IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

ORACLE CORPORATION and ORACLE U.S.A. INC.,	)	
Plaintiffs/Counterdefendants,	)	C.A. No. 06-414 (SLR)
v.	)	REDACTED PUBLIC VERSION
EPICREALM LICENSING, LP,	)	
Defendant/Counterclaimant.	)	

ORACLE'S OPENING CLAIM CONSTRUCTION BRIEF

MORRIS, NICHOLS, ARSHT & TUNNELL LLP
Mary B. Graham (#2256)
James W. Parrett, Jr. (#4292)
1201 North Market Street
P.O. Box 1347
Wilmington, DE 19899
mgraham@mnat.com
jparrett@mnat.com
(302) 658-9200

Attorneys for Oracle Corporation and Oracle U.S.A. Inc.

OF COUNSEL:

James G. Gilliland
Theodore T. Herhold
Joseph A. Greco
Robert J. Artuz
Eric M. Hutchins
TOWNSEND AND TOWNSEND AND CREW LLP
379 Lytton Avenue
Palo Alto, CA 94301
(650) 326-2400

Dorian Daley
Peggy E. Bruggman
Matthew M. Sarboraria
ORACLE CORPORATION
ORACLE U.S.A., INC.
500 Oracle Parkway
Redwood Shores, CA 94065

DATED: July 31, 2008

Redacted Filing Date: August 7, 2008

TABLE OF CONTENTS

Page

I. NATURE AND STAGE OF PROCEEDINGS 1

II. SUMMARY OF ARGUMENT 1

III. STATEMENT OF FACTS 2

A. Early Development and Failed Efforts to Commercialize the Patented Technology 2

B. The Questionable Validity and Enforceability of the Patents-in-Suit 4

C. The Current Litigation and Prior Claim Construction 5

D. The Accused Oracle Products 5

E. The Subject Technology 6

F. The Patents-In-Suit 9

IV. LEGAL STANDARDS FOR CLAIM CONSTRUCTION 13

V. ORACLE'S PROPOSED CLAIM CONSTRUCTIONS 16

A. Releasing Terms 16

1. Releasing 16

2. Said page server receiving said request and releasing said Web server [second computer system or HTTP-compliant device] to process other requests 16

B. Intercepting Terms 21

1. Interceptor 21

2. Intercepting said request at said Web server 21

3. Intercepting said request at said second computer system 22

4. Intercepting said request at said HTTP-compliant device 22

C. Dispatching Terms 27

1. Dispatcher 27

2. Dispatching / Dispatching said request to said page server 27

TABLE OF CONTENTS (*Continued*)

Page

3. The "Dispatcher" of the Claims is a Software Program Distinct From Other Required Elements of the Claims 27

4. EpicRealm's Construction of "Dispatching" Improperly Incorporates Limitations Into the Claim From the Specification 30

D. General Web Terms 33

1. Page Server 33

2. Web Page 34

3. Web Server/HTTP-Complaint Device 35

4. Request 36

5. General Web Terms for Which EpicRealm Offers No Construction 36

E. Means-Plus-Function Terms 37

1. Means for Generating 37

2. Means for Receiving 37

3. Page Server Processing Means 37

VI. CONCLUSION 39

TABLE OF AUTHORITIES

Page

Cases

Bicon, Inc. v. Straumann Co.,
441 F.3d 945 (Fed. Cir. 2006) 14, 15, 20, 21, 29

Markman v. Westview Instruments, Inc.,
52 F.3d 967 (Fed. Cir. 1995) 13, 14, 18, 26

Medrad, Inc. v. MRI Devices Corp.,
401 F.3d 1313 (Fed. Cir. 2005) 14, 21

Medtronic, Inc. v. Advanced Cardiovascular Sys., Inc.,
248 F.3d 1303 (Fed. Cir. 2001) 16, 38

O2 Micro Int'l Ltd. v. Beyond Innovation Tech. Co. Ltd.,
521 F.3d 1351 (Fed. Cir. 2008) 18, 26, 29, 37

Phillips v. AWH Corp.,
415 F.3d 1303 (Fed. Cir. 2005) (en banc) 14, 15, 28, 30, 32, 35

United States Surgical Corp. v. Ethicon, Inc.,
103 F.3d 1554 (Fed. Cir. 1997) 14, 18, 20

Vitronics Corp. v. Conceptronic, Inc.
90 F.3d 1576 (Fed. Cir. 1996) 14, 15

Statutes

35 U.S.C. §112, ¶6 ..... 15, 16, 38

Other Authorities

The American Heritage Dictionary,
3d Ed. (1996) 28, 30

Plaintiffs and Counterdefendants Oracle Corporation and Oracle U.S.A. Inc. ("Oracle") submit the following brief in support of their proposed claim constructions for various disputed claim terms of the patents-in-suit.

I. NATURE AND STAGE OF PROCEEDINGS

This is a patent infringement action, originally brought by Oracle as a declaratory judgment action for non-infringement and invalidity of patents allegedly owned by Defendant and Counterclaimant epicRealm Licensing, LP ("epicRealm"). The patents-in-suit, U.S. Patent Nos. 5,894,554 ("the '554 patent") and 6,415,335 ("the '335 patent"), pertain to systems and methods for managing dynamic web page generation requests.[1] Fact and expert discovery have both been completed. A hearing on claim construction, summary judgment, and *Daubert* motions is scheduled for October 3, 2008. The pre-trial conference is scheduled for December 5, 2008, and trial is scheduled for January 12, 2009. In addition to the present action, epicRealm filed infringement counterclaims in a separate declaratory judgment action initiated by Quinstreet, Inc. in Delaware, Case No. 06-495-SLR. That action is currently scheduled for trial in October, 2009. EpicRealm also filed additional infringement actions against various parties in the Eastern District of Texas. Trial is currently scheduled for August against two of the defendants there.

II. SUMMARY OF ARGUMENT

This brief sets forth Oracle's proposed constructions for various disputed claim terms of the patents-in-suit. As shown below, Oracle's proposed constructions are fully supported by, and

---

[1] A description of the subject technology and patents is contained in section III, E-F, *infra*.

consistent with, both the intrinsic and extrinsic evidence and should therefore be adopted by this Court.[2]

III. STATEMENT OF FACTS[3]

A. Early Development and Failed Efforts to Commercialize the Patented Technology

InfoSpinner, Inc., the purported predecessor to epicRealm, was started in late 1995 by Keith Lowery, the lead inventor of the patents-in-suit, to commercialize Lowery's idea for an Internet-related software product. ForeSite, the embodiment of the patented technology, was developed and sold by InfoSpinner until its reincorporation as epicRealm, Inc. in the Fall of 1999. By that time, while other Internet-based companies were in the midst of the Internet boom, InfoSpinner had yet to turn a profit on its flagship product – ForeSite. As a result, the company changed its name to epicRealm, Inc. and jettisoned its four-year endeavor as a software company based on the patented technology, which had proved to be an epic-flop in the marketplace. The company instead moved into a completely new line of business as a "content delivery network" (CDN) provider. [redacted]

---

[2] In addition to the present brief, Oracle also files herewith: 1) Oracle's Motion for Summary Judgment of Noninfringement; 2) Oracle's Motion for Summary Judgment of Invalidity; 3) Oracle's Motion for Partial Summary Judgment of No Willful Infringement; 4) Oracle's Motion to Strike Expert Testimony of EpicRealm's Damages Expert, Michael J. Wagner; and 5) Oracle's Motion to Exclude Damages Based on Foreign Sales. Exhibits in support of Oracle's motions are contained in the Appendix of Exhibits in Support of Oracle's Opening Claim Construction Brief and Motions ("Appendix"), filed herewith. Evidentiary support for each of the exhibits is contained in the accompanying Declaration of Theodore T. Herhold In Support of Oracle's Opening Claim Construction Brief and Motions.

[3] The following Statement of Facts is intended as background information for all of Oracle's motions. Evidentiary citations are to the exhibits contained in the accompanying Appendix and are referred to by number as "A__."

[redacted]

In March 2005, epicRealm Licensing, LLC was formed by a former Brobeck lawyer, Terry Fokas, who saw an opportunity to use the patents-in-suit to extract licensing fees from companies that were selling successful web-based products. The assets of epicRealm, Inc. were transferred to epicRealm Licensing, LLC in exchange for an interest in the proceeds of any licensing or enforcement actions.

In December 2005, epicRealm Licensing, LLC was merged into yet another entity formed by Fokas named epicRealm Licensing, LP, the named party in this action, to facilitate a tax issue. [redacted]

EpicRealm now seeks hundreds of millions of dollars in damages from Oracle, and

---

[4] The long and convoluted tale of the InfoSpinner/epicRealm history is included in Oracle's Response to EpicRealm's Motion to Substitute Parties, filed on January 2, 2008 (D.I. 62), by which epicRealm seeks to substitute Parallel Networks as the real party in interest in this action. While that motion is currently pending before this Court, the court in the Texas actions – citing the numerous inconsistencies and uncertainties regarding the various corporate transactions that lead to the formation of Parallel Networks and its alleged ownership of the patents-in-suit – denied the same motion filed by epicRealm there. A34.

hundreds of millions of dollars more from the defendants in the Texas actions, based on patented technology that proved to be an utter failure in the marketplace and has now been found invalid by the USPTO (*see* below).

B. The Questionable Validity and Enforceability of the Patents-in-Suit

The patents-in-suit were highly criticized by the Internet community when they were first issued and publicly announced by InfoSpinner in April, 1999. As one industry observer sarcastically put it at the time: [redacted] A16, Exh. 23 at p. EPIC071985. Even epicRealm's own expert, hired in late 2003 to give a written opinion on the enforceability of the patents-in-suit for purposes of epicRealm's contemplated licensing program, concluded: [redacted] A19, Exh. 5.[5]

More recently, six different reexamination requests were filed by various industry participants and subsequently granted by the USPTO. In a lengthy and detailed Office Action dated June 30, 2008, the examiner rejected all of the claims of the '554 patent citing multiple §102 and 103 prior art references that were not cited during the original prosecution of the patents-in-suit. A8. Because of the large number of references cited by the examiner and the comprehensiveness of the office action, it is likely epicRealm will have to amend the claims – assuming it can – in order to overcome the rejections. If epicRealm does so, it will have

---

[5] When epicRealm's hired expert, Dr. Chen, was asked at this deposition what advice he gave to epicRealm concerning its intention to enforce the patents-in-suit against potential infringers, he testified: [redacted] A19, at 106:8-12.

significant consequences not only on the scope and enforceability of the patents-in-suit, but on whether it can even continue to pursue its claims against Oracle and the other defendants.

C. The Current Litigation and Prior Claim Construction

Notwithstanding the suspect validity and enforceability of the patents-in-suit, Fokas proceeded with epicRealm's licensing and enforcement program, eventually initiating numerous lawsuits in Texas and filing infringement counterclaims in Delaware against Oracle and Quinstreet seeking hundreds of millions of dollars in damages. Trial in the Texas actions is scheduled against two of the defendants in August, 2008.

As part of the Texas proceedings, the court construed a number of claim terms from the patents-in-suit. A33. Oracle was not a party to, and did not participate in, the claim construction proceedings in those cases. As set forth more fully below, while the parties here have adopted many of the constructions promulgated by the Texas court, the parties disagree over whether other constructions should be adopted. For example, Oracle agrees to the Texas court's construction of "releasing," a key claim limitation in all of the asserted claims, while epicRealm does not. There are certain additional constructions that Oracle believes are necessary, as well as modifications to those already provided by the Texas court.

D. The Accused Oracle Products

Founded in 1977, Oracle is an enterprise software company that develops, manufactures and sells database and middleware software, as well as applications software designed to help customers manage and grow their businesses. While Oracle's Database software accounts for the vast majority of Oracle's revenues, Oracle also offers various web-based "middleware" products and services which it began to develop and market well before the filing of the patents-in-suit. Oracle's Internet Application Server ("IAS") software, which forms the foundation for Oracle's larger Fusion Middleware product offerings, enables customers to access their back-end database

information over a network such as the Internet.

EpicRealm has accused the following Oracle products of infringing the patents-in-suit:

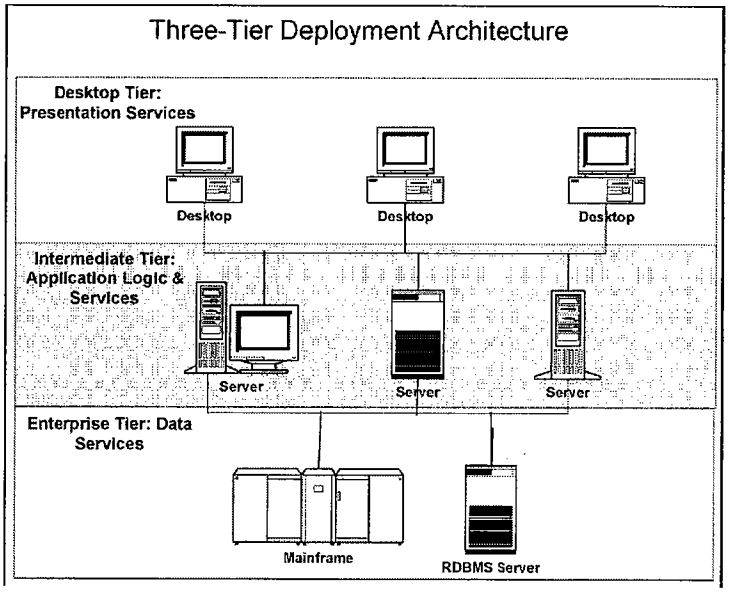
Accused Oracle Products	Releases at Issue	'554 Patent Claims Infringed	'335 Patent Claims Infringed
Accused Oracle Web Cache Products	Release 1.0.2 (November 2000) and all subsequent releases	1-5, 7-11	2 and 16
Accused Oracle Application Server Products	10gR1 (9.0.4) (April 2003) and all subsequent releases	1-5, 7-11	2 and 16
Accused Oracle Database Products	for RAC and JDBC: 10gR2 (10.2.0.1.0) (May 2005) and all subsequent releases; for RAC and OCI: 11g (11.1) (October 2007) and all subsequent releases	1-3, 7-11	2 and 16

A summary of the accused Oracle products and epicRealm's shifting infringement theories is set forth in Oracle's Motion for Summary Judgment of Non-infringement, filed herewith.

E. The Subject Technology

The technology at issue in this case pertains to systems and methods for managing dynamic web page requests sent and received over the Internet. The Internet was widely available for commercial use by at least the mid-1990s. The basic three-tier client/server architecture employed by the Internet, however, was known and used long before then. Tier 1 of the three-tier architecture, known as the "desktop" or "presentation" tier, is comprised of a client program (web browser) located on a user's desktop computer which sends and receives requests for information over the Internet and displays the information on the user's computer. Tier 2,

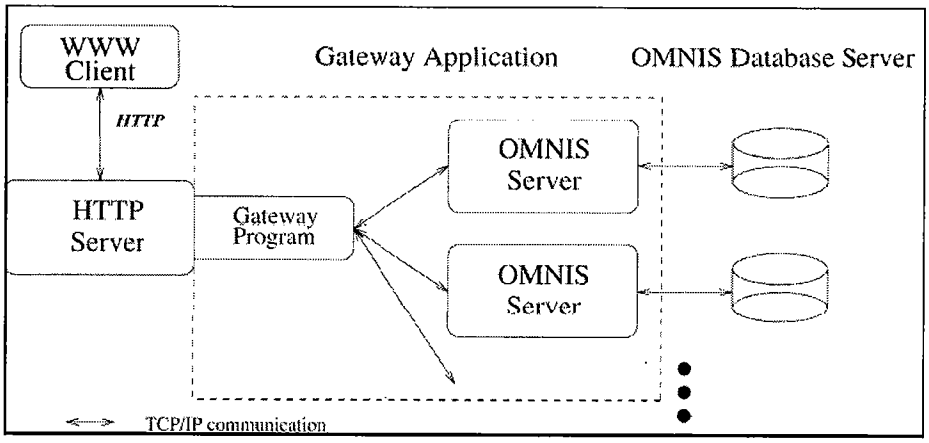
known as the "middle" or "intermediate" tier, is comprised of one or more web servers which receive and process requests sent by the client over the Internet and return completed web pages to the client for viewing. Tier 3, known as the "data" tier, is comprised of one or more back-end database servers which store the information used to make up web pages. The basic three-tier architecture was depicted in, for example, W. Eckerson, "Three Tier Client/Server Architecture: Achieving Scalability, Performance, and Efficiency in Client Server Applications" (January, 1995). A11.



Web pages are comprised of either "static" or "dynamic" information. Static web pages contain relatively fixed content that changes little over time and can be stored in memory in the web server itself for easy and efficient access. An example of a static web page is the home page of a company providing basic information regarding the company's address, telephone number and product offerings. Dynamic web pages contain content that changes frequently over time, such as the price of a stock, or that changes based upon the identity of the requesting party, such as a user's shopping cart at a retail website.

To maintain acceptable response times, it was well known by the mid-1990s to utilize

more than one server in the middle tier to process dynamic web page requests. In such a "partitioned architecture" configuration, instead of the web server handling all of the processing of dynamic web page requests itself, it will off-load them to one or more page-generating servers in order to lighten the processing load on the web server and release it to process additional requests. Such a system was described in, for example, A. Clausnitzer, et al., "A WWW Interface to the OMNIS/Myriad Literature Retrieval Engine" (April, 1995). A12.[6]



As depicted in Clausnitzer, dynamic web page requests are diverted from the "HTTP Server" (web server) to one or more "OMNIS Servers" (page servers) within the middle tier so that these page servers, rather than the HTTP Server (web server), processes the requests. The OMNIS server, in turn, is connected to a database server from which the OMNIS server dynamically retrieves query results and generates the web page for ultimate transmission back to the client. *Id.* at p. 1023.

---

[6] The Clausnitzer article, along with other prior art references, was recently cited by the examiner in rejecting all of the claims of the '554 patent in the ongoing reexamination proceedings before the USPTO. As stated by the examiner, the architecture disclosed in Clausnitzer is "virtually identical" to the partitioned architecture disclosed and claimed in the patents-in-suit. A10, at pp. 20-21.

F. The Patents-In-Suit[7]

The parent application for the patents-in-suit was filed on April 23, 1996. The '554 patent issued on April 13, 1999, and the '335 patent, a division of the '554 patent which shares a common specification, issued on July 2, 2002. The patents-in-suit purport to cover a computer-implemented method and apparatus for managing dynamic web page requests using a partitioned architecture. Requests for dynamic web pages are intercepted at a web server and diverted to a dispatcher, which in turn routes them to a page server for processing, and the page server releases the web server to concurrently process additional requests. A1 ('554 patent: "System for Managing Dynamic Web Page Generation Requests by Intercepting Request at Web Server and Routing to Page Server Thereby Releasing Web Server to Process Other Requests.") Figure 4 of the '554 patent illustrates the basic architecture:

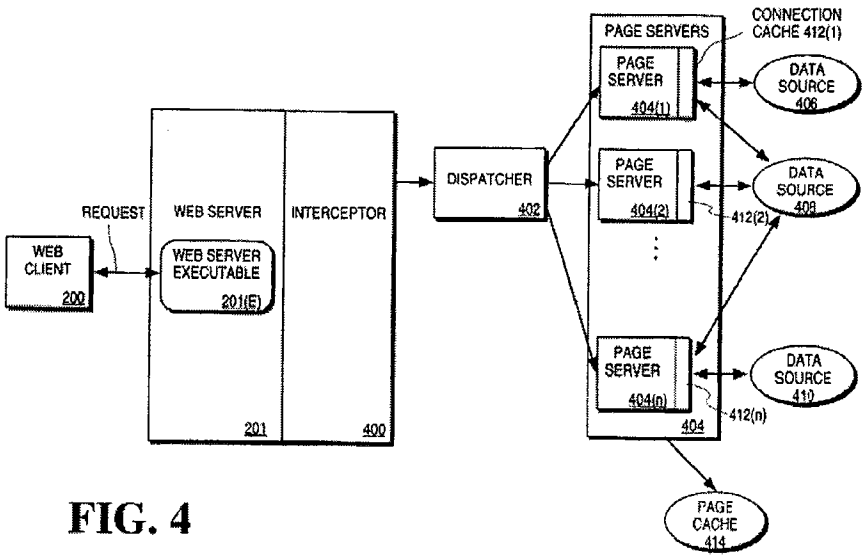


FIG. 4

[7] Copies of the patents-in-suit can be found at A1 and A2. Excerpts from the file histories can be found at A3 and A4. Because the patents contain a common specification, all citations are to the '554 patent.

The specification describes the prior art web server architecture as lacking "any mechanism for managing the Web requests of the Web Sites." The specification states that "[a]lthough these [web server] machines may be running 'multi-threaded' operating systems that allow transactions to be processed by independent 'threads,' all the threads are nevertheless on a single machine, sharing a processor." The specification further describes the object of the purported invention: "When numerous requests are being simultaneously processed by multiple threads on a single machine, the Web server can slow down significantly and become highly inefficient. The claimed invention addresses this need by utilizing a partitioned architecture[8] to facilitate the creation and management of custom Web sites and servers." Col. 4:48-53.

To fulfill the objective of the purported invention, the specification teaches that once a dynamic web page request is off-loaded from the web server (through the dispatcher) to a page server for processing, the selected page server "receive[es] the request and release[es] the Web server to process other requests." Col. 2:20-27 (Summary of the Invention). The specification teaches that this "releasing" step by the page server increases efficiency. *See, e.g.* Col. 5:12-30 ("By routing the request to Dispatcher 402 residing on a different machine than the Web server executable 201(E), the request can then be processed by a different processor than the Web server executable 201(E). Web server executable 201(E) is thus free to continue servicing client requests on Web server 201 while the request is processed 'off-line,' at the machine on which Dispatcher 402 resides."); 6:22-28 ("While Page server 404(2) is processing the request, Web server executable 201(E) can concurrently process other Web client requests. This partitioned

---

[8] The "partitioned architecture" described in the patents-in-suit refers to the off-loading of dynamic web page requests from a web server to a page server, much as depicted in Clausnitzer, *supra*.

architecture thus allows both Page server 404(2) and Web server executable 201(E) to simultaneously process different requests, thus increasing the efficiency of the Web site.").

Claim 1 of the '554 patent exemplifies the asserted claims of the patents-in-suit:

> 1. A computer-implemented method for managing a dynamic Web page generation request to a Web server, said computer-implemented method comprising the steps of:
>
> routing said request from said Web server to a page server, said page server receiving said request and releasing said Web server to process other requests, wherein said routing step further includes the steps of intercepting said request at said Web server, routing said request from said Web server to a dispatcher, and dispatching said request to said page server;
>
> processing said request, said processing being performed by said page server while said Web server concurrently processes said other requests; and
>
> dynamically generating a Web page in response to said request, said Web page including data dynamically retrieved from one or more data sources.

As described in the claims, in order for the page server to receive a request for a dynamic web page and release the web server, the request must be "intercepted at said web server."[9] This "intercepting" requires a diverting of the dynamic web page request so that the web server executable software will not process it. *See, e.g.*, Col. 4:57-60 ("[T]he request is routed to Web server 201. *Instead of Web server executable* 201(E) processing the URL request,[10] however, Interceptor 400 intercepts the request and routes it to Dispatcher 402.") (emphasis added). The Interceptor (or web server) must make a decision to divert the dynamic web page request rather than allow the web server executable software to process it. The Interceptor or web server

---

[9] All the asserted claims require "intercepting." Claim 10 of the '554 patent specifically requires that an "interceptor" perform the "intercepting." The other asserted claims only require that the intercepting be done "at the web server" (or at the "second computer system" or "HTTP-compliant device," which acts as a web server).

[10] A "URL" is a "Uniform Resource Locator," which is a string of characters used to represent a location or resource available on the Internet, e.g., http://ded.uscourts.gov/Index.htm, which is the website for this Court.

cannot merely route whatever requests it receives to the Dispatcher without some decision, based on the URL, that the request will be diverted to the Dispatcher. When prosecuting the '335 patent, for example, the inventors specifically told the Patent Office that the "mere routing" of a request from a web server does not involve interception. EpicRealm made this representation in an attempt to overcome the examiner's rejection of the claims as anticipated by Leaf, U.S. Patent No. 5,754,772. The inventors stated:

> Leaf does not teach or suggest 'intercepting said request.' Instead Leaf teaches that the web server routes the request directly to the transaction gateway client. Leaf, col. 4, lines 55-57. *Leaf does not teach or suggest 'intercepting said request at said Web server' because merely routing a request from a web server to the transaction gateway does not involve interception.*

A4, at EPIC000497 - 498 (emphasis added).

Once a request for a dynamic web page has been intercepted and diverted to the dispatcher, the dispatcher then routes the request to a page server, which both receives the request and then releases the web server to process other requests. The page server processes the request and generates a dynamic web page, which includes data dynamically retrieved from one or more data sources, while the web server concurrently processes other requests. Col. 4:54-6:32.

As demonstrated in Oracle's Motion for Summary Judgment of Noninfringement, Oracle's accused products do not employ the functionality claimed in the patents-in-suit and therefore do not infringe. While Oracle's products employ the basic three-tier client/server architecture long known and used in the Internet, they utilize modern memory and processing systems which do not require or use the outdated functionality recited by the patents-in-suit for managing dynamic web page requests. For example, Oracle's products do not have a page server that "releases" a web server to concurrently process other requests. The two structures that epicRealm alternatively labels "web servers" in Oracle products (Web Cache and OHS) both

have a large capacity for simultaneously processing numerous requests for dynamic web pages on separate "fibers" or "threads" and each of those multiple fibers or threads waits until the web page is returned from the data source. Neither the alleged "web server" nor any fiber or thread on the alleged web server is ever "released" by what epicRealm labels a "page server" to concurrently process other requests. Both Web Cache and OHS continue processing other requests regardless of the action, or inaction, of a "page server."

Nor do the accused Oracle products perform the function of "intercepting" dynamic web page requests at the web server and routing them to a "dispatcher." Oracle products do not "intercept" or "divert" requests to a "dispatcher" based on the nature of the request (*e.g.*, based on the URL). Rather, Oracle's accused "web servers" process *all* requests, whether or not they are for dynamic web pages, in the *same* way. Nor do Oracle products have an "interceptor" that "routes" dynamic web page requests to a "dispatcher" or a "dispatcher" that "dispatches" dynamic web page requests to a "page server." These and other fundamental differences between the patented technology and the accused Oracle products are described more fully in Oracle's Motion for Summary Judgment of Noninfringement, filed herewith.

IV. LEGAL STANDARDS FOR CLAIM CONSTRUCTION

Patent infringement analysis entails two essential steps. First, the meaning and scope of the asserted claims are determined, and second, the properly construed claims are compared to the accused product. *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 976 (Fed. Cir. 1995) (en banc), *aff'd* 517 U.S. 370 (1996). The first step in the analysis, claim construction, is a question of law for the court to decide. *Id.*

"Claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims, for use in the determination of infringement." *United States Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554,

1568 (Fed. Cir. 1997). Claim construction begins with the words of the claims. *Vitronics Corp. v. Conceptronic, Inc.* 90 F.3d 1576, 1582 (Fed. Cir. 1996). Claim terms are generally given their ordinary and customary meaning. *Id.* 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." *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc). "To begin with, the context in which a term is used in the asserted claim can be highly instructive." *Id.* at 1314. The person of ordinary skill in the art views the claim term in light of the entire intrinsic record which includes the claims, the specification and the file history. *Id.* Thus, the claims "must be read in view of the specification, of which they are a part." *Markman*, 52 F.3d at 979; *Medrad, Inc. v. MRI Devices Corp.*, 401 F.3d 1313, 1319 (Fed. Cir. 2005) ("We cannot look at the ordinary meaning of the term…in a vacuum. Rather, we must look at the ordinary meaning in the context of the written description and prosecution history.").

The claims themselves provide substantial guidance in determining the proper construction to give claim terms. *Phillips*, 415 F.3d at 1314. The context in which a term is used in a claim, as well as the claim language of other asserted and unasserted claims, can aid in determining the proper meaning and scope of the claims. *Id.* For example, when a dependent claim recites an additional limitation to the independent claim, it is presumed that the independent claim does not contain that limitation. *Id.* at 1314-15. A claim should not be construed to render a claim limitation superfluous. "[C]laims are interpreted with an eye toward giving effect to all terms in the claim." *Bicon, Inc. v. Straumann Co.*, 441 F.3d 945, 950 (Fed. Cir. 2006).

Claims must also be read in view of the specification. Because the patent laws require an inventor to provide a "full" and "exact" description of the claimed invention, the specification necessarily informs the proper construction of claims. *Id.* at 1316. "The construction that stays

true to the claim language and most naturally aligns with the patent's description of the invention will be, in the end, the correct construction." *Id.*; *Vitronics*, 90 F.3d at 1582 (The 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."). This is true because the patentee may define terms in the specification, provide a different meaning than a term may otherwise have, or disavow claim scope. In these situations, an inventor's lexicography governs. *Phillips*, 415 F.3d at 1316. Although claims must be interpreted in light of the specification of which they are a part, the court must be careful not to read particular embodiments of the invention into the claims. *Id.* at 1324.

In addition to the written description (specification) of the patented invention, "the prosecution history can often inform the meaning of the claim language by demonstrating how the inventor understood the invention and whether the inventor limited the invention in the course of prosecution, making the claim scope narrower than it would otherwise be." *Phillips*, 415 F.3d at 1317.

In discerning the proper construction to be given claim terms, the court may also consult evidence external to the patent documents, including expert and inventor testimony, dictionaries and learned treatises. *Id.* However, extrinsic evidence is viewed as "less reliable" than the patent and its file history in construing claims. *Id.* at 1318. "Undue reliance on extrinsic evidence poses the risk that it will be used to change the meaning of claims in derogation of the 'indisputable public records consisting of the claims, the specification and the prosecution history, thereby undermining the public notice function of patents.'" *Id.* at 1319 (citations omitted).

Where a claim limitation is drafted in "means-plus-function" language and does not recite definite structure in support of its function, the limitation is subject to 35 U.S.C. §112, ¶6, which

provides that "such a claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof." To construe a means-plus-function claim limitation, the court must first ascertain the limitation's function and then determine the corresponding structure disclosed in the specification and equivalents thereof. *Medtronic, Inc. v. Advanced Cardiovascular Sys., Inc.*, 248 F.3d 1303, 1311 (Fed. Cir. 2001). To qualify as corresponding structure under §112, ¶6, it must be "clearly linked or associated" with the recited function not merely whether it is capable of performing that function.

V. ORACLE'S PROPOSED CLAIM CONSTRUCTIONS

Oracle's proposed constructions are discussed herein by way of groupings into five categories: (A) "releasing" terms; (B) "intercepting" terms; (C) "dispatching" terms; (D) general web terms; and (E) means-plus-function terms. *See* A37.

A. Releasing Terms

1. Releasing

Term	Oracle's Construction	epicRealm's Construction
Releasing ('554 patent – claims 1, 9, 11) ('335 patent – claims [1], [15])	said page server receiving said request and said page server performing an act (separate from merely receiving the request) to free the Web server to process other requests.	Freeing

2. Said page server receiving said request and releasing said Web server [second computer system or HTTP-compliant device] to process other requests

Term	Oracle's Construction	epicRealm's Construction
said page server receiving said request and releasing said Web server [second computer system or HTTP-compliant device] to process other requests ('554 patent –	said page server receiving said request and said page server performing an act (separate from merely receiving the request) to free the Web server to process other requests.	[EpicRealm does not construe the entire phrase, but proposes that "releasing should be construed as: Freeing.]

Term	Oracle's Construction	epicRealm's Construction
claims 1, 9, 11) ('335 patent – claims [1], [15])		

The term "releasing" is used in all of the asserted claims. Claim 1 of the '554 patent, which is exemplary of the other claims, recites: "...routing said request from said Web server to a page server, said page server receiving said request and releasing said Web server to process other requests..."[11] The claims recognize two separate acts occurring at the page server: 1) receiving the request; and 2) releasing the web server to process other requests. Oracle's proposed construction reflects this claim language, while epicRealm's construction obfuscates the distinction.[12]

The specification supports Oracle's construction. The specification describes the page server's act of releasing the web server to process other dynamic web page requests as central to the purported efficiency the claimed invention provides over the prior art. Col. 2:21-34 (Summary of the Invention); 4:48-53 ("When numerous requests are being simultaneously processed by multiple threads on a single machine, the Web server can slow down significantly and become highly inefficient. The claimed invention addresses this need by utilizing a partitioned architecture to facilitate the creation and management of custom Web sites and servers."); 6:21-28 ("While Page server 404(2) is processing the request, Web server executable 201(E) can concurrently process other Web client requests. This partitioned architecture thus

---

[11] Other asserted claims require that the page server release a "second computer system" ('554 claims 9-10) or an "HTTP-compliant device" ('335 claim 16). Such "system" or "device" operates as web server because, for example, it receives web page requests and returns web pages in response to those requests. EpicRealm accuses the same Oracle components as being "web servers," "second computer systems," and "HTTP-compliant devices."

[12] Oracle's construction is identical to that adopted by the Texas court. A33.

allows both Page server 404(2) and Web server executable 201(E) to simultaneously process different requests, thus increasing the efficiency of the Web site."). To accomplish the intended purpose of the claimed invention, the specification makes clear that after receiving the request, the page server then releases the web server to process other requests. *See* 2:25-26 ("the page server receiving the request *and* releasing the Web server to process other requests") (emphasis added); Abstract, line 8 (same). Oracle's and the Texas court's construction, which makes clear that the page server performs an act – separate from the mere receiving of the request – to release the web server is consistent with, and supported by, the specification. *See* Declaration of Dr. Paul C. Clark in Support of Oracle's Opening Claim Construction Brief and Oracle's Motion for Summary Judgment of Noninfringement ("Clark Decl."), ¶¶32-34.

