IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

ORACLE CORPORATION and ORACLE U.S.A. INC.,	)	
Plaintiffs/Counterclaim Defendants,	)	C.A. No. 06-414-SLR
v.	)	**JURY TRIAL DEMANDED**
EPICREALM LICENSING, LP,	)	**PUBLIC VERSION**
Defendant/Counterclaim Plaintiff.	)	

APPENDIX TO EPICREALM'S MEMORANDUM IN OPPOSITION TO ORACLE'S MOTION FOR SUMMARY JUDGMENT OF INVALIDITY

VOLUME I of II

EXHIBITS 1 to 26

Richard L. Horwitz (#2246)
David E. Moore (#3983)
POTTER ANDERSON & CORROON LLP
Hercules Plaza, 6th Floor
1313 N. Market Street
Wilmington, DE 19899
Tel: (302) 984-6000
rhorwitz@potteranderson.com
dmoore@potteranderson.com

Attorneys for Defendant/Counterclaim Plaintiff epicRealm Licensing, LP

OF COUNSEL:

Harry J. Roper
George S. Bosy
Aaron A. Barlow
Patrick L. Patras
David R. Bennett
Paul D. Margolis
Benjamin J. Bradford
Emily C. Johnson
JENNER & BLOCK
330 N. Wabash Avenue
Chicago, IL 60611-7603
Tel: (312) 923-8305

Dated: August 21, 2008
Public Version Dated: August 28, 2008
880392 / 31393 / Oracle

TABLE OF CONTENTS

Exh.	*Document*
1.	Excerpt from Deposition transcript of Paul C. Clark, July 11, 2008
2.	Excerpt from deposition transcript of Carol Colrain, April 15, 2008
3.	Excerpt from deposition transcript of Paul M. Dantzig, May 22, 2008
4.	Excerpt from deposition transcript of Dr. David Finkel, June 20, 2008
5.	Excerpt from deposition transcript of Steve Harris, April 18, 2008
6.	Excerpt from deposition transcript of Hal Hildebrand, Dec. 21, 2007
7.	Excerpt from deposition transcript of Ronald L. Howell, April 17, 2008
8.	Excerpt from deposition transcript of Keith Lowery, April 25, 2008
9.	Excerpt from deposition transcript of Katrina Montinola, Feb. 19, 2008
10.	Excerpt from deposition transcript of Mark Nelson, Feb. 22, 2008
11.	Excerpt from deposition transcript of Matthew Sarboraria, May 15, 2008
12.	Excerpt from deposition transcript of Michael Shamos, June 30, 2008
13.	Excerpt from deposition transcript of Douglas Surber, Feb. 28, 2008
14.	Atlanta Olympics WOMplex, March 1997, EPIC435843-435853
15.	USSN 08/794,269, Office Action, 8/23/2000, ORCL01623264-01623276
16.	Oracle10iAS Planning Update, ORCL01066383-01066413
17.	Oracle World, ORCL01050220-01050280
18.	Excerpt from Expert Report of Michael J. Wagner, 5/2/08
19.	Wagner Expert Report Volume 1 Tab 2, Schedules
20.	epicRealm's Second Amended Complaint, Jan. 27, 2006, *EpicRealm Licensing, LLC v. Autoflex Leasing, Inc. et al.,* (E. D. Tex.)
21.	Answer and Counterclaims of Safelite Group, Inc in Response to Plaintiff's Second Amended Complaint, February 8, 2006, *EpicRealm Licensing, LLC v. Autoflex Leasing, Inc. et al.,* (E. D. Tex.)
22.	Stipulation of Dismissal between epicRealm, LP and Safelite Group, Inc., June 26, 2006, *EpicRealm Licensing, LLC v. Autoflex Leasing, Inc. et al.,* (E. D. Tex.)

23. Order, Stipulation of Dismissal between epicRealm, LP and Safelite Group, Inc., June 29, 2006, *EpicRealm Licensing, LLC v. Autoflex Leasing, Inc. et al.,* (E. D. Tex.)

24. Settlement and License Agreement between Safelite Group, Inc and epicRealm Licensing, LP, June 7, 2006, EPIC435865-435872

25. US 5,761,673 Bookman, MAC003523-531

26. USSN 90/008,342, Reexam of US 5,894,554, Information Disclosure Citation, June 30, 2008

27. USSN 90/008,342, Reexam of US 5,894,554, Office Action, June 30, 2008

28. Excerpt from Report of Plaintiff's Expert Michael I Shamos

29. Excerpt from Oracle WebServer User's Guide, Release 1.0 A34986-2, ORCL00941713-00941960, *Ref.* 153 (also MAC003037-284)

30. Excerpt from Understanding SQL*Net Release 2.2, ORCL01806620-01806818

31. Functional Specification for Runtime Load Balancing, RDBMS, 10gR2, ORCL01060148-01060193

32. U.S. Patent Application 2005/0262183, Colrain et al., Nov. 24, 2005

33. InfoSpinner product information and correspondence, EPIC000133-000173

34. Letter re InfoSpinner products, EPIC014932-014936

35. Facsimile from R. West to S. Dettmer et al., EPIC014986-014991

36. Distributorship Agreement between Software AG of North America, Inc. and InfoSpinner, Inc., 2/26/96, EPIC000006-000034

37. Letter from R. West to J. Daly, 3/12/96, EPIC365224-365225

38. USSN 09/234,048, US 6,415,335, Paper 9, Response to Office Action, EPIC000489-501

39. Oracle WebServer 2.0 Technical Note March 1996, ORCL000761-000781, *Ref.* 605

40. Lagoze, et al., Dienst: Implementation Reference Manual, Cornell University Technical Report TR95-1514 (May 5, 1995). MAC 000270-000338

41. epicRealm Supplemental Response to Plaintiffs' Interrogatory Nos. 3-5, 9-15, 17-20, and 22-25, May 2, 2008

42. NTIS, Implementing Distributed Server Groups for the World Wide Web, 1/25/95, ORCL01818175-01818194

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

CERTIFICATE OF SERVICE

I, David E. Moore, hereby certify that on August 28, 2008, the attached document was electronically filed with the Clerk of the Court using CM/ECF which will send notification to the registered attorney(s) of record that the document has been filed and is available for viewing and downloading.

I further certify that on August 28, 2008, the attached document was Electronically Mailed to the following person(s):

Mary B. Graham
Morris, Nichols, Arsht & Tunnell LLP
1201 North Market Street
P.O. Box 1347
Wilmington, DE 19899
MGraham@MNAT.com

James G. Gilliland
Igor Shoiket
Townsend and Townsend and Crew LLP
Two Embarcadero Center
Eighth Floor
San Francisco, CA 94111-3834
OracleEpicrealm@townsend.com

Theodore T. Herhold
Robert J. Artuz
Eric M. Hutchins
Eric A. Mercer
Joseph A. Greco
Nitin Gupta
Townsend and Townsend and Crew LLP
379 Lytton Avenue
Palo Alto, CA 94301
OracleEpicrealm@townsend.com

Chad E. King
Townsend and Townsend and Crew LLP
1200 Seventeenth Street
Suite 2700
Denver, CO 80202
OracleEpicrealm@townsend.com

/s/ David E. Moore
Richard L. Horwitz
David E. Moore
Potter Anderson & Corroon LLP
Hercules Plaza – Sixth Floor
1313 North Market Street
P.O. Box 951
Wilmington, DE 19899-0951
(302) 984-6000
rhorwitz@potteranderson.com
dmoore@potteranderson.com

Exhibit 1

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 2

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 3

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 4

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 5

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 6

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 7

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 8

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 9

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 10

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 11

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 12

Redacted

IN THE UNITED STATES DISTRICT COURT FOR THE
WESTERN DISTRICT OF PENNSYLVANIA

- - - -

ORACLE CORPORATION and ORACLE U.S.A. INC.,	)	
Plaintiffs,	)	
-vs-	)	C.A. No. 06-cv-414 (SLR)
EPICREALM LICENSING, LP,	)	
Defendant.	)	

- - - -

VIDEOTAPED DEPOSITION OF:
MICHAEL SHAMOS, Ph.D., J.D.

- - - -

DATE: June 30, 2008
Monday, 9:12 a.m.

LOCATION: JONES DAY
One Mellon, 31st Floor
500 Grant Street
Pittsburgh, PA 15219

TAKEN BY: Defendant

REPORTED BY: Heidi H. Willis, RPR, CRR
Notary Public
Reference No. HW07873

VIDEOTAPED DEPOSITION OF MICHAEL SHAMOS, Ph.D., J.D., a witness, called by the Defendant for examination, in accordance with the Federal Rules of Civil Procedure, taken by and before Heidi H. Willis, RPR, CRR, a Court Reporter and Notary Public in and for the Commonwealth of Pennsylvania, at the offices of Jones Day, One Mellon, 31st Floor, 500 Grant Street, Pittsburgh, Pennsylvania, on Monday, June 30, 2008, commencing at 9:12 a.m.

APPEARANCES:

FOR THE PLAINTIFFS:
Robert J. Artuz, Esq.
rjartuz@townsend.com
TOWNSEND and TOWNSEND and CREW, LLP
379 Lytton Avenue
Palo Alto, CA 94301-1431
P 650-326-2400
F 650-326-2422

FOR THE DEFENDANT:
Patrick L. Patras, Esq.
ppatras@jenner.com
Benjamin Bradford, Esq.
bbradford@jenner.com
JENNER & BLOCK, LLP
One IBM Plaza
Chicago, IL 60611-7603
P 312-222-9350
F 312-222-2945

FOR THE TEXAS DEFENDANTS FRED FINDER NETWORK and VARIOUS, INC.:
Hector J. Ribera, Esq.
hribera@fenwick.com
FENWICK & WEST, LLP
Silicon Valley Center
801 California Street
Mountain View, CA 94041
P 650-988-8500
F 650-938-5200

APPEARANCES (CONT'D):

FOR THE TEXAS DEFENDANT HERBALIFE INTERNATIONAL OF AMERICAS, INC.:
Andrey Belenky, Esq.
abelenky@jonesday.com
JONES DAY
222 East 41st Street
New York, NY 10017-6702
P 212-326-3939
F 212-755-7306

FOR PARALLEL NETWORKS and EPICREALM:
Kevin J. Meek, Esq.
kevin.meek@bakerbotts.com
BAKER BOTTS, LLP
200 Ross Avenue
Dallas, TX 75201-2980
P 214-953-6500
F 214-953-6503
(Appearing Telephonically)

ALSO PRESENT:
Matthew Sarboraria, Esq.
matthew.sarboraria@oracle.com
Oracle
500 Oracle Parkway
M/S 5op7
Redwood Shores, CA 94065
P 650-506-1372
F 650-506-7114

EXAMINATION INDEX

MICHAEL SHAMOS, Ph.D., J.D.
BY MR. PATRAS 7

Certificate of Reporter 329
Errata Sheet 330

EXHIBIT INDEX

PAGE

Shamos
1 Notice of Deposition 7
2 Report of Plaintiff's Expert Michael I. Shamos, Ph.D., J.D., Concerning Invalidity 10
3 Exhibit 3 to Expert Report of Michael Shamos 10
4 Rebuttal Expert Report of Dr. David Finkel 106
5 Expert Report of Dr. David Finkel 126
6 United States Patent No. 5,894,554 133
7 Reference 605, ORCL 00000761-00000781 280

PROCEEDINGS

THE VIDEOGRAPHER: My name is Adam Werner of Veritext. The date today is June 30th, and the time is approximately 9:10 a.m.

This deposition is being held in the office of Jones Day located at 500 Grant Street, Pittsburgh, PA.

The caption of this case is Oracle Corporation V. epicRealm Licensing in the United States District Court for the District of Delaware.

The name of witness is Michael Shamos.

At this time the attorneys will identify themselves and the parties they represent, after which our court reporter, Heidi Willis of Veritext, will swear in the witness and we can proceed.

MR. PATRAS: Patrick Patras and Benjamin Bradford from Jenner & Block on behalf of epicRealm Licensing. Also on the telephone today is Kevin Meek from Baker Botts also representing epicRealm Licensing.

Page 74

horse, and I stopped doing AAA work at that point. I mean you get paid if you do the arbitration, but you have pay a fee to be listed as an arbitrator, and I wasn't comfortable with that.

Q. Did you view your -- strike that.

Do you view your role as an expert in this case as being something different than the role you served as an arbitrator?

A. Sure.

Q. How so?

A. Well, the responsibilities and activities are completely different. The arbitrator has power; the expert doesn't. The arbitrator is supposed to look at two arguments and apply the law and decide who is right; it's not within the can of the expert. The arbitrator issues awards; the expert doesn't. I'm not sure what the relationship would be.

Q. How much did you earn as an expert witness in 2007?

A. I don't know. If you wanted me to guess at it, I could probably guess at it.

Q. Sure.

A. It was probably in the range of $400,000.

Page 75

[redacted]

Q. How many cases did you work on in 2007 as an expert witness?

A. That's a difficult question. You'll have to qualify. The way it works is I get engaged in a case, and then typically, unless there's some horrible circumstance, I remain engaged until there's a final determination in the case.

During that period there are spurts of activity, when expert reports have to be prepared, for example, and then there are long periods where there is complete inactivity. There was one case I was involved in where it took two and a half years to get a Markman Order. There was zero work on the case for two and a half years, but technically I was, you know, working on the case. I was -- it was pending.

And so I've been -- I'm involved --

Page 76

I was involved in a lot of pending cases on which there was no activity at all. So I'm not exactly sure how to answer the question.

Q. Okay. Let me ask a slightly different question. How many cases did you bill time to as an expert witness in 2007?

A. I don't know, but I would -- I would guess it to be in the range of 20. But, again, the total billing on a case might be eight hours for the whole year. So I don't know what it represents.

Q. Okay. Are you serving as an expert witness in any other cases involving the '554 and '335 patents?

A. Well, other than this case and the Texas case, no. Well, when you say involving, you mean in which infringement or validity of those patents is at issue?

Q. Correct.

A. No. It's conceivable that they have been cited as prior art in some other cases.

Q. Is your rate of pay for your work in the Texas case the same as your rate of pay for your work in this case?

A. No.

Page 77

Q. What are you being paid in the Texas case?

A. I have to qualify that too. In the Texas case -- well, I bill through an entity called Expert Engagements, which is a limited liability company that my wife and I own. Expert Engagements bills $475 an hour for me in the Texas case, and of that I receive 425 from Expert Engagements.

In the Delaware case, because I was engaged later and my rate was higher, it's 525, billed by Expert Engagements, and I actually don't even know how much I receive out of that. My wife makes the determination, but I assume that it's a -- the percentage is on a par with that in the Texas case.

Q. How much have you earned so far on the Texas and Delaware cases?

A. A small fortune. I don't actually know the number. I think in the Oracle case it's around -- it's a little over 100,000. I'm sure it's more in the Texas case because it's been pending much longer.

Q. Approximately how much have you earned in the Texas case?

A. I don't know. It might be double that. I

don't know.
Q. So you may have earned over $300,000 on these two cases?
A. I might have.
Q. Did you know Dr. Clark previous to working on these matters?
A. I didn't know him until this morning.
Q. Had you ever worked on a case with Dr. Clark, either on the same side or the opposite side?
A. Not to my knowledge.
Q. Did you know Dr. Finkel prior to this case?
A. No. I still don't know him.
Q. Did you ever work on a case with Dr. Finkel to your knowledge, either on the same side or opposite sides?
A. Well, I mean other than Delaware and -- these two --
Q. Other than these two cases?
A. Not to my knowledge.
Q. Do you believe that Dr. Clark is qualified to be an expert witness in this case?
A. I have no information about Dr. Clark's qualifications. I can't say one way or another.
Q. Do you believe that Dr. Finkel is qualified to

be an expert witness in this case?
A. As a general matter, I believe the level of qualification of experts to be -- the necessary level to be relatively low.
If you look at the Federal Rules as to what an expert's supposed to do, it's supposed to assist the trier of fact in arriving at conclusions.
Dr. Finkel has been a computer scientist for a long time. He has specific knowledge in certain areas that are relevant to this -- to this litigation. I know nothing about his demeanor, but strictly based on qualifications, I'm sure he has the ability to say relevant things in the case.
Q. Have you ever done any work, Dr. Shamos, designing a system that would process dynamic web page requests?
A. I certainly didn't work on designing a system for which that was the whole purpose. There have been numerous systems in which there were dynamic page requests that were handled that I was involved in the design of, sure.
I mean I have served as advisor for a huge number of what we have referred to as

the practicum projects as part of our master of science in e-commerce degree program and the e-business techologies degree program. These have required the creation of dynamic pages, and so, yeah, sure.
But, no, I never went and designed a system specifically for the purpose of creating a dynamic web page.
Q. With respect to your last answer and the work that you specified there, when was the first time that you did that type of work?
A. Oh, probably -- well, with respect to the CMU's academic program, it would have been the year 2000. As part of my work in representing clients, not necessarily system design, but evaluating them for potential patentability, I would have gone back to the early '90s.
Q. What systems did you evaluate in the early 1990s that related to this area?
A. Well, for example -- when you say related to this area, not systems that were designed specifically for the creation, efficient creation of dynamic web pages, but systems which, by their nature, had to create dynamic web pages.

And so there was a patent recently litigated actually that I drafted on holding municipal bond auctions over the internet, and the pages that are displayed to the bidders are dynamically generated because they are based on real-time information about bids that have been received.
So it's part and parcel, but it was not a system that focused on the generation of dynamic web pages.
Q. Have you done any design work relating to any load balancing of requests for dynamic web pages?
MR. ARTUZ: Objection, vague.
A. I don't have any specific recollection. It's conceivable that there may have been systems that I was consultant on where load balancing was done or I advised it to be done, but I don't have any specific recollection.
Q. Could you identify for me, Dr. Shamos, what you believe to be your most closely related experience with respect to the issues in this case prior to working on this case or the Texas case?
MR. ARTUZ: Objection, vague.

from what already existed in dealing with client server architectures; is that correct?
A. We are talking about at the architecture level of load balancing, it's exactly the same thing. The algorithms that you might use for doing a load balancing might be different, but the fact that you are load balancing from web servers to page servers is no different from the previous load balancing that you were doing between clients and server and client server computing. It's distributing load to optimize some parameter, typically a performance parameter.
Q. Are there any different considerations that need to be taken into account in the web server context as opposed to the client server context that you just described?
A. Not as revealed in the patents.
Q. Outside the context of what you say is revealed in the patents, are there any such different considerations?
A. Well, different considerations as to what? Of course if you are passing web requests, you are going to be using a certain protocol. If you are not passing web requests, you are

going to be using a different protocol, but that has nothing to do with the fundamental nature of the load balancing.
The load balancing is to balance load so that servers don't become overburdened with the requests that they are handling, thus slowing down response time. It's the same consideration that we've had in load balancing since the 1940s.
Q. Do you believe that what is achieved by the epicRealm patents-in-suit is something that was known since the 1940?
A. What is achieved? I'm afraid you are going to have to tell me what was achieved. Then I'll let you know whether it was known in the '40s.
Q. Efficient management of dynamic web page requests.
A. Obviously that was not known in the 1940s.
Q. Are you familiar with something called sticky sessions?
A. Yes.
Q. What is a sticky session?
A. In an architectural decision that has been revisited many, many, many times, the designers of the Hypertext Transfer Protocol

decided to make it connectionless, which means that once an HTTP session is opened and the request is handled, the connection is closed and the next connection has no knowledge that there was ever a previous one.
What that means is that if I make a request to a web server and then it's handed off to some page server, and then a millisecond later I make another request to the same web server, it may assign the second request to a totally different -- to a totally different page server.
And so the idea of a sticky session is to ensure that subsequent requests within the context of a session -- and it's often very difficult to define what a session is -- but within a session all the requests are going to go to the same server. There can be benefits to that. One of the benefits is that if we already know that that server has the necessary resources to answer my request, we don't have to go and waste time figuring out whether that's the one to use.
Another is that that server may actually maintain a record of the interaction

between the client and itself and, therefore, not have to go and make an inquiry over the network to learn this information. So that's what a sticky session is. It's ensuring that subsequent requests in the session are handled by the same server.
Q. In your view is the use of a sticky session something that would be considered to be dispatching, as that term is used, for example, in claim 1 of '554 pattern?
A. Yes.
MR. ARTUZ: Objection, vague, incomplete hypothetical.
A. Oops, I should have waited, but I'm answering anyway.
So there is a dispatching element to it, and the dispatching element is in determining an initial opportunity which server should be assigned. That's the dispatching.
Now, it becomes a philosophical question whether the second request is dispatched because an intelligent choice has already been -- or let's say an informed choice within the claim construction -- an

informed choice has already been made of which server to use, and so subsequent requests are simply taking the benefit of that informed selection.

Q. Are your comments here in paragraph 102 on dispatching based on the Texas claim construction for dispatching, or did you also consider the parties' proposed claim constructions in this case?

A. Okay. So the first thing that's quoted there is the claim construction from the Texas action, and what I said earlier was that the proposed claim constructions in this action are at least as general. And so if something satisfies the limitations of this dispatching and satisfies the dispatching of in the Oracle case.

Q. In your opinion can a system with a single page server perform dispatching?

MR. ARTUZ: Objection, vague, incomplete hypothetical.

A. It's a -- I believe it to be a purely philosophical question and not a technical question. This comes up all the time in other cases, not this particular question, but

vacuous cases of things. Is a list a list if it only has one thing on it. Is a list a list if it has zero things on it. Is a blank piece of paper a list. Is a page server -- can you dispatch to, within the meaning of the construction, to a single page server, okay.

Ordinarily I would think no. To the extent that epicRealm previously accused systems that only had a single page server of infringement, it made me rethink it. The issue being a purely semantic one of are you making an informed choice if there's only one thing you can choose. I don't know.

Q. What's your opinion?

A. We'll call in -- we'll call in the philosophers.

Q. Do you have an opinion on that question?

A. My opinion was at first blush when I saw it, it didn't look to me like it was an informed selection. But then, you know, I looked at -- I looked at the question again because I was informed that epicRealm had accused single page server systems, and then I retreated to the position that it's a purely philosophical question. It's not a technical question, you

know. We are not going to find in the literature a discussion of whether a single page server can be dispatched to. It's just not there.

Q. What will your opinion be at trial on that question?

A. What will my opinion be at trial. I -- probably just as I said, which is at first blush it didn't seem as though it would be dispatching because there's no informed selection going on, to the extent informed selection is required. If it's just determining which page server should process the request and sending the request to the -- to that page server, it satisfies that.

Q. Do you have an opinion, Dr. Shamos, that you can share with us -- because if you are going to have one at trial, I'd like to learn about it today -- as to whether dispatching can be accomplished within the meaning of that term in claim 1 of the '554 patent if there is only a single page server?

MR. ARTUZ: Objection, vague, incomplete hypothetical.

A. I'm not sure based on our colloquy that anyone

will put me up to offer an opinion on that subject; however, as I said, if an informed selection is required, then it's difficult for me to ascertain what the information is on which the selection is being made. If no informed selection is necessary, because of claim construction, then, yeah, a single page server can be dispatched to, because dispatching has an ordinary meaning to me of sending, sending off, like I dispatch a letter, I put it in the post box.

Q. So you can't give me a yes or no answer I guess is what you are saying?

A. No, I can't, because we don't have a claim construction yet.

Q. Well, let's take all the possibilities. There's Oracle's proposed claim construction and there's epicRealm's proposed claim construction.

A. Okay. My recollection, and it's always dangerous to try to recall statutes or claim constructions from memory --

Q. Would you like to see the claim constructions to answer the question?

A. I would, yes.

Page 126

Q. We'll get those for you in a minute.
A. You can help me out in the meantime by telling me whether a tree falling in a forest makes a sound if no one's around to hear it. Give me the answer to that question and I think I can give you the answer to the other one.
Q. I would hire an expert witness who is very smart and ask him that question, ask him what his opinion is.
A. I think we are going to end up philosophers to answer this in. If it would help, I have this document which I may exhibit to you.
Shall I show it to him?
Q. I'll go ahead and mark an exhibit that has both in them here.
A. Okay. Fine.
Q. This is -- we'll mark as Shamos Exhibit 5 a copy of another report entitled Defendant epicRealm Licensing, LP's Expert Report of Dr. David Finkel.
- - - -
(Exhibit No. 5 marked for identification.)
- - - -
Q. Dr. Shamos, we have handed you what's been marked as Exhibit 5, and I believe if you look

Page 127

at Tab A and Tab B, you will see the epicRealm proposed claim construction and Oracle proposed claim construction, so you could look at each of those for dispatching and then answer for me the question of whether in your opinion a system with only a single page server can perform dispatching.
A. Okay. Under epicRealm's proposed claim construction, no, because it requires an informed selection, and if there's no choice, it doesn't seem to me to be an informed selection, but that's not a technical opinion of a computer scientist. That's an opinion of a hopefully rational human being considering the philosophical issues.
Q. Okay. How about under Oracle's proposed claim construction?
A. Okay. So dispatching said request to said page server, Analyzing a request to make an informed selection of which page server should process the request and sending the request to that pager server.
No, still requires informed selection, so I don't see what -- I don't see how the selection is informed.

Page 128

Q. If there's only one page server?
A. There's no selection. It's rote. But, again, it's not a technical opinion.
Q. You mentioned I think earlier in your answers on this topic that you had been told that epicRealm had accused certain systems having only a single page server of infringing --
A. Yes.
Q. -- is that correct?
A. I have been told that.
Q. What is your basis for saying that epicRealm has made those accusations?
A. Because I was told that.
Q. By whom?
A. By counsel for Texas defendants.
Q. What systems having only a single page server were accused of infringing to your knowledge?
A. I do not know.
Q. Were they systems of the Texas Defendants as opposed to Oracle?
A. They were of the Texas Defendants.
Q. Are you aware of any similar situation with respect to any of the Oracle products and accusations?
A. When you say a similar situation, meaning?

Page 129

Q. Has anyone told you that epicRealm has made an accusation against Oracle that would involve single server systems being accused of infringing the claims?
A. I don't recall. It might have happened.
Q. But not that you recall?
A. Right.
MR. PATRAS: Okay. Let's take a break.
- - - -
(There was a recess in the proceedings.)
- - - -
THE VIDEOGRAPHER: We are now back on the record. The time indicated on the screen is 12:03 p.m.
BY MR. PATRAS:
Q. Dr. Shamos, I wanted to go back and just clarify one thing. Earlier you were looking on your computer at a copy of your report?
A. Yes.
Q. It's the report for the Delaware case; correct?
A. Yes.
Q. Dr. Shamos, do you have an understanding of what dynamic information is?

Page 170

implied that if all the packets have come from the same IP address that that means the IP address is part of the message? I don't know. There's not enough fine structure to answer the question.
Q. Do you know, Dr. Shamos, how Oracle's Mod OC4J works?
A. Not in detail.
Q. Enough to testify about it at trial?
A. Maybe about some aspect of it.
Q. What aspects about it are you familiar with to that degree?
A. I don't know. It depends on -- it depends on what I'm asked.
Q. Well, tell me --
A. I don't have any anticipation of testifying about accused systems at trial.
Q. Dr. Shamos is load balancing a broader concept than dispatching as used in the claims here?
MR. ARTUZ: Objection, vague.
A. It is -- I wouldn't characterize it either as broader or narrower. It's different and related. You can have load balancing without dispatching. You can have dispatching without load balancing. You can have dispatching that

Page 171

has nothing to do with balancing. You can have load balancing that doesn't have to do with dispatching as construed here. They are related but not the same.
Q. Okay. Could you give me an example of load balancing that is not dispatching?
A. Yes, sure, if there are no page servers, it's not dispatching.
Q. Can you give me an example, any other examples of load balancing that are not also dispatching as proposed by, say, epicRealm's construction?
A. Well, epicRealm's claim construction, if the -- if load balancing is done by -- does not base -- if the informed selection is not based on dynamic information maintained about page servers, then it's not dispatching under epicRealm's construction.
Q. Let's say, for example that --
A. It can still be load balancing.
Q. -- that a product were to use a random distribution of requests to suitable page servers. Would that be dispatching in your view?
MR. ARTUZ: Objection, incomplete

Page 172

hypothetical, lacks foundation.
Q. Let me actually add on under epicRealm's construction first.
MR. ARTUZ: Same objection.
A. By random distribution I assume you mean as a request comes in, it is randomly assigned to some pool of servers and the size of that pool does not change. If that -- under --
Q. You can make that assumption.
A. Under those assumptions, then I can't see in there what the dynamic information maintained about page servers is. If the number of page servers was changing and the dispatcher had to inform itself of what is the available set to which I may dispatch, that would be dynamic information, but just as I laid it out, I don't see how that fits dispatching.
Q. Okay. As you first laid it out, I guess a set number of page servers --
A. A known, fixed, constant number and purely random selection among them.
Q. -- would that system be using load balancing?
A. Sure.
Q. What if the system were to use -- again, if you want to assume a fixed number of page

Page 173

servers -- a round-robin algorithm to distribute requests? Would that be dispatching under epicRealm's construction in your opinion?
MR. ARTUZ: Objection, vague, lacks foundation, incomplete hypothetical.
A. I can see an argument either way.
Q. What are those arguments?
A. The argument is it's clearly based on dynamic information maintained about page servers because the information that it's maintaining is which was the least recently used server that I sent a request to, okay?
Q. Can I interrupt right there just for a second? I'm sorry.
A. Yes.
Q. You said which was the least recently used.
A. Yes.
Q. How would round-robin determine the least recent -- you mean time-wise?
A. Yes, least recent in time.
Q. I got you. I misunderstood, sorry.
A. And the other argument is while that maintains no information -- no dynamic information whatsoever about page servers, it only

Page 174

maintains dynamic information about the dispatcher, where did you least recently send the request to, and it almost becomes a matter of philosophy at that point.
Q. Which argument do you think is correct?
A. I haven't -- I mean I've batted both of them about in my mind. I haven't come to a conclusion as to which one I prefer. It's not even clear that either is correct, since arguments, decent arguments can be made for both and there is no arbiter that we are going to go to that is going to give us the final God-like determination, well, except a judge, they are god-like.
Q. Dr. Shamos, are you familiar with a reference that you've referred to I believe as the Garland reference, your Reference 13?
A. Yes.
Q. Do you contend that the Garland reference anticipates any of the asserted claims of the patents-in-suit?
A. No.
Q. Why not?
A. Because it does not have express mention of web pages or dynamic web pages, and,

Page 175

therefore, I can't find it in the four corners of the reference, so it can only be an obviousness reference.
Q. Dr. Shamos, if I can direct your attention back to your report and in particularly to paragraph 71, please.
A. Yes.
Q. You state in paragraph 71, Every asserted claim requires the page server to expressly release the web server or HTTP-compliant device. Do you see that?
A. Yes.
Q. What is your basis for that statement?
A. Because every claim contains an expressed step, an expressed act of releasing.
Q. Can you explain where you get the expressly portion of that from?
A. In -- I'm in the '554 patent.
Q. Okay.
A. I see in the first step of claim 1, Releasing said web server to process other requests. The next independent claim in that patent is 9. It's not worth talking about 9 since it's a hybrid claim, but in any case the releasing is also recited in claim 9.

Page 176

Claim 11, the next independent claim, Releasing said web server to process other requests, and likewise with the '335. Go through every claim of the '335, there's an express releasing step.
Q. In your report did you talk about expressly routing the request, for example?
A. Expressly routing?
Q. Well, if the basis for putting the term "expressly" in paragraph 71 is because the word "release" is in the claim, I mean there's lots of other words in the claim too; right? So I'm not understanding why "expressly" you believe is appropriate here, but not with respect to apparently to other claim limitations?
A. Because I don't understand epicRealm to contend that those other steps do not occur in the prior art, whereas you do contend that the releasing step doesn't occur in the prior art, so that's why I discussed releasing specifically in that way.
We have the All Elements Rule. In order to infringe a claim, you have to perform -- if it's a process claim, you have

Page 177

to perform all the steps. One of the steps at each one of the process claims in these patents is the step of releasing the web server. That's what I mean by it's an expressed step, expressly releasing.
Now, later on there's a discussion of how you may release, okay, and there I use a -- I also use the word "express" in a different sense, which is an actual message which says "Thou are hereby released" is one way of doing it, but there are other ways of doing it that still fall within an act of releasing.
Q. So, for example, what I think you might refer to somewhere as implicit releasing --
A. Yes.
Q. -- that might still be something to be pointed to for purposes of paragraph 71 as expressly released?
A. No, implicitly releasing is not expressly released.
Q. Then I don't understand the two different ways you say you are using the word "express" in paragraph 71 versus your discussion of how to release?

Page 210

detail in the construction.
Q. Your statement here in paragraph 94 that I read, was that under epicRealm's proposed construction or under Oracle's construction?
A. It's under both. Okay. If the web server doesn't process the request at all, doesn't even look at it, then the intercepting has happened before the web server, and that's Oracle's, that's Oracle's construction.
If the web server processes the request a little bit but it doesn't complete the processing of the request, then that can fall within the definition of the web server not handling the request.
See, the word "handling" wasn't construed. Can you spend three microseconds handling it before you decide you are not handling it anymore? I don't know, and I actually cannot believe that the outcome of this case is going to rest on that -- on that determination.
Q. Under Oracle's proposed construction here, how would the web server receive a request?
A. What's the web server? Is it the machine or the web server executable? You tell me.

Page 211

Q. It's Oracle's construction.
A. No, the language of the claim is epicRealm's language in the claim.
Q. Is it possible under Oracle's proposed construction for the web server to receive a request that is going to be handled ultimately by a page server?
A. Vague and ambiguous. What is the web server? Is it the machine, or is it the web server executable?
Q. You can look at both proposed constructions and answer for each of them.
A. Oh, okay. I can probably do that.
So I'm looking at epicRealm's construction of intercepting, Intercepting said request at said web server, and the epicRealm proposed construction is, intercepting the handling of a request at a web server.
Now, I'm not going to chide people more than usual on this, but you know how it goes with constructions. You saw from my resume that I've written a number of dictionaries. I'm a lexicographer, and I'm really careful about trying to define things,

Page 212

and I really know that any time you try to condense a lot of meaning into a small number of words, you are missing something.
And this construction -- and I'm sure it's also true of Oracle's -- misses something, and intercepting the handling of a request to the web server simply asks the question now what does handling mean. Does handling the request mean doing anything at all to it, or does handling mean completing the processing of the request.
If the handling is simply examining it to decide whether we want to do further handling, then it's doing some handling but it's not doing all the handling.
Oracle wants it so that no handling whatsoever is done by the web server executable. Some other code looks at it first and kicks it away before it gets to the executable.
I can't determine what happens or what's right based on the word "handling." There's just not enough information in the construction to make that determination.
Now let's look at -- now let's look

Page 213

at Oracle's.
Q. Could I ask you a question --
A. Yeah.
Q. -- just based on what you said so far?
A. Sure.
Q. How did you determine that 30 references anticipated the claim if you couldn't tell whether or not something meets intercepting?
A. So I have my own understanding of what intercepted is. I think that it's okay if the web server looks at the request long enough to determine that it shouldn't handle that. I like that. Oracle doesn't like that. I think it's going to be six of one, half dozen of the other.
If Oracle's right, then there's not infringement and maybe the invalidity arguments fall apart. If Oracle's wrong, the invalidity arguments are correct, the patent's invalid, but maybe they would infringe the literal words of the claim. I don't know. It often happens, but it's one or the other. You can't have both.
Q. And do you know how the accused Oracle products work with respect to this issue?

Page 214

A. To --
Q. To the intercepting issue?
A. To save time, I will repeat that I know none of the details of any of the accused instrumentalities. I'm not going to be able to answer that question other than to say no, I don't know. So --
Q. I'm sorry, could I ask one follow up on that?
A. Sure.
Q. Do you know how the Apache web server works with respect to the intercepting issue?
A. I'll say it again. What exactly is the Apache web server? Is it a machine on which Apache happens to be running, or is it the Apache web server executable, and is it the entirety of the executable, or can we delineate certain lines of code that check for dynamic web requests and say that's not really part of the web server, that's part of something else. It all depends on where you draw the line. There isn't enough information to answer the question.
Q. Under what I think you described as your view of intercepting --
A. Yes.

Page 215

Q. -- do you believe that the Apache web server meets that construction, if you will, for intercepting?
A. I believe it does.
Q. Okay. Thank you. I'm sorry, I interrupted your answer.
A. No, no, no, it's your depo. Okay. Now I'm looking at Oracle's construction of intercepting said request at said web server.
Receiving a request at the web server and diverting the request before the web server executable can process the request. That's as I said. They want -- whatever processing you are going to do to determine whether the web server is going to process it, you do it before the web server even gets the teensiest look at one character of the request, okay. That's I think a pretty good lay vision of what intercepting is.
So the quarterback has thrown a pass to you, and I catch it before you do. I have intercepted it. You never even had a chance to try, and then somebody else comes up with the objection that what happens if we both touch it and it's a jump ball and you touched

Page 216

it a little bit but I ended up intercepting it, is that still intercepting. I mean that's what patent cases are about, making arguments like this.
Q. Am I correct that you disagree with Oracle's proposed construction here for intercepting?
A. It's not an issue of agree or disagree. There are sound arguments to be made in favor of either construction. I think, as I said, the lay understanding of interception is the recipient never even touches it. It's been grasped away by somebody else before he gets it. I think that's fine.
Q. But that's not what you personally believe intercepting should be construed to mean; correct?
A. Should be. You know, what I believe is largely irrelevant at this point. The Court's going to make a determination.
Q. But in your opinion you believe that intercepting should be construed differently than what Oracle has proposed here; correct?
A. So when I look at the spec, I have to make sense of a jumble, and the jumble is that it shows the interceptor on completely the wrong

Page 217

end of everything. It shows the interceptor past the web server instead of before the web server.
And I read the words of the spec, and I look at the real world as to how web servers handle CGI requests, and I know that at least based on -- based on the prior art, the web server looks at the request a little bit before it says get the hell out, okay.
And that's the basis on which I think of the two, if you forced, put me against the wall and said you are the judge, which one of these are you going to pick, I'd probably pick the one in which gets to look a little bit, but I see the argument on the other side. And, you know, this is the lexicographer's nightmare. You never come up with the perfect definition of anything.
Q. What construction for intercepting did you use in doing your analysis as to whether the prior art anticipated the claims?
A. I used -- well, it's not clear to me that I used either of them, but I used yours.
Q. Were there any items of prior art that you reviewed, that you recall that would have met

Page 218

Oracle's proposed construction for intercepting?
A. So I don't recall specifically, but there certainly are conceivable, conceivable configurations. For example, suppose I have a load balancing server, which we haven't discussed in -- in this case, but it's common in server farms where all requests initially come to a load balancing server. That server does nothing else except pass the request on somewhere else.
The load balancing server can look at the request, see if it's a static page request, in which case it sends it to one set of servers. If it determines that it's a dynamic request, then it sends it to a different set of servers. That would be intercepting under Oracle's construction because the web server itself never looked at the request, not even one character of it.
Q. And what you've called the web server in that answer --
A. Would be the thing that would be delivering the page. There's either one set of servers for delivering static pages, a different set

Page 219

of servers for delivering dynamic pages, and the load balancing server is none of those. It would be the intercepter.
Q. Could I direct you in your report, Dr. Shamos, to paragraph 186.
A. Yes.
Q. Beginning at the bottom line on page 52, you state, "One of skill in the art would understand that a web page has not been generated until it is complete; that is, if data is to be retrieved and surrounded by HTML tags so it can be sent to a browser, then the page has not been generated until the tags have been added."
A. If there are tags, yes.
Q. Do you believe that a web page has to include HTML?
A. You've just raised a question that is one of the most difficult issues in this case. No, it doesn't have to be HTML, but it -- I believe that one of skill in the art would understand that a web page contains some web markup. It could be XML, it could be something else, but it's not simply text. I realize that the Texas Court didn't go along

Page 220

with that, and thereby created a mess.
Q. What about a flash page?
MR. ARTUZ: Objection, vague.
A. I don't actually have a good understanding of how flash -- what flash pages look like when they are sent to -- when they are sent to a browser. I don't know what the header looks like, whether there is any HTML in the header or not. If there is, then there's HTML there.
Q. Did flash exist in April 1996? Do you know?
A. I don't know. I would now agree that flash is web conference. The whole issue here is pure text, is the digit 2 a web page, because the digit 2 is displayable by a web browser. I don't like a construction that makes the digit 2 into a web page, don't like it.
Then we have the issue of partial web pages. Suppose the dynamic portion of the web page consists of a stock quotation that's going to be inserted in a template that, you know, has the ticker symbol and all that stuff.
When you generate or when you retrieve the stock quote, is that a web page? The quote itself, without any HTML, could be

Page 221

displayed in a browser because it's just a number, but we understand, typically, that if there's other stuff that goes on the page, you are not done generating it until you are done, until you have everything, yet the constructions in this case leave open the possibility that pieces of web pages can be web pages. I don't like it, but I have to live with it.
Q. Do any of the references that you rely on as anticipating claim 11 of the '554 patent disclose a machine readable medium as you understand that term's use in claim 11?
A. Yeah, all of them, at least by inherency. They are all computer systems. Computers read their instructions from computer readable medium. If you don't have a computer readable medium, you've got no computer these days. It was different in the old days when you hardwired computers to do things, but you don't hardwire them anymore. You read the instructions off a medium.
Q. Do you have an understanding, Dr. Shamos, as to what the test is for enablement of a patent claim?

A. Yes, I think I do.
Q. What do you understand the test for enablement of a patent claim to be?
A. A claim is enabled if one of skill in the art is able to practice the claimed invention without undue experimentation, make --
Q. Does the -- sorry.
A. -- make and use the invention without undue experimentation.
Q. Does enablement need to be determined on a claim-by-claim basis as opposed to the patent as a whole?
A. Well, it applies to claims. The claim is either enabled or it isn't. It's quite possible to have a patent in which no claims are enabled, but it is an individual determination.
Q. Is it your opinion that all the asserted claims in this case are not enabled?
A. I think it is.
Q. You think it is?
A. Yep.

- - - -

(There was a discussion off the record.)

- - - -

A. And the reason I say that is that because releasing is required in every claim and there is no enablement of releasing, there is no claim that's enabled.
Intercepting, is intercepting required in every claim too? There could be lots of reasons that the claims aren't enabled. Let's go to the report and let's see.
Q. You can of course look where you like. I might suggest looking at page 163 as a starting point.
A. That might help us. Thank you. Invalidity -- okay. Yeah, I think I characterized it properly.
Q. Does the examiner in the patent office have an opportunity to consider enablement issues during prosecution?
A. Yes.
Q. Do you believe that the examiner made a mistake in concluding that each of the asserted claims in this case was properly enabled?
A. I don't have information that the examiner considered enablement. You asked me whether

he had the opportunity to consider enablement.
Q. Was the examiner supposed to consider enablement in reviewing the claims to determine whether they should be allowed as a patent?
A. Well, I think the way it really works is that if it's clear to the examiner that there's no enablement, that is if it's an egregious case, then he'll issue a nonenablement rejection, but if it requires him to put himself in the role of system designer and attempt to be one of skill in the art and actually go and imagine what he would have to do to make and use the invention, I don't think a lot of examiners do that.
Q. Does the MPEP instruct examiners that part of their job is to ensure that the claims are properly enabled?
A. It's been a bit since I've bedded down with the MPEP. I wouldn't be surprised to learn that there are instructions to the examiner to do that.
My belief is that if the examiner's followed every instruction in the MPEP, they might get through about one patent application

per year.
Q. So you don't believe that the examiner of the '554 patent did a good job in determining enablement of the asserted claims of the '554 patent; is that correct?
A. I don't want to characterize it that way. Examiners are very hard working people. I've even practiced law with some of them. It is possible that the examiner didn't consider it or didn't realize that there was an issue.
If he considered it and came to the conclusion that there was enablement, I would disagree with his conclusion. It doesn't mean he did a bad job. Examiners come to the wrong result all the time, and that's what patent litigation is about, and that's how patents get found invalid, et cetera.
Q. You've got a statement in paragraph 779 of your report. The last sentence reads, "In addition, far from containing enabling disclosure for a claim that would support such a contention, the express words of the specification directly contradict epicRealm's attempt to apply the claims this broadly."
Do you see that?

Page 310

opinion.
Q. Am I correct that your report includes over 150 paragraphs on obviousness and only six paragraphs on secondary considerations?
A. Oh, but those are wonderful paragraphs. They are dynamite paragraphs.
Q. Am I correct?
A. Yes, I believe so.
Q. You state in paragraph 800 that, in part, epicRealm's allegations are completely controverted by the published prior art.
A. Yes.
Q. What did you mean by that?
A. Okay. So the -- you can glean from a large number of litigated cases what factors are considered in secondary considerations. It is routine in some experts' reports to, by rote, simply make a list of every one of the secondary considerations that a Court has recognized and allege, without factual basis, that that secondary consideration applies in this case.
Now, you can argue all you want to that there are secondary considerations of nonobviousness; however, when the prior art

Page 311

clearly shows that the invention is at least obvious over numerous pairs of references and anticipated, the secondary considerations do not come into play because they are secondary, they are not primary, and so they are irrelevant, just not so, because the entire art was veering in this direction.
Q. You believe --
A. I'll give you an example. I'll give you an example. Long-felt need, long-felt need was amply satisfied by a prior art. It was no longer in need at the time the inventions were made.
Finkel even argues against long-felt need. He says the web was in its infancy, and so how can you have a long-felt need in a field where it's in its infancy.
Q. Do you understand that secondary considerations are to be treated secondarily? Is that what you said in your last answer?
A. Yes.
Q. Did you review the transcript of the deposition of IBM in this case?
A. I think I looked at it. I didn't rely on it.
Q. Did you consider the information in that

Page 312

deposition in reaching your obviousness conclusions?
A. I probably did.
Q. Were you aware that IBM set out to build the busiest website in history for the 1996 Atlanta Olympics?
A. Yes.
Q. IBM devoted millions of dollars and enlisted everyone at IBM who knew about dynamic web pages to work on the project?
A. Yes.
Q. IBM did not use the technology of the patents-in-suit and by its own admission IBM failed?
A. And it's not remotely clear that IBM would have succeeded if they used the technology of the patents-in-suit. The reason was I was watching TV, and I was very interested in the web at the time of the Atlanta Olympics. The load on the IBM website was vastly greater than even IBM ever anticipated. We are talking about a complete outlier here. We are talking about a website that nobody in history had ever tried to develop or had ever anticipated.
That has nothing to do with the

Page 313

run-of-the-mill, rote response to dynamic web page requests that were being serviced by thousands of tens of thousands of websites. Talking about a completely unique situation.
Q. And IBM didn't do what the patents tell you to do, did it?
A. I don't think it did, but there's no evidence that if they did what the patents tell you to do the Atlanta games website would have succeeded.
Q. Were you aware in conducting your obviousness analysis that IBM took a license to epicRealm's's patented technology?
A. Yes.
Q. How did that affect your opinions?
A. Not much.
Q. Why not?
A. Because there are plenty of reasons that one takes licenses. One takes licenses, for example, to avoid the cost of litigation, to avoid bad press, to avoid the allegation that big, nasty IBM is pushing on, you know, enterprising little company.
It would be different if an entire industry lined up at the door to take

Page 322

balancing it didn't get from these patents.
Q. In your view is there any secondary consideration that points in any way to nonobviousness of any of the asserted claims?
A. I don't recall that there is. You know, you can make a list of them. I'm sure they were elaborately listed in Dr. Finkel's report. I remember looking at them and not being concerned at any of them. There can't be secondary considerations of nonobvious when the whole invention is anticipated. There's no such thing as nonobvious if it's anticipated.
Q. Does that mean you can ignore the secondary considerations in your obviousness analysis?
A. No. It means that the secondary considerations are unlikely to be applicable, because the entire art already knew of these things. So there can't be anything about it that's not obvious.
So if you point to increased sales, for example, it didn't come from this, it may have come from something else. Long-felt need, there was no long-felt need. The need was satisfied very rapidly, et cetera, et

Page 323

cetera. We can go through each one of the long, lengthy catalog of secondary considerations. None are applicable.
Q. Did you consider in reaching your conclusions in this case a report by a Dr. Chen?
A. I read it. I can't say that it affected my conclusions or that my conclusions were based upon it or relied upon it. I did read it. I read it with amusement. I agree with him. Ask me what you will.
Q. When you say you read it with amusement, why do you say that?
A. Because he's not a native speaker of the English language, and I say that many of the sentences in there were inartfully expressed, as was even the title of his report, and so it caused a chuckle, but he was right in his conclusions.
Q. What conclusions was he right about?
A. The conclusion was that this -- I think the words were very close to this -- patent would hardly be enforceable, these patents would hardly be enforceable.
Q. Did you review Dr. Chen's deposition testimony in this case?

Page 324

A. I don't think I did.
Q. Are you aware that Dr. Chen is not a patent attorney?
A. I don't think that has any relevance at all. I didn't rely on his opinion. I just said that I agree with that. One might wonder why folks are soliciting patentability or enforceability opinions from non-patent attorneys.
Q. Were you aware that Dr. Chen did not identify any claim construction for his analysis?
A. I was aware of that. His report was brief.
Q. Is that a problem in your view?
A. I don't understand the context in which it would be a problem. I don't rely on his report, and so it's not a problem for me. The only party in the whole world I can imagine it being a problem for is epicRealm.
Q. Why do you say that?
A. Because he concluded the patents weren't enforceable and you went out and tried to enforce the patents, that why.
Q. Are you aware that Dr. Chen testified that he couldn't say that he had even reviewed the file histories of the patents?

Page 325

MR. ARTUZ: Objection, assumes facts, lacks foundation.
A. It doesn't matter whether I'm aware of it or not. I haven't relied on his report.
Q. Why not?
A. Because I came to my own conclusion. There's no need for me to rely on his report. His report was brief. My report is long.
Q. Do you believe his report was competent?
MR. ARTUZ: Objection, vague, calls for a legal conclusion.
A. It certainly does not rise to the standards that one would expect of an attorney being asked to give an enforceability report or a validity report, no. Whether it was competent based on what he was asked to do, I don't know how the task was phrased. I don't know how he was chosen as the person to make the report.
My impression was that it was believed that he was someone of skill in the art, not a patent attorney, and they were asking does this sound new to you given what you know of the art.
Q. Do you know of any prior art references that disclose dispatching using metric-based load

Exhibit 13

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 14

Atlanta Olympics WOMplex



By Andy Stanford-Clark

BEST AVAILABLE COPY

This article describes the major components of the Atlanta Olympics WOMplex and attempts to give a flavor of some cutting-edge technology developed for the project.

For the summer of '96, a team of IBM programmers and engineers, based in Southbury, Connecticut, built the official Internet Web site for the Atlanta Olympic Games. The project was immense in every respect—from the amount of content in the site, to the volume of traffic through the site, to the amount of equipment deployed, and the number of hours worked by the team.

WOMplex is an umbrella term that describes the combination of the Web Object Manager (WOM) Internet content and application server, the scalable architecture, the requisite scalable parallel processing hardware, the content authoring environment, the load-balancing systems, the user-level applications on the Web site, and the subsystems that support those applications. In other words, a WOMplex is a scalable, manageable, globally distributable Internet content and application server comprising both hardware and software.

Statistics

The URLs http://www.atlanta.olympic.org and http://results.atlanta.olympic.org together constituted the largest Web server in the world. During the 17 days of the Olympics, the site received just over 192 million hits, with peak daily hits of 16.9 million. Drilling down to the "Sneak Peek" application, which provided a constantly changing montage of captured images from 38 video feeds from the Games in Atlanta, the servers handled up to 21,000 images per hour. What makes these figures especially impressive is that every page served was built dynamically from its component objects at the instant the page was requested.

To provide this level of service 24 hours a day required non-trivial amounts of equipment. The Atlanta site was served by 50 nodes of a distributed IBM RS/6000™ Scalable POWERparallel™ system, spread across five locations in four countries. Most of the processors were at the primary WOMplex site at one of Integrated Systems Solutions Corporation's (ISSC's) major Facilities Management Centers in Southbury. This site provided the required infrastructure for a project of this scale—Uninterruptable Power Supplies(UPS), good physical security, high-bandwidth connectivity into major U.S. Network Access Points (NAPs), and raised floor.

Several nodes—typically some portion of an RS/6000 SP frame—joined the WOMplex at four mirror sites:

- Cornell Theory Center at Ithaca, New York
- IBM U.K. Laboratories at Hursley, U.K.
- University of Karlsruhe near Mainz, Germany
- Keio University near Tokyo, Japan

Each mirror site replicated most of the content of the Olympic site. Notable exceptions were the results system and the rapidly changing Sneak Peek images, which were served only from the primary site at Southbury.



Andy Stanford-Clark

Presented at Get Connected Technical Interchange '96, IBM Hursley U.K., October 1996

IBM 000026

EPIC435843

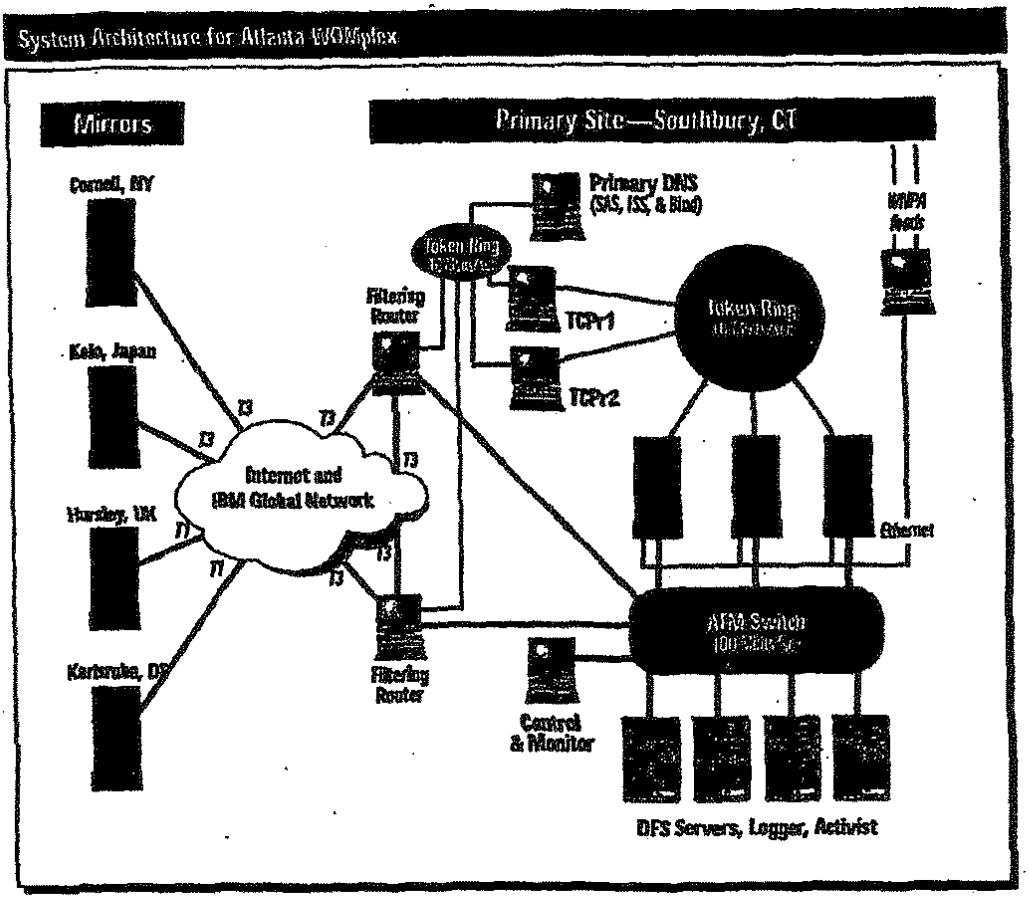


Figure 1. System architecture used for the Olympic games in Atlanta

During the last few months before the games, tickets for seats to events at the games were sold over the Internet by a server based on the IBM Net.Commerce server. In all, this generated over $5 million in ticket sales.

System Architecture

Figure 1 shows the overall system architecture for the Atlanta WOMplex. The "cloud" combines the Internet and the IBM Global Network. T3 connections are DS3 links providing 45 Mbits per second. Filtering routers are Cisco® 7500 Series routers with sets of carefully crafted filter rules to allow only the desired packets in and out from desired sources and destinations. TCPr1 and TCPr2 are the TCP routers, or WOMsprayers—these will be discussed later. Together SAS, ISS, and Bind form the name resolution service. The WNPA feed is a 56 Kbits per second leased-line connection to Atlanta, which provided the feed of results to the databases that were used to drive the Internet Results System. Several leased lines were installed for redundancy. The roles of the machines labeled Logger and Activist are discussed below.

Infrastructure

The IBM RS/6000 SP system provided the WOMplex infrastructure. At the main Southbury site, three frames of the RS/6000 SP were filled with mainly "thin" nodes (equivalent to the RS/6000 Model 390). We did not use the RS/6000 SP high-speed switch in this application—all communication between processors was done over the Asynchronous Transfer Mode (ATM) network.

Control workstations for the main SP frames were in the machine room near the RS/6000 SP. Access to the SP control workstations at the remote mirror sites was handled through X-Windows sessions running across the Internet, using

IBM 000027

REST AVAILABLE COPY

EPIC435844

the machines were generated and then frequently changed. The apparent inconvenience of complex password schemes was easy to justify in the high profile environment of the Olympics.

Transmission of system control information, including machine root passwords, across the Internet was a potential major exposure. Early adoption of Secure SHell (SSH) for all terminal communication between machines solved this problem. AIX® and Windows® 95 clients for SSH allowed developers to use SSH as a direct replacement for the telnet they would have used otherwise. A version of the SSH client that worked through a SOCKS firewall allowed developers inside the IBM firewall to access the servers and development lab machines.

ISS provided another level of security with the "caching, routing name server," which was deployed as the primary name server for the Olympics site. In this Domain Name Server (DNS) emulation mode, ISS had authority over one or more specified subdomains. If ISS received a name resolution request for a subdomain over which it did not have authority or a name that was not being used for load balancing, then ISS passed the request to one of two backend, ordinary, name servers—typically knowledgeable DNSs that could resolve nearly any request.

The WOM system has been iteratively developed to support high-volume Web sites for sporting events.

The IP address of the original request determines the backend name server that receives the request. ISS configuration has a table of subnet mask specifications. The client IP address, or name server, requesting the name resolution is matched against this table. If a match occurs, the client is considered internal, and passed to one backend DNS; otherwise, it is considered external and sent to the other one.

From the practical standpoint, even with a distributed, multi-site operation, the appropriate client machines can access all of the main site's internal server names and addresses, whereas the rest of the world can only access the limited set of names and addresses (possibly only the Internet address) that the DNS administrator made available. The ISS name server also blocks the DNS Zone Transfer operation, a popular method to access lists of internal machines that are considered hacking targets.

The WOM Web Server

The Atlanta Web site used the latest version of IBM's Web Object Manager system, known as the *WOMserver*. The WOM system has been iteratively developed to support high-volume Web sites for sporting events and IBM external Web sites since 1994. Notable events have included Wimbledon® Tennis, Masters Golf, U.S. Open™ Tennis, and the ACM Chess Challenge (Kasparov versus Deep Blue).

Object-oriented Perl was the language used for the Atlanta server. Supporting applications and subsystems on the site were written in C, Perl, and Java™. Several prototype versions of the WOMserver written entirely in Java were also deployed in the WOMplex for experimental, demonstration, and testing purposes. A specialized binary server, based on the Apache Web server and run on a separate port number, handled binary objects—GIF, JPEG, AVI, MPEG, WAV, Java classes, and Bamba audio and video. All references to binary objects in the HTML source dynamically served as pointers to this port (for example, http://www.atlanta.olympic.org:8080/).

The data flow for entering authored content into the WOMserver had several steps. Content was authored by several types of systems, including traditional HTML editors, image editing tools, and a new generation of WOMtools for directly authoring into the WOM internal object format. This content was then "checked in" to the WOM database. Other systems, based on IBM Digital Libraries technology, generated Web pages from dynamically changing data, such as sports results.

WOMwire was the protocol used for communication among the various tools, including several batch upload tools. This protocol defines the way in which objects and their metadata are transmitted to the WOM system. The WOM database used the DB2/6000™ relational model to store the data and metadata object.

The WOMserver can access data and metadata stored in several formats. For the Atlanta WOMplex, the data and metadata were exported as flat files, called *servables*, from the DB2® database into the DFS filesystem. The DB2 export mechanism also triggered the DFS fileset replication system to push the updated files out to all servers at each WOMplex site.

Web-based applications for the Atlanta site would traditionally have been written as Common Gateway Interface (CGI) programs. However, they were written for the WOMserver in an object-oriented Perl or Java environment. They were run inside the server to avoid the overheads associated with executing CGI applications on traditional Web servers.

WOM architecture allows an executable program to generate any part of a served page;

the applications also fit within this paradigm. Detailed discussion of the WOM Network Application Platform is outside the scope of this article.

The WOMserver is *multi-homed*—it can support several distinct Web sites concurrently in the same server. This makes the WOMplex, with global load-balancing systems, a potential home for many Web sites at the same time—ideal for providing content-hosting services.

Centralized Logging

An important aspect of WOMplex operation is collecting HTTP access and error logs from the servers distributed across the world and bringing them into a central location for monitoring and analysis. The "Grim Reaper" performs this function. The Grim Reaper, a tree-structured plumbing of sockets and UNIX® pipes, streamed log records from each server to a confluence point at each site, then across the Internet to a central collection point on a machine called the *logger* (see Figure 1).

When everything is streaming nicely, the task of the Grim Reaper is straightforward. The problems occur when one of three events occurs:

- The servers produce log data faster than the local Grim Reaper can take it from them.
- Network delays or failures cause holdups in the Grim Reaper plumbing, causing a back-log of data.
- The final receiver of the logs cannot cope with the rate at which they are arriving from the Grim Reaper.

Any of these scenarios, coupled with the simple mandate that no log records can be discarded, make the Grim Reaper's job very difficult. As a result, the Reaper is a sophisticated technology. At any stage, log streams can be diverted into a file on a local disk to hold data, which for some reason, cannot be sent on. Later, when connections are restored or congestion eases, files are spooled back into the network. The Grim Reaper also can deal with requirements such as off-line transfer of batched log data (for example, late at night with less network traffic) and on-the-fly data compression for transmission across slower network connections.

After arriving at the logger machine, the log streams were transferred across a high-speed protocol Common Link Access to Workstation (CLAW) link to an 8-way CMOS 390 Squadron mainframe. The logs were then pre-processed by a series of pipes and filters, then implemented using MVS® Pipes. Individual log records were added to a table in a DB2/MVS database. From here, SQL queries could extract useful information from the logs, including hit counts and rates, online data mining using Activist, and real-time "click tracking." The raw log data was spooled to tape for later analysis.

Content Authoring Environment

The WOMwire protocol defines how authoring and site management tools communicate across the wire with the WOM database. The client part of the WOMwire protocol was implemented as an ActiveX control. The authoring tools, developed for the Windows 95 and Windows NT™ environments in Visual Basic®, C, or C++, could trivially access the WOM database by including the WOMwire control.

WOM objects can contain text, graphics, sound, video, page layout information, or executable code. These objects and their associated metadata can be checked in and out of the system, similar to a programmer's source code control system. The WOMbler created and edited content and managed metadata. WOMexplorer was used to navigate and reorganize a WOMplex site.

WOM objects can contain text, graphics, sound, video, page layout information, or executable code.

Each page on the site fits into an information taxonomy, described by a set of taxonomy tags. These tags position the page somewhere in an *information space*, which has a different hierarchy than the *data space* hierarchy imposed by the HTML links between a set of pages. This exciting new approach to Web site design is outside the scope of this article.

A Visual Basic application called the *Wu-tool classifier* was used to classify existing pages on the site. Users could classify each page into the information taxonomy by choosing from sets of predefined categories that applied to the page in question.

With the WOMpix tool, users could process images by cropping, resizing, reformatting, and labeling, before checking them into the WOM database. WOMpix dealt with many thousands of images that appeared on the site during the Olympics. It also allowed metadata, such as the source agency and photographer, to be recorded with the image. WOMpix could also generate thumb-nail images and image water-marking.

Load Distribution

Load distribution and balancing within the WOMplex uses a two-tier architecture. At each site, local load balancing distributes incoming traffic toward a single IP address across multiple backend WOMservers. Between sites, a global load-balancing system allows logical Web sites to be moved between physical server locations.

Interactive Session Support

Both tiers of the load-balancing system use software called Interactive Session Support (ISS). This is the interactive component of the RS/6000 SP LoadLeveler™ package for batch and interactive load balancing across a set of RS/6000 SP nodes or across other UNIX workstations, notably RS/6000, HP™, Sun®, and Silicon Graphics.

ISSmonitor collates agent responses, starts and stops remote agents (using inetd), and reacts appropriately if agents do not report in for an extended period of time.

Each server runs the ISSagent, which runs a user-defined metric that determines how busy the server is. The metric can be any executable program or pipeline of UNIX commands, providing it returns an integer at the end. For example, a popular choice is `ps -ef | wc -l`, which reports the number of processes currently running on the system and provides a good measure of machine activity. Far more complex metrics can be contrived, including an executable program to generate the metric value. The metric used for the Olympics counted the number of established socket connections on port 80 of each server.

ISSagents periodically report to a central ISSmonitor, which collates agent responses, starts and stops remote agents (using `inetd`), and reacts appropriately if agents do not report in for an extended period of time. The ISSmonitor also uses Internet Control Message Protocol (ICMP) ping packets to ensure that servers are still "alive" before recommending them for use. The ISSmonitor transfers a *load profile* of server activity to the WOMsprayer via a socket. The WOMsprayer uses the load profile to compile a weighted round-robin schedule for incoming traffic. Any server whose ISSagent does not respond is eliminated immediately from the load-balancing schedule to prevent incoming traffic from hitting a non-responsive server and receiving `Connection Refused` error message.

In another operational mode, ISS works as a load-balancing Domain Name Server (DNS). A generic Internet name, such as www.atlanta.olympic.org, is assigned to the cluster of servers. When a client machine or another name server in the hierarchical DNS system requests the IP address for that name, ISS ensures that the name server returns the IP address of the least heavily loaded server to provide the required service. ISS can work in DNS mode in one of two ways.

ISS can cooperate with an existing DNS, by regularly updating the "name to address mapping" database of the name server. Although this works well for low volume, long-lived connections, it is not responsive enough for high-volume, short-lived transactions usually experienced by a Web server.

ISS also operates as a high-performance, load-sensitive DNS, totally replacing the existing name server. This mode of operation can handle Web server traffic and provide a weighted round-robin mechanism for distributing high-volume load evenly across the backend servers.

ISS also offers an enhanced security, caching name server facility, described in the Security section above.

Local Load Balancing—WOMsprayer

Load-balancing software technology from IBM Watson Research has several names: TCP router, Encapsulated Cluster, WOMsprayer, ShockAbsorber, and Network Dispatcher. The *WOMsprayer*, a network router between two physical networks, offers a single IP address to the world. When a TCP connection is made to that address on the inbound network, the WOMsprayer software modifies the Medium Access Control (MAC) address field of the incoming connection request packet and transfers the packet to its outbound network. The MAC address change redefines the final packet destination, and the WOMsprayer arranges for it to be delivered to one of several backend servers.

The backend machines have unique IP addresses and also have the IP address of the WOMsprayer aliased onto their inbound network interface. Consequently, when this modified connection request packet arrives at a backend machine, it appears that the packet was destined for this server; the server does the required "accept" to create the TCP session. The WOMsprayer maintains a table of currently active connections, so any further incoming packets for that TCP session, such as packet acknowledgments, are routed to the correct backend server.

A weighted round-robin schedule determines which backend server receives an incoming session request. This schedule is constructed from the server activity profile that the ISSmonitor periodically supplies to the WOMsprayer. If ISS has not heard from a server or has reason to believe that a server is having problems, the

BACK TO CONTENTS

server is quickly excluded from the WOMsprayer schedule. No new connections are routed to that server until it rejoins the cluster.

The backend server's default route serves as the network path from the server to the client browser. This route generally does not use the same machine as the WOMsprayer. Even if the same server is used, the WOMsprayer does not intercept the returning packets.

Global Load Balancing

Two mechanisms allow Web traffic to be spread effectively across the multiple sites comprising the WOMplex: ISS-DNS and the WOMbat.

ISS-DNS

In its DNS emulation mode, ISS allows the DNS administrator to configure a table of Internet domain names to IP address mappings. This differs from an ordinary DNS in that more than one IP address may be associated with any given Internet name. An ISSagent runs on each server that is designated to serve any given name. According to the health, load, and the configured balancing policy for that name, incoming resolution requests for that name receive the IP address of the most appropriate server.

In the WOMplex, the ISSagents run on the machine that hosts the WOMsprayer for each site. Two balancing policies can be configured: Round and Fail.

Round. This policy does a weighted round-robin of the IP addresses in the list associated with that name. It is distributed according to two factors:

- The server activity profile, as determined from the input provided by the ISSagents
- A server weighting factor, allowing for different server capacity at each site—different numbers of WOMserver nodes in the RS/6000 SPs at each site, in the case of the WOMplex

Fail. This attempts to use the first IP address in the list—the "preferred" site—until that site becomes unavailable. Then, ISS switches to a weighted round-robin of the remaining IP addresses in the list, the same as in the Round policy. This system allows a single, logical Web site (defined by a specific Internet name) to be distributed across multiple physical locations, which could be a cluster of servers controlled by a WOMsprayer. Adjustments to the system configuration can be made dynamically, allowing extra servers to be switched-in, if necessary, to cope with excess load. The system enables automatic fail-over to other servers if a server experiences downtime, which can be fail-over to other servers from a preferred or primary server or omission of unresponsive servers from the round-robin schedule.

WOMbat

WOMbat uses ping to "echo-locate" a client machine and determine the nearest WOMplex site in network round-trip terms. With this knowledge, WOMserver can then divert the client to its nearest server site, and hopefully receive better service, thanks to the better network connectivity to that site.

When a client browser visits a WOMplex site for the first time, the server knows it has not seen that client before, because the client does not have a browser *Cookie* for that site. The server generates a Cookie, a unique key to identify that client in the future, and sends it to the browser to use whenever it visits that site. The Cookie is also stored in the Global Profile Daemon (GPD), a database of profile information, keyed by browser Cookie. The server also invokes— asynchronously in the background—the WOMbat Driver before composing the HTML page requested by the client and returning it to the client.

The WOMbat Driver sends details of the client's IP address and its browser Cookie to a WOMbat daemon at each site in the WOMplex. The WOMbat daemon sends a volley of ICMP echo request packets (ping) to the client IP address and measures the round-trip time of any responses that return within a pre-specified time-out period.

WOMbat daemons send back to the WOMbat Driver the average network response time taken to ping the client machine. The WOMbat Driver waits for a specified length of time for the WOMbat daemons to respond, then determines which server is closest to the client by comparing the responses. Server availability information also can be factored into this calculation, so that the closest server might be rejected in favor of a more distant, but less busy server. The user's profile in the GPD contains the identity of the preferred server.

The next time the user requests a page from the Web server, the server daemon interrogates the GPD using the browser Cookie as a key, and discovers that the preferred server for that client is not this server. The WOMserver then builds the requested page for the client, but rewrites all embedded links to follow-on pages and to

In its DNS emulation mode, ISS associates more than one IP address with any given Internet name.

[redacted] BACK TO CONTENTS

embedded images as URLs on the preferred server rather than on local links. For example, if the preferred server was www.hursley.atlanta.olympic.org and the local link would normally have been the following:

```
<IMG SRC="/data/image1.gif">
```

then, the link sent back to the client is rewritten as follows:

```
<IMG SRC="http://www.hursley.atlanta.
olympic.org/data/image1.gif">
```

When the client browser follows any of these links, they are retrieved from the preferred server. From that point on, all links will point back to that server, so the client has effectively been "moved" to that server.

WOMplex separates logical services from physical servers, which allows a Web site to be hosted from any site in the WOMplex.

Virtual Addressing

AIX and other UNIX operating systems can alias multiple IP addresses onto a single network interface; that is, the server can "pretend" to have several different IP addresses on the same network at the same time. This was used two ways in the Atlanta WOMplex: multi-homing and service virtualization.

Multi-Homing. A single WOMserver instance can serve several Web sites. The IP site addresses are aliased onto the server's Internet network interface. The WOMserver configuration has separate "trees" of URL data for each hosted address. When a Web page request arrives, the WOMserver interrogates the socket connection to identify which IP address it was sent to, then serves the appropriate page from the correct data tree. Many "standard" Web servers consider this a standard feature, such as Virtual Hosting in Apache.

Service Virtualization. WOMplex separates logical services from physical servers, which allows a Web site to be hosted from any site in the WOMplex. IP aliasing within a site separates logical from physical and frees any dependence on specific hardware. All externally visible IP addresses are aliases, not the "real" base IP address of any machine. Therefore, if a server fails, the service IP addresses supported by the failed server can be transferred readily to another machine. One additional requirement, apart from adding the IP address aliases, is to flush the arp cache on the nearest network router, which forces it to reestablish the new IP to MAC address mappings. The System Network Management Protocol (SNMP) interface to the router can often automate this operation.

Global Profile Daemon

The Global Profile Daemon stores and caches client profile information, using a client browser Cookie as a retrieval key. Any WOM application or the WOMserver itself can store profile information in GPD. Profile data is held as a file of lines containing `name = value` pairs. Keys used for the Atlanta WOMplex were constructed from the concatenation of the following:

- Three randomly chosen letters
- Cookie creation timestamp (seconds since Epoch, plus an offset)
- Cyclic Redundancy Check (CRC) checksum digits

GPD uses a multi-level cache to balance the global distribution requirements of profile data and fast access to user profiles by the WOMserver. Closest to the WOMserver, GPD maintains an in-memory cache with a Least Recently Used (LRU) policy; therefore, people who are known to be currently moving around the Web site are most likely to be in memory.

To locate profiles efficiently, the directory structure has directory names 'A' through 'Z'—three levels deep. For example, the profile with the key ADF1234567123 would be found in the file with the following full path: /dfs/profiles/A/D/F/3217654321FDA.

Note that the name given to the profile file was the reverse of the key string, that is, 3217654321FDA. This is because DFS uses its own hashing algorithm for efficient searching of directory contents—based on the first character of the filename. Consequently, the first character of the filename should be a random character. This avoids having all files in a directory starting with the same first character, and thus all falling into the same hash bucket. We chose three directory levels and 26 entries per level, knowing that DFS caching is optimized for a certain maximum directory content size.

At the next level, DFS caches recently retrieved profiles on the local disk of the server machine. At a site level, recently accessed profiles are stored in a DFS fileset that physically resides on disks at that site. This avoids trans-Atlantic connections for many profiles. Finally, the main DFS server at the primary WOMplex site in Southbury stores the global GPD repository.

When a client comes back to the Web site (at any physical location) with a previously assigned Cookie, the WOMserver requests information on that client's profile from its local (on the same machine) GPD instance. GPD checks the in-memory LRU cache and returns the information if it is there. If not, GPD must fetch the profile from the global repository because that client has not been to this physical site recently.

GPD reads the profile from the global DFS directory and copies the file to a local DFS fileset located physically on the local site. Then GPD sends a UDP broadcast to other servers at its local site informing them that the client with a specified Cookie has come to the site, and may soon request pages from them. Other GPD instances at that site receive the broadcast and go to the local DFS fileset to read the profile information into the memory cache. Thus, only the first GPD at a site has to do the trans-Atlantic journey to get the profile data of a returning client. The other servers (who are likely to receive requests from that client in the near future, thanks to the WOMsprayer) pick it up from a fileset that is local to their physical site.

The CRC digits are built into the Cookie. This makes it more difficult for anyone to tamper with a Cookie while trying to obtain profile information about another user by reverse engineering their Cookie. Cookies with invalid checksums are not accepted by the WOM system.

Applications

This section describes applications that were designed and built for the Atlanta Web site. For various reasons, not all of these went into full production on the site. They were all working in the lab, so we have included them here for the sake of completeness.

Sneak Peek

Sneak Peek was the largest "Net Cam" ever. Live video feeds from 38 locations at the Atlanta Games were fed to "The Fishbowl" in Atlanta. Here they went through a bank of "frame grabbers" and were checked into the WOM system. Images were captured approximately every five seconds from each feed. Because of the high volatility of this data, the images were not stored into the WOM DB2 database, but went directly into DFS. HTML pages to render the latest set of images and to navigate between Olympic venues were generated on-the-fly to pick up the most recent image file names.

The human operator in the Fishbowl received all feeds concurrently and could select channels that currently had activity on them. The navigation options reflected the choices presented by the dynamic HTML pages in the Sneak Peek section of the site. The operator could choose the most stunning images and build a daily "scrapbook" of images worth keeping. These images were titled, cropped, assigned metadata using the WOMpix tool, and checked into the main WOM database.

Sneak Peek images were not mirrored across the world because of their highly volatile nature—they were all served from the primary WOMplex site at Southbury. This became an extremely popular application, and many hundreds of thousands of images were downloaded.

Bamba

Bamba is IBM's streaming audio and video technology for the Web based on G.723 low bit-rate streaming standards. For more information about the technology and to download the decoders and encoders, visit the Bamba home page on the IBM AlphaWorks™ Web site (http://www.alphaworks.ibm.com/).

Bamba has three advantages over other Internet audio and video formats:

- Data is streamed, so you can view and download it simultaneously, avoiding huge waits while the whole file downloads.
- The content is encoded at a low bit rate, making it suitable for transfer over a standard 28.8 Kbit modem.
- G.723 is a standard.

Three forms of Bamba content were used at the Atlanta Olympics:

- **Audio clips:** Recordings of many interviews with athletes, Olympics organizers, and news reports. A regular Web server handles Bamba audio clips in the same way as GIF images. On the Atlanta site, they were one data type served by the binary server.
- **Video clips:** Encoded highlights of the sports events. Bamba video clips are also regular binary content that can be served by an ordinary Web server. They were served from the binary server during the Olympics.
- **Live audio:** Atlanta local radio station, WGST Atlanta, was broadcast over the Internet 24 hours a day during the Olympics. Bamba

Sneak Peek images were all served from the primary WOMplex site at Southbury.

Reflector allowed many people to listen to the radio station concurrently.

Bamba Reflector

In typical use, a small Pentium™ PC would do real-time Bamba audio encoding. It is not realistic to have hundreds or thousands of clients connecting to this machine to listen to the audio stream, so Bamba Reflector was created. The Reflector connects to the transmitter as the only client to that machine. It then accepts connections from Bamba clients. Whatever the Reflector receives from the transmitter is transferred to the clients. There must be a certain amount of buffering and flow-control to deal with clients connected over different speed links. The Reflector intelligently drops Bamba packets without losing stream synchronization for any clients who fall too far behind in receiving Bamba packets. Each Reflector can support many hundreds of client connections concurrently. One Reflector can act as a source to other reflectors, so a whole world-wide tree of Reflectors can potentially serve many thousands of clients across the globe.

Users could specify to the WOMserver the specific sports, countries, and athletes of interest to follow as the Games progressed.

The Reflector configuration can limit connections to a specified upper limit or it can be off-line—not broadcasting at the moment. When a connection is not possible, it can send out an information message in one of three forms: a pre-recorded audio message built into the client, a text message sent from the Reflector, or an audio message sent from the Reflector. Various combinations are also possible.

ISS load balancing can be used in front of a bank of Bamba Reflectors to balance incoming connection requests to a single advertised Internet name across the available servers to ensure an even load across the Reflectors.

WOMplex Boiler Room

The Boiler Room monitoring system allowed statistics to be gathered from various parts of the WOMplex, its applications, and subsystems. The system structure is described below.

Each application to be monitored had a `funneld` (funnel daemon) client. To make this a simple task, developers had client code for Java, Perl, and C available. The protocol used throughout this system was Internet Relay Chat (IRC), transmitting data in a textual format with each monitored system using a different IRC channel number. The client code connected to a `funneld` instance ran on each server. The `funneld` instances connected to other `funnel` daemons in a tree structure. An IRC server was at the root of the tree of `funnel` daemons.

The Boiler Room monitors were Java applets embedded in HTML pages on the WOMplex Web site. These applets used IRC client classes written in Java that connected back to the IRC server. The monitor applications then tuned in to the IRC channels required to construct their displays and waited for information to arrive. A variety of dials, counters, and indicators were implemented as Java classes that could be combined in various ways to display the statistics for any system.

Since the Boiler Room system could also tail log files and receive data through sockets and Interprocess Communications (IPC) message queues, it was easy to hook up monitors for almost anything. A special *chart recorder* client logged IRC channel data for subsequent analysis.

Information Taxonomy

Early in this project we realized that a very large site needs some kind of information taxonomy, or hierarchy, mapped to the site; otherwise, it becomes completely unmanageable and almost impossible to navigate. This important issue required the skills of an information architect.

The metadata for every page, image, and multimedia clip checked into the WOM database included a set of category tags. These tags position the object in the *information space* of the Web site. This information space is dimensionally exclusive from the *data space* imposed by the HTML hyperlinks built into the documents. Tools, such as the Wu-tool, categorize pages on the site. Once the whole site has been categorized, facilities can be provided to navigate the site in information space, as well as the traditional HTML navigation. For example, someone looking at pages about Men's Canoeing might also be interested in other pages that are "close" in the information space. For example, in this case, Kayak events might be of interest.

As an interesting new area for Web site navigation and management, this will undoubtedly play an important role in the future.

Personal Home Page

The combination of generating dynamic pages in the WOMserver and identifying individual clients from their browser Cookie allowed a personal home page to be created for each visitor to the site.

Users could specify to the WOMserver the specific sports, countries, and athletes of interest to follow as the Games progressed. If they provided

Secure SHell (SSH) (described below) for secure connections. The Operations Center in Southbury could control every processor in the WOMplex—right down to power cycling.

ATM Network

The outbound network for data from the Web site was routed over a 100 Mbits per second ATM network. ATM uses centralized switching technology to provide a high-bandwidth network. The site networking was highly complex, with multiple Token-Ring, Ethernet™, and ATM networks connected to the Internet through two filtering routers, IBM 8260 switching hubs enabled this level of connectivity. A 16 Mbits per second Token Ring handled the inbound networking. *Note:* Network requirements for a Web site are highly asymmetrical—one packet that contains an HTTP request can provoke the transfer of potentially many megabytes of data from the servers.

Transarc DFS Cell

Transarc's Distributed File System (DFS) ensured that all WOMplex Web servers had access to up-to-date information. (For those familiar with Andrew File System (AFS®), think of DFS as the "grown up" AFS.) DFS is based on the fileset concept, a collection of files handled in the same manner as a single group. Each fileset in a DFS cell resides on a nominated server; it may have mirrors on other nominated servers. Fileset replication happens in two ways: on a timer interval or (often more useful) on a manual "push" from the primary server, when the application or user knows that a batch of updates has been completed.

Each WOM server in the WOMplex was a DFS client configured to request files from the nearest DFS server. At remote mirror locations, the clients retrieved their data from a local server that was kept in sync with the main servers in Southbury. One big advantage of DFS in the Web server environment is client caching. During the Olympics, most clients achieved a 98% cache hit rate.

The primary DFS server in Southbury was distributed across three RS/6000 Model 570 servers and two RS/6000 SP wide nodes. It used 128 GB of IBM 7137 RAID storage, configured with replication and mirroring to give 32 GB of effective storage with three-way redundancy in addition to the RAID failsafe.

The entire cell was administered centrally from Southbury. Problems with filter rules on routers and TCP/IP routing across the Internet make setting up a very large distributed DFS cell a complex process; although when it is running, the system works extremely well. The single DFS cell spanned six locations: Southbury (the main server), Cornell, Keio, Hursley, Karlsruhe (the mirror sites), and Atlanta, where the content was being generated for the Web site.

Internet Connectivity

To ensure the availability of enough bandwidth for the Olympics, four DS3 circuits (T3—45 Mbits per second) were connected into four major U.S. Network Access Points (NAPs): Mae-East, Mae-West, Sprint NJ, and Ameritech™ Chicago. Regardless of the origin of a request, by using a routing algorithm that minimized hop counts, the response was sent to the major NAP closest to the source of the request. Most of the world's major non-U.S. service providers connect directly into one of these four NAPs, so the site was well connected. Multiple circuits also protected the site against any link failure. Packets were automatically routed over other links or could be manually routed around trouble spots, which occurred from time to time at various NAPs.

Security

The entire Atlanta Internet site, the software development environment, and all DFS infrastructure were located outside the IBM firewalls. Consequently, site security was always a high priority for everyone because such a high visibility site was an obvious target for hackers.

High security must be built into a site from the ground up; it cannot be retrofitted onto an existing site. The filtering routers provided the main protection against invasion, and the filter rules for these machines were a major design and maintenance effort. Interworking with the remote mirror sites involved setting filter rules at the remote sites to allow only User Datagram Protocol (UDP) and Transmission Control Protocol (TCP) packets on specific port numbers in and out from specified Internet Protocol (IP) addresses.

A major problem for the primary site at Southbury was the need for extensive access to systems inside the DMZ ("De-Militarized" Zone) at the remote sites; the rest of the world could only have port 80 World Wide Web access. Heterogeneous router technology across the world ensured that many people learned a lot about writing filter rules in a short space of time!

Kerberos and authentication servers were used to secure DFS. Sophisticated password schemes for

High security must be built into a site from the ground up.

broadcast. The client applet received the play list and modified its display behavior accordingly, adding new items as they were transmitted and deleting old items that were no longer in the play list. The producers relied heavily on the information taxonomy to determine which new items being checked into the site were of interest to their programs.

The client applet used HTTP to retrieve non-text items, such as images, from the main Web site, using a URL sent over IRC. The applet also used HTTP to retrieve a file containing the initial set of current content when the applet was first invoked.

Conclusion

The Atlanta Olympics Web site was the largest Web site in the world and handled an immense amount of traffic during the 1996 Summer Games. The WOMplex technology was pioneered to support large sporting events on the Internet. The Olympics gave the WOMble Team a chance to push forward the technology frontiers, and has paved the way for products using WOM technology in the future.

The WOMserver was designed as an Internet application platform, of which a sporting event server was just an application. The suitability of this architecture as an application environment has been proven in the "trial by fire" experienced during the Olympics.

This experience greatly advanced load balancing and global scalability. The systems deployed for the Olympics showed that a Global WOMplex is a viable concept.

The author gratefully acknowledges the contributions of Sean Martin and David Grossman to this article, and also wishes to thank the Team at the IBM Southbury lab.



Andy Stanford-Clark, IBM Corporation, 150 Kettletown Road, Southbury, CT 06488. Internet: andysc@vnet.ibm.com. Dr. Stanford-Clark is an Internet advanced technology consultant in IBM's Internet Division. His responsibilities include exploring emerging Internet technologies and working with customers to implement Internet-based projects. Dr. Stanford-Clark is currently in the U.S. on international assignment from IBM Hursley, U.K. He has a BS in Computing and Mathematics and a PhD in Parallel Computing from the University of East Anglia in the U.K.

Exhibit 15

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

CH MF

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.
08/794,269	02/03/97	ADINUTHULA S	3015-060

EXAMINER	
LAU, S	
ART UNIT	PAPER NUMBER
2755	19

LMC1/0830

HICKMAN PALERMO TRUONG & BECKER
1600 WILLOW STREET
SAN JOSE CA 95125-5106

DATE MAILED: 08/30/00

Please find below and/or attached an Office communication concerning this application or proceeding.

Commissioner of Patents and Trademarks

TD

Office Action Summary

Application No.	Applicant(s)
08/794,269	Adunuthula, et al
Examiner	**Group Art Unit**
S. Lao	2755

[X] Responsive to communication(s) filed on *Jun 8, 2000*

[ ] This action is **FINAL.**

[ ] Since this application is in condition for allowance except for formal matters, **prosecution as to the merits is closed** in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11; 453 O.G. 213.

A shortened statutory period for response to this action is set to expire *3* month(s), or thirty days, whichever is longer, from the mailing date of this communication. Failure to respond within the period for response will cause the application to become abandoned. (35 U.S.C. § 133). Extensions of time may be obtained under the provisions of 37 CFR 1.136(a).

Disposition of Claim

[X] Claim(s) *1-12 and 14-36* is/are pending in the applicat

Of the above, claim(s) _____ is/are withdrawn from consideration

[ ] Claim(s) _____ is/are allowed.

[X] Claim(s) *1-12 and 14-36* is/are rejected.

[ ] Claim(s) _____ is/are objected to.

[ ] Claims _____ are subject to restriction or election requirement.

Application Papers

[ ] See the attached Notice of Draftsperson's Patent Drawing Review, PTO-948.

[ ] The drawing(s) filed on _____ is/are objected to by the Examiner.

[ ] The proposed drawing correction, filed on _____ is [ ] approved [ ] disapproved.

[ ] The specification is objected to by the Examiner.

[ ] The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. § 119

[ ] Acknowledgement is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d).

[ ] All [ ] Some* [X] None of the CERTIFIED copies of the priority documents have been

[ ] received.

[ ] received in Application No. (Series Code/Serial Number) _____.

[ ] received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

*Certified copies not received: _____

[ ] Acknowledgement is made of a claim for domestic priority under 35 U.S.C. § 119(e).

Attachment(s)

[X] Notice of References Cited, PTO-892

[ ] Information Disclosure Statement(s), PTO-1449, Paper No(s). _____

[ ] Interview Summary, PTO-413

[ ] Notice of Draftsperson's Patent Drawing Review, PTO-948

[ ] Notice of Informal Patent Application, PTO-152

--- SEE OFFICE ACTION ON THE FOLLOWING PAGES ---

ORCL01623264

DETAILED ACTION

1. Claims 1-12, 14-36 are pending. This action is in response to applicant's response filed 6/8/2000.

2. The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior Office action.

3. Claims 1, 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al in view of Heizer and Kahn et al.

As to claim 1, Putz teaches (abstract, col. 9, lines 21-30; col. 10, lines 20-37; col. 12, lines 33-55) a method for responding to a request (remote procedure calls) issued to a server (main server program 55) from a client (client 52-54) over a network system, the method comprising the steps of:

the server obtaining the request over a network (receive remote procedure calls);

the server identifying (check program dictionary) a program type corresponding to the request (named database operation program) among a plurality of program types (operation programs NewDocDecs, Render, DescriptionSearch),

if no currently executing instance of the program type (named operation program) exists, selectively initiating an instance of the program type (fork a child process of the named operation program),

the server dispatching the request to the instance (fork, then pass the remaining operating program arguments),

executing the instance to process the request (invoke operating program for performing),

responding to the request (return results of the operation).

Putz does not teach (1) the step of initiating is based on a comparison between how many instances are executing of the program type is executing and a prescribed number

of the program type, (2) determining whether an instance of the program type currently exists that is not currently processing a previously received request.

As to (1), Heizer teaches a method for responding to a request (request) issued to a server program type (server 122) from a client (client 103), including initiating an instance of the program type (server process 202, 203) based on a comparison between how many instances are executing of the program type is executing (fig. 3) and a prescribed number of instances of the program (low/high water mark); see col. 2, line 20 - col. 4, line 55; fig.s 1-5.

Putz creates an instance of a named program type for each client request of that named program type, which eventually will cause server overloading. This the aspect of the request dispatching that Heizer addresses to improve. (Col. 1, lines 21-41). Therefore, one of the ordinary skill in the art would have been motivated to apply the teaching of Heizer to Putz's dispatching of each named program, to prevent server overloading. The combination would have provided maximum number of instances for each named program type.

As to (2), Kahn teaches management of server resource (internal reader), wherein if there is a server instance currently exists (previously created internal reader) that is not currently processing a previously received request (which has finished processing and waiting for additional work, idle), an incoming request/job is assigned to such server instance. Otherwise, new server instances are initiated/created to handle the request/job. See abstract, col. 13, lines 40-63, col. 15, lines 35-61, col. 18, lines 11-35. Kahn determines whether there is an available/idle/not_currently_processing server instance by checking the IDLE flag for an internal reader. It would have been obvious to include the teaching of Kahn into Putz as modified so as to provide handling of changing work flow without excessive system overhead (Kahn, col. 18, lines 31-35).

As to claim 25, it is program product claim of claim 1.

4. Claims 2-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer and Kahn et al and further in view of Donica et al.

As to claim 2, Donica teaches responding to a client request, including obtaining the request by a server process (scheduler) executing in a first address space (scheduler address space 30), initiating an instance of a service program (initiator) in a second address space that is separate from the first address space (initiator address space 40 separate from 30), see col. 5, lines 31-34. Since Putz, Heizer, Kahn et al and Donica address load balancing and reducing operating system overhead, it would have been obvious to combine the teachings.

As to claim 3, Donica teaches after executing an instance, maintaining it in memory for executing a subsequent request (idle initiators already assigned to a class), see col. 7, lines 19-24. Note discussion of claim 2 for motivation to combine.

5. Claims 4, 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer and Kahn et al as applied to claims 1, 25 and further in view of Tobe et al.

As to claim 4, Tobe teaches network resource management, including detecting (detect by the system) a fault in an instance during execution thereof (node goes down); and aborting the instance in response to detecting the fault (back up node takes over), see col. 9, lines 42-52. Since Putz, Heizer, Kahn et al and Tobe address resource management and overhead, it would have been obvious to combine the teachings.

As to claim 26, it is program product claim of claim 4.

6. Claims 5-9, 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer and Kahn et al as applied to claims 1, 25 and further in view of Donica et al and Atkins et al.

As to claim 5, Donica teaches specifying at least one of a maximum (maximum number of initiators) and a minimum (minimum number of initiators) number of instances (col. 6, lines 3-5, fig. 2).

As to that each instance executes within an address space that is separate from the address spaces used by other instances of the program, it is taught by Atkins in that each node holds an instance (copy) of a service program (shared object), see pg. 3, left col.,

lines 7-20. Since Putz, Heizer, Donica and Atkins address server resource management, it would have been obvious to combine the teachings.

As to claims 6-9, Heizer teaches initiating a specified minimum number of instances (one server process 202) prior to obtaining the request over the network (on start up, col. 3, lines 16-17), after obtaining the request over the network, determining if one of the minimum number of instances is available for processing said request (first request, col. 4, lines 3-13), after obtaining the request over the network, initiating a new instance of the program if none of the minimum number of instances is available for processing said request and the number of instances is less than the maximum number of instances (third request, col. 4, lines 17-29), executing said instance includes executing an available instance from the minimum number of instances (first request, col. 4, lines 3-13), returning the request to the network without processing said request based on the prescribed number of said instances exceeding a maximum value (21st request, col. 4, lines 52-56).

As to claim 27, it is program product claim of claim 5.

7. Claims 10-11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer and Kahn et al as applied to claim 1 and further in view of Donica et al and Travis et al.

As to claims 10-11, Donica teaches specifying a maximum number of instances, as discussed on claim 5. Travis teaches registering a plurality of programs (applications) with the server (register, col. 17, line 58 - col 18, line 45; server registration, col. 18), specifying a virtual path for a program (names of classes/messages), request (invocation, message) identifies a virtual path (message names), identifying the program (method to perform the operation) based on the virtual path identified in the request (select from the database, Invoker Operation, col.s 25-29), Since Putz, Heizer, Kahn et al, Donica and Travis address request allocation, it would have been obvious to combine the teachings.

Travis teaches specifying other data such as performance information (parameters supported, col. 18, lines 20-29), it would have been obvious to specify a maximum number of instances which is commonly regarded as performance information.

8. Claims 12, 21-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer and Kahn et al as applied to claim 1 and further in view of Djakovic.

As to claim 12, Heizer teaches receiving the request from a transport protocol process (protocol/network software 113), but fails to teach receiving is according to a first protocol and converting the request to a second protocol independent from the first protocol.

However, Djakovic teaches receiving a request (job submitted by client) according to a first protocol (source protocol) and converting the request to a second protocol independent from the first protocol (form a new job in target protocol), see col. 1, lines 40-51; fig. 3. Since Putz, Heizer, Kahn et al and Djakovic address server processing clients request/job, it would have been obvious to combine the teachings.

As to claim 21, note discussion of claim 1 for obtaining, step A)-C), and note discussion of claim 12 for a transport adaptor / means configured to perform protocol conversions. Heizer further teaches forwarding the request to a dispatcher (control process 204 accepts new client) executed by the server; and processing the request by causing the dispatcher to perform the step D) if no instance is available and the existing number of instances exceeds the maximum prescribed number, then sending said reply over the network indicating the request was not processed (when the maximum number of server processes is set at 20, request 21st is rejected, step 317), see col. 3, lines 10-23; col. 4, lines 3-56. Note discussions of claims 1, 12 for motivations to combine.

As to claim 22, note discussion of claim 6.

As to claim 23, Heizer teaches sending a replying indicating the request is not processed (reject, step 317), and sending such a reply when no program configured to handle the request would have been an obvious condition to apply.

As to claim 24, Heizer teaches server process (control process 204), and plug-in routine added to the server process (library 121) .

9. Claims 28-30, 34, 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al as applied to claim 1 and further in view of admitted prior art (pages 1-3).

As to claim 28, note discussion of claim 1, in particular, note claim 1 for receiving request and sending response, for dispatcher selectively dispatching / dispatching and sending the response / responding, for not currently processing and each program having a prescribed number of instances / maximum / minimum. The instances (server process) of a program type of Heizer reside at respective address spaces (202, 203) since they execute on their respective clients.

Further, Heizer teaches network listener (protocol/network software 113), each program type configured to perform an operation that generates an output in response to receiving a request (a server process is to receive a client request, process the request and send a response).

As to that the dispatcher is in a plug-in format, it is taught by admitted prior art (plug-in extensions, page 2, last para.). It would have been obvious to include the plug-in format of admitted prior art into the system/method of Heizer as modified to provide dynamic response to request (page 2, last para.).

As to claims 29-30, admitted prior art teaches receiving the request and sending the response based on Hypertext Transfer Protocol (HTTP) (HTTP, page 1, lines 9-10), to output a static Hypertext Markup Language (HTML) page (page 1, last para.). A daemon is a well known means for performing services.

As to claim 34, Heizer teaches initiating a prescribed minimum number of said instances of the corresponding program (one instance, see rejection of claim 13). Each execution engine is taught by well known distributed network managers.

As to claim 36, Putz teaches registering (the program directory) program types (named database operations) with a dispatcher (main server program 55) for dispatching client requests. (col. 10, lines 31-37). Typically registering with a program directory includes registering the locations/addresses.

10. Claim 35 is rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al in view of admitted prior art as applied to claim 28 and further in view of Donica.

As to claim 35, Donica teaches configuration library (PARMLIB) identifying for each of the programs the corresponding object type (class) and the prescribed number of instances (maximum and minimum numbers of initiators), see col. 6, lines 1-5; col. 8, lines 27-31. As to plug-in format, note discussion of claim 28. Note rejection of claim 2 for motivation to combine.

11. Claim 31 is rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al in view of admitted prior art as applied to claim 28 and further in view of Djakovic.

As to claim 31, note the discussion of claim 12.

12. Claim 32-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al in view of admitted prior art as applied to claim 28 and further in view of Page et al.

As to claims 32-33, Page teaches a plurality of execution engines to control the execution of program instances and to receive the request (adapters to DCE, DDE, LU6.2) (distributed brokers), API specifying predetermined operations, including at least one of initialization (REGISTER), execution of the request (SEND), and shutdown (EOC, ABENDS), see fig.s 8, 17B, 21, 23. Since Putz, Heizer, Donica, Atkins, APA, Travis and Page address client server management, it would have been obvious to combine the teachings.

13. Claims 14, 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al as applied to claim 1 and further in view of Donica et al.

As to claim 17, note discussions of claims 1, 10-11 (with respect to Donica), and Heizer further teaches forwarding the request to a dispatcher (control process 204 accepts

new client) executed by the server; and processing the request by causing the dispatcher to perform the steps of A) determining whether an available instance of a program configured to handle the request is available among an existing number of instances of the program (decides which of 202, 203 should handle the client), B) if an available instance is available, then dispatching the request for execution by the available instance (first request is given to 202 since it is activated, step 307), C) if no instance is available, then initiating a new instance of the program for execution of the request if the existing number of instances does not exceed a maximum prescribed number (for the third request, a new server process 203 is started, step 313), and D) if no instance is available and the existing number of instances exceeds the maximum prescribed number, then sending said reply over the network indicating the request was not processed (when the maximum number of server processes is set at 20, request 21st is rejected, step 317), see col. 3, lines 10-23; col. 4, lines 3-56.

Further, Putz teaches registering (the program directory) program types (named database operations) with a dispatcher (main server program 55) for dispatching client requests. (col. 10, lines 31-37). Typically registering with a program directory includes registering the locations/addresses. Registering with a dispatcher (control process) information of a maximum number of instances (high water mark) of a service is taught by Heizer (see discussion of claim 1).

As to claim 14, Heizer teaches receiving by the dispatcher (control process/server program) a reply (response) from the instance executing and sending information contained to a client (send response). See fig. 1.

14. Claims 15-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al in view of Donica et al as applied to claim 17 and further in view of Tobe et al.

As to claim 15, note discussion of claim 4 for detecting and terminating/aborting and dispatcher (management node, col. 4, lines 49-51).

As to claim 16, sending a reply is covered by claim 17, step D).

15. Claims 18-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Putz et al, Heizer, Kahn et al in view of Donica et al as applied to claim 17 and further in view of Atkins et al.

As to claims 18-19, note discussion of claim 5.

16. Claim 20 is rejected under 35 U.S.C. 103(a) as being unpatentable over Putz, Heizer, Kahn in view of McChesney et al.

As to claim 20, note discussions of claims 1, and Heizer further teaches forwarding the request to a dispatcher (control process 204 accepts new client) executed by the server; and processing the request by causing the dispatcher to perform the steps of A) determining whether an available instance of a program configured to handle the request is available among an existing number of instances of the program (decides which of 202, 203 should handle the client), B) if an available instance is available, then dispatching the request for execution by the available instance (first request is given to 202 since it is activated, step 307), C) if no instance is available, then initiating a new instance of the program for execution of the request if the existing number of instances does not exceed a maximum prescribed number (for the third request, a new server process 203 is started, step 313), and D) if no instance is available and the existing number of instances exceeds the maximum prescribed number, then sending said reply over the network indicating the request was not processed (when the maximum number of server processes is set at 20, request 21st is rejected, step 317), see col. 3, lines 10-23; col. 4, lines 3-56.

As to E), McChesney teaches a process/object life cycle management system, including delaying and automatically deallocating an instance after a predetermined time interval (API Shutdown 223, automatically shutdown an idle process after a time interval). See fig. 2, col. 30, line 52 - col. 31, line 3. Since McChesney addresses object/resource management as does Putz as modified, it would have been obvious to combine the teachings.

17. Applicant's arguments filed 6/8/2000 have been considered but are moot in view of the new ground(s) of rejection.

In the response, applicant's representative argued in substance that the motivations to combine the cited references must be expressly found in the cited references and not based on hindsight/applicant's disclosure.

The examiner's response is as follows.

MPEP § 2143.01 requires that the suggestion or motivation to combine references must be found in the references themselves, OR, in the knowledge generally available to one of ordinary skill in the art. The proper test for the relevance of a cited combination of references is: "whether the teachings of the prior art, taken as a whole, would have made obvious the claimed invention," In re Gorman, 933 F.2d at 986, 18 USPQ2d at 1888. Subject matter is unpatentable under section 103 if " 'would have been obvious ... to a person having ordinary skill in the art.' While there must be some teaching, reason, suggestion, or motivation to combine existing elements to produce the claimed device, it is not necessary that the cited references or prior art specifically suggest making the combination: In re Nilssen, 851 F.2d 1401, 1403, 7 USPQ2d 1500, 1502 (Fed. Cir. 1988)." Such suggestion or motivation to combine prior art teachings can derive solely from the existence of a teaching, which one of ordinary skill in the art would be presumed to know, and the use of that teaching to solve the same [or] similar problem which it addresses. In re Wood, 599 F.2d 1032, 1037, 202 USPQ 171, 174 (CCPA 1979). "In stun, it is off the mark for litigants to argue, as many do, that an invention cannot be held to have been obvious unless a suggestion to combine prior art teachings is found in a specific reference." In re Oetiker, 24 USPQ2d 1443 (CAFC 1992).

In response to Applicant's argument that the Examiner's conclusion of obviousness is based upon improper hindsight reasoning, it must be recognized that any judgement on obviousness is in a sense necessarily a reconstruction based upon hindsight reasoning. But so long as it takes into account only knowledge which was within the level of ordinary skill at the time the claimed invention was made, and does not include knowledge gleaned

only from the applicant's disclosure, such a reconstruction is proper. In re McLaughlin, 443 F.2d 1392; 170 USPQ 209 (CCPA 1971).

18. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sue Lao whose telephone number is (703) 305-9657. The fax number for this Group is (703) 305-9731.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Group receptionist whose telephone number is (703) 305-3900.

Sue Lao
August 23, 2000

MAJID BANANKHAH
PRIMARY EXAMINER

Notice of References Cited

Application No.	Applicant(s)		
08/794,269	Adunuthula, et al		
Examiner		Group Art Unit	Page 1 of 1
S. Lao		2755	

U.S. PATENT DOCUMENTS

	DOCUMENT NO.	DATE	NAME	CLASS	SUBCLASS
A	5,894,554	4/1999	Lowery et al	395	200.33
B	6,070,191	5/2000	Narendran et al	709	226
C	6,098,093	8/2000	Bayeh et al	709	203
D					
E					
F					
G					
H					
I					
J					
K					
L					
M					

FOREIGN PATENT DOCUMENTS

	DOCUMENT NO.	DATE	COUNTRY	NAME	CLASS	SUBCLASS
N						
O						
P						
Q						
R						
S						
T						

NON-PATENT DOCUMENTS

	DOCUMENT (Including Author, Title, Source, and Pertinent Pages)	DATE
U		
V		
W		
X		

Exhibit 16

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 17

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 18

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 19

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 20

U.S. DISTRICT COURT
EASTERN DISTRICT OF TEXAS
06 JAN 27 PM 4:29
TX EASTERN-MARSHALL
BY

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

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-163
	§	
AUTOFLEX LEASING, INC., et al.,	§	
	§	
Defendants.	§	

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-356
	§	
FRANKLIN COVEY CO., et al.,	§	
	§	
Defendants.	§	

PLAINTIFF'S SECOND AMENDED COMPLAINT

Plaintiff, epicRealm Licensing, LP, brings this action for patent infringement and alleges the following:

I. PARTIES

1. Plaintiff epicRealm Licensing, LP ("epicRealm") is a Delaware limited partnership with its principal place of business in Dallas, Dallas County, Texas.

2. On information and belief, defendant Autoflex Leasing – Dallas, LP ("Autoflex") is a Texas limited partnership, with its principal place of business at 558 S. Central Expressway, Richardson, Texas 75080-6126, and is doing business in the Eastern District and elsewhere in the

State of Texas. Autoflex may be served with process by service on its registered agent, Andrew L. Adams, located at 558 S. Central Expressway, Richardson, Texas 75080-6126.

3. On information and belief, defendant eHarmony.com, Inc., ("eHarmony") is a California corporation, with its principal place of business at 300 North Lake Avenue, Suite 1111, Pasadena, California 91101. Although eHarmony has done — and continues to do — business in the Eastern District and elsewhere in the State of Texas, it has not designated or maintained an agent for service of process in Texas. Accordingly, pursuant to Section 17.044 of the Texas Civil Practice and Remedies Code, eHarmony may be served with process by service upon the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701.

4. On information and belief, defendant Friendfinder Network, Inc., ("Friendfinder") is a California corporation, with its principal place of business at 445 Sherman Avenue, Suite T, Palo Alto, California 94306. Although Friendfinder has done — and continues to do — business in the Eastern District and elsewhere in the State of Texas, it has not designated or maintained an agent for service of process in Texas. Accordingly, pursuant to Section 17.044 of the Texas Civil Practice and Remedies Code, Friendfinder may be served with process by service upon the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701.

5. On information and belief, defendant Grande Communications Networks, Inc., ("Grande") is a Delaware corporation, with its principal place of business at 401 Carlson Circle, San Marcos, Texas 78666-6730, and is doing business in the Eastern District and elsewhere in the State of Texas. Grande may be served with process by service on its registered agent, Andrew Kever, located at 401 Carlson Circle, San Marcos, Texas 78666-6730.

6. On information and belief, Transplace Texas, L.P., ("Transplace") is a Texas limited partnership, with its principal place of business at 5800 Granite Parkway, Suite 1000,

Plano, Texas 75024, and is doing business in the Eastern District and elsewhere in the State of Texas. Transplace may be served with process by service on its registered agent, Ellis McKinley, located at 5800 Granite Parkway, Suite 1000, Plano, Texas 75024.

7. On information and belief, defendant Franklin Covey Co. ("Franklin Covey") is a Utah corporation, with its principal place of business at 2200 West Parkway Boulevard, Salt Lake City, Utah 84119, Attention: Tami Donavon, Director of Legal Services. Although Franklin Covey has done — and continues to do — business in the Eastern District and elsewhere in the State of Texas, it has not designated or maintained an agent for service of process in Texas. Accordingly, pursuant to Section 17.044 of the Texas Civil Practice and Remedies Code, Franklin Covey may be served with process by service upon the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701.

8. On information and belief, defendant Clark Consulting, Inc., ("Clark") is a Delaware corporation, with its principal place of business at 102 S. Wynstone Park Dr. #100, North Barrington, Illinois, 60010, and is doing business in the Eastern District and elsewhere in the State of Texas. Clark may be served with process by service upon its registered agent, CT Corporation System, 350 N. St. Paul St., Dallas, Texas 75201.

9. On information and belief, defendant The Macerich Company ("Macerich") is a Maryland corporation, with its principal place of business at 401 Wilshire Blvd., Suite 700, Santa Monica, California 90401, Attention: Arthur M. Coppola, President and CEO. Although Macerich has done — and continues to do — business in the Eastern District and elsewhere in the State of Texas, it has not designated or maintained an agent for service of process in Texas. Accordingly, pursuant to Section 17.044 of the Texas Civil Practice and Remedies Code,

Macerich may be served with process by service upon the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701.

10. On information and belief, defendant Safelite Group, Inc., ("Safelite") is a Delaware corporation, with its principal place of business at 2400 Farmers Drive, Columbus, Ohio 43235, Attention: Dan Wilson, President and CEO. Although Safelite has done — and continues to do — business in the Eastern District and elsewhere in the State of Texas, it has not designated or maintained an agent for service of process in Texas. Accordingly, pursuant to Section 17.044 of the Texas Civil Practice and Remedies Code, Safelite may be served with process by service upon the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701.

11. On information and belief, defendant Herbalife International of America, Inc., ("Herbalife") is a California corporation, with its principal place of business at 1800 Century Park East, Los Angeles, California 90067, and is doing business in the Eastern District and elsewhere in the State of Texas. Herbalife may be served with process by service upon its registered agent, United States Corp. Company, 701 Brazos Street, Suite 1050, Austin, Texas 78701.

12. On information and belief, defendant Pink Sheets, LLC, ("Pink Sheets") is a Delaware limited liability company, with its principal place of business at 304 Hudson St. 2nd Floor, New York, New York 10013, Attention: R. Cromwell Coulson, Chairman and CEO. Although Pink Sheets has done — and continues to do — business in the Eastern District and elsewhere in the State of Texas, it has not designated or maintained an agent for service of process in Texas. Accordingly, pursuant to Section 17.044 of the Texas Civil Practice and Remedies Code, Pink Sheets may be served with process by service upon the Texas Secretary of State, 1019 Brazos Street, Austin, Texas 78701.

EPIC398189

II. JURISDICTION AND VENUE

13. This infringement action arises under the patent laws of the United States, title 35, United States Code. This Court has jurisdiction of this action under 28 U.S.C. §§ 1331, 1338(a).

14. All of the defendants have done — and continue to do — business in the Eastern District of Texas. All defendants have minimum contacts with the Eastern District of Texas such that this venue is a fair and reasonable one. The defendants have committed purposeful acts or transactions in the State of Texas such that they reasonably knew and expected that they could be haled into a Texas court as a consequence of such activity. Accordingly, venue in the Eastern District of Texas is proper under 28 U.S.C §§ 1391(b), 1400(b).

III. PATENT INFRINGEMENT

15. On April 13, 1999, and July 2, 2002, United States Patent Nos. 5,894,554 and 6,415,335 B1, which are collectively referred to as the "epicRealm Patents," duly and legally issued. These two patents concern, among other things, systems and methods for managing dynamic Web page generation requests. Copies of the epicRealm Patents are attached hereto as Exhibits "A" and "B" and made a part hereof.

16. EpicRealm is the owner of the epicRealm Patents and has the right to enforce those patents with respect to the defendants.

17. On information and belief, defendants use systems and methods for managing dynamic Web page generation requests within the scope of one or more of the claims of the epicRealm Patents. As a result, all of the defendants have been and still are infringing one or more of the claims of the epicRealm Patents as defined by 35 U.S.C. § 271 (a), (b), and/or (c). EpicRealm has suffered damage by reason of defendants' infringement and will continue to suffer additional damage until this Court enjoins the infringing conduct.

18. To the extent that defendants have continued or do continue their infringing activities after receiving notice of the epicRealm Patents, such infringement is willful, entitling epicRealm to the recovery of increased damages under 35 U.S.C. § 284.

19. This is an "exceptional case" justifying an award of attorneys' fees and costs to epicRealm pursuant to 35 U.S.C. § 285.

20. EpicRealm believes that defendants will continue to infringe the epicRealm Patents unless enjoined by this Court. Such infringing activity causes epicRealm irreparable harm and will continue to cause such harm without the issuance of an injunction.

IV. JURY DEMAND

21. Plaintiff requests trial by jury pursuant to Federal Rule of Civil Procedure 38.

V. PRAYER FOR RELIEF

22. EpicRealm requests that the Court find in its favor and against defendants and that the Court grant the following relief:

a. Judgment that one or more of the claims of the epicRealm Patents have been infringed, either literally and/or under the doctrine of equivalents, by defendants;

b. Judgment in favor of epicRealm for the full amount of its actual damages caused by defendants' infringing activities, including an assessment of interest and costs;

c. Judgment for increased damages for willful infringement pursuant to 35 U.S.C. § 284;

d. Judgment that this is an "exceptional case" and awarding epicRealm its reasonable attorneys' fees and costs pursuant to 35 U.S.C. § 285;

e. That defendants be permanently enjoined from further activity or conduct that infringes the claims of the epicRealm Patents; and

f. That the Court award epicRealm such other and further relief as is just and proper under the circumstances.

Respectfully submitted,

/s/ Larry D. Carlson

Larry D. Carlson, Attorney-in-Charge
Texas State Bar No. 03814500
E-Mail: larry.carlson@bakerbotts.com
Kevin Meek
Texas State Bar No. 13899600
E-Mail: kevin.meek@bakerbotts.com
Thomas Frame
Texas State Bar No. 24027318
E-Mail: thomas.frame@bakerbotts.com
Jeff Moles
Texas State Bar No. 24041504
E-Mail: jeff.moles@bakerbotts.com
BAKER BOTTS L.L.P.
2001 Ross Avenue
Dallas, Texas 75201
Telephone: (214) 953-6500
Facsimile: (214) 953-6503

Otis W. Carroll
Texas State Bar No. 03895700
E-mail: otiscarroll@icklaw.com
IRELAND, CARROLL & KELLEY, P.C.
6101 S. Broadway, Suite 500
P.O. Box 7879
Tyler, Texas 75711
Telephone: (903) 561-1600
Facsimile: (903) 581-1071

S. Calvin Capshaw
Texas State Bar No. 03783900
E-Mail: ccapshaw@mailbmc.com
BROWN McCARROLL LLP
1127 Judson Road, Suite 220
P.O. Box 3999
Longview, Texas 75601-5157
Telephone: (903) 236-9800
Facsimile: (903) 236-8787

Franklin Jones, Jr.
Texas State Bar No. 00000055
E-Mail: maizieh@millerfirm.com
JONES & JONES, Inc., P.C.
201 West Houston Street
P.O. Drawer 1249
Marshall, Texas 75671-1249
Telephone: (903) 938-4395
Facsimile: (903) 938-3360

ATTORNEYS FOR PLAINTIFF
EPICREALM LICENSING, LP

CERTIFICATE OF SERVICE

I certify that on the _____, all counsel are being served with a copy of this document either: (1) by the Court's Electronic Filing System, pursuant to Local Rule CV-5(a)(3)(A), if they have consented to electronic service, or (2) by electronic mail, pursuant to Local Rule CV-5(e), if they have not so consented.

/s/ Larry D. Carlson
Larry D. Carlson

Exhibit 21

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

EPICREALM LICENSING, LP, Plaintiff, v. AUTOFLEX LEASING, INC., *et al.*, Defendants.	CIVIL ACTION NO. 2-05CV-163 Jury Trial Demanded
EPICREALM LICENSING, LP, Plaintiff, v. FRANKLIN COVEY CO., *et al.*, Defendants. SAFELITE GROUP, INC., Third Party Plaintiff, v. ORACLE CORPORATION AND INTERNATIONAL BUSINESS MACHINES CORPORATION, Third Party Defendants.	CIVIL ACTION NO. 2-05CV-356 Jury Trial Demanded

ANSWER AND COUNTERCLAIMS OF SAFELITE GROUP, INC. IN RESPONSE TO PLAINTIFF'S SECOND AMENDED COMPLAINT

Defendant Safelite Group, Inc. ("Safelite") respectfully answers the Second Amended Complaint of epicRealm Licensing, LP ("epicRealm") ("Second Amended Complaint"), filed on January 27, 2006, in accordance with the numbered paragraphs thereof, and asserts counterclaims against epicRealm as follows:

I. PARTIES

1. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 1 of epicRealm's Second Amended Complaint.

2. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 2 of epicRealm's Second Amended Complaint.

3. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 3 of epicRealm's Second Amended Complaint.

4. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 4 of epicRealm's Second Amended Complaint.

5. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 5 of epicRealm's Second Amended Complaint.

6. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 6 of epicRealm's Second Amended Complaint.

7. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 7 of epicRealm's Second Amended Complaint.

8. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 8 of epicRealm's Second Amended Complaint.

9. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 9 of epicRealm's Second Amended Complaint.

10. Safelite admits that it is a Delaware corporation with its principal place of business at 2400 Farmers Drive, Columbus, Ohio 43235. Safelite also admits that it does and has done business in the State of Texas, but otherwise denies any and all other material

allegations of fact that may be contained in paragraph 10 of epicRealm's Second Amended Complaint.

11. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 11 of epicRealm's Second Amended Complaint.

12. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 12 of epicRealm's Second Amended Complaint.

II. JURISDICTION AND VENUE

13. Responding to paragraph 13 of epicRealm's Second Amended Complaint, Safelite admits that this action arises under the patent laws of the United States, Title 35, United States Code. Safelite admits that this Court has subject matter jurisdiction over this action under 28 U.S.C. §§ 1331 and 133 8(a). Safelite denies epicRealm's allegations of patent infringement with respect to Safelite.

14. Responding to paragraph 14 of epicRealm's Second Amended Complaint, Safelite admits that it has done and continues to do business in the Eastern District of Texas, but denies that venue is appropriate in this District. To the extent that any remaining allegations contained in paragraph 14 of epicRealm's Second Amended Complaint are not specifically denied, Safelite lacks knowledge or information sufficient to form a belief as to the truth of said allegations.

III. PATENT INFRINGEMENT

15. Responding to paragraph 15 of epicRealm's Second Amended Complaint, Safelite admits that United States Patent Nos. 5,894,554 and 6,415,335 B1 (the "epicRealm Patents") were issued on April 13, 1999 and July 2, 2002, respectively. Safelite denies that the epicRealm Patents were validly issued. To the extent that any remaining allegations contained in paragraph

15 of epicRealm's Second Amended Complaint are not specifically denied, Safelite lacks knowledge or information sufficient to form a belief as to the truth of said allegations.

16. Safelite lacks knowledge or information sufficient to form a belief as to the truth of the allegations contained in paragraph 16 of epicRealm's Second Amended Complaint.

17. Responding to paragraph 17 of epicRealm's Second Amended Complaint, Safelite denies that it uses systems and methods that are within the scope of one or more of the claims of the epicRealm Patents. Safelite denies that it has been or still is infringing one or more of the claims of the epicRealm Patents as defined by 35 U.S.C. § 271. Safelite denies that epicRealm has suffered damage or will suffer damage arising from Safelite's alleged use of systems and methods to manage dynamic Web page generation requests. Safelite denies that epicRealm is entitled to an injunction against Safelite's alleged use of such systems and methods and denies that an injunction against Safelite is an appropriate remedy for the acts of which epicRealm complains. To the extent that any remaining allegations contained in paragraph 17 of epicRealm's Second Amended Complaint are not specifically denied, Safelite lacks knowledge or information sufficient to form a belief as to the truth of said allegations.

18. Responding to paragraph 18 of epicRealm's Second Amended Complaint, Safelite denies that it received notice of epicRealm's claims of infringement prior to the filing of epicRealm's Second Amended Complaint. Safelite denies any allegations that it has or will infringe the epicRealm Patents. Safelite denies that its past, current or future activities constitute willful infringement and denies that epicRealm is entitled to the recovery of increased damages under 35 U.S.C. § 284. To the extent that any remaining allegations contained in paragraph 18 of epicRealm's Second Amended Complaint are not specifically denied, Safelite lacks knowledge or information sufficient to form a belief as to the truth of said allegations.

19. Responding to paragraph 19 of epicRealm's Second Amended Complaint, Safelite denies that this is an "exceptional case" justifying an award of attorney's fees and costs to epicRealm pursuant to 35 U.S.C. § 285.

20. Responding to paragraph 20 of epicRealm's Second Amended Complaint, Safelite denies that it has or will continue to infringe the epicRealm Patents. Safelite denies that its allegedly infringing activity causes epicRealm irreparable harm and will continue to cause such harm without the issuance of an injunction, and denies that epicRealm is entitled to an injunction against Safelite. To the extent that any remaining allegations contained in paragraph 20 of epicRealm's Second Amended Complaint are not specifically denied, Safelite lacks knowledge or information sufficient to form a belief as to the truth of said allegations.

IV. JURY DEMAND

21. Safelite acknowledges and joins epicRealm's request for a trial by jury pursuant to Federal Rule of Civil Procedure 38.

V. SAFELITE'S RESPONSE TO EPICREALM'S PRAYER FOR RELIEF

22. Safelite denies that epicRealm is entitled to any of the relief requested in its prayer for relief, or any relief whatsoever.

VI. ADDITIONAL DEFENSES

Safelite, without waiver, limitation, or prejudice, further answers epicRealm's Second Amended Complaint by asserting the following additional defenses:

23. Safelite has not infringed and is not infringing (either directly, contributorily, or by inducement) any claim of the epicRealm Patents.

24. On information and belief, each asserted claim of the epicRealm Patents is invalid and void for failure to comply with the conditions for patentability specified in Title 35 of the United States Code, including without limitation at least Sections 102, 103 and 112.

25. On information and belief, epicRealm and its predecessors in interest, as owners of the epicRealm Patents, failed to comply with the requirements of 35 U.S.C. § 287, and epicRealm is, therefore, precluded from seeking any recovery for alleged damages accruing prior to the filing of this action.

26. On information and belief, epicRealm is precluded from seeking any recovery for alleged damages accruing prior to the filing of this action and should be enjoined from enforcing the epicRealm patents against Safelite by the doctrine of laches. epicRealm unreasonably delayed since 1999 and 2002 in enforcing the epicRealm patents against businesses using systems and methods to manage dynamic Web page generation requests. epicRealm seeks, now, to assert these patents only after the accused systems and methods have become widely used in the field of Web site management and development.

27. Venue is improperly placed in this District and Division, and properly belongs in Dallas Texas.

28. Any claim for damages or relief against Safelite is subject to the Order of the United States Bankruptcy Court for the District of Delaware, in In re Safelite Glass Corp., Case No. 00-2252 (MFW), entered on September 12, 2000, confirming Safelite's First Amended Plan of Reorganization, dated July 31, 2000. Safelite's Petition was filed on June 9, 2000. The Plan was consummated and became effective on September 29, 2000 (the "Effective Date"). The case was closed and a Final Decree entered on January 31, 2002. As a consequence, any obligation of Safelite that may be asserted or assessed herein has been discharged, at least in part.

VII. COUNTERCLAIMS

Defendant Safelite Group, Inc. ("Safelite") for its Counterclaims against Plaintiff epicRealm Licensing, LLC ("EpicRealm"), states as follows:

29. Safelite is a Delaware corporation with its principal place of business at 2400 Farmers Drive, Columbus, Ohio 43235.

30. On information and belief, and based upon epicRealm's allegations, epicRealm is a Delaware limited partnership with its principal place of business in, Dallas, Texas 75201.

31. This Court has subject-matter jurisdiction over these Counterclaims in accordance with 28 U.S.C. §§ 1331, 1338, and 2201 as they arise under an Act of Congress relating to patents.

32. Although Safelite denies that venue is initially proper in this district, if the Court ultimately holds otherwise, then venue for these Counterclaims will also be proper under 28 U.S.C. §§ 1391(b), (c) and 1400.

A. FIRST COUNTERCLAIM

(Declaratory Judgment of Non-infringement and Invalidity of U.S. Pat. No. 5,894,554)

33. Safelite repeats and re-alleges paragraphs 23-32 as if fully set forth herein.

34. This declaratory judgment counterclaim is asserted against epicRealm under the patent laws, Title 35, United States Code, and under Title 28, United States Code, §§ 2201 and 1338(a).

35. epicRealm has averred that U.S. Patent No. 5,894,554 ("the '554 patent") was duly and legally issued by the United States Patent & Trademark Office ("USPTO"), that it is the owner of all right, title, and interest in the '554 patent, including the right to sue for past and future infringement, and that Safelite infringes the '554 patent.

36. Safelite denies that it has infringed, directly, contributorily, or by inducement, either literally or under the doctrine of equivalents, and denies that it is infringing, either literally or under the doctrine of equivalents, any valid and/or enforceable claim of the '554 patent, and Safelite further asserts that the '554 patent is invalid for failing to satisfy the conditions for patentability set forth in Part II of Title 35 of the United States Code, including but not limited to, §§ 102, 103, and 112.

37. Therefore, there has been and is now an actual controversy between Safelite and epicRealm as to the invalidity and non-infringement of the '554 patent.

B. SECOND COUNTERCLAIM

(Declaratory Judgment of Non-infringement and Invalidity of U.S. Pat. No. 6,415,335 B1)

38. Safelite repeats and re-alleges paragraphs 23-37 as if fully set forth herein.

39. This declaratory judgment counterclaim is asserted against epicRealm under the patent laws, Title 35, United States Code, and under Title 28, United States Code, §§ 2201 and 1338(a).

40. epicRealm has averred that U.S. Patent No. 6,415,335 B1 ("the '335 patent") was duly and legally issued by the United States Patent & Trademark Office ("USPTO"), that it is the owner of all right, title, and interest in the '335 patent, including the right to sue for past and future infringement, and that Safelite infringes the '335 patent.

41. Safelite denies that it has infringed, directly, contributorily, or by inducement, either literally or under the doctrine of equivalents, and denies that it is infringing, either literally or under the doctrine of equivalents, any valid and/or enforceable claim of the '335 patent, and Safelite further asserts that the '335 patent is invalid for failing to satisfy the conditions for

patentability set forth in Part 11 of Title 35 of the United States Code, including but not limited to, §§ 102, 103, and 112.

42. Therefore, there has been and is now an actual controversy between Safelite and epicRealm as to the invalidity and non-infringement of the '335 patent.

VIII. DEMAND FOR JURY

43. Safelite requests trial by jury pursuant to Federal Rule of Civil Procedure 38 for all issues triable of right by a jury.

