics the claim language:

Claim Language: Element 1(c)	Bedrock's Infringement Contention
the record search means including a means for identifying and removing at least some of the expired ones of the records from the linked list when the linked list is accessed, and	Specifically, code contained within function rt_intern_hash, as invoked by functions ip_route_input_mc, ip_mkroute_input, ip_route_input_slow, ip_mkroute_output, ip_rt_redirect, and/or ip_route_output_slow in module /net/ipv4/route.c, comprises record search means including a means for identifying and removing at least some of the expired ones of the records from the linked list when the linked list is accessed or its equivalent.

Thus, Bedrock merely contends that "code contained within function rt_intern_hash . . . comprises . . ., and then repeats the claim language. In the publicly available route.c file from the latest accused version of Linux (2.6.31), the function rt_intern_hash contains 186 lines of code. Within those lines of code, the function makes calls to at least 20 other functions, some of which are not defined in the route.c file. Each of those functions, in turn, calls other functions. For instance, the function rt_garbage_collect, one of the functions called by the function rt_intern_hash, itself has 121 lines of code and calls at least 13 other functions. And so forth. As becomes apparent in this unraveling of the code, the seemingly "specific" citation to the rt_intern_hash function does nothing to narrow the scope of potentially infringing code, much less affirmatively identify allegedly infringing code, and thus runs afoul of the specificity requirements of P.R. 3-1.

This Court recently ruled that mere identification of functions in source code was not enough to satisfy a plaintiff's Rule 3-1 obligations. *Michael S. Sutton Ltd.,* No. 6:07-cv-00203-LED. With respect to the identification of functions, the Court stated: "While this may provide some information on Sutton's infringement theory, it does not point specifically to where each step is found. Thus Nokia is left guessing as to the alleged location of the steps in the source code. Consequently, Sutton's claim chart does not show the location of each element and thus does not meet the standard set by P.R. 3-1(c)." *Id.* at *5.

Bedrock's identification of source code functions is no different than Sutton's identification of such functions. Bedrock not only has failed to specifically cite the accused code in the identified functions, but also has disclaimed any responsibility to identify the accused code, whether contained in the identified functions or elsewhere. In footnote 1 of its chart, Bedrock states:

> Bedrock's identification of specific routines is ***not intended to identify all of the code necessary to satisfy the claim limitation at issue***. Other source and/or object code components, such as routines, functions, methods, macros, classes, data structures, libraries etc. may be necessary. Furthermore, these identified code components may be called by or call other code components. Also, the code components identified are normally the first or the most relevant in a series of code components and are intended to reference the entire function path.[12]

In other words, far from identifying the allegedly infringing code, Bedrock has cited only code "components" – and not even all relevant code components but, rather, only "the first or most relevant in a series of code components" – leaving the identity and the number of the other allegedly relevant components to the imagination. Bedrock's refusal to cite any allegedly infringing source code or even to identify all of the places where it might be found leaves Defendants guessing as to the code involved in the alleged infringement. Bedrock apparently is attempting to leave itself in a position where it can tailor its infringement contentions to any of a myriad of possible interpretations.

Presumably in its pre-filing investigation and in preparing its infringement contentions, Bedrock examined the publicly available Linux code and thus already knows exactly which lines of code it contends infringe the patent-in-suit. This information – which is in Bedrock's possession and which would clearly assist Defendants in crystallizing their non-infringement and invalidity theories – is part of the specific identification required by Rule 3-1. Thus, it should have been disclosed in Bedrock's infringement contentions. *See Network Caching Tech., LLC v. Novell, Inc.*, 2002 WL 32126128 at *4 (N.D. Cal. Aug. 13, 2002) (stating plaintiff "must provide in its [preliminary infringement contentions] the relevant facts it obtained in its pre-filing inquiry").

Defendants are prejudiced by Bedrock's failure to provide adequate disclosure of its infringement theories. Without specific citations to source code, they are "unable to crystallize [their] non-infringement and invalidity theories, and . . . are hindered in identifying what claim terms need construction." *ConnecTel*, 391 F. Supp. 2d at 528; *Amer. Video Graphics,* 359 F. Supp. 2d at 560 ("To the extent defendants are given vague infringement contentions, they are hampered in their ability to prepare their defense.").

[12] Exhibit A to Bedrock's Infringement Contentions at 1 (emphasis added).

2. Bedrock's Infringement Contentions Do Not Provide Separate Claim Charts For "Each Accused Instrumentality"

Bedrock's infringement contentions also fail to comply with Rule 3-1 because Bedrock supplied only one claim chart for 375 versions of Linux. These versions represent at least six years of development of the Linux code. Over this lengthy period of time, the accused route.c module was changed many, many times.