EpicRealm's proposed construction, on the other hand, is vague and does no more than substitute one unconstrued word ("freeing") for another ("releasing"). It does not resolve "disputed meanings and technical scope" or "clarify" or "explain what the patentee covered by the claims, for use in the determination of infringement," as required by the Federal Circuit. *United States Surgical,* 103 F.3d at 1568; *see also O2 Micro Int'l Ltd. v. Beyond Innovation Tech. Co. Ltd.*, 521 F.3d 1351, 1360-61 (Fed. Cir. 2008) ("When parties raise an actual dispute regarding the proper scope of these claims, the court, not the jury, must resolve that dispute.") (*citing Markman*, 52 F.2d at 979).

Before epicRealm realized its problems in contending that any alleged Oracle page server "released" any alleged Oracle web server, epicRealm's construction, proffered in the Texas action, was "freeing *a process to service other requests*." A32, at p. 11 (emphasis added). EpicRealm understood then that the page server must perform some act to release "a process" (e.g., web server) to "service other requests." In an effort to overcome a fatal flaw in its infringement case, EpicRealm is now trying to use the lone word "freeing" in place of

"releasing" to argue that when the web server offloads the request to the page server, the web server is somehow implicitly "freed" to process other requests because it is no longer responsible for processing the offloaded request. While even under this strained construction of "releasing" Oracle's products do not infringe, epicRealm's construction is wrong.[13] As the Texas court held:

> "Releasing" is used in each claim as part of the phrase "said page server receiving said request and releasing said Web server to process other requests." It is therefore the Page server that does the releasing. The larger context of the use of releasing in the claims themselves indicates that the Page server has a role in the releasing as in the claims themselves it is the Page server that releases the Web server. This language also conforms to the Summary of Invention and Abstract. 2:25-26; Abstract, line 8.

A33, at p. 28. As the Texas court's construction properly reflects, the page server must perform two separate acts: (1) receive the request; *and* (2) release the web server to process other requests. *See* 2:25-26 ("the page server receiving the request *and* releasing the Web server to process other requests") (emphasis added); Abstract, line 8 (same).

EpicRealm's construction would effectively render the "releasing" limitation superfluous and read it out of the claim. Because the claims already require that the page server "receive" the request, if "releasing" simply means implicitly "freeing" the web server to process other requests – without the performance of some act on the part of the page server – then the mere act of "receiving" would also satisfy the act of "releasing" and the claim would have the same scope as if the "releasing" limitation did not appear all. The Federal Circuit has held such a construction

---

[13] As set forth in Oracle's non-infringement summary judgment motion, the alleged "page servers" of the accused Oracle products do not "release" the alleged web servers to process other requests. Instead, the structures that epicRealm labels "web servers" in accused products (Web Cache and OHS) are capable of simultaneously processing numerous requests for dynamic web pages on separate "fibers" or "threads" and each of those multiple fibers or threads waits until the web page is returned from the data source. Neither the alleged "web server" nor any fiber or thread on the alleged web server is ever "released" by what epicRealm labels a "page server."

improper:

> Allowing a patentee to argue that physical structures and characteristics specifically described in a claim are merely superfluous would render the scope of the patent ambiguous, leaving examiners and the public to guess about which claim language the drafter deems necessary to his claimed invention and which language is merely superfluous, nonlimiting elaboration. For that reason, claims are interpreted with an eye toward giving effect to all terms in the claim.

Bicon, 441 F.3d at 950. The construction proposed by Oracle, and adopted by the Texas court, is the only one supported by the claims and specification. EpicRealm's construction – "freeing" – does little, if anything, to clarify the meaning of the claim term. *U.S. Surgical,* 103 F.3d at 1568 ("Claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims, for use in the determination of infringement.").

Oracle's proposed construction is also supported by the file history. The inventors obtained the patents by using the importance of the page server's act of "releasing" the web server to distinguish over the prior art. For example, in responding to the examiner's rejection of the claims of the '554 patent in view of the Barbari and Goldberg references, the inventors clarified that these references did not teach "receiving a request on a page server ***and*** releasing the Web server to process other requests, as claimed." A3, at pp. EPIC000265-266 ("a page server according to one embodiment of the presently claimed invention intercepts Web request from a Web client and appropriately routes the requests *after releasing the Web server to service other requests*") (emphasis added). Thus, in distinguishing the purported invention over the prior art, the inventors not only made clear that it is the page server that releases the web server to process other requests, but that the page server's act of "releasing" the web server is separate from the page server "receiving" the request.

The inventors further clarified the meaning and scope of the purported invention in

responding to the examiner's rejection of the claims in view of the Rogers reference during prosecution of the '335 patent:

> Rogers involves organizing distributed sub-agents as distributed integration solution servers to support a web server. [citation omitted] Rogers also discusses a means for accepting web client requests for information, obtaining data from one or more databases and presenting that information to the web clients. [citation omitted] At no time does Rogers teach or suggest 'concurrently' processing other requests or '*releasing said Web server to process other requests*' because merely retrieving data from multiple sources does not teach or suggest these elements.

A4, at pp. EPIC000551 (emphasis added). In distinguishing the claimed invention over Rogers, the inventors again made clear that the page server not only receives the request but also performs a separate act of "releasing [the] Web server to process releases other requests." EpicRealm's proposed construction in which the word "freeing" is substituted for the word "releasing," without more, does not adequately or properly describe the purported invention as claimed. *Bicon*, 441 F.3d at 950 ("[C]laims are interpreted with an eye toward giving effect to all terms in the claim."); *Medrad*, 401 F.3d at 1319 ("We cannot look at the ordinary meaning of the term…in a vacuum. Rather, we must look at the ordinary meaning in the context of the written description and prosecution history.").

B. Intercepting Terms

1. Interceptor

Term	Oracle's Construction	epicRealm's Construction
Interceptor ('554 patent – claim 10)	a software component that performs intercepting	no proposed construction

2. Intercepting said request at said Web server

Term	Oracle's Construction	epicRealm's Construction
intercepting said request at said Web server	receiving a request at the Web server machine and diverting the request before the Web server executable can process the request	intercepting the handling of a request at a Web server

Term	Oracle's Construction	epicRealm's Construction
('554 patent – claims 1, 11) ('335 patent – claims [1])		

3. Intercepting said request at said second computer system

Term	Oracle's Construction	epicRealm's Construction
intercepting said request at said second computer system ('554 patent – claims 9, 10)	receiving a request at the second computer system and diverting the request before the second computer system executable can process the request	no proposed construction

4. Intercepting said request at said HTTP-compliant device

Term	Oracle's Construction	epicRealm's Construction
intercepting said request at said HTTP-compliant device ('335 patent – claim [1], [15])	receiving a request at the HTTP-compliant device and diverting the request before the HTTP-compliant device executable can process the request	at least intercepting the handling of a request at a said HTTP-compliant device

The term "intercepting" appears in all of the asserted claims of the patents-in-suit. The term "interceptor" appears in '554 claim 10. Claim 1 of the '554 patent exemplifies how the term "intercepting" is used in other claims. Claim 1 recites that dynamic web page requests are routed from the web server to a page server, "wherein said routing step further includes the steps of intercepting said [dynamic web page] request at said Web server,[14] routing said request from said Web server to a dispatcher, and dispatching said request to said page server." Col. 8:63-9:5.

Consistent with how the term is used in the claims and specification, Oracle's proposed

[14] Various claims of the patents-in-suit use the terms "web server," "HTTP-compliant device," and "said second computer" to refer to HTTP-compliant server software, or a machine running such software, that receives web page requests and returns web pages in response. *See* discussion *infra* re "web server."

construction states that the dynamic web page request must be received at the web server (second computer system/HTTP-compliant device) and "diverted" before the web server executable can process it. As explained above with respect to "releasing," the patent specification focuses on the importance of the page server receiving a dynamic web page request and releasing the web server to process other requests. This releasing cannot take place unless the dynamic web page request is diverted away from the web server and sent on to the page server so the web server executable does not have to process the request. Oracle's proposed construction – receiving a request at the Web server machine and diverting the request before the Web server executable can process the request – captures this central aspect of the claimed invention.

For example, the specification provides that a request from a client for a dynamic web page – as opposed to a static web page request – is routed to a web server (201) for processing. "*Instead of the Web server executable (201E) processing the URL request, however, Interceptor (400) intercepts the request and routes it to Dispatcher (402).* In one embodiment, Interceptor (400) resides on the Web server machine as an extension to Web server (201)." Col. 4:55-62 (emphasis added). As the specification makes clear, the request is diverted from the web server so the web server executable software does not process it, and instead the request is sent on to the dispatcher which in turn dispatches it to a page server.[15]

The specification goes on to state (referencing Fig. 5): "A Web browser sends a URL request to a Web server in processing block (500). In processing block (502), the Web server receives the URL request, and an interceptor then intercepts the handling of the request in processing block (504). The interceptor connects to a dispatcher and sends the URL request to

---

[15] Oracle's construction is in accord with a standard dictionary definition of "intercepting" which defines "intercept" as "[t]o stop, deflect, or interrupt the progress or intended course of." A13.

the dispatcher in processing block (506)." Col. 8:28-34. "By routing the request to Dispatcher (402) residing on a different machine than *the Web server executable (201E), the request can then be processed by a different process than the Web server executable* (210E). *Web server executable (201E) is thus free to continue servicing client requests* to Web server (201) while the request is process 'off-line,' at the machine on which Dispatcher (402) resides." *Id.* at 5:12-19 (emphasis added).

Thus, consistent with Oracle's proposed construction, the specification makes clear that a dynamic web page request is received at the web server machine (or HTTP-compliant device/said second computer system) and diverted to the dispatcher before the web server executable can process the request. *See* Clark Decl., ¶¶35-39. In this manner, "the request can be processed by a different processor than the Web server executable" and the "Web server executable is thus free to continue servicing client requests." Col. 5:12-19.

The inventors' representations during prosecution of the patents-in-suit also support Oracle's construction. During prosecution of the '335 patent, the inventors attempted to overcome the examiner's rejection of the claims as anticipated by Leaf, U.S. Patent No. 5,754,772, stating:

> Leaf does not teach or suggest "intercepting said request." Instead Leaf teaches that the web server routes the request directly to the transaction gateway client. Leaf, col. 4, lines 55-57. Leaf does not teach or suggest "intercepting said request at said Web server" because *merely routing a request from a web server to the transaction gateway does not involve interception.*

A4, at EPIC000497-498 (emphasis added).

As the inventors made clear, the purported invention does not merely "route" or pass on requests for dynamic web pages from the web server to the dispatcher. Rather, the web server (or an "interceptor" associated with the web server) makes a determination that the request is for a dynamic web page (as opposed, for example, to a request for a static web page) that the web

server executable software will not itself process and then diverts "said request" and routes it to the dispatcher before the web server executable actually processes the request. In this way, the "partitioned architecture" of the purported invention can fulfill the objective the off-loading dynamic web page requests from the web server to the page server (via the dispatcher) so the page server can release the web server to take on additional requests. Col. 2:21-32 (Summary of the Invention) (dynamic web page requests are routed "from the Web server to a page server, the page server receiving the request and releasing the Web Server to process other requests"); 4:38-54 ("When numerous requests are being simultaneously processed by multiple threads on a single machine, the Web server can slow down significantly and become highly inefficient. The claimed invention addresses this need by utilizing a partitioned architecture to facilitate the creation and management of custom Web sites and servers."); 5:12-30 (the "advantage" of the purported invention described as "off-loading the processing of Web requests from the Web server machine").

EpicRealm's proposed construction, in which "intercepting" means "intercepting," does nothing to clarify the meaning of the term. EpicRealm is seeking to obtain an ambiguous claim construction that it hopes will read on the accused Oracle products. However, before it understood that Oracle's products do not divert any requests from the alleged "web servers" but instead process all of them the same way (*see* Oracle's Motion for Summary Judgment of Noninfringement), epicRealm proposed a construction of "intercepting" in the Texas action – "stopping, deflecting or interrupting the processing of a request" – which included the "deflecting" requirement that is substantially the same as the "diverting" requirement in Oracle's construction. A32, at p. 8; *see also* A71, at p. 1 of 12 (epicRealm's Preliminary Claim Constructions in Texas, defining "intercepting" as "receiving and *diverting* a request") (emphasis added). Realizing it could not prove infringement by the accused Oracle products under such a

construction, epicRealm dropped that construction and now proposes a construction in which "intercepting" means nothing more than "intercepting."[16]

EpicRealm's circular construction provides no insight or clarity into the meaning of this key claim term. Presumably, epicRealm will contend that the plain and ordinary meaning of the term "intercepting" should apply and that no further construction is necessary. This approach, however, was recently rejected by the Federal Circuit in *O2 Micro*. In that case, the district court declined to construe the term "only if," holding that the ordinary and customary meaning of the claim term should apply. The Federal Circuit, however, disagreed: "When the parties raise an actual dispute regarding the proper scope of these claims, the court, not the jury, must resolve that dispute." *O2 Micro,* 521 F.3d at 1360-61 (*citing Markman*, 52 F.2d at 979). The court went on to state: "A determination that a claim term 'needs no construction' or has the 'plain and ordinary meaning' may be inadequate when a term has more than one 'ordinary' meaning or when reliance on a term's 'ordinary' meaning does not resolve the parties' dispute." *Id.* Here, the parties dispute the meaning and scope of the term "intercepting." Because the term may have more than one "ordinary" meaning, this Court must resolve the dispute and construe the term as a matter of law, and not leave it up to the jury to decide. *O2 Micro,* 521 F.3d at 1360-61.[17]

---

[16] EpicRealm's position is reminiscent of the conversation between Alice and Humpty Dumpty in *Through the Looking Glass*: "When I use a word," Humpty Dumpty said, in a rather scornful tone, "it means just what I choose it to mean - neither more nor less." "The question is," said Alice, "whether you can make words mean so many different things."

[17] The Texas court construed "intercepting" as "at least intercepting." A33, at pp. 20-25. [Tex claim construction order] Regardless of how the Texas court may have viewed the claim construction disputes in that case, such a construction would clearly not resolve the dispute over the meaning of the term in this case. Moreover, the Texas construction was issued before *O2 Micro*, which mandates that the term be construed in light of the present dispute between the parties in this case.

C. Dispatching Terms

1. Dispatcher

Term	Oracle's Construction	epicRealm's Construction
Dispatcher ('554 patent – claim 1, 9, 11) ('335 patent – claims 2, 16)	a software program for determining which page server should be used to process a dynamic web page generation request	no proposed construction

2. Dispatching / Dispatching said request to said page server

Term	Oracle's Construction	epicRealm's Construction
Dispatching / dispatching said request to said page server ('554 patent – claims 1, 9, 11) ('335 patent – claims 2, 16)	analyzing a request to make an informed selection of which page server should process the request, and sending the request to that page server	examining a request to make an informed selection of which page server should process the request based on dynamic information maintained about page servers, the dynamic information indicating which page server can more efficiently process the request, and sending the request to the selected page server.

The terms "dispatcher" and "dispatching" appear in all of the asserted claims. For example, claim 1 of the '554 patent recites the steps of: "intercepting said [dynamic web page] request at said Web server, routing said request from said Web server to a dispatcher, and dispatching said request to said page server." Col. 9:2-5. As shown below, although the "dispatcher" is the software component that performs the act of "dispatching," the two terms are separate claim elements that require separate constructions.

3. The "Dispatcher" of the Claims is a Software Program Distinct From Other Required Elements of the Claims

The "dispatcher" as used in the claims is a separate software component or program that determines which page server should process a dynamic web page request. The claims all require that a "dispatcher" both receive requests "routed" from another required software

program or device – e.g., a "web server" or "interceptor" – and "dispatch" (send) the request to the required page server software program.[18] For example, claim 1 of the '554 patent requires "routing said request from said Web server to a dispatcher, and dispatching said request to said page server." Similarly, '554 patent claim 10 requires "a dispatcher receiving said routed request from said interceptor and dispatching said request to said page server." The context in which "dispatcher" is used is highly instructive. *Phillips*, 415 F.3d at 1314. That context unequivocally requires a "dispatcher" element that is separate from the web server [19] and/or interceptor elements because the "dispatcher" must receive requests routed *from* that other element.

The specification supports Oracle's construction. Fig. 4 shows the dispatcher (402) as separate from the interceptor (400) and the web server (201), which includes the web server executable 201(E), a software program. The specification explains:

> In one embodiment of the invention, Dispatcher 402 resides on a different machine than Web server 201. This embodiment overcomes the limitation described above, in prior art Web servers, wherein all processing is performed by the processor on a single machine. By routing the request to Dispatcher 402 residing on a different machine than the Web server executable 201(E), the request can then be processed by a different processor than the Web server executable 201(E). Web server executable 201(E) is thus free to continue servicing client requests on Web server 201 while the request is processed "off-line," at the machine on which Dispatcher 402 resides.

---

[18] A "dispatcher" is defined in *The American Heritage Dictionary*, 3d Ed. (1996) as a separate software program or "routine that controls the order in which input and output devices obtain access to the processing system." "Dispatching" is defined as: "The act of sending off, as to a specific destination." A13.

[19] The parties agree, in part, on the construction of web server, to the extent that the term can refer to either "software that receives Web page requests and returns Web pages in response to the requests" or to the "machine" on which that software is running. As discussed below, and as provided by the specification, even if the "dispatcher" were on the same machine as the "web server," it still must be a separate program from the web server software, which the patents call the "Web server executable" and which receives web pages and returns web pages in response.

Col. 5:8-19. This description shows that the claimed dispatcher is a separate software program from the web server executable, or any other software residing on the web server machine, because the dispatcher is on a separate machine and runs on a separate processor than the web server executable software program.

The specification provides that in another embodiment, "Dispatcher 402 can also reside on the same machine as the Web server." Col. 5:20-21. The specification suggests that placing the web server software program and the dispatcher software program on the same machine may "allow flexibility for a small Web site to grow." *Id.* at 5:29-30. The specification leaves no doubt, however, that the web server software program and dispatcher software program are separate elements: "For example, a small Web site administrator can use a single machine for both Dispatcher 402 and Web server 201 initially, *then off-load Dispatcher 402 onto a separate machine as the Web site grows.*" *Id.* at 5:30-33 (emphasis added). The dispatcher software program could not be "off-loaded" to a separate machine if it were the same program as the web server software. If this alternative embodiment were construed to mean that the dispatcher software program were not separate from the web server software, then it would not be consistent with the language of the claims because the web server could not "route" requests to the dispatcher. Instead, there would only be one software program, which could not, of course, "route" requests to itself. *See* Clark Decl., ¶¶40-43.

EpicRealm offers no construction for "dispatcher." EpicRealm apparently contends that if the function of "dispatching" is present in the accused products, then whatever performs that function must ipso facto be a "dispatcher." But as with epicRealm's construction of "releasing," this construction is wrong because it effectively reads the claimed "dispatcher" limitation out of the claims and does not resolve the dispute between the parties over the proper construction of the claim term. *Bicon*, 441 F.3d at 950; *O2 Micro*, 521 F.3d at 1360-61. It is also wrong

because it contradicts the clear language of the claims, which require that the dispatcher be a separate component that receives a request routed from a web server or interceptor. Thus, for example, even if a web server or interceptor performed the "dispatching" function, it could not also be a "dispatcher" in the context of the claims because it could not "route" a request to itself. *Phillips*, 415 F.3d at 1314. Such an application of the claim would read out the requirement that the request be "routed," i.e., sent from one place to another.[20]

4. EpicRealm's Construction of "Dispatching" Improperly Incorporates Limitations Into the Claim From the Specification

In defining the scope of the claimed invention, the patentee chose the word "dispatching" to define how the "dispatcher" sends requests to the selected page server. For example, in claim 1 of the '554 patent, a dynamic web page request is routed from a web server to a page server, wherein the routing step includes: "intercepting said request at said Web server, routing said request from said Web server to a dispatcher, and dispatching said request to said page server." As set forth above, the act of "intercepting" a web page request first involves the identification of the request as dynamic (as opposed to static) and then diverting "said request" to the dispatcher for processing. As taught by the specification:

> Dispatcher 402 maintains a variety of information regarding each Page server on the network, and dispatches requests based on this information. For example, Dispatcher 402 retains dynamic information regarding the data sources that any given Page server can access. Dispatcher 402 thus examines a particular request and determines which Page servers can service the URL request. Dispatcher 402 then hands off the request to the appropriate Page server.

Col. 5:51-59. Therefore, "dispatching said request to said page server" as used in the claims and as taught by the specification means: "analyzing a request to make an informed selection of

---

[20] To "route" is defined in *The American Heritage Dictionary*, 3d Ed. (1996) as "to send or forward by a specific route." A13.

which page server should process the request, and sending the request to that page server." *See* Clark Decl. ¶¶40-43.

While Oracle's proposed construction is very similar to the first and last parts of epicRealm's proposed construction,[21] Oracle objects to epicRealm's additional language as unsupported by either the claims or the specification: "...based on dynamic information maintained about page servers, the dynamic information indicating which page server can more efficiently process the request...."[22]

Through its proposed construction, epicRealm is attempting to distinguish over the prior art with an unsupported construction of "dispatching" that improperly reads examples from the specification into the claim – *i.e.,* the term "dispatching," standing alone, necessarily includes the additional limitation of selecting which page server will process the request "based on dynamic information maintained about page servers, the dynamic information indicating which page server can more efficiently process the request." None of the asserted claims, however, include any such limitation. As discussed further below, the only claim that includes such a limitation is claim 29 of the '335 patent, which epicRealm *dropped* because it does not support its proposed construction.

The specification makes clear that the dispatcher "maintains *a variety of information* regarding each Page server on the network, and dispatches requests based on this information." Col. 5:51-53 (emphasis added). While the specification goes on to offer examples of the type of

---

[21] While the first part of Oracle's construction calls for "*analyzing* a request" and epicRealm's calls for "*examining* a request," there does not appear to be any substantive difference between these terms for purposes of the non-infringement and invalidity arguments.

[22] In the Texas actions, the parties stipulated to the construction that epicRealm proposes here. Thus, the issue was never litigated before the Texas court.

information maintained about page servers, including "dynamic information," neither the claims themselves nor the specification limits the "variety of information" that can be maintained to "dynamic information maintained about page servers, the dynamic information indicating which page server can more efficiently process the request," as proposed by epicRealm. This is a classic example of improperly reading particular embodiments from the specification into the claims. *Phillips*, 41 F.3d at 1324.

If the inventors of the patents-in-suit wanted claims limited to the narrow type of dispatching that epicRealm now proposes, they easily could have included such a limitation in the claim language. In fact, that's exactly what they did with claim 29 of the '335 patent. Claim 29 provides in relevant part:

> 29. A computer-implemented method comprising the steps of:
>
> transferring a request from an HTTP-compliant device to a dispatcher;
>
> ***maintaining dynamic information regarding data sources a given page server may access;***
>
> dispatching said request to an appropriate page server based on said request and ***based on said dynamic information,*** said page server receiving said request and releasing said HTTP-compliant device to process other requests…

'335 patent, claim 29 (emphasis added). Claim 29 contains the identical "dispatcher" and "dispatching" elements present in all the other claims of the '554 and '335 patents. But unlike those claims, it also contains the limitation that the dispatching must be done based on "dynamic information regarding data sources a given page server may access." Thus, in claim 29, the inventors recognized that mere "dispatching" did not require the use of such "dynamic information" and they specifically included it as an additional claim limitation. *Phillips*, 415 F.3d at 1314 ("Other claims of the patent in question, both asserted and unasserted, can also be valuable sources of enlightenment as to the meaning of a claim term. Because claim terms are

normally used consistently throughout the patent, the usage of a term in one claim can often illuminate the meaning of the same term in other claims.").

There is no indication that the inventors considered the general term "dispatching" to have the limited meaning epicRealm now ascribes to it. Both the specification and the claims, including claim 29 of the '335 patent, prove the contrary. The Court should, therefore, adopt Oracle's proposed construction.

D. General Web Terms

1. Page Server

Claim Term	Oracle's Construction	epicRealm's Construction
Page Server ('554 patent – claims 1, 4, 7, 9, 10, 11) ('335 patent – claim [1], [15])	page generating software, running on a processor separate from that of the Web server, that generates dynamic Web pages	page-generating software that generates a dynamic Web page

The only difference between the parties' proposed constructions is the clarification in Oracle's construction that the page generating software "run(s) on a processor separate from that of the web server." Claim 1 of the '554 patent, for example, recites that the dynamic web page request is routed "from said Web server to a page server, said page server receiving said request and releasing said Web server to process other requests..." The claim goes on to recite that processing of the request is performed by the page server "while said Web server concurrently processes said other requests..." As the claims make clear, the request is routed from one process (the web server) to another (the page server). After receiving the request, the page server releases the web server to process other requests. The page server then processes the request while the web server "concurrently processes" other incoming requests. The ordinary and customary meaning of the term, as used in the context of the claims, necessarily requires that the page server run on a separate processor from that of the web server since a single processor

cannot: (1) route requests to itself; (2) release itself to process other requests; and (3) process the original request while concurrently processing additional requests. *See* Clark Decl., ¶¶44-45.

The specification supports this construction. Prior art web servers processed requests "on a single machine, namely the Web server machine. Although these machines may be running 'multi-threaded' operating systems that allow transactions to be processed by independent 'threads,' all the threads are nevertheless on a single machine, sharing a processor. As such, the Web executable thread may hand off a request to a processing thread, but both threads will still have to be handled by the processor on the Web server machine." Col. 4:39-48. The specification teaches that claimed invention makes use of a "partitioned architecture" which purportedly "allows both Page server 404(2) and Web server executable 201(E) to simultaneously process different requests, thus increasing the efficiency of the Web site." *Id.* at 6:24-28. This "partitioned architecture" cannot be achieved if both the web server and page server are running on the same processor. Clark Decl., ¶45. EpicRealm's proposed construction does not make this distinction and only obfuscates the issue. Accordingly, Oracle's proposed construction is the only one consistent with both the claims and the specification and should therefore be adopted.

2. Web Page

Claim Term	Oracle's Construction	epicRealm's Construction
Web Page ('554 patent – claims 1, 3, 6, 7, 9, 11) ('335 patent – claim [1])	a file containing embedded commands in a Web formatting language such as HTML capable of being displayed on a Web browser	Web content displayable through a Web browser

The ordinary and customary meaning of "web page" as applied to the World Wide Web is a "file stored on a Web server that contains formatted text, graphics and hypertext links to other pages on the Internet. A Web page is created using HTML codes and is viewed with a browser."

See A14, at p. 200; *see also* A68, at p. 506. The specification states that web pages "contain hypertext links that are used to connect any combination of graphics, audio, video and text, in a non-linear, non-sequential manner. Hypertext links are created using a special software language known as HyperText Mark-Up Language (HTML)." Col. 1:15-21. As the ordinary and customary meaning of the term suggests, and as taught by the specification, "web page" should be construed as "a file containing embedded commands in a Web formatting language such as HTML capable of being displayed on a Web browser," as proposed by Oracle. *See* Clark Decl., ¶46. EpicRealm's construction, on the other hand, does not provide the necessary clarification regarding the proper scope and meaning of the term. *Phillips*, 415 F.3d at 1314. Oracle's construction should therefore be adopted.

3. Web Server/HTTP-Complaint Device

Claim Term	Oracle's Construction	epicRealm's Construction
Web Server ('554 patent – claims 1, 11) ('335 patent – claim [1])	HTTP-compliant server software, or a machine running such software, that receives Web page requests and returns Web pages in response to the requests	software, or a machine having software, that receives Web page requests and returns Web pages in response to the requests
Http-compliant device ('335 patent – claims 15, 16)	a device running software that implements the Hypertext Transfer Protocol	a device that is compliant with the communication protocol known as HyperText Transport Protocol (HTTP)

While there is little difference between the parties' proposed constructions, Oracle maintains that its constructions are more consistent with the specification. For example, with respect to the term "Web server," the specification states that "Web pages reside on the Web, on Web servers or Web sites…Web client machines running Web browsers can access these Web pages at Web sites via a communications protocol known as HyperText Transport Protocol (HTTP)." Col. 1:23-27. Unlike epicRealm's proposed construction which uses the general term

"software," Oracle's construction more closely aligns with the specification which states that Web clients communicate with Web servers via the HTTP protocol; hence, Web servers run "HTTP-compliant server software," not any old software. There is no disclosure in the specification of any other type of software or protocol used. The parties' proposed constructions for "Http-complaint device" captures this distinction and, Oracle submits, also supports Oracle's construction for "Web server."

4. Request

Claim Term	Oracle's Construction	epicRealm's Construction
Request ('554 patent – all) ('335 patent – all)	a message that asks for a Web page specified by a URL	a message that asks for a Web page

The only difference between the parties' proposed constructions of "request" is that Oracle's explains that it is a message that asks for a Web page "specified by a URL." As the specification makes clear, "A Web browser allows a Web client to request a particular Web page from a Web site by specifying a Uniform Resource Locator (URL)." Col. 1:30-32. The specification states further that a Web browser "sends a URL request to a Web server in processing block 500. In processing block 502, the Web server receives the URL request, and an interceptor then intercepts the handling of the request in processing block 504." *Id.* at 8:28-32. Thus, the specification makes clear that a "request" as used in the claims is a *URL request*, not some other type of request as epicRealm's proposed construction would imply. There is nothing in the specification or file history that teaches otherwise. *See* Clark Decl., ¶¶47-49. Therefore, Oracle's construction should be adopted.

5. General Web Terms for Which EpicRealm Offers No Construction

Oracle has proposed constructions for the following web terms, while epicRealm has proffered no construction for these terms: (1) "connection cache;" (2) "data sources:" (3)

"logging into;" (4) "machine-readable medium;" and (5) "router." Oracle's proposed constructions and support are contained in A15. Because the parties dispute the proper construction of these claim terms, it is incumbent on this Court to construe them as a matter of law. *O2 Micro,* 521 F.3d at 1360-61. Oracle submits that because its proposed constructions are fully supported by the intrinsic and extrinsic record, they should be adopted by the Court.

E. Means-Plus-Function Terms

1. Means for Generating

Claim Term	Oracle's Construction	epicRealm's Construction
Means for generating [said request] ('554 patent – claim 9)	§112 ¶6 corresponding function: generating said request structure: a processor of a computer that is, or has, a Web client running a Web browser	"a processor of a computer that is, or has, a Web client running a Web browser" or equivalents thereof

2. Means for Receiving

Claim Term	Oracle's Construction	epicRealm's Construction
Means for receiving [said request from said first computer] ('554 patent – claim 9)	§112 ¶6 corresponding function: receiving said request from said first computer structure: a processor of a computer that is, or has, a Web server running a Web server executable	"a processor of a computer that is, or has, a Web server running Web server executable" or equivalents thereof

3. Page Server Processing Means

Claim Term	Oracle's Construction	epicRealm's Construction
Page server processing means ('554 patent – claim 9)	§112 ¶6 corresponding function: processing dynamic Web page generation requests corresponding structure: a processor of a computer that runs software for generating dynamic Web pages	"a processor of a computer that runs page server software (wherein page server software is page-generating software that generates a dynamic Web page)" or equivalents thereof

All of the means-plus-function claim terms are contained in asserted claim 9 of the '554 patent. Pursuant to 35 U.S.C. §112, ¶6, the Court must first ascertain the function performed by the respective limitations, then determine the corresponding structure disclosed in the specification that is "clearly linked or associated" with the recited function. *Medtronic*, 248 F.3d at 1311. The parties' proposed claim constructions are virtually identical, except Oracle's constructions first recite the function and then identify the corresponding structure as required under a proper §112, ¶6 analysis.[23] EpicRealm's proposed constructions also append the phrase "or equivalents thereof" to the end of the construction (which is presumed in Oracle's construction). Oracle submits that if the function of the limitation is first recited, followed by the corresponding structure, the resulting construction will be proper (whether or not "and equivalents thereof" is added to the end of the construction).

---

[23] For "page server processing means," there does not appear to be any substantive difference between the parties' recited structure and either one could be selected.

VI. CONCLUSION

For the foregoing reasons, Oracle respectfully requests that its proposed constructions for the patents-in-suit be adopted by the Court.

MORRIS, NICHOLS, ARSHT & TUNNELL LLP

/s/ James W. Parrett, Jr.
Mary B. Graham (#2256)
James W. Parrett, Jr. (#4292)
1201 North Market Street
P.O. Box 1347
Wilmington, DE 19899
mgraham@mnat.com
jparrett@mnat.com
(302) 658-9200

Attorneys for Oracle Corporation and Oracle U.S.A. Inc.

OF COUNSEL:

James G. Gilliland
Theodore T. Herhold
Joseph A. Greco
Robert J. Artuz
Eric M. Hutchins
TOWNSEND AND TOWNSEND AND CREW LLP
379 Lytton Avenue
Palo Alto, CA 94301
(650) 326-2400

Dorian Daley
Peggy E. Bruggman
Matthew M. Sarboraria
ORACLE CORPORATION
ORACLE U.S.A., INC.
500 Oracle Parkway
Redwood Shores, CA 94065

Dated: July 31, 2008

2432443

61437190 v1

CERTIFICATE OF SERVICE

I hereby certify that on July 31, 2008, I caused the foregoing to be electronically filed with the Clerk of the Court using CM/ECF which will send electronic notification of such filing to the following:

Richard L. Horwitz
David Ellis Moore
POTTER ANDERSON & CORROON, LLP

Additionally, I hereby certify that true and correct copies of the foregoing were caused to be served on July 31, 2008 upon the following individuals in the manner indicated:

BY E-MAIL AND HAND DELIVERY

Richard L. Horwitz
David Ellis Moore
POTTER ANDERSON & CORROON, LLP
1313 N. Market St., Hercules Plaza, 6th Floor
P.O. Box 951
Wilmington, DE 19899-0951

rhorwitz@potteranderson.com
dmoore@potteranderson.com

BY E-MAIL

George S. Bosy
JENNER & BLOCK
330 N. Wabash Avenue
Chicago, IL 60611-7603

gbosy@jenner.com

/s/ James W. Parrett, Jr.
James W. Parrett, Jr. (#4292)

61437190 v1