IX. SAFELITE'S PRAYER FOR RELIEF

Wherefore, Safelite Group, Inc. respectfully requests that this Court:

(a) Dismiss the claims of epicRealm against Safelite with prejudice;

(b) Deny epicRealm all relief it has requested in its Second Amended Complaint;

(c) Deny any permanent injunctive relief in favor of epicRealm and against Safelite;

(d) Declare that Safelite has not and does not infringe any valid claim of the '554 patent and the '335 patent, directly or indirectly, literally or by equivalents;

(e) Declare that each claim of the '554 patent and the '335 patent is invalid;

(f) Award Safelite costs, together with reasonable attorneys' fees and all other expenses for this suit because this case is exceptional under 35 U.S.C. § 285; and

(g) Award Safelite such other relief as this Court may deem just and proper.

Respectfully submitted,

Safelite Group, Inc.

By: s/John F. Prescott, Jr.
John F. Prescott, Jr.
Indiana State Bar # 5813-49
(Admitted herein *pro hac vice*)
ATTORNEY-IN-CHARGE
ICE MILLER LLP
One American Square
Suite 3100
Indianapolis, IN 46282
Telephone: 317-236-2398
Facsimile: 317-592-4736
Email: John.Prescott@icemiller.com

ATTORNEY FOR DEFENDANT
SAFELITE GROUP, INC.

CERTIFICATE OF SERVICE

I hereby certify that on February 8, 2006, a copy of the foregoing ANSWER AND COUNTERCLAIMS OF SAFELITE GROUP, INC. IN RESPONSE TO PLAINTIFF'S SECOND AMENDED COMPLAINT was filed electronically. Notice of this filing will be sent to all counsel of record by operation of the Court's electronic filing system. Parties may access this filing through the Court's system.

s/John F. Prescott, Jr.

INDY 1681455v.1

Exhibit 22

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

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-163
	§	
AUTOFLEX LEASING, INC., et al.,	§	
	§	
Defendants.	§	

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-356
	§	
FRANKLIN COVEY CO., et al.,	§	
	§	
Defendants.	§	

STIPULATION OF DISMISSAL

Plaintiff epicRealm Licensing, LP ("epicRealm") and defendant Safelite Group, Inc. ("Safelite") have agreed to a settlement of all claims asserted between them in this case, and, pursuant to that agreement, have agreed to dismiss all such claims with prejudice. Accordingly, pursuant to Federal Rule of Civil Procedure 41(a)(1), epicRealm stipulates to the dismissal with prejudice of all its claims asserted in this case against Safelite, and Safelite stipulates to the dismissal with prejudice of all its claims asserted in this case against epicRealm.

Respectfully submitted,

/s/ Jeff Moles

Larry D. Carlson, Attorney-in-Charge
Texas State Bar No. 03814500
E-Mail: larry.carlson@bakerbotts.com
Kevin Meek
Texas State Bar No. 13899600
E-Mail: kevin.meek@bakerbotts.com
Thomas Frame
Texas State Bar No. 24027318
E-Mail: thomas.frame@bakerbotts.com
Jeff Moles
Texas State Bar No. 24041504
E-Mail: jeff.moles@bakerbotts.com
BAKER BOTTS L.L.P.
2001 Ross Avenue
Dallas, Texas 75201
Telephone: (214) 953-6500
Facsimile: (214) 953-6503

Otis W. Carroll
State Bar No. 03895700
E-Mail: otiscarroll@icklaw.com
IRELAND, CARROLL & KELLEY, P.C.
6101 S. Broadway, Suite 500
P.O. Box 7879
Tyler, Texas 75711
Telephone: (903) 561-1600
Facsimile: (903) 581-1071

S. Calvin Capshaw
State Bar No. 03783900
E-Mail: ccapshaw@mailbmc.com
BROWN McCARROLL LLP
1127 Judson Road, Suite 220
P.O. Box 3999
Longview, Texas 75601-5157
Telephone: (903) 236-9800
Facsimile: (903) 236-8787

Franklin Jones, Jr.
State Bar No. 00000055
E-Mail: maizieh@millerfirm.com
JONES & JONES, Inc., P.C.
201 West Houston Street
P.O. Drawer 1249
Marshall, Texas 75671-1249
Telephone: (903) 938-4395
Facsimile: (903) 938-3360

ATTORNEYS FOR PLAINTIFF
EPICREALM LICENSING, LP

/s/ John F. Prescott, Jr.
John F. Prescott, Jr.
Indiana State Bar # 5813-49
(Admitted herein *pro hac vice*)
ATTORNEY-IN-CHARGE
ICE MILLER LLP
One American Square
Suite 3100
Indianapolis, IN 46282
Telephone: 317-236-2398
Facsimile: 317-592-4736
Email: John.Prescott@icemiller.com

ATTORNEY FOR DEFENDANT
SAFELITE GROUP, INC.

CERTIFICATE OF SERVICE

I certify that on June 26, 2006, all counsel are being served with a copy of this document either: (1) by the Court's Electronic Filing System, pursuant to Local Rule CV-5(a)(3)(A), if they have consented to electronic service, or (2) by electronic mail, pursuant to Local Rule CV-5(e), if they have not so consented.

/s/ Jeff Moles
Jeff Moles

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

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-163
	§	
AUTOFLEX LEASING, INC., et al.,	§	
	§	
Defendants.	§	

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-356
	§	
FRANKLIN COVEY CO., et al.,	§	
	§	
Defendants.	§	

ORDER

Recognizing the Stipulation of Dismissal filed by plaintiff epicRealm Licensing, LP and defendant Safelite Group, Inc., it is

ORDERED that the claims asserted herein by plaintiff epicRealm Licensing, LP against defendant Safelite Group, Inc. be, and hereby are, dismissed with prejudice;

ORDERED that the claims asserted herein by counter-plaintiff Safelite Group, Inc. against counter-defendant epicRealm Licensing, LP be, and hereby are, dismissed with prejudice; and

ORDERED that the parties shall bear their own attorneys' fees, expenses, and costs.

Exhibit 23

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

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-163
	§	
AUTOFLEX LEASING, INC., et al.,	§	
	§	
Defendants.	§	

EPICREALM LICENSING, LP,	§	
	§	
Plaintiff,	§	
	§	
v.	§	2:05-CV-356
	§	
FRANKLIN COVEY CO., et al.,	§	
	§	
Defendants.	§	

ORDER

Recognizing the Stipulation of Dismissal filed by plaintiff epicRealm Licensing, LP and defendant Safelite Group, Inc., it is

ORDERED that the claims asserted herein by plaintiff epicRealm Licensing, LP against defendant Safelite Group, Inc. be, and hereby are, dismissed with prejudice;

ORDERED that the claims asserted herein by counter-plaintiff Safelite Group, Inc. against counter-defendant epicRealm Licensing, LP be, and hereby are, dismissed with prejudice; and

ORDERED that the parties shall bear their own attorneys' fees, expenses, and costs.

SIGNED this 29th day of June, 2006.

DAVID FOLSOM
UNITED STATES DISTRICT JUDGE

Exhibit 24

THIS EXHIBIT HAS BEEN REDACTED IN ITS ENTIRETY

Exhibit 25

US005761673A

United States Patent [19]

Bookman et al.

[11] **Patent Number: 5,761,673**

[45] **Date of Patent: Jun. 2, 1998**

[54] **METHOD AND APPARATUS FOR GENERATING DYNAMIC WEB PAGES BY INVOKING A PREDEFINED PROCEDURAL PACKAGE STORED IN A DATABASE**

[75] Inventors: **Matthew Bookman**, Los Gatos; **John Francis Haverty**, La Honda; **Magnus Mard Lonnroth**, Redwood City; **Teresita Katrina Rodriquez Montinola**; **Joseph Charles Pistritto**, both of Belmont, all of Calif.

[73] Assignee: **Oracle Corporation**, Redwood Shores, Calif.

[21] Appl. No.: **594,686**

[22] Filed: **Jan. 31, 1996**

[51] **Int. Cl.**[6] **G06F 17/30**

[52] **U.S. Cl.** **707/104**; 707/10; 395/680

[58] **Field of Search** 395/793, 603, 395/610, 184.01, 682, 356, 733, 357, 617, 680; 707/10, 104

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,530,852	6/1996	Meske, Jr. et al.	395/610
5,537,586	7/1996	Amram et al.	395/603
5,572,643	11/1996	Judson	395/793
5,701,451	12/1997	Rogers et al.	707/104
5,710,918	1/1998	Lagarde et al.	395/680

OTHER PUBLICATIONS

"Oracle Version 7", Computergram International, Jun. 17, 1992.

"Oracle Rides the Internet with Web", Computergram International, Feb. 10, 1995.

"Sun Java, Netscape Livescript Married as Javascript", Computergram International, Dec. 1995.

Marshall et al. "Industry Lines Up for Java—Vendors Set Stage for Dynamic", Commications Week, Dec. 1995.

"Oracle Unveils Oracle Websystem Internet Line", Computergram International, Nov. 1995.

Marshall, Martin "Sybase Sites to Get Web Links", Communciations Week, Nov. 1995.

(List continued on next page.)

Primary Examiner—Thomas G. Black

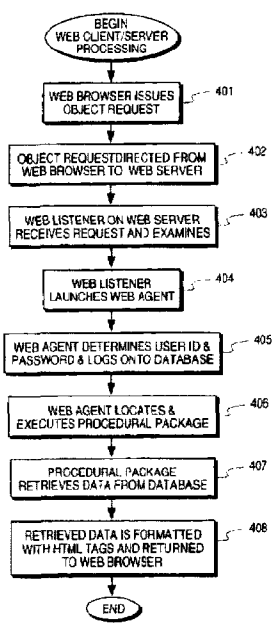
Assistant Examiner—Charles L. Rones

Attorney, Agent, or Firm—Blakely, Sokoloff, Taylor & Zafman LLP

[57] **ABSTRACT**

A method and apparatus for generating dynamic Web pages is disclosed. Specifically, the present invention claims a method and apparatus for generating dynamic Web pages on a Web server by invoking and executing predefined procedural packages stored in a database. The claimed invention receives an object request on the Web server and activates a Web agent on the Web server based on the object request. The Web agent invokes and executes the predefined procedural package to retrieve data from a data repository, and then formats the retrieved data as HTML output.

7 Claims, 4 Drawing Sheets



OTHER PUBLICATIONS

Karpinski, Richard "Oracle Readies Web Initiative–Businesses Can Tie Their Databases Directly to the Web", Interactive Age, n207, p. 1, Jan. 30, 1995.
"Hypercard–Based Oracle Card for Macintosh, Windows 3.0", Computergram International, Jan. 1991.
Karpinski, Richard "Web–Based Help Desks Debut", Communications Week, Oct. 1995.
"Next Leaps Aboard the Internet with Webobjects", Computergram International, Aug. 15, 1905.
Clinton et al. "Intranet Tools: Coprorations Seek Internal Web Applications—and Major Vendors are Happy to Help", Informationweek, n552, p14(2), Nov. 6, 1992.
Nash, Kim S. "Web Browser Makers Follw Modular Trend", News, Apr. 29, 1996.
"Netscape & Sun Announce Javascript", Newsbytes, Dec. 4, 1995.
"Oracle Challlenges Netscape with Free Powerbrowser", Computergram International, Dec. 1995.
"Oracle Set Powerbrowser", Computergram International, Dec. 1995.
"Oracle Begins Internet Initiative:Multimedia, Hypertext Development Tools", Seybold Report on Desktop Publishing, v9, n7, p27(1), Mar. 6, 1995.
Menefee, Craig "Oracle Intros Database Products", Newsbytes, Jan. 23, 1995.
"Sybase: Sybase Introduces Powersite Enterprise Web Development Environment", M2 Communications, Dec. 1997.
"ObjectShare IntroducesjKit/Grid; First Offering in a New Line of Advanced Java Components; Extensible New Software Provides Powerful Java Classes for Grids, Tables and More; Advanced Component Automatically Reacts to Data Changes", Buisness Wire, pp. 101, Oct. 1996.

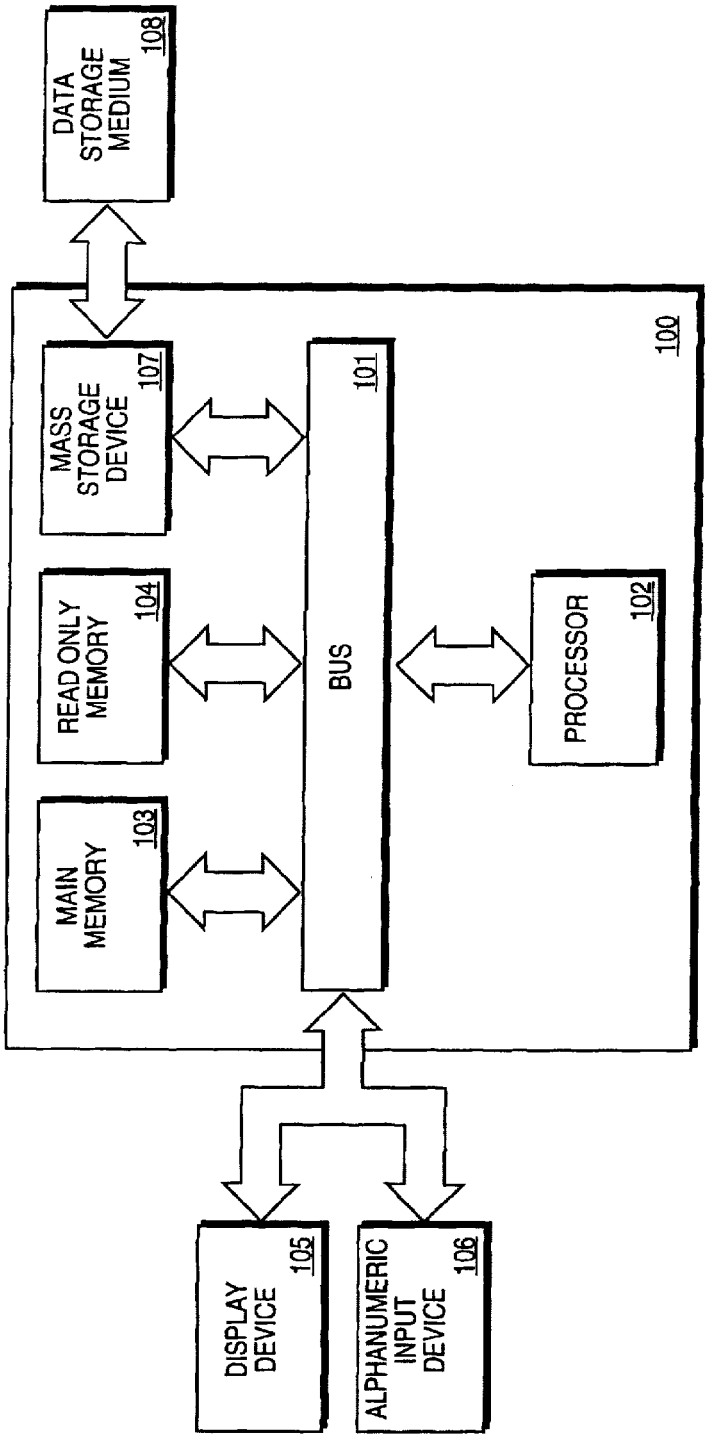

DATA STORAGE MEDIUM
108
100
MASS STORAGE DEVICE
107
101
READ ONLY MEMORY
104
BUS
PROCESSOR
102
MAIN MEMORY
103
DISPLAY DEVICE
105
ALPHANUMERIC INPUT DEVICE
106

FIG. 1

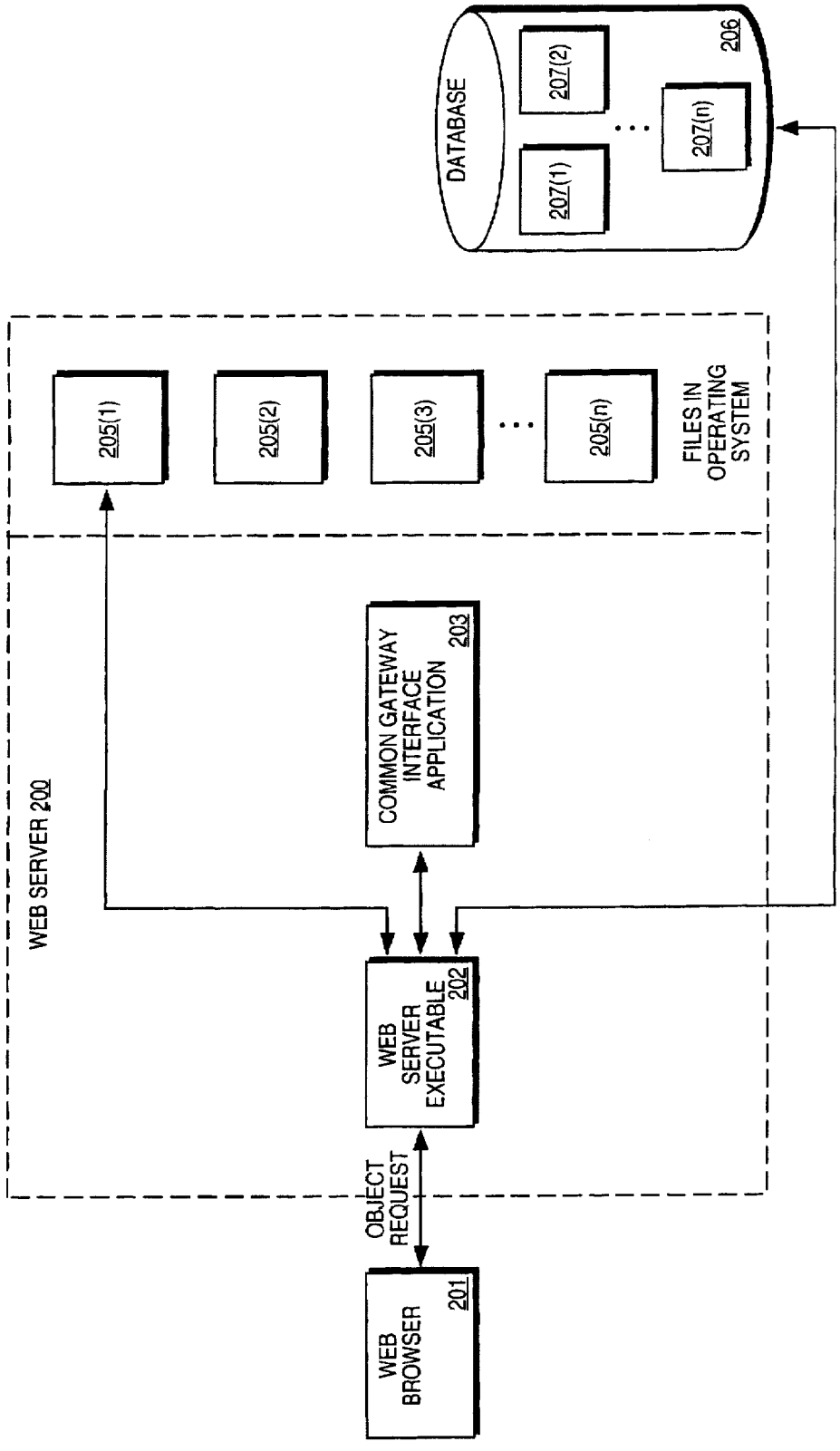


FIG. 2 (PRIOR ART)

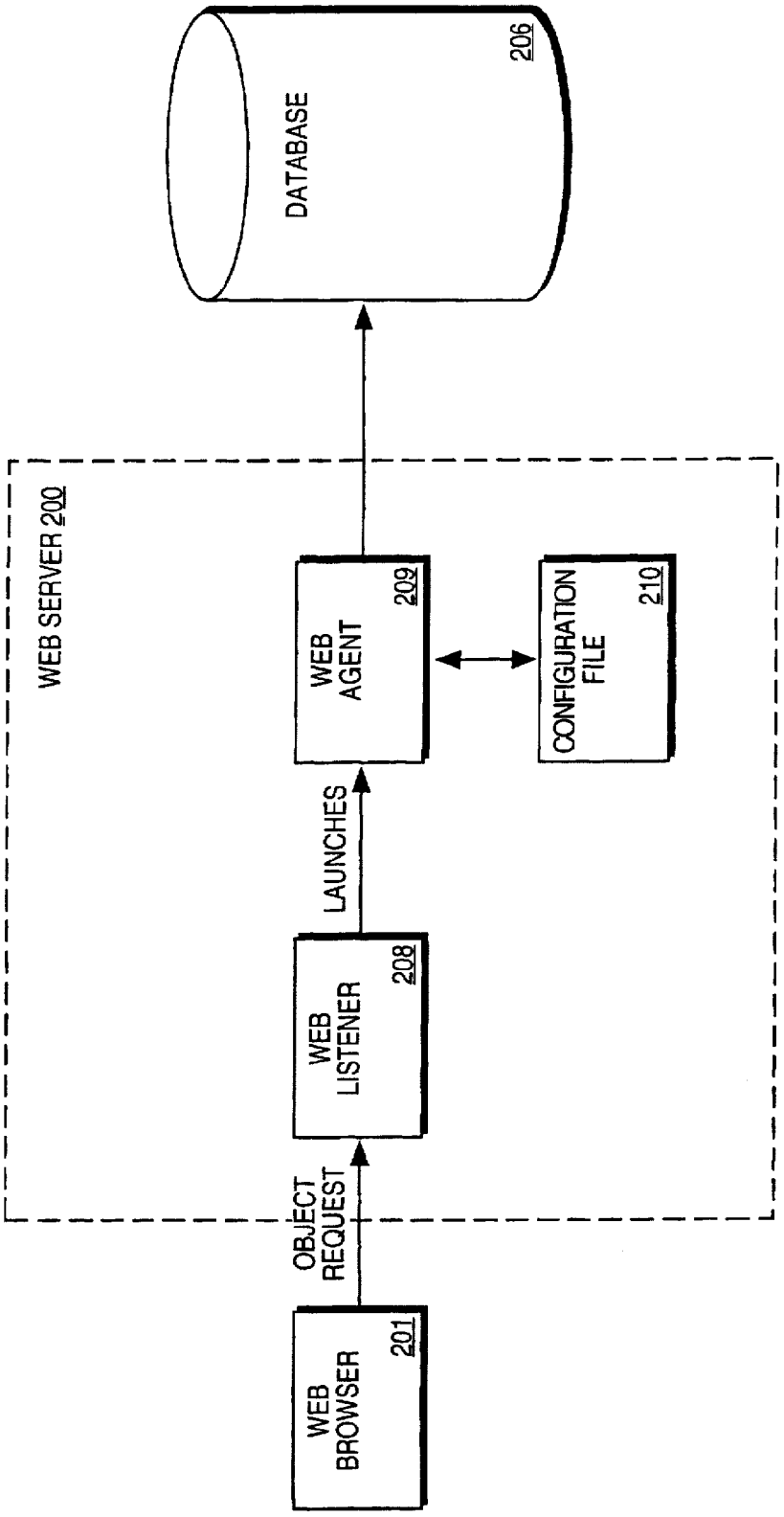

206
DATABASE
WEB SERVER 200
WEB AGENT
209
CONFIGURATION FILE
210
LAUNCHES
WEB LISTENER
208
OBJECT REQUEST
WEB BROWSER
201

FIG. 3

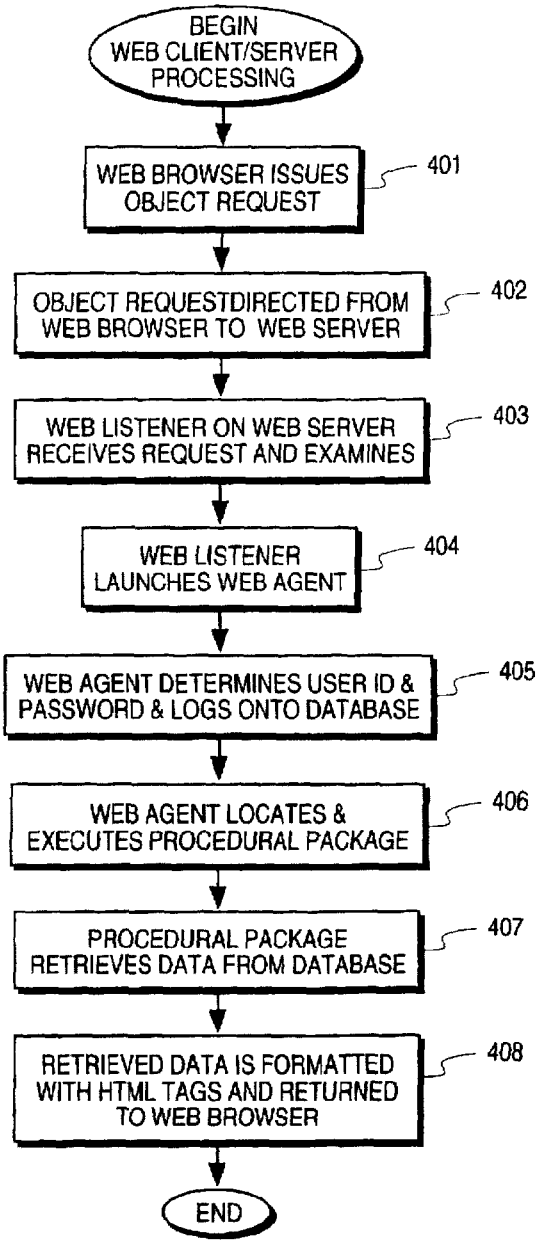

BEGIN WEB CLIENT/SERVER PROCESSING
WEB BROWSER ISSUES OBJECT REQUEST
401
OBJECT REQUESTDIRECTED FROM WEB BROWSER TO WEB SERVER
402
WEB LISTENER ON WEB SERVER RECEIVES REQUEST AND EXAMINES
403
WEB LISTENER LAUNCHES WEB AGENT
404
WEB AGENT DETERMINES USER ID & PASSWORD & LOGS ONTO DATABASE
405
WEB AGENT LOCATES & EXECUTES PROCEDURAL PACKAGE
406
PROCEDURAL PACKAGE RETRIEVES DATA FROM DATABASE
407
RETRIEVED DATA IS FORMATTED WITH HTML TAGS AND RETURNED TO WEB BROWSER
408
END

FIG. 4

METHOD AND APPARATUS FOR GENERATING DYNAMIC WEB PAGES BY INVOKING A PREDEFINED PROCEDURAL PACKAGE STORED IN A DATABASE

FIELD OF THE INVENTION

The present invention relates to the field of Internet server technology. Specifically, the present invention relates to a method and apparatus for generating dynamic Web pages.

DESCRIPTION OF RELATED ART

The World Wide Web ("the Web") represents all the computers on the Internet that offer users access to information and documentation on the Internet via interactive "hypermedia." Hypermedia describes a document in which "hypertext links" are used to connect any combination of graphics, audio, video and text in a non-linear, non-sequential manner. Words, phrases and icons in documents become links that enable a user to jump at will to a new location in a document or to a new document, either on the same computer or on a different computer on the Web.

Hypermedia authors use a special software language known as HyperText Markup Language (HTML) to create these hyperlinks and to create and format these hypermedia documents. These hypermedia documents are generally referred to as "Web pages" and are stored on Web servers as text files, containing the information encoded in HTML. The Web pages can also be stored in databases, in database tables. Examples of currently available Web servers include NETSITE™ from NETSCAPE™, CERN™ from the European Particle Physics Laboratory and NCSA™ from the National Center for Supercomputing Applications (NCSA™) at the University of Illinois, Urbana-Champaign.

Web client machines running "Web browsers" can access Web pages stored on Web servers via a communications protocol known as HyperText Transport Protocol (HTTP). Web browsers are software interfaces that run on Web clients to allow access to Web servers via a simple user interface. A variety of Web browsers are available today, the most popular ones being "NAVIGATOR™" which can be obtained from NETSCAPE™, and "MOSAIC™" which can be obtained from NCSA™.

A Web browser allows a Web client to request a particular hypermedia document from a Web server by specifying a Universal Resource Locator (URL). An URL is a "Web address" that identifies the Web page and its location on the Web. When the appropriate Web server receives the URL, the server locates the document corresponding to the requested URL and, if required, takes action to create HTML output.

Although Web pages were traditionally stored as static files on the Web server operating system, today Web pages can also be generated dynamically using the Common Gateway Interface (CGI). CGI is a standard interface for running external programs on a Web server. It allows Web servers to create dynamic documents when the server receives a request from the Web browser. When the Web server receives a request for a dynamic document, the Web server executes the appropriate CGI script and transmits the output of the execution back to the requesting Web browser. The Web browser does not differentiate between static and dynamic documents. It simply displays the output of the request.

With CGI, it is becoming increasingly common to extract dynamic information from databases as well as from text files. As Web sites continue to grow, storing Web pages as text files in the operating system becomes unwieldy and inefficient. Databases provide significant advantages over traditional file systems for storage of large amounts of information. Database capabilities may also be exploited in other ways to increase the efficiency of generating dynamic Web pages.

SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a more efficient method and apparatus for generating dynamic Web pages. Specifically, the present invention discloses a method and apparatus for generating dynamic Web pages on a Web server by invoking and executing predefined procedural packages stored in a database.

The claimed invention comprises the steps of receiving an object request on the Web server, activating a Web agent on the Web server based on the object request, the Web agent invoking the predefined procedural package, executing the predefined procedural package to retrieve data from a data repository, and formatting the retrieved data as HTML output.

Other objects, features and advantages of the present invention will be apparent from the accompanying drawings and from the detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** illustrates a typical prior art computer system.

FIG. **2** is an illustration of a typical prior art Web server environment.

FIG. **3** illustrates the interaction between the Web browser, the Web listener, the Web agent and the database, as claimed in the preferred embodiment of the present invention.

FIG. **4** illustrates the preferred embodiment of the presently claimed invention, in block diagram form

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The present invention relates to a method and apparatus for generating dynamic Web pages on a Web server. Specifically, the present invention discloses a Web agent for generating dynamic Web pages by invoking and executing procedural packages that are stored in a database. A procedural package, for the purposes of this invention, includes procedural blocks such as Oracle™ Corporation PL/SQL™ blocks. It will, however, be apparent to one of ordinary skill in the art that other types of procedural packages may also be utilized. In the following detailed description, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent to one of ordinary skill in the art that these specific details need not be used to practice the present invention. In other instances, well-known structures, interfaces, and processes have not been shown in detail in order not to unnecessarily obscure the present invention.

FIG. **1** illustrates a typical computer system **100** in which the present invention operates. The preferred embodiment of the present invention is implemented on a SUN™ Workstation manufactured by SUN MICROSYSTEMS™ of Mountain View, Calif. Alternate embodiments may be implemented on an IBM™ Personal Computer manufactured by IBM Corporation of Armonk, N.Y. or a MACINTOSH™ computer manufactured by APPLE™ Computer, Incorporated of Cupertino, Calif. It will be apparent to those of

ordinary skill in the art that other computer system architectures may also be employed.

In general, such computer systems as illustrated by FIG. 1 comprise a bus **101** for communicating information, a processor **102** coupled with the bus **101** for processing information, main memory **103** coupled with the bus **101** for storing information and instructions for the processor **102**, a read-only memory **104** coupled with the bus **101** for storing static information and instructions for the processor **102**, a display device **105** coupled with the bus **101** for displaying information for a computer user, an alphanumeric input device **106** coupled with the bus **101** for communicating information and command selections to the processor **102**, and a mass storage device **107**, such as a magnetic disk and associated disk drive, coupled with the bus **101** for storing information and instructions. A data storage medium **108** containing digital information is configured to operate with data storage device **107** to allow processor **102** access to the digital information on data storage medium **108** via bus **101**. In addition, a CD-ROM drive (not shown) may also be used for the storage of high resolution images for display on the display device **105**.

Processor **102** may be any of a wide variety of general purpose processors or microprocessors such as the SPARC™ manufactured by SUN MICROSYSTEMS™ the MOTOROLA™ 68040 or POWERPC™ brand microprocessor manufactured by MOTOROLA™ Corporation or the PENTIUM® microprocessor manufactured by INTEL® Corporation. It will be apparent to those of ordinary skill in the art, however, that other varieties of processors may also be used in a particular computer system. Display device **105** may be a liquid crystal device, cathode ray tube (CRT), or other suitable display device. Mass storage device **107** may be a conventional hard disk drive, floppy disk drive, or other magnetic or optical data storage device for reading and writing information stored on a hard disk, a floppy disk, a magnetic tape, or other magnetic or optical data storage medium.

In general, processor **102** retrieves processing instructions and data from a data storage medium **108** using data storage device **107** and downloads this information into random access memory **103** for execution. Processor **102**, then executes an instruction stream from random access memory **103** or read-only memory **104**. Command selections and information input at alphanumeric input device **106** are used to direct the flow of instructions executed by processor **102**. Equivalent input device **106** may also be a pointing device such as a conventional mouse or trackball device. The results of this processing execution are then displayed on display device **105**.

The preferred embodiment of the present invention is implemented as a software module, which may be executed on a computer system such as computer system **100** in a conventional manner. Using well known techniques, the application software of the preferred embodiment is stored on data storage medium **108** and subsequently loaded into and executed within computer system **100**. Once initiated, the software of the preferred embodiment operates in the manner described below.

FIG. **2** illustrates a typical prior art Web server environment, containing conventional objects. An object includes HTML files, GIF or JPEG files, any files supported by Web browsers and Common Gateway Interface (CGI) scripts. Each object has an attribute associated with it that identifies the type of object. For example, a AVI extension on a Web object indicates that the object is a Microsoft™ Video object. Web browser **201** interacts with Web server **200** via Web server executable **202**. Web browser **201** makes an object request from Web server executable **202**. Web server executable **202** locates the object either in a text file on the operating system of Web server **200**, or a table in database **206**, based on the object request URL. If the object is an HTML file, a GIF file or other files supported by Web browsers, Web server executable **202** retrieves and returns object **205**(1) or **207**(1) to Web Browser **201**. If the object requested is a CGI script, however, CGI script **203** will generate dynamically created HTML output and transfer the output to Web server executable **202**, and back to requesting Web browser **201**.

The present invention discloses a method and apparatus for generating dynamic Web pages on a Web server by invoking and executing procedural packages such as ORACLE™ Corporation, PL/SQL™ that are stored in a database. For the purposes of this invention, a "database" is defined as an actual database or any repository that can be made to look like a database to the Web server. Procedural packages provide an object oriented approach to database programming. A package consists of a public header declaration and a private body containing the definitions. The public package header contains function and procedure prototypes as well as other procedural objects such as variables, cursors and exceptions. The private package body contains definitions for objects declared in the package header as well as private functions, procedures, cursors, variables and exceptions, which are only accessible from other functions and procedures in the package.

Each procedure or function may have zero or more parameters and each parameter may be defined as being read-only or writeable by the program unit. One advantage of packaging functions and procedures is an object oriented feature that allows multiple versions of the same function or procedure, with different "signatures." Signatures in this context are parameter lists which can differ in datatype and/or quantity. This feature greatly simplifies the coding effort for developers because they can use the same procedure for different types of data. It also improves code readability and makes code less prone to break when, for example, datatypes in tables are changed. Generating dynamic Web pages using procedural packages also benefits from the simple programming involved in creating procedural packages as opposed to traditional CGI scripts.

A virtual path is the pathname pointing to an HTML document presented as if it were located in the file system rather than in the database. The Web server maps the virtual path to the actual path based on a configuration file. The HTML document may be in a directory on the file system or in the database. Use of this virtual pathname ensures that the actual location of the HTML document will be transparent to a client Web browser that requests the document. The Web browser will be able to use the request an HTML document without knowing whether it is attempting to retrieve the HTML document from the file system or a database. The Web agent takes advantage of two special features available in all Web servers namely the ability to pass extra PATH information after the name of a CGI executable, and the ability to map virtual directories to the physical file system.

FIG. **3** illustrates the interaction between the Web browser **201**, Web listener **208**, Web agent **211** and Database **206**, in the preferred embodiment of the presently claimed invention. For clarity, Database **206** is illustrated as residing at a different location from Web server **200**. It will be apparent to one of ordinary skill in the art, however, that Database **206** may reside either on the same computer as Web Server **200** or on a different computer.

Web browser **201** first makes an object request from Web listener **208** on Web Server **200**. For example, if the object request specifies URL http://nhl.oracle.com/hr/owa/fireemp, the following actions will occur. The first part of the URL, http://nhl.oracle.com identifies the Web server **200**. The object request will therefore be directed from Web browser **201** to Web server **200**. Web listener **208** on Web server **200** receives the object request and examines the rest of the URL. /hr/owa, the first portion of the address, is defined on the Web server as a CGI application with a virtual path pointing to a directory containing Web agent **209**. Web listener **208** launches Web agent **209**. The preferred embodiment of the present invention implements the Web agent as an Oracle Call Interface program. It will be apparent to one of ordinary skill in the art, however, that any interface program that is similar to the Oracle Call Interface may be used to implement Web agent **209**.

After activation, Web agent **209** examines the portion of the URL that precedes "owa." In this example, Web agent **209** finds "/hr" and uses this name to determine which database service to use. A list of various database services will be stored on Web server **200**, in configuration file **210**, each service providing a list of userids and passwords for logging into Database **206**. Once Web agent **209** determines the appropriate userid and password to use for the database service, it logs into Database **206**. Web agent **209** then examines the URL to determine what follows "owa." In the example above, Web agent **209** finds "fireemp" and uses that to locate the predefined procedural package to execute. In the preferred embodiment of the presently claimed invention, "fireemp" is a PL/SQL package stored in an Oracle7™ database. It will be apparent to one of ordinary skill in the art, however, that other types of procedural packages and databases may also be utilized.

Web agent **210** then executes "fireemp" which retrieves data from Database **206**. Although in the preferred embodiment of the claimed invention, the data is retrieved from Database **206**, it will be apparent to one of ordinary skill in the art that the procedural package "fireemp" can also retrieve data from alternate data repositories. The retrieved data is then formatted with HTML tags. The formatting information may be specified in PL/SQL procedure "fireemp." Alternatively, the user can include in PL/SQL procedure "fireemp" a call to a standard set of PL/SQL packages called HTP (HyperText Procedure) and HTF (HyperText Function). HTP and HTP automatically generate HTML tags. The formatted output is then returned to requesting Web browser **201**.

FIG. **4** illustrates the preferred embodiment of the presently claimed invention, in block diagram form, using the same example as above. The Web browser issues an object request having URL http://nhl.oracle.com/hr/owa/fireemp in processing block **401**. The object request is then directed from Web browser to Web server in processing block **402**. The Web listener on the Web server receives the object request and examines the rest of the URL in processing block **403**. /hr/owa, the first portion of the address, is defined on the Web server as a CGI application with a virtual path pointing to a directory containing Web agent. In processing block **404**, the Web listener launches the Web agent "owa" in the directory corresponding to virtual path /hr. After activation, in processing block **405**, the Web agent determines an appropriate userid and password and logs into the database. The Web agent then locates and executes the procedural package "fireemp" in procedural package **406**. The procedural package retrieves data from the database in procedural block **407**. The retrieved data is then formatted with HTML tags and returned to requesting the Web browser in procedural block **408**.

Thus, a method and apparatus for generating dynamic Web pages on a Web server using predefined procedural packages stored in a database is disclosed. These specific arrangements and methods described herein are merely illustrative of the principles of the present invention. Numerous modifications in form and detail may be made by those of ordinary skill in the art without departing from the scope of the present invention. Although this invention has been shown in relation to a particular preferred embodiment, it should not be considered so limited. Rather, the present invention is limited only by the scope of the appended claims.

We claim:

1. A Web agent for generating dynamic Web pages comprising:

means for invoking a predefined procedural package stored in a database;

means for executing the predefined procedural package to retrieve data from a data repository; and

means for formatting the retrieved data to conform with a selected format.

2. A computer-readable medium having stored thereon a plurality of sequences of instructions, the plurality of sequences of instructions including sequences of instructions which, when executed by a processor, cause the processor to perform the steps of:

receiving a request on a Web server;

activating a Web agent on the Web server based on the request, the Web agent invoking a predefined procedural package stored in a database;

executing the predefined procedural package to retrieve data from a data repository; and

formatting the retrieved data to conform with a selected format.

3. A computer-implemented method for generating dynamic Web pages on a Web server using a predefined procedural package stored in a database, the computer-implemented method comprising the steps of:

receiving a request on the Web server;

activating a Web agent on the Web server based on the request, the Web agent invoking the predefined procedural package stored in the database;

executing the predefined procedural package to retrieve data from a data repository; and

formatting the retrieved data to conform with a selected format.

4. The computer-implemented method as described in claim **3** wherein the step of executing the predefined procedural package to retrieve data from the data repository retrieves data from the database.

5. The computer-implemented method as described in claim **4** wherein the step of activating the Web agent further includes the step of identifying a database service.

6. The computer-implemented method as described in claim **5** further including the step of logging into the database using the database service.

7. The computer-implemented method as described in claim **3** wherein the step of formatting the retrieved data is performed by the predefined procedural package.

* * * * *

Exhibit 26

PTO-1449 Information Disclosure Citation in an Application	Application No. 90/008342	Applicant(s) Lowery, et al.	
	Docket Number 066241.0125	Group Art Unit 3992	Filing Date November 27, 2006

U.S. PATENT DOCUMENTS

		DOCUMENT NO.	DATE	NAME	CLASS	SUBCLASS	FILING DATE
MX	C	5,701,451	12/22/97	Rogers, et al.			
MX	F	5,721,908	02/24/98	Lagarde, et al.			
MX	E	5,742,762	04/21/98	Scholl, et al.			
MX	D	5,761,673	06/02/98	Bookman, et al.			
MX	A	6,249,291	06/19/01	Popp, et al.			
MX	B	6,651,108	11/18/03	Popp, et al.			
	G						
	H						
	I						
	J						
	K						
	L						

		DOCUMENT NO.	DATE	COUNTRY	CLASS	SUBCLASS	TRANSLATION YES	TRANSLATION NO
	M							
	N							
	O							
	P							
	Q							

NON-PATENT DOCUMENTS

		DOCUMENT (Including Author, Title, Source, and Pertinent Pages)	DATE
MX	S	Knudson, Richard, "Application Development with Microsoft's Internet Information Server", IMG The Information Management Group; 4 pages; http://www.image.com/IMG/What's+New/2.6.96.htm, 01/20/2006.	
MX	T	"Microsoft SQL Server in the Active Internet", Microsoft Corporation, 7 pages; http://msdn.microsoft.com/archive/en-us/dnarsqlsg/html/msdn_sqlintwp.asp?frame=true, 12/29/2005.	03/12/1996
MX	U	"Oracle® WebServer User's Guide"; Release 1.0, Part No. A34986-2, 1-248 pages. 1995	
MX	V	"head 1.1", 27 pages. 1994	

M. Shelton 6·10·08

EXAMINER: Initial if citation considered, whether or not citation is in conformance with MPEP § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to the applicant.

p. 1 of 8