That fact that some of the functions Bedrock contends are infringing do not even appear in all of the accused versions of the route.c code highlights the shortcomings of Bedrock's "one size fits all" approach. For example, for claim element 1(d), Bedrock's infringement contention (which again mimics the claim language of the element) is as follows:

Claim Language: Element 1(d)	Bedrock's Infringement Contention
means, utilizing the record search means for accessing the linked list and, at the same time, removing at least some of the expired ones of the records in the linked list.	Specifically, the functions ip_route_mc, ip_mkroute_input, ip_route_input_slow, ip_mkroute_output, ip_rt_redirect, and/or ip_route_output_slow in module /net/ipv4/route.c include means, utilizing the record search means, for accessing the linked list, and, at the same time, removing at least some of the expired ones of the records in the linked list or its equivalent.

Many of the accused versions of route.c do not have either of the functions ip_mkroute_input or ip_mkroute_output. These functions were apparently added in Linux kernel version 2.6.12. The Defendants are left guessing as to whether these functions are a part of Bedrock's contentions.

This is just one of many differences in code among the many accused versions of Linux. A document comparison between the earliest accused version of the route.c file, from Linux 2.4.22, and the most recent accused version of the route.c file, Linux 2.6.31, yields 3451 total changes in the file.[13] This comparison includes code within the cited functions, some of which was substantially changed during this span of time.

[13] Both files are available at http://www.kernel.org/pub/.

Given that Bedrock has failed to identify the specific lines of code that it believes to be infringing, Defendants cannot determine whether any of the changes made in the identified functions are relevant to the parties' respective infringement and non-infringement positions.

C. Bedrock's Infringement Contentions Fail To Comply With Patent Rule 3-1 For Additional Reasons

Means-Plus-Function Limitations: Many of the asserted claims include means-plus-function limitations. Yet Bedrock has failed to comply with Rule 3-1(c), which requires it to provide the "identity of the structure(s), act(s), or material(s) in the Accused Instrumentalit[ies]" for claim elements governed by 35 U.S.C. § 112(6). *See Alberta Telecomm. Research Centre v. Rambus, Inc.*, 2007 WL 4170564 (N.D. Cal. 2007). For example, with respect to claim element 1(c), Bedrock merely provides a list of named functions for the entire element. Bedrock fails to identify any specific code that is the "record search" means. Similarly, Bedrock fails to identify any specific code that provides the "means for identifying" or the "means for removing." This lack of specificity extends to each and every means-plus-function limitation in the claims. Defendants should not be forced to guess which portions of the code Bedrock is interpreting as the structure corresponding to these claim limitations.

Direct and Indirect Infringement Allegations: Bedrock broadly alleges that each Defendant "makes, uses, sells, offers to sell ***or*** imports (***or*** actively induces ***or*** contributes to same)" the accused instrumentalities.[14] The string of disjunctives in this assertion does not specifically identify Bedrock's direct or indirect infringement theories. Indeed, it does not appear that any theory has been ruled out. This kitchen-sink approach of alleging every potential direct and indirect infringement theory does not comply with Rule 3-1.

[14] *See, e.g.*, Exhibit A to Bedrock's Infringement Contentions at 1 (emphasis added).

Literal Infringement and Doctrine of Equivalents Allegations: Finally, Bedrock improperly alleges both literal infringement and infringement under the doctrine of equivalents. P.R. 3-1(d) provides that the patentee must disclose "whether each element of each asserted claim is claimed to be literally present or present under the doctrine of equivalents." Rather than stating whether it alleges literal infringement or infringement under the doctrine of equivalents, Bedrock has decided to cover all bases and alleges that each limitation is infringed under both theories.[15] Such allegations in the alternative are improper under the Local Patent Rules. Defendants request that Bedrock correct its infringement contentions to allege either literal infringement or infringement under the doctrine of equivalents. In the event Bedrock later wishes to change its allegations, it can seek to amend its contentions pursuant to the Local Rules.

D. Bedrock's Deficient Infringement Contentions Preclude Defendants from Developing Invalidity Contentions

As explained to Bedrock's counsel, Defendants are severely prejudiced by Bedrock's failure to comply with the Local Patent Rules. Without proper infringement contentions and without the ability to determine how Bedrock interprets the claims, it is impossible for the Defendants to properly construct their invalidity contentions. Further, because Defendants are unable to determine the specific functions Bedrock believes its claims read on, Defendants cannot determine the scope their invalidity contentions should take.

To avoid this prejudice, Defendants request that the date to serve their invalidity contentions be moved to the latter of (i) 45 days after the date Bedrock serves compliant infringement contentions, or (ii) the date set forth in the Docket Control Order.

[15] *See, e.g.*, Exhibit A to Infringement Contentions, at 1-2, 2-3, 3, 4, 5, 5-6, 6-7, 7.

IV. CONCLUSION

Bedrock has had access to each accused version of the publicly available Linux source code long before this case was filed, but has failed to provide the highly specific infringement contentions required by Rule 3-1 and this Court's prior decisions. Consequently, Defendants respectfully request that the Court compel Bedrock to serve infringement contentions that provide:

- Detailed identifications of where each element of each asserted claim is found, including in each instance specific references to source code;
- A separate claim chart for each version of Linux where the code implicated by the infringement contentions has changed;
- Detailed explanations of the how the code performs the alleged infringement that do not merely "mimic" the language of the asserted claims;
- Specific identification of the "the identity of the structure(s), act(s), or material(s) . . . that performs the claimed function" for each claim element that Bedrock contends is governed by 35 U.S.C. § 112(6);
- Bedrock's specific theories of direct or indirect infringement; and
- A statement whether Bedrock alleges literal infringement or infringement under the doctrine of equivalents for each claim limitation.

Defendants further request that the date to serve their invalidity contentions be moved to 45 days after the date on which the Court requires Bedrock to serve compliant infringement contentions, or the date set by the Docket Control Order, whichever is later.

Dated: December 9, 2009

Respectfully submitted,

By: */s/ Alan L. Whitehurst*
Frank G. Smith
frank.smith@alston.com
ALSTON & BIRD LLP
One Atlantic Center
1201 West Peachtree Street
Atlanta, GA 30309
Telephone: (404) 881-7240
Facsimile: (404) 256-8184

Alan L. Whitehurst
alan.whitehurst@alston.com
Marissa R. Ducca
marissa.ducca@alston.com
ALSTON & BIRD LLP
The Atlantic Building
950 F Street, N.W.
Washington, DC 20004
Telephone: (202) 756-3300
Facsimile: (202) 756-3333

Michael J. Newton (SBN 24003844)
mike.newton@alston.com
ALSTON & BIRD LLP
Chase Tower
2200 Ross Avenue, Suite 3601
Dallas, TX 75201
Telephone: (214) 922-3423
Facsimile: (214) 922-3839

Louis A. Karasik (*pro hac vice*)
lou.karasik@alston.com
ALSTON & BIRD LLP
333 South Hope Street
16th Floor
Los Angeles, CA 90071
Telephone: (213) 576-1148
Facsimile: (213) 576-1100

By: */s/ Michael E. Jones (with permission)*
Michael E. Jones
State Bar No. 10929400
mikejones@potterminton.com
POTTER MINTON
110 N. College
Tyler, Texas 75702
Telephone: (903) 597-8311
Facsimile: (903) 593-0846

Claude M. Stern
claudestern@quinnemanuel.com
Todd M. Briggs
toddbriggs@quinnemanuel.com
QUINN EMANUEL URQUHART OLIVER & HEDGES, LLP
555 Twin Dolphin Dr., Suite 560
Redwood Shores, CA 94065
Telephone: 650-801-5000
Facsimile: 650-801-5100

Attorneys for Defendants Google Inc. and Match.com, LLC

By: */s/ Yar R. Chaikovsky (with permission)*
Yar T. Chaikovsky
ychaikovsky@sonnenschein.com
SONNENSCHEIN NATH & ROSENTHAL LLP
1530 Page Mill Road, Suite 200
Palo Alto, CA 94304
Telephone: (650) 798-0300
Facsimile: (650) 798-0310

Attorney for Defendant Yahoo! Inc.

Attorneys for Defendants MySpace Inc.,
Amazon.com Inc., AOL LLC,
CME Group Inc, and PayPal Inc.

Michael E. Jones
State Bar No. 10929400
Allen Gardner
State Bar No. 24043679
POTTER MINTON
A PROFESSIONAL CORPORATION
110 N. College, 500 Plaza Tower
P.O. Box 359
Tyler, Texas 75710
Tel: (903) 597-8311
Fax: (903) 593-0846
mikejones@potterminton.com
allengardner@potterminton.com

Attorneys for Defendant PayPal, Inc.

Dated: December 9, 2009

CERTIFICATE OF SERVICE

The undersigned hereby certifies that counsel of record who are deemed to have consented to electronic service are being served with a copy of this **DEFENDANTS' MOTION TO COMPEL PLAINTIFF TO COMPLY WITH PATENT RULE 3-1 AND TO EXTEND THE TIME TO SERVE INVALIDITY CONTENTIONS**, via the Court's CM/ECF system per Local Rule CV-5(a)(3) on December 9, 2009.

By: *__/s/Alan L. Whitehurst________*
Alan L. Whitehurst

CERTIFICATE OF CONFERENCE

I HEREBY CERTIFY that, in accordance with the requirements of Local Rule CV-7(h), I have conferred with lead and local counsel for all of the parties affected by this motion. Plaintiff Bedrock Computer Technologies LLC opposes this motion.

/s/ Alan L. Whitehurst
Alan L. Whitehurst