Patent Claims Revisited

Dargaye Churnet
Patent Claims Revisited

Dargaye Churnet
Patent Claims Revisited

By Dargaye Churne*

This paper proposes that the most beneficial patent reform begins with claim drafting regulations. Part I serves as an introduction. Part II highlights the problems with the nation’s current patent system. This section discusses how each of these problems is caused in part by the current claim drafting regulations. Part III reviews the changes made by the America Invents Act. Part IV proposes new regulations for claim drafting that will offer more significant benefits than those provided by the America Invents Act. Specifically, this paper argues that by requiring applicants to include a claim chart defining each claim limitation, examiners at the PTO will need less time to understand the patent’s scope, the PTO will issue higher quality patents, and patent litigation costs will be diminished because courts will devote less time to claim construction.

I. INTRODUCTION ................................................................. 501

II. PATENT PROCESS .............................................................. 503
    A. Claim Drafting .................................................................. 503
    B. Patent Examination .......................................................... 505
    C. Patent Litigation ............................................................... 507
       1. Claim Construction ......................................................... 507
       2. Patent Trolls .................................................................. 509

III. AMERICA INVENTS ACT ....................................................... 510

IV. PROPOSED SOLUTION .......................................................... 512
    A. Implementation ............................................................... 512
    B. Benefits ........................................................................... 517

V. CONCLUSION ..................................................................... 519

APPENDIX ........................................................................... 520

I. INTRODUCTION

On September 16, 2011, President Obama signed into law the most influential patent reform legislation in nearly sixty years. The Leahy-Smith America Invents Act (“Act”) is Congress’s attempt to overhaul a beleaguered patent system, which many believe was long overdue for reform. The Act does just that. It drastically changes the

* The author wrote this article in 2012 as a student at Northwestern University School of Law. Prior to attending Northwestern, the author served as a patent examiner at the United States Patent and Trademark Office for three years. The author has experience drafting and litigating patents. The author wishes to thank James Ferguson, an adjunct professor at Northwestern, for guidance in preparing this article.
filing system for U.S. patents along with the procedures for challenging applications filed to the United States Patent and Trademark Office (PTO). The Act further permits the PTO to set its own fees and maintain these funds in a separate account, thereby allowing the PTO to hire more examiners to attack the tremendous patent application backlog.

The Act was designed to fix a broken patent system. The U.S. patent system’s problems include patent pendency (the time it takes the PTO to respond from the date on which the applicant files the application), the PTO’s application backlog, the patent examination quality at the PTO, patent litigation costs, and abuse of the patent system by patent trolls, to name a few.

Although the Act addresses many of these issues peripherally, it fails to address the cause of most problems in the patent system. The problems faced in litigation are the result of a system that allows an inventor to amorphously define the metes and bounds of her invention. Far too often, patents—and, more specifically, the patent’s claims—offer little guidance to third parties as to what exactly has been invented. Such confusion leaves even well-meaning manufacturers unaware that their devices or processes infringe upon another’s intellectual property rights. This, in turn, leads the patentee to bring the infringer to court in an attempt to recover damages.

Before a court can address the issue of damages, it must first analyze the limitations of the asserted claims through claim construction. Through this process, the court reviews the patent’s claims along with the prosecution history in an attempt to accurately ascertain the metes and bounds of the invention. Once the claim terms are defined, the court can then determine whether the defendant has infringed. Thus, claim construction is a pivotal element of patent litigation.

The claims are, similarly, the central focus of the patent examiner’s review at the PTO. When the applicant has conceived of an invention and drafted a patent application, she submits it to the PTO for examination. An examiner must review the entire application under significant time constraints, and then search for relevant prior art references and draft an Office Action explaining why he has rejected or allowed the patent. The examiner’s determination of whether the patent will be issued is based almost exclusively on the claims. The examiner must interpret the claims in light of the entire specification.

Reading an entire patent application and gaining a thorough understanding of the claims may take weeks. Patent examiners, however, are expected to do so in less than 24 hours. It is no wonder, then, that many have questioned the quality of patents the PTO has issued. It is unreasonable to expect a patent examiner to adequately review patent claims vaguely linked to a lengthy and technical specification in such a short amount of time. These “bad patents” the PTO grants then become the issue of litigation and claim construction.

This paper proposes that the most beneficial patent reform begins with claim drafting regulations. Part II highlights the problems with the nation’s current patent system. This section discusses how each of these problems is caused in part by the current claim drafting regulations. Part III reviews the changes made by the America Invents Act. Part IV proposes new regulations for claim drafting that will offer more significant benefits than those provided by the America Invents Act. Specifically, this paper argues that by requiring patent applications to include a claim chart defining each claim limitation, the examiner at the PTO will need less time to understand the patent’s
scope, the PTO will issue higher quality patents, and patent litigation costs will be
diminished because courts will devote less time to claim construction.

II. PATENT PROCESS

A. Claim Drafting

To understand the value of claim drafting reform, one must first understand the
critical role that claims play throughout the patent process. The process begins when an
inventor conceives of a novel method, machine, manufacture, or composition of matter.\(^1\) The inventor will likely seek to protect her intellectual property rights to the invention. She does so by applying for a patent, which rewards her full disclosure of the innovation with a temporary monopoly on the rights to the invention.\(^2\)

The inventor—or more often, her patent attorney—must then draft a patent application to submit to the PTO. The application includes, in relevant parts: an abstract, drawings, a brief description of the drawings and invention, a specification describing the invention in detail, and, most importantly, the claims.\(^3\)

Each section of the patent application plays a different role in providing as full a
description of the invention as possible. The application begins with an abstract that
provides the reader with a single-paragraph description of the invention, the details of
which will be expounded upon throughout the application.\(^4\) Next, the application must
include drawings that are “necessary to understand the subject matter to be patented.”\(^5\) These drawings “show every feature of the invention as specified in the claims.”\(^6\) Depending on the invention, the drawings often display the invention from multiple views, with identifying symbols and references to allow the reader to associate the drawings with the claims and detailed specification.\(^7\)

Immediately following the drawings is a section briefly describing each drawing in
one or two sentences, providing the reader with a greater understanding of the aspects of
the invention being displayed in the drawings.\(^8\) Next, the inventor provides a brief
summary of the invention. This section “should present the substance or general idea of
the claimed invention in summarized form.”\(^9\) The brief summary may identify the
invention’s benefits and how they overcome preexisting problems in the field of art.\(^10\)

Each of the previous sections provides support for the claimed invention, but it is
the next section—the detailed description of the invention—that provides the most
support for the claims. In this section, “the invention must be explained along with the

\(^4\) Id.
\(^5\) Id.
\(^6\) Id.
\(^7\) See 37 C.F.R. § 1.84 (2011).
\(^9\) Id.
\(^10\) Id.
process of making and using the invention in full, clear, concise, and exact terms.’’ Most notably, this section must (1) enable a person of ordinary skill in the art to practice the invention, (2) provide a written description of what is being claimed, and (3) describe the best mode for practicing the invention. As the U.S. Court of Appeals for the Federal Circuit pointed out, the “specification aids in ascertaining the scope and meaning of the claims inasmuch as the words of the claims must be based upon the description. The specification is, thus, the primary basis for construing the claims.”

The patent application concludes with a list of the claims. The patent claims’ central function is to define the scope of legal protection that the government grants the inventor in return for her disclosure of the invention. Therefore, the patent attorney must reduce the inventor’s conception that has been described in a specification, sometimes hundreds of pages long, to a numbered list of one-sentence claims that provide adequate legal protection for the invention. In so doing, the attorney walks a tightrope as he attempts to draft claims that are simultaneously broad and narrow.

On the one hand, the attorney must ensure that the claims are broad enough to protect the inventor’s intellectual property rights to the invention. The broader an attorney drafts the claims, the more coverage the inventor has when suing third parties for infringing the patent. Therefore, broader claims provide the inventor with a more valuable patent.

On the other hand, excessively broad claims run a greater risk of being rejected by the PTO. The lack of specificity in broad claims provides patent examiners with more room for claim interpretation and a more expansive wealth of prior art that anticipate the claims. Thus, while broad claims are preferable to draft the most valuable patent to the inventor, attorneys must balance this interest with the need for drafting claims narrow enough to avoid an examiner’s rejection at the PTO.

Regardless of how broad the claims may be, their scope cannot extend beyond what is disclosed in the rest of the specification. To satisfy this requirement, the claims simply “must find clear support or antecedent basis in the description so that the meaning of the terms in the claims may be ascertainable by reference to the description.” Furthermore, in drafting the claims, the patentee may be her own “lexicographer,”

---

11 Id.
12 See 35 U.S.C. § 112 (2006) (“The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.”).
14 See Motion Picture Patents Co. v. Universal Film Mfg. Co., 243 U.S. 502, 510 (1917) (describing the interpretive rules used by the Court in interpreting patent law).
16 See MPEP § 608.01(m) (8th ed. Rev. 10, July 2010).
17 See Steven W. Lundberg et al., Crafting the Claims, in ELECTRONIC AND SOFTWARE PATENTS: LAW AND PRACTICE, § 6.02.C (Steven W. Lundberg et al. eds., 2d ed. 2005).
18 Id.
19 See MPEP § 608.01(i).
20 Id.
defining terms outside of their plain and ordinary meaning. In so doing, the patentee may refer to elements disclosed in the specification using different terms in the claims.

Given these considerations, it is essential that one read the entire specification to gain an accurate understanding of the claimed invention. A third party must often read a specification multiple times to gain a thorough understanding of the claims. The lax claim drafting regulations—specifically, for tying the claimed terms to their exact location in the specification—cause many of the current problems with the nation’s patent system. Regulations linking the claimed terms with their precise definition will resolve many of the problems presented in patent examination and litigation.

B. Patent Examination

Once the inventor and her attorney have completed drafting the patent application, they submit it to the PTO for review. A patent examiner knowledgeable in the invention’s field of art reviews the application. The examiner must read the entire application and review the drawings. Once the examiner has reviewed the entire specification to gain an understanding of the invention, he reads the claims, giving them “their broadest interpretation consistent with the specification.”

Next, the examiner conducts a search of the prior art in an attempt to find references that anticipate or obviate the claims. This search includes patents, publications, and any other evidence showing that the invention was in the public domain before the application was filed or conceived. More likely than not, the examiner will find references that he believes can be used to reject the claims. Once the search is complete, the examiner will draft an Office Action to the applicant explaining why the claims were rejected or why the patent was granted.

The applicant has an opportunity to amend the claims to overcome the prior art rejections or can argue that the rejections are improper. The examiner will receive the Office Action response from the applicant and perform a new prior art search. The examiner will then send a second Office Action to the applicant similar to the first. This

21 See In re Bass, 314 F.3d 575, 577 (Fed. Cir. 2002) (“Words in a claim are to be given their ordinary and accustomed meaning unless the inventor chose to be his own lexicographer in the specification.”); Teleflex, Inc. v. Ficosa N. Am. Corp., 299 F.3d 1313, 1325 (Fed. Cir. 2002) (“The patentee may demonstrate an intent to deviate from the ordinary and accustomed meaning of a claim term by including in the specification expressions of manifest exclusion or restriction, representing a clear disavowal of claim scope.”).
22 See Pager, supra note 15, at 778.
23 See Rowe v. Dror, 112 F.3d 473, 480 (Fed. Cir. 1997) (Examiners have the “task of examining the entire patent disclosure to discern the meaning of claim words and phrases.”).
24 In re Buszard, 504 F.3d 1364, 1366 (Fed. Cir. 2007); see also In re Yamamoto, 740 F.2d 1569, 1571 (Fed. Cir. 1984) (employing the specification analysis).
27 See MPEP § 706 (8th ed. Rev. 10, July 2010) (“The goal of examination is to clearly articulate any rejection early in the prosecution process so that the applicant has the opportunity to provide evidence of patentability and otherwise reply completely at the earliest opportunity.”).
28 Id. § 708.
29 Id.
process generally continues until the PTO grants the patent or the applicant abandons the application.

¶21 Two major problems have arisen from the PTO’s current process for examining applications. First, the PTO faces a backlog of about one million patent applications. This backlog has lengthened pendency to an average of over two years. The backlog and pendency problem result in courts congested with low quality patent disputes. Second, examiners do not have enough time to gain a complete understanding of the claimed inventions. This leads to (a) examiners rejecting applications using references that do not read on the claims and (b) examiners allowing patents when a more thorough understanding of the claims would have led them to find a reference that rejects the claims.

¶22 The PTO’s internal flaws are, in part, the cause of these problems. The average age of newly-hired examiners is around twenty-seven to twenty-eight years old. These young examiners are generally on their first or second job and use the PTO as a docking point in their careers. So, many of these new examiners only stay at the PTO for one to three years. New hires generally spend their first eight months in a patent examining training program and do not examine their first application until their sixth month at the PTO. Many of these examiners leave the PTO and are replaced by an influx of new examiners, who, in turn, leave the PTO after one to three years. Therefore, examiners with very little work experience, let alone patent examining experience, review many patent applications. Furthermore, a new examiner is often put in charge of an application reviewed by an examiner that left the PTO. The new examiner is forced to spend valuable examination time getting familiar with the application and prosecution history.

¶23 Although a supervisor reviews the junior examiner’s Office Action and search history, the supervisor is under time constraints and cannot review all of the prior art noted by the junior examiner. Thus, many applications are left to an extremely inexperienced examiner’s discretion to determine whether they are worthy of a patent.

¶24 Though not all examiners at the PTO are inexperienced, they all face the examination time constraint. On average, an examiner is expected to review an application within sixteen to seventeen hours. This includes reading the application, searching the prior art, and drafting an Office Action. Because many of the examiners lack technical expertise in their field, much of their examination time is spent sifting through the applicant’s documents and reading secondary sources to understand the art presented in the application. Furthermore, examiners often spread the sixteen to seventeen examination hours over three to four years in back and forth correspondence.

---

33 Id.
34 Id.
35 Id. at 445.
with the applicant. Simply put, examiners are not given enough time to thoroughly review most patent specifications to gain an accurate understanding of the claims.

The PTO’s problems come down to speed and quality. Critics of the current patent system desire a shorter pendency, which would diminish the application backlog, and to have the PTO issue higher quality patents. Given the PTO’s internal flaws, coupled with the rapid increase of patent applications filed to the PTO, claim drafting reform would greatly benefit patent examiners and, in turn, the entire patent process. If examiners could more quickly determine the limitations of each claim, they would both spend less time reviewing excessive specifications and have more time to search for the most relevant prior art. In so doing, examiners would be able to reject patents that are anticipated or obviated by the prior art, thereby reducing the number of bad patents granted. However, under the current system, many bad patents are granted. This leads to unwanted effects in patent litigation—namely, rising litigation costs through time spent in claim construction and the emergence of patent trolls abusing the patent system.

C. Patent Litigation

1. Claim Construction

Claim construction is the court’s process of interpreting patent claims to determine their proper scope and meaning. As described above, the PTO must construe an applicant's patent claims to determine patentability in view of novelty, obviousness, enablement and written description. Similarly, manufacturers and innovators may review and interpret the patent claims in order to determine how best to design around or improve upon the claimed invention. Claim interpretation further affects patent licensing negotiations, as the value of patent licenses depends on patent claim scope.

During patent litigation, claim construction serves the dual purpose of determining whether the defendant has infringed the patent and determining whether the patent is valid. Before a court can determine whether the patent has been infringed, it must first determine the patent claim scope by construing the claims. A validity analysis requires the court to compare the construed claims to the prior art as well as to the patent disclosure itself. Claim construction, therefore, is a critical factor in patent litigation and is often the first step in resolving patent disputes.

In order to determine whether an accused action infringes the patent or if prior art invalidates the patent, the court must know what the claims in the patent mean. Courts generally give claim terms their plain and ordinary meaning. This interpretation is

---

40 See Miller, supra note 38, at 199.
41 See Kimberly A. Moore, Are District Court Judges Equipped to Resolve Patent Cases?, 15 HARV. J.L. & TECH. 1, 2 (2001) (“[C]laim construction is the touchstone for any infringement or validity analysis.”).
42 See Markman v. Westview Instruments, Inc., 52 F.3d 967, 976 (Fed. Cir. 1995) (en banc).
43 See Cotropia, supra note 39, at 74–75.
supplemented by the patent's specification and the prosecution history; it may also include the context of other claims in the same patent application.

Courts do not always apply the plain and ordinary meaning to claim terms. If the disclosure provides specific definitions, the court will apply those definitions to the claim terms. However, patentees are limited in their ability to be their own lexicographer. For instance, they cannot disclaim definitions or prior art from the claims. Further, the court must always construe the claims in light of the prosecution history and prior art. Accordingly, courts will not construe claims to mean something that the PTO rejected or the patentee eliminated through amendments during patent prosecution.

If ambiguity persists after applying these techniques, the Federal Circuit has indicated that courts can rely on extrinsic evidence, such as technical dictionaries or expert testimony. If a claim is still unclear after a court applies all of the above claim construction rules, it should construe the claim so as to be valid if possible. Doing so usually results in the court applying a narrow claim construction.

Patent litigation is notoriously costly; some studies estimate that the median cost is as much as $4 million for a case in which the stakes are between $1 million and $25 million. A portion of this cost is attributable to time spent on claim construction. To prepare for the Markman hearing at which the court considers evidence and arguments that it uses to construe the claims, the patentee will spend time carefully reviewing all prior art in order to propose a construction that avoids the prior art and encompasses the accused product.

The defendant will also review the prosecution history to determine what interpretations the patentee has disclaimed. In addition, the defendant will review the prior art in order to propose a construction that encompasses the prior art and avoids the accused product. The Markman hearing and resulting claim construction ruling by the court is the most important part of most cases.

After the court issues a claim construction ruling, the parties must proceed based on that ruling. Since claim construction is a legal question, the Federal Circuit reviews a district court's claim construction de novo with no deference given to the lower court's factual findings. If, as happens in a substantial percentage of all reported appeals, the Federal Circuit reverses the district court based on the claim construction ruling,

\[(2005).\]

45 Phillips v. AWH Corp., 415 F.3d 1303, 1321 (Fed. Cir. 2005) (en banc).
46 See id. at 1325.
47 See id. at 1315–16, 1319.
48 See id. at 1316.
49 See Microsoft Corp. v. Multi-Tech Sys., Inc., 357 F.3d 1340, 1350 (Fed. Cir. 2004).
50 See Phillips, 415 F.3d at 1317–18.
51 See id. at 1327.
52 See Miller, supra note 38, at 198.
55 See Lemley, supra note 44, at 101–02.
57 See Cybor Corp. v. FAS Techs., Inc., 138 F.3d 1448, 1454 (Fed. Cir. 1998) (en banc).
parties must repeat all of their trial preparation and, perhaps, even the trial. This is, at least in part, attributable to differing claim construction standards and can substantially increase litigation costs.

2. Patent Trolls

The ambiguity of patent claims has contributed to the emergence of patent trolls. This group, often referred to as “non-practicing entities,” acquires patents with no intention of practicing the invention. Instead, the troll simply waits for a manufacturer to sufficiently commercialize a product that could arguably read on the troll’s patent and then seeks to extract exorbitant licensing fees. Patent trolls thrive in conditions where they can easily acquire bad patents, patent litigation costs are extremely high, and the risk to a defendant of losing a patent suit is potentially crippling. As a result, U.S. companies face a plethora of patent suits brought by plaintiffs with arguably substandard patents. In fact, a Boston University study has revealed that patent trolls have cost U.S. innovators $500 billion in lost wealth from 1990 to 2010.

The mere threat of litigation can be a powerful tool for the patent troll to force licensing or settlement agreements from profitable manufacturers that cannot afford to stop production of the potentially infringing device or process. Consequently, the settlement or licensing fee is often extremely high, even when the asserted patent most likely would not read on the innovator’s device or process. Trolls can then use the fees obtained through licensing agreements to create a steady cash inflow to fund future legal threats. In this way, patent trolls create a disincentive to innovate and stifle research and development.

Claim drafting reform would diminish the harmful effect of patent trolls on the patent system in at least two ways. First, clearly defined claims allow third parties to more accurately determine the patent scope. Presently, manufacturers sued by patent trolls have the option of settling a potentially meritless claim or continuing through the costly and uncertain nature of patent litigation and claim construction. Parties opt for settlement when they are both uncertain of the asserted claim scope and of how the court

---

65 E.g., Myers, supra note 61, at 334.
66 Id. at 335.
will construe the claims. Rather than entering a lengthy and expensive litigation process in which they have little guidance as to how a court will construe the asserted claims, manufacturers enter settlement agreements with the trolls.

¶37 Manufacturers, then, are entering settlement agreements because they are cheaper than litigation costs and because the court could construe the claims broadly to hold the manufacturers liable for infringement. Thus, trolls are using the manufacturer’s uncertainty as to how a court will interpret a needlessly ambiguous claim and fear of exorbitant litigation costs to extort settlement agreements. More clearly defined claims would significantly limit a troll’s ability to extort funds from manufacturers because both manufacturers and courts would be able to identify a single patent scope. If the manufacturer’s device or process reads on that scope, then he will likely opt for settlement. If, in the more likely case, the troll is asserting a patent that does not read on the manufacturer’s device or process, the manufacturer can proceed through litigation and claim construction with confidence that the court will apply the same meaning to the claim terms and rule in the manufacturer’s favor.

¶38 Secondly, more clearly defined claims will reduce the time courts spend in claim construction. An attenuated claim construction period leads to reduced litigation costs. With litigation costs diminished, a major concern for manufacturers faced with infringement suits from trolls is eliminated. Currently, however, the manufacturer might still be tempted to accept a settlement agreement if it requires the manufacturer to pay far less than it would in litigation, even if the manufacturer is confident that the court will rule in its favor. Reduced litigation costs through clearly defined claims incentivize manufacturers to challenge the troll’s meritless claims through litigation rather than accepting unfavorable settlements.

¶39 This is not to say that more clearly defined claims would eliminate the troll’s presence in the patent landscape altogether. Rather, regulations requiring applicants to draft clearly defined claims would limit the troll’s harmful impact on the patent system. Unfortunately, the America Invents Act did not address the claim drafting reform necessary to fix our nation’s patent system.

III. AMERICA INVENTS ACT

¶40 In an effort to overhaul the flawed patent system, Congress passed, and the President signed into law, the America Invents Act.68 Congress has wrestled with this Act since 2005 to address issues in both patent prosecution and litigation. As described in greater detail below, the Act changes the filing system at the PTO, institutes new procedures for challenging patents, and creates a new fee collection structure for applications at the PTO.69 Although the America Invents Act makes beneficial changes to the U.S. patent system, it does not reach the root of the problem at the claim drafting level.

¶41 Most notably, the America Invents Act moves U.S. patent law away from a “first-to-invent” system. Under this system, the courts and PTO granted patent rights to the first party to conceive of and reduce to practice the invention. Even if one party filed for

69 Id.
a patent before another, the latter would be entitled to the patent rights if he could prove that he was the first to conceive of the invention. If the two parties disputed who was the first to conceive of the invention, the parties would present evidence in court or interference proceedings.

¶42 For patent applications having an effective filing date on or after March 16, 2013, conception and reduction to practice are no longer relevant in patentability analysis. Instead, the U.S. will follow the system more consistently applied internationally—the “first-to-file” system. This eliminates the need to hold interference or court proceedings to determine which inventor independently conceived of their invention within a span of a few weeks or months. The first-to-file system should therefore reduce both litigation costs and patent examination time. However, while the new rule is more straightforward than the first-to-invent rule, some argue it favors big businesses that have the money and lawyers to quickly file for patents over small businesses and entrepreneurs. Still, the change is the most significant in the America Invents Act, and one that will at least moderately improve the U.S. patent system.

¶43 The Act also provides new ways for third parties to challenge bad patents through pre-issuance submissions and post-grant review. Pre-issuance submissions will allow third parties to provide the PTO with potentially invalidating prior art, but only while a patent application is pending. Post-grant review will allow a third party to present legal challenges to a patent to the PTO, but only in the first nine months after the patent issues. Both processes should have the intended effect of minimizing the number of bad patents the PTO issues without depleting judicial resources.

¶44 To take advantage of these changes, however, parties must constantly monitor the activity of the PTO. Critics argue that such legislation once again benefits big business with the resources to monitor activity within the PTO and only provides more jobs for patent attorneys rather than entrepreneurs. Therefore, though pre-issuance submissions and post-grant review offer new avenues to challenge bad patents, they are unlikely to make serious improvement to the patent system unless the general public becomes more cognizant of the PTO’s inner workings.

¶45 Many have argued that the best way to improve the quality of patents issued by the PTO is for Congress to provide more funding to the PTO to hire more examiners. By hiring more examiners, the PTO could reduce its application backlog. Currently, Congress controls the PTO’s budget and sets its fees. The America Invents Act, however, enables the PTO to set its own fees in an effort to improve its patent

---

70 Id. § 3.
72 Leahy-Smith America Invents Act § 8.
73 Id. § 6.
74 Id. § 8.
75 Id. § 6.
77 See, e.g., Allen E. Hoover, Let’s Run the PTO as a Business, 14 INTELL. PROP. TODAY 12, 27 (2007).
examination process.\textsuperscript{79} However, Congress will continue to have some budgetary power and be able to appropriate funds that the PTO will place in escrow.\textsuperscript{80} For this reason, critics question how much the Act will actually increase funding at the PTO to overhaul IT and hire more examiners.

The America Invents Act failed to address other areas of the patent system. The Act does nothing to limit patent damages by aligning them with any actual value of a patented invention. Similarly, patent trolls are not deterred from extorting more funds from innovators and manufacturers. Furthermore, although the Act makes beneficial changes to improve the patent system, it does so peripherally, without reaching the root of the problem: claim-drafting regulation.

IV. PROPOSED SOLUTION

A. Implementation

This paper makes a simple proposal to improve many flaws of the U.S. patent system. Inventors applying for a patent with the PTO should be required to submit a claim chart included in their application. This procedural alteration would enhance a third party’s understanding of the invention’s scope in a much more timely fashion than the present system. The change will improve patent examination quality at the PTO and reduce litigation costs spent in claim construction.

Parties generally draft claim charts in litigation to argue their position that a device or process does or does not infringe on the asserted claims. Therefore, the plaintiff will provide a broad definition of the claims in order to persuade the court that the defendant has infringed on the claim. Conversely, the defendant will provide a narrow interpretation to avoid infringement. Instead of courts continuing this time-consuming practice of requiring competing claim charts to determine an ex post definition of the claims, they should require the claim chart and associated definitions within the patent itself.

The claim chart included within the application would provide great benefits for patent examiners and those who must interpret the claims. As an example, consider U.S. Patent No. 7,269,636 (see \textit{infra} Appendix). Claim 1 reads:

A method of operating a computer network to add function to a Web page comprising:

- downloading said Web page at a processor platform, said downloading step being performed by a Web browser;
- when said Web page is downloaded, automatically executing a first code module embedded in said Web page;
- said first code module issuing a first command to retrieve a second code module;

\textsuperscript{79} Leahy-Smith America Invents Act § 10.\textsuperscript{80} Id.
assembling in response to said issuing operation, said second code module having a service response;

said first code module issuing a second command to initiate execution of said second code module; and

initiating execution of said second code module at said processor platform in response to said second command.\textsuperscript{81}

\textsuperscript{\textsection50}This is the first of 29 claims that will legally define the patent scope. However, without more, it is almost impossible for a third party to determine the invention’s scope from the language in Claim 1. To do so, the third party would have to meticulously examine the twenty pages of support in the highly technical specification. This would most likely require multiple readings of the specification while noting where each claim term is defined or described.

\textsuperscript{\textsection51}Claim 1 is not necessarily a poorly written claim, and its ambiguity is certainly not an anomaly in claim drafting. Patent drafting is a difficult process. It is a great skill for one to be able to transform each of the invention’s features into words. Furthermore, as described earlier, those drafting the claims must balance the interests of using language narrow enough to avoid rejection by the PTO and broad enough to protect the inventor’s intellectual property rights and ability to sue infringers. Claim 1 has achieved both goals. The patent has been issued and the claim’s language is ambiguous and broad enough for the patentee to assert it against third parties performing a wide variety of processes.

\textsuperscript{\textsection52}Now, consider the proposed claim chart below, tying each of Claim 1’s limitations to its definition within the specification, along with an example of the limitation:

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\textbf{CLAIM 1} & \textbf{SPECIFIC DEFINITION} & \textbf{EXAMPLE} & \textbf{PRIOR ART}\textbf{ (OPTIONAL)} \\
\hline
A method of operating a computer network to add function to a Web page comprising & “function, such as streaming media or other media services” – col. 5, l. 38-40 & A method for adding to a web page, like Yahoo.com, a pop-up that looks like a radio and plays streaming music & U.S. Patent No. 5,796,952 – also includes a method within a computer network adding different functions to a web page col. 2, l. 40-45 \\
\hline
See Fig. 4 (111) & downloading said Web page at a processor platform, said & & \\
\hline
“Second processor platform 24 includes a CPU 40, a memory 42, input/output lines 44, an input device 46, such as a & Yahoo.com is downloaded by Internet Explorer at a personal & U.S. Patent No. 5,796,952 – web browser downloads a &
\hline
\end{tabular}
\end{table}

\textsuperscript{81}U.S. Patent No. 7,269,636 (filed July 1, 2003).
<table>
<thead>
<tr>
<th>downloading step being performed by a Web browser</th>
<th>keyboard or mouse, a display device 48, such as a display terminal, and speakers 50.” – col. 4, l. 9-12</th>
<th>computer</th>
<th>web page at client col. 5, l. 12-16</th>
</tr>
</thead>
<tbody>
<tr>
<td>See Fig. 1 (24)</td>
<td>“Web browser 52 is software which navigates a web of interconnected documents on the World Wide Web via Internet 28.” – col. 4, lines 23-25</td>
<td></td>
<td></td>
</tr>
<tr>
<td>See Fig. 1 (52)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>when said Web page is downloaded, automatically executing a first code module embedded in said Web page</td>
<td>“First code module 36 executes enough functionality to act as a “bootstrap loader” in order to load second code module 90” – col. 5, l. 9-11</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td>See Fig. 1 (36) and Fig. 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>said first code module issuing a first command to retrieve a second code module</td>
<td>“A first command line (LINE NO. 1) 92 contains an exemplary initialization for a first command 93, i.e., a script, that will activate a Web address 94 for contacting server system 26 (FIG. 1) and call CGI program 84 into execution. In addition, first command line 92 communicates Web address 38 to server system 26 via a network connection 96 (FIG. 1) over Internet 28…CGI program 84 initiates the downloading of second code module 90 to a second processor platform.” – col. 5, l. 14-24</td>
<td>The first piece of code within Yahoo.com loads the second piece of code by issuing command</td>
<td>N/A</td>
</tr>
<tr>
<td>See Fig. 2 (92, 93, 94)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>assembling in response to</td>
<td>“Task 144 causes processor 62 (FIG. 2) to form a service”</td>
<td>Once the command to</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>said issuing operation, said second code module having a service response</td>
<td>response indicating a denial of service. In a preferred embodiment, a desired service response is media appliance metaphor 111 functioning to provide streaming media, in this case music, along with Web page 34. However, with respect to task 144, the service response indicating denial of service may be the media appliance metaphor 111 having a slash through it. Alternatively, the service response may simply be an absence of any media appliance metaphor.” – col. 7, l. 60 – col. 8, l. 1</td>
<td>retrieve the second piece of code is issued, the second piece of code is assembled to include the radio graphic for Yahoo.com</td>
<td></td>
</tr>
</tbody>
</table>

| said first code module issuing a second command to initiate execution of said second code module | “Fourth command line 104 contains a second command 106 that initiates execution of second code module 90 that was downloaded to temporary memory 54 of second processor platform 24.” – col. 5, l. 30-35 | The first piece of code within Yahoo.com issues a second command to initiate execution of the second piece of code |

| initiating execution of said second code module at said processor platform in response to said second command | See Fig. 3 (246, 248) | The second piece of code is executed and the radio graphic is displayed on Yahoo.com at the personal computer in response to the second command to initiate execution |

As seen above, the first column displays Claim 1, with claim limitations separated by rows. The second column serves dual purposes—it provides support for the limitations in the specification and, more importantly, defines certain claim terms using the specification. Notice that not all terms from column 1 are defined in column 2. Only
those terms for which the applicant was the lexicographer are defined. All other terms should be given their plain and ordinary meaning.

¶54 Take the limitation recited in row 1 as an example. The limitation is “A method of operating a computer network to add function to a Web page comprising.” The only term in this limitation that is described in the specification beyond its plain and ordinary meaning is “function.” Therefore, the definition from the specification for “function” is quoted verbatim in column 2. Further, the inventor cites the quotation by column and line number to allow claim chart readers to quickly locate the definition in the specification.

¶55 The second column also cites relevant figures representing the claim limitation. This is another aid to help readers more quickly ascertain the claim’s scope. A representative figure may not always be available, but when one exists, the inventor should similarly cite it in the claim chart. Looking again at row 1, the citation reads “See Fig. 4 (111),” meaning element 111 within Figure 4.

¶56 The first two columns are fairly standard for claim charts. Most claim charts separate claim limitations in a manner similar to column 1. Column 2 generally recites a portion of a specification that one can interpret to read on the claim limitation. However, the specification in other claim charts is usually one of a prior art reference used to invalidate the patent. The proposed claim chart, instead, cites the asserted patent’s specification.

¶57 The final two columns are unique to the proposed claim chart. Column 3 provides a “real world” example of the claim limitation. This column’s purpose, similar to the first two, is to provide the reader with a quicker, more thorough understanding of the claim. The example provided for row 1 is “A method for adding to a web page, like Yahoo.com, a pop-up that looks like a radio and plays streaming music.” A reader, after reviewing column 3, now has a clear idea of what the first claim limitation was attempting to convey.

¶58 The first claim limitation is not exclusively referring to radio graphics that play streaming music. The scope goes further than that. Therefore, examples listed in column 3 of the claim chart will not limit the invention’s scope. Instead, applicants should recognize that they are simply providing one of the possibly many embodiments of the invention. Still, a real world example of the embodiment described in layman’s terms will give the patent reader a quicker understanding of the limitation and the ability to envision similar embodiments.

¶59 Ideally, the PTO will require the claim chart as a section of the application after the “Detailed Description of the Invention” and before the claims. This claim chart would only include the first three columns. However, the PTO could instead require the claim chart to be a separate form that the applicant submits to the PTO. The examiner would then receive the application along with the claim chart form including column 4. If the examiner finds a prior art reference that reads on the claim limitation, then she would cite that portion of the reference in column 4. The applicant would receive the updated claim chart along with, or in lieu of, the Office Action rejecting the application.

¶60 The claim chart above was created for independent Claim 1. In some cases, the dependent claims may be self-explanatory and a separate claim chart for each claim may be excessive. Therefore, applicants do not necessarily need to submit charts for all the
claims. Rather, the PTO could require claim charts for all independent claims and make them optional for dependent claims.

B. Benefits

Compare Claim 1 alone with the sample claim chart, and the benefit to this paper’s proposal becomes apparent. Third parties reviewing the claim for the first time will more quickly understand its scope after reading the claim chart. After reading the claim alone, a third party would have no clue what the inventor meant by “function” or “service response,” for example. The reader could gain an understanding of the claim by reviewing the specification and drawings. However, this is an arduous, time-consuming process. The claim chart does the work for the reader so he can quickly and easily access definitions and examples of the claim terms.

Employing the claim chart within the patent application should be a minor modification for the patent applicant. A patent applicant is already required to support each claim element in the specification. However, currently, applicants have very lax regulations for tying their claim terms to the specification. So, applicants or their attorneys can draft very long and dense specifications and use ambiguous terms in the claims that third parties could reasonably interpret in a variety of ways from the specification. This is especially beneficial when the PTO construes the claims narrowly, thereby avoiding prior art rejection, and the patentee then asserts the claims in an infringement suit as broadly as possible.

Although prosecution history estoppel prevents applicants from limiting claim scope in prosecution and then expanding it in litigation, the estoppel only applies when the applicant expressly limited the scope in prosecution. If the patent examiner reviewing the application interprets the claims narrowly, prosecution history estoppel does not apply. Examiners are taught to give claim terms their “broadest reasonable interpretation,” but given the ambiguity of the claim terms in view of the specification, examiners overlook prior art references that can be used to reject a broad claim. Certainly, the examiners’ stringent time constraint makes it even more difficult to review the application and search for relevant prior art references to reject it in an Office Action. The PTO and U.S. government should recognize the PTO’s internal flaws and adapt claim-drafting regulation to ease the PTO’s burden.

The claim chart forces the applicant to define the claim terms with clarity. An applicant’s focus will no longer be on the narrow/broad art of claim drafting. Ambiguous claim terms will lose their ability to transform between prosecution and litigation. Instead, claims will be easily understandable, and patents will be granted on their merits. Applicants may still be their own lexicographers, but the new terms must be defined in the claim chart. Otherwise, they will be given their plain and ordinary meaning.

The claim chart’s benefit to patent examiners is tremendous. With less than eighteen hours on average to examine an application, it is unreasonable to expect a patent examiner to review and understand entire applications, let alone to find the most pertinent prior art. The time the examiner saves by reviewing the claim chart and the clarity he gains from the chart’s definitions and examples provide him with extra valuable hours to

---

82 See Phillips v. AWH Corp., 415 F.3d 1303, 1317–18 (Fed Cir. 2005) (en banc).
search relevant prior art that might be used to reject the claims. This results in the PTO issuing fewer bad patents.

¶66 With fewer bad patents in the market, needless patent suits will be diminished. This is particularly true for patent trolls. Patent trolls thrive in conditions in which litigation is lengthy and expensive and in which they can essentially gamble on the chance that the court will construe ambiguous claim terms in their favor for exorbitant damages recovery. Trolls use these conditions to extort settlements from manufacturers.

¶67 The proposed claim chart adjusts these conditions by removing the claims’ ambiguity. This has the positive benefit of reducing litigation, or more specifically claim construction, and the associated litigation costs. Further, the manufacturer’s concern that the courts may read the claims broadly enough to encompass the alleged infringement is eliminated because all relevant parties will have the single patent scope at their convenience within the claim chart. Consequently, the proposed claim chart reduces the troll’s incentive to threaten bad faith litigation in an attempt to extort settlement agreements.

¶68 This paper’s proposal will drastically reduce the preparation time, and associated attorney’s fees, for Markman hearings because parties will no longer need to provide their own claim charts. Rather than spending months submitting competing claim charts to the court and to one another, the parties will simply refer to the claim chart presented in the patent. The court will then define each limitation as it is listed in column 2 of the chart. If the patentee did not include a definition in the chart, the court will give the limitation its plain and ordinary meaning.

¶69 The proposal does not eliminate the court’s need for Markman hearings. Instead, the proposal reduces litigants’ preparation time and the hearing’s length, which can be up to six months.83 Parties in litigation will not need to pay fees as their attorneys draft charts in an attempt to identify the most beneficial claim construction. The proposal provides the courts and all other interested parties with the claim construction. Parties will now use patent litigation, as they should, arguing that the defendant’s device or process does or does not read on the asserted claims, not arguing what those claims mean.

¶70 As previously discussed, the claim chart will be a tremendous aid for examiners reviewing patent applications at the PTO. This, in turn, will reduce the number of bad patents that the PTO issues. Nevertheless, the PTO will still issue a number of patents on which a prior art reference already reads. When plaintiffs assert these patents in infringement suits, the defendants often counter with invalidity contentions.

¶71 Through invalidity contentions, defendants compare each limitation of the asserted claims to the prior art to show why the claims are invalid. Just as the proposed chart aids the court in its claim construction, it further aids the court in its invalidity analysis. Specifically, in cases in which the prior art reference is a patent or published application, the court can compare the claim charts within those references with the claim chart included with the asserted patent to determine whether it should invalidate the claims. Currently, both parties submit their own claim charts in litigation to argue whether the prior art reference reads on the asserted claims. However, the proposed claim chart will, once again, provide the court with an unbiased, previously supplied definition of the

relevant claim limitations. Accordingly, the proposed claim chart will aid the court in its invalidity analyses.

V. CONCLUSION

¶72 The current regulations that allow patentees to draft malleable claims that can change depending on a party’s interest are at the root of the nation’s patent system problems. This paper has offered a proposal to reform patent law by requiring patent applicants to clarify their claim limitations. Specifically, the PTO should require applicants to submit a claim chart defining each claim element and to link it to the specification along with a real world example of the claim limitation.

¶73 If implemented, the proposal will drastically improve the PTO’s patent examination quality. Rather than scouring the specification for support in understanding the claim terms, the examiner can quickly determine the metes and bounds of the invention. Therefore, examiners can spend less time reviewing each application and make a significant dent in the current backlog. Furthermore, a quicker understanding of the claims allows examiners to spend more time searching for relevant prior art—time that they would have before spent interpreting the claims.

¶74 Similarly, the proposal will diminish patent litigation costs because courts will have to spend less time in claim construction. The America Invents Act addressed many areas of patent law in need of reform. However, these changes failed to address the greatest problem with our nation’s patent law—the claims. This paper’s proposal offers tremendous improvements to U.S. patent law at almost negligible cost.
**APPENDIX**

(12) United States Patent  
McCollum et al.  
(45) Date of Patent: *Sept. 11, 2007*

<table>
<thead>
<tr>
<th>Patent Numbers</th>
<th>Inventors</th>
<th>Assignee</th>
<th>Notice</th>
<th>Prior Publication Data</th>
<th>Related U.S. Application Data</th>
<th>Int. Cl.</th>
<th>U.S. Cl.</th>
<th>Field of Classification Search</th>
<th>References Cited</th>
</tr>
</thead>
<tbody>
<tr>
<td>6,009,430 A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>709/205, 769/217, 218, 219, 220, 222, 224, 225, 227, 228, 769/229</td>
<td></td>
</tr>
<tr>
<td>6,112,340 A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>709/218, 709/220, 709/217, 709/221, 709/223, 709/224, 709/225, 709/227, 709/229, 769/229</td>
<td></td>
</tr>
<tr>
<td>6,126,615 A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>709/205, 769/217, 218, 219, 220, 222, 224, 225, 227, 228, 769/229</td>
<td></td>
</tr>
<tr>
<td>6,132,564 B1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>709/227, 769/205</td>
<td>5,932,499 B1 12/2001 Ludwig et al.</td>
</tr>
<tr>
<td>6,005,086 B2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>709/229</td>
<td>7,042,991 B1 7/2007 Ambrotik et al.</td>
</tr>
</tbody>
</table>

A computer network (26) includes a first processor (22) for maintaining a Web page (34) having an embedded file code module (50) and accessible through a Web address (38). A second processor (24) supports a Web browser (32) for downloading the Web page (34) and executing the first code module (50). When executed, the first code module (50) issues a first command (53) to retrieve a second code module (90) from a server system (26). The server system (26) includes a database (68) having a service response (142, 176, 184) associated with the Web address (38). A processor (62) assembles the second code module (90) having the service response (142, 176, 184). When the second code module is retrieved, the first code module (50) issues a second command (190) to initiate execution of the second code module (90) to provide added function to the Web page (34).
<table>
<thead>
<tr>
<th>LINE NO.</th>
<th>CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>92</td>
<td><code>&lt;script src='http://bsiserver:domainname.com/cgi-bin/bsiservercall.cgi'&gt;</code></td>
</tr>
<tr>
<td>94</td>
<td></td>
</tr>
<tr>
<td>98</td>
<td><code>&lt;/script&gt;</code></td>
</tr>
<tr>
<td>100</td>
<td><code>&lt;script&gt;</code></td>
</tr>
<tr>
<td>102</td>
<td></td>
</tr>
<tr>
<td>104</td>
<td>BSISet (i);</td>
</tr>
<tr>
<td>108</td>
<td><code>//--&gt;&lt;/script&gt;</code></td>
</tr>
</tbody>
</table>

**FIG. 2**
FIG. 5
### FIG. 7

<table>
<thead>
<tr>
<th>URL 1</th>
<th>Recreational Golf</th>
<th>Denial of Access to Service</th>
<th>Denial of Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>URL 2</td>
<td>Cooking</td>
<td>Conditional Access to Service</td>
<td>Conditional Content (including URL 5)</td>
</tr>
<tr>
<td>URL 3</td>
<td>Weddings</td>
<td>Predetermined Access to Service</td>
<td>Predetermined Content</td>
</tr>
<tr>
<td>URL 4</td>
<td>Football</td>
<td>Predetermined Access to Service Flag-Conditional Service for Tracking Index 60</td>
<td>Predetermined Content</td>
</tr>
<tr>
<td>URL n</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### FIG. 8

```
VISITOR REGISTRATION SUBPROCESS

APPLY TRACKING INDEX TO FIRST PLATFORM

GENERATE ENTRY IN VISITOR DATABASE TO STORE BROWSER AND PLATFORM INFO IN ASSOCIATION WITH TRACKING INDEX

RETURN
```
<table>
<thead>
<tr>
<th>Tracking Index</th>
<th>Browser ID</th>
<th>Platform ID</th>
<th>Visitor Preferences</th>
<th>Visitor Specified Parameter Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>202</td>
<td>204</td>
<td>206</td>
<td>~208</td>
<td>~212</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Second Platform</td>
<td>Browser Info</td>
<td>Platform Info</td>
<td>Visitor Specified Parameter Set</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
<td></td>
<td>56</td>
<td></td>
</tr>
<tr>
<td>210</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIG. 9**
VISITOR PRE-REGISTRATION PROCESS

RECEIVE REQUEST FOR PRE-REGISTRATION FROM SECOND PLATFORM

RECEIVE BROWSER INFO. AND PLATFORM INFO.

APPLY TRACKING INDEX TO SECOND PLATFORM

QUERY SECOND PLATFORM TO ESTABLISH VISITOR SPECIFIED PARAMETERS

GENERATE ENTRY IN VISITOR DATABASE TO STORE BROWSER AND PLATFORM INFO. IN ASSOCIATION WITH TRACKING INDEX

APPEND ENTRY WITH VISITOR SPECIFIED PARAMETERS

EXIT

FIG. 10
METHOD AND CODE MODULE FOR ADDING FUNCTION TO A WEB PAGE

RELATED INVENTION

The present invention is a continuation of "Method And System For Adding Function To A Web Page," U.S. patent application Ser. No. 09/429,357, filed Oct. 28, 1999, now U.S. Pat. No. 6,504,691, which is incorporated by reference herein.

TECHNICAL FIELD OF THE INVENTION

The present invention relates to the field of computer networks and particularly to methods and systems for adding function to Web pages that are accessible through the Internet.

BACKGROUND OF THE INVENTION

The worldwide network of computers commonly referred to as the "Internet" has seen explosive growth in the last several years. The Internet is expected to evolve with the adoption of new forms of interactive technology, such as the basic Internet infrastructure which consists of many elements, not the least of which are the Web browsers and Web pages.

Groups of Web pages, forming Web sites, are evolving to a high level of sophistication at an staggering pace. Email to large corporations are taking advantage of this trend, and electronic commerce (E-Commerce), that is, business transactions taking place over the Internet is consuming at a rapid pace. It is highly desirable for those who would like to carry out commerce on the Internet (to have a very sophisticated Web site that can perform numerous functions and services to an increasingly sophisticated class of Web site visitors. Such Web sites may desirably include such information services as searchable databases for price, stock, shipping, etc., product information, competitive comparisons, and so forth.

In order for such information services to be successfully communicated to potential customers, it is imperative to garner the interest of large numbers of Internet users. As with most traditional forms of commerce, advertising plays an important role in attracting customers. Accordingly, what is needed is economical, yet effective, advertising and publicity in order to attract the interest of Internet users.

A recent advance in Web site technology is the addition of streaming media, as well as other more sophisticated functional enhancements, to Web sites. The concept of streaming media is defined broadly as audio and video being delivered to a Web site visitor in packets over the Internet. The streaming media can be delivered so quickly that audio sounds and graphic images can be heard and seen almost immediately, comparable in quality to commercials, over-the-air radio or television. Examples of streaming media include banners, informational forms using a "marquee", audio-based commercials, and so forth.

Unfortunately, it is expensive to add such enhancements to Web sites. Bandwidth costs for delivering streaming media may be prohibitively expensive. In addition, there are problems associated with the complexity of producing the streaming media that is to be "broadcast" over the Web sites, and licensing of the streaming media if it's proprietary.

A typical example of adding function to a Web site is the addition of an "affiliates" program. An affiliate program, provided by a third party, may be desired by the Web site developer to add functionality to their Web site for the purpose of offering the appeal of the site for or revenue sharing in which they will receive a percentage of sales. In order to obtain such an affiliate program, the Web site developer may be required to register with the supplier of the affiliate program in order to obtain and execute the affiliate program in connection with their Web site. Unfortunately, such a registration process typically requires the Web site developer to fill out lengthy on-line electronic forms. Such forms may be cumbersome and time consuming, that filling out such forms leads to their abandonment on the part of the Web site developer. If the Web site developer successfully manages to register, the Web site developer must then wait for the implementing code for the affiliate program to be e-mailed to them. Once the Web site developer acquires the implementing code, the code is then copied and pasted into the HyperText Markup Language (HTML) for the Web site whose insertion.

Unfortunately, unless capability with the Web browsers of those subsequently access the Web site with the enhanced function provided by the affiliate program is installed, then the Web site developer has successfully added the implementing code for the affiliate program, all Web browsers accessing the Web site may not be able to interpret the affiliate program and the Web site visitor may not be able to experience the added function.

SUMMARY OF THE INVENTION

Accordingly, it is an advantage of the present invention that a method and system for adding function to a Web page are provided.

It is another advantage of the present invention that a method and system are provided that are compatible with Web browsers which adhere to the standards for HyperText Transfer Protocol (HTTP).

It is another advantage of the present invention that a method and system are provided that allow function to be added to a Web page through an easily distributed software code module.

It is yet another advantage of the present invention that a method and system are provided that allows the client demand that are specific to predetermined parameters.

The above and other advantages of the present invention are carried out in one form by a method of operating a computer network to add function to a Web page.

The method calls for downloading the Web page at a processor platform. When the Web page is downloaded, a second code module is automatically executed so that second code module can execute a first code module embedded in the Web page.

The first code module moves a first command to retrieve a second code module, this second code module being embedded in the server system, and the second code module is a second command to initiate execution of the second code module at a processor platform.

The above and other advantages of the present invention are carried out in another form by a computer readable code module for adding function to a Web page. The code module is configured to be embedded in the Web page which is generated in a HyperText Markup Language (HTML), and is configured for automatic execution when the Web page is downloaded to a client machine supporting a graphical user interface and a Web browser. The computer readable code module includes means for communicating a Web address of the Web page to a server system via a network connection to initiate a download of a second computer readable code module to the client machine. The computer readable code module further includes means for communicating first information characterizing said Web browser to said server...
and means for communicating second information characterizing said client machine to said server. In addition, the computer readable code module includes means for initiating downloading of said and computer readable code module following the downloading of the second computer readable code module and means for providing a comment tag informing the Web browser to ignore the initiating means.

BRIEF DESCRIPTION OF THE DRAWINGS

A more complete understanding of the present invention may be had by referring to the detailed description and drawings in connection with the figures, wherein like reference numbers refer to similar items throughout the Figures, and:

FIG. 2 shows a block diagram of a computer network in accordance with a preferred embodiment of the present invention;

FIG. 3 shows a flow chart of a Web page display process;

FIG. 4 shows an electronic display presenting a Web page including a media appliance metaphor;

FIG. 5 shows a flow chart of a service response provision process;

FIG. 6 shows a registration sub-process of the service response provision process;

FIG. 7 shows a Web address database generated by a server system of the computer network;

FIG. 8 shows a visitor registration sub-process of the service response provision process;

FIG. 9 shows a visitor address database generated by the server system of the computer network;

FIG. 10 shows a visitor pre-registration process performed prior to the Web page display process of FIG. 3.

FIG. 11 shows the electronic display presenting the media appliance metaphor detached from the Web page, and FIG. 12 shows the electronic display presenting another Web page including the media appliance metaphor.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 shows a block diagram of a computer network in accordance with a preferred embodiment of the present invention. Computer network 30 includes a first processor platform 22, a second processor platform 24, and a server system 26. First processor platform 22, second processor platform 24, and server system 26 are connected together via a network 28. In a preferred embodiment, network 28 is the Internet. However, network 28 can also represent a LAN, a MAN, a wireless local area network, or a combination of a wireless and wireline cumulative network. It should be readily apparent to those skilled in the art that computer network 28 also includes many more processors and network systems which are not shown for the sake of clarity.

First processor platform 22 includes a central processing unit (CPU) 26, a memory 52. Memory 52 includes a Web page 34 in which first code module 90 is embedded. A Web address 38 in memory 52 is associated with Web page 34. In a preferred embodiment, Web page 34 is generated in HyperText Markup Language (HTML). HTML is the authoring software language used on the Internet's World Wide Web for creating Web pages. Web address 38 is a Universal Resource Locator (URL), or a string expression used to locate Web page 34 via a network 28. It should be readily apparent to those skilled in the art that first processor platform 22 also includes additional components such as input/output (I/O) devices, a keyboard and/or mouse, a display device 48, such as a display terminal and speakers 60. Memory 52 includes Web browser software 62 and a temporary memory 54. A first portion of memory 52 is designated for browser information (BROWSER INFO) 56 and a second portion of memory 52 is designated for platform information (PLATFORM INFO) 58. In addition, a third portion of memory 52 is designated for a tracking index 60 or co-oc, which will be discussed in detail below. Those skilled in the art will understand that memory 52 also contains additional information, such as application programs, operating system, data, etc., which are not shown in FIG. 1 for the sake of clarity.

Second processor platform 24 includes a CPU 46, a memory 42, an input/output (I/O) device 44, an input device 40, such as a keyboard or mouse, a display device 48, such as a display terminal and speakers 60. Memory 42 includes Web browser software 62, and a temporary memory 54. A first portion of memory 42 is designated for browser information (BROWSER INFO) 56 and a second portion of memory 42 is designated for platform information (PLATFORM INFO) 58. In addition, a third portion of memory 42 is designated for a tracking index 60 or co-oc, which will be discussed in detail below. Those skilled in the art will understand that memory 42 also contains additional information, such as application programs, operating system, data, etc., which are not shown in FIG. 1 for the sake of clarity.

Web browser 52 is software which navigates a Web of interconnected documents on the World Wide Web via Internet 28. When a Web site, such as Web page 34, is accessed through Web address 38, Web browser 52 moves a copy of Web page 34 into temporary memory 54. Web browser 52 uses HyperText Transfer Protocol (HTTP) for communicating over Internet 28 in a preferred embodiment. Web browser 52 supports the HyperText Markup Language 1.0 and the Internet Extensible standards, such as Netscape 3.0 and above, Internet Explorer 3.0 and above, and the like.

Browser information 56 is information specific to first processor platform 22. Platform information 58 includes, for example, make and version of Web browser 52, what plug-ins are currently present, and so forth. Platform information 58 is information specific to second processor platform 24. Platform information 58 includes, for example, make and version of platforms 24, make and version of the operating system operating on platform 24, and so forth.

Server system 26 includes a processor (CPU) 62, a memory 64, a database structure 66 having a Web address database 68 and a visitor database structure 70, and a server structure 72 for accommodating streaming media servers 74 and other media servers 76. Ports 78 are in communication with server structure 72 and Internet 28 and are used by the Transmission Control Protocol/Internet Protocol (TCP/IP) transport protocol for providing communication between computer systems. Server structure 72 and Internet 28 are used by the Transmission Control Protocol/Internet Protocol (TCP/IP) transport protocol for communicating with network 28. Server structure 72 includes a database or database instructions 82, a common gateway interface program 84, code assembler instructions 86, and communication instructions 88. Web address database instructions 82 are executed by processor 62 for maintaining and accessing Web address database 68. Likewise, visitor database instructions 82 are executed by processor 62 for maintaining and accessing visitor database 70. CGI interface program 84 executes functions at server system 26 including, among other things, checking if Web site 34 is registered. Code assembler instructions 86 are executed by processor 62 to assemble a second code module 90 which is subsequently communicated to second processor platform 24 through the execution of CGI interface program 84 and communication instructions 88. Second code module 90 is communicated from ports 78 over Internet 28 and downloaded to temporary memory 54 at second processor platform 24.
FIG. 2 shows an example of a form of code that can be used in the implementation of the present invention. First, code module M1 is generated in HTML and embedded in the HTML of Web page P1 (FIG. 1) on a Web page developer's system Web page P2. In a preferred embodiment, a first code module is generally distributed. In other words, a Web page developer's system Web page P2 on the Internet 28, and copied and pasted into a Web page during Web page development. First, code module M1 executes enough functionality to act as a "bootstrap loader" in order to load second code module M2 (FIG. 1) into temporary memory 54 (FIG. 1) of the second processor platform 24 (FIG. 1) for subsequent execution.

A first command line (LINE NO. 1) 92 contains an exemplary initialization for a first command 94, i.e., a script, that will activate a Web address 96 for calling a server system 26 (FIG. 1) and calls CGI program 84 into execution. In addition, first command line 92 communicates Web address 38 to server system 26 via a network connection 98 (FIG. 1) over Internet 28. CGI program 84 executes multiple functions at server system 26 (FIG. 1). For example, CGI program 84 checks to see whether or not Web page P4 is registered. In addition, CGI program 84 initiates the downloading of second code module M2 to second processor platform 24. A second command line (LINE NO. 2) 98 terminates the script stored in first command line 92.

A third command line (LINE NO. 3) 100 starts a new script. Third command line 100 also contains a comment tag 102 used to allow Web browser 52 to ignore a fourth command line (LINE NO. 4) 104. Fourth command line 104 contains a second command 106 that initiates execution of second code module M9 that was downloaded to temporary memory 54 of second processor platform 24. A fifth command line (LINE NO. 5) 108 contains a comment tag 102 and terminates the script begun on third command line 100.

FIG. 3 shows a flow chart of a Web page display process 110. Web page display process 110 is performed by second processor platform 24 in response to a function, such as examining multimedia or other multimedia services to Web page P2 when downloaded to second processor platform 24.

Web reference 48 (FIG. 1) is a device 34 in connection with Web page display process 110. Media appliance metaphor 111 is a software device that exists in the realm of electronic communication and has a counterpart in the real world. When displayed with Web page P2 on display device 48 of second processor platform 24, media appliance metaphor 111 is a graphic representation of something that looks and behaves like a media appliance. In the exemplary embodiment, media appliance metaphor 111 represents a radio image. Other examples of media appliance metaphors include television images, computer images, computer game toy images, and so forth. When displayed on Web page P2, media appliance metaphor 111 gives the viewer to Web page P2 the impression that they already know how to use the device because it looks and acts like something that they are already familiar with.

Metaphors take any form desired for which practical programming constraints can be met. This includes, but is not limited to, interactive video games, network games, network information appliances such as Web-based telephones or call centers, and annotation service appliances, like beepers. First, code module M2 (FIG. 1) is used to apply an appliance metaphor on a Web page is a universal program interface, and acts as a bootstrap loader capable of activating and executing programs capable of such a purpose.

Although the present invention is described in connection with the presentation of media appliance metaphor 111 as applied to Web page P2, it need not be limited to such a media appliance metaphor. Rather, first code module M1 (FIG. 1) can be embedded in a Web page to be executed by a visiting processor platform in order to activate other code modules not associated with media appliance metaphors.

With reference back to FIG. 3, Web page display process 110 begins with a task 112. Task 112 accesses Web browser 52 to download Web page P4 of second processor platform 24. In other words, Web browser 52 moves the copy of Web page P4 with the embedded first code module M1 into temporary memory 54 (FIG. 1) of second processor platform 24.

When Web page P4 is downloaded to second processor platform 24 in task 114, a task 116 is performed. Task 116 causes Web browser 52 to automatically execute first code module M1 embedded in Web page P4, a copy of which is now stored in temporary memory 54.

Following task 114, a task 118 is performed. At task 118, first code module M1 executes first command line 92 (FIG. 1) to retrieve second code module M9 by issuing first command 93 to activate Web address 96, context server system 26 (FIG. 1), and call CGI program 84 into execution.

A task 118 is performed in connection with task 116. Task 118 causes second processor platform 24 to communicate Web address 38 to server system 26 through the execution of first command line 92, as discussed previously.

Next, a task 120 is performed. Like task 118, task 120 causes second processor platform 24 to communicate browser information P5 (FIG. 1) and platform information P6 (FIG. 1) through the execution of first command line 92, to server system 26. Following task 120, second processor platform 24 performs additional activities (not shown) pertinent to the downloading and presentation of Web page P4 on display device 48 (FIG. 1). Furthermore, an indication of displayable text following task 120, and relevant to display process 110, second processor platform 24 awaits communication from server system 26 before display process 110 can proceed.

FIG. 5 shows a flow chart of a service response provision process 122 performed by server system 26 (FIG. 1) in response to display process 110 (FIG. 3). Process 122 begins with a task 124. Task 124 causes processor 42 (FIG. 1) of server system 26 to receive first command 98 (FIG. 3).

In response to receipt of first command in task 124, a task 126 is performed. At task 126, server system 26 receives Web address 38 communicated by second processor platform 24 at task 118 (FIG. 1) and display process 110 (FIG. 1). Task 126 includes a task 128, a task 130, a task 132, a task 134, a task 136, a task 138, and a task 140. Task 128 causes processor 42 of server system 26 to perform a registration subprocess 132 (FIG. 1). Process 122 proceeds to a task 130. Task 130 causes processor 42 of server system 26 to perform a registration subprocess 132 (FIG. 1). Registration subprocess 132 is performed by server system 26 to register Web page P4 with the controlling entity of server system 26. In addition, registration...
subprocess 132 is performed to determine a service response. Alternatively, registration subprocess 132 is invoked immediately following the design of Web page 24 by a Web page developer. For example, following the design of Web page 24, the Web page developer may download Web page 24 at a processor platform to review the graphed, textual, and multimedia content of Web page 24 before Web page 24 is completed. Immediately upon acknowledgment that there is no entry in Web address database 68 (FIG. 1), response task 132 begins as with task 134. Task 134 causes server system 26 to retrieve Web page 24. Task 134 may also cause server system 26 to retrieve Web page 24 for any entry that are named in association with Web page 24.

In response to task 134, a task 136 is performed. Task 136 causes processor 62 of server system 26 to execute a portion of Web address database instructions 80 to return information content of Web page 24. The information content of Web page 24 is derived from all characters and words that are written on Web page 24, and that are publicly accessible. The information content may then be reduced by extracting navigation elements, or HTML tags, embedded in Web page 24 that are used to specify information about Web page 24. In particular, that "keyword" and "description" meta elements usually contain inforamtion that search engines and other search engines need to capture. Web page 24. Other informational contexts which may be extracted are sites, other URLS, domain names, domain name extensions (such as .com, .net, .gov, etc.), and so forth.

Following task 136, a task 138 is performed. Task 138 causes processor 62 to archive the information content described in connection with task 136.

In response to task 138, a task 140 is performed. Task 140 causes processor 62 to execute Web address database instructions 80 to produce a particularized "signature" for each of Web page 24. The signature is important for determining the course of the interest by a user using second processor platform 24 to display Web page 24 from where the profile is produced in order to perform a service response (discussed below) related to the profile.

Following task 140, a query task 142 is performed. Query task 142 determines whether or not Web page 24 can be registered. Processor 62 (FIG. 1) may determine that Web page 24 cannot be registered if the information content of Web page 24 is objectionable or otherwise unacceptable to be displayed with added function, i.e., media appliance metaphor 111 (FIG. 4). When query task 142 determines that Web page 24 is not to be registered, subprocess 132 proceeds to task 144.

Task 144 causes processor 62 (FIG. 2) to form a service response indicating a denial of service. In a preferred embodiment, a denied service response to media appliance metaphor 111 is provided in a manner similar to the one described above involving Web page 34. However, in response to task 144, the service response indicating denial of service may be the media appliance metaphor 111 having a slash through it. Alternatively, the service response may simply be an absence of any media appliance metaphor.

Following task 144, subprocess 132 proceeds to task 146. Referring to FIG. 7 in connection with task 146, FIG. 7 shows Web address database 68 of server system 26 (FIG. 1). Web address database 68 includes a minimum, a Web address field 106, a Web page profile field 152, a service response field 154, and a parameter set field 156. Task 146 (FIG. 8) causes processor 62 (FIG. 1) to generate an entry, for example, a first exemplary entry 158, in Web address database 68. Web address field 106 is designated for a Web address, or URL. Profile field 152 contains the profile of the Web address produced in task 140 (FIG. 6) of registration subprocess 132. Service response field 154 is designated for a service response, and parameter set field 156 is designated for parameters used to assemble second code module 90.

First entry 158 generated in response to task 144 (FIG. 6) includes Web address 38 identified simply as URL 1 in Web address database 68 (FIG. 6), a profile Web page field 152 associated with URL 1 indicates Web page 34 as being directed toward RECREATION-ARTGOELF. A service response field 154 indicates a denial of service is caused to service response field 154 for entry 158, and a denial content parameter set 144 associated with service response 154 used to form an audible, visual, or other presentation of denial service response 156.

Referring back to query task 142 (FIG. 6) of registration subprocess 132, when query task 142 determines that Web page 24 is registered, subprocess 132 proceeds to a query task 166. At query task 166 (processor 62 (FIG. 1) may execute a portion of Web address database instructions 80 to determine if a service response for Web page 24 is to be customized. That is, the Web page developer of Web page 24 may indicate in advance that the appearance of metaphor 111, the manner and formats of the audio channels, the banners that are displayed, the specific type of informational needs, and so forth.

When processor 62 (FIG. 1) determines that the service response is to be customized, registration subprocess 132 proceeds to task 168. At task 168, processor 62 (FIG. 1) establishes a parameter set for customization of media appliance metaphor 111 to be applied to Web page 34. The custom metaphor is defined by the parameter set. Establishment of the parameter set may be performed through a query exercise performed between server system 26 and the Web page developer of Web page 24. Customization can include references to commercials targeted to Web page 34, custom configuration data, custom Web page metaphor preferences, Web page cursor preferences, and so forth.

In response to task 168, task 170 is performed. Task 170 causes processor 62 to form a service response indicating conditional service, i.e., presentation of media appliance metaphor 111 that has been customized as a result of the activities associated with task 168. Following task 170, registration subprocess 132 proceeds to task 146 for generation of an entry in Web address database 68 (FIG. 7) to store the service response in association with the Web address.

Referring momentarily to FIG. 7, Web address database 68 includes a second exemplary entry 172. Second entry 172 generated in response to task 170 (FIG. 7) includes a Web address 38 in Web address field 106 identified merely as URL 2. A profile Web page field 152 associated with URL 2 indicates Web page 34 as being directed toward
Texas Cooking: A service response to a user request generally involves a coordination of service response 156 for entity 172, and a content recommendation set 178 associated with conditional service response 176. In this embodiment, the service response is not to be customized, registration subprocess 152 proceeds to task 160. Task 160 causes processor 62 to perform a service response registration is performed, or default, service. Such a service response is determined by the entity controlling server system (Fig. 26): it being directed toward WESCOING. A service response 166 indicating a predefined request is stored in service response field 154 for entity 182, and a pre-definition content exists. In this embodiment, service response 186 is used to form an image, or other presentation of predefined content to service response 186.

Following task 160, the formation of service response 142 indicating denial of service, the formation of service response 176 indicating conditional service, or the formation of service response 186 indicating predefined service, Web page 34 is registered, and subprocess 132 exits.

Returning back to service response provision process 122 (Fig. 26), following task 130, a request registration subprocess 132 (Fig. 37) has been performed, or when query task 128 determines that Web page 34 (Fig. 30) is not included in the previous register provision process 122 continues with task 199. Task 199 causes processor 62 (Fig. 4) to receive browser information 36 (Fig. 3) and platform information 38 (Fig. 4) from second processor platform 24 (Fig. 11). As discussed previously, browser information 36 includes, for example, make and version of Web browser 22, as well as user's current location. Similarly, platform information 38 includes, for example, make and version of platform 24, and date and version of the operating system operating platform 24, and so forth.

In response to task 199, a query task 192 is performed. Query task 192 examines processor 62 to receive a portion of visitor database instructions 32 (Fig. 4) to determine if there is any entry in visitor database 70 related to the browser information 36 and platform information 38. When query task 192 determines that there is no entry in visitor database 70, indicating that a user of second processor platform 24 has not previously downloaded a Web page containing first code module 36, provisioning process 122 proceeds to a task 194.

Task 194 causes processor 62 to further execute visitor database instructions 32 to perform a visitor registration subprocess. Fig. 8 shows a visitor registration subprocess performed to a task 194. Visitor registration subprocess 196 performs for tracking visitor's interests that may be useful for targeting advertising and tailoring added function to Web pages.

Visitor registration subprocess 196 generates visitor database 70 in response to task 198. Task 198 causes processor system 26 (Fig. 4) to apply tracking index 60 to second processor platform 24 via network connection 56. Tracking index 60 also includes a user profile or personalization information is a feature of HTTP server 26 which allows the entity controlling server system 26 to place information in memory 42 (Fig. 4) of second processor platform 24. Tracking index 60 allows server system 26 to both store and retrieve information on second processor platform 24. Tracking index 60 is persistent, meaning it remains in memory 42 (Fig. 4) of second processor platform 24, for subsequent use by server system 26. Since tracking index 60 is persistent, tracking index 60 can be used by server system 26 to track a visitor, using second processor platform 24, to any Web page that has embedded therein first code module 36.
and platform information 58 from a second processor platform 24 via network connection 96.

Following task 220, task 222 is performed. In a manner similar to task 159 of visitor registration process 109 (Fig. 8), server system 26 applies a tracking index or cookie, such as tracking index 60, to second processor platform 24.

Next, task 224 is performed. To task 224, processor 62 and second processor platform 24 perform an interactive process to obtain visitor specified parameters for establishing visitor specified parameter set 212 (Fig. 9). Such visitor specified parameters may include, for example, the appearance of specified metaphors, specific audio channels, format preferences, such as location on the Web page, size, color, and so forth.

Following task 224, a task 226 is performed. Task 226 causes processor 62, through the execution of visitor database instructions 82 (Fig. 9), to generate an entry, such as entry 210, in database 70 to store browser information 56 and platform information 58 in association with tracking index 60.

In addition, a task 228 is performed in cooperation with task 226. Task 228 causes processor 62, extracting visitor database instructions 82, to append entry 210 with visitor specified parameter set 212, as illustrated in visitor database 79 (Fig. 9). Following task 228, visitor pre-registration process 216 exits.

Referring back to query task 214 of service response provision process 122 (Fig. 8), when processor 62 determines that entry 210 (Fig. 9) includes visitor specified parameter set 212 obtained through the execution of visitor pre-registration process 216 (Fig. 9), processor 122 proceeds to a task 230.

Task 230 causes processor 62 to access Web address database 68 to access a service response in service response field 154 (Fig. 7) to indicate a visitor specified conditional service is to be performed in cooperation with second processor platform 24. Referring momentarily to Web address database 68 (Fig. 7), database 68 includes an example entry 222 for a Web address. The address is identified using URL 404 and pre-fixed with "http://" to indicate that address 159 is a profile 234 in profile field 152 associated with URL 404 indicates Web page 34 as being directed toward "FOOT-411."

Service response 106 indicating predetermined service is to be executed in service response field 154 for fourth entry 234 (Fig. 7). Following predetermined context set 218 associated with service response 106 is stored in parameter set field 156.

In response to task 230, service response field 154 also includes a flag 159 associated with tracking index 60 indicating that predetermined service response 106 is amended to conditional service response 176 for second platform 24. Fig. 206 indicates if processor 62 to access visitor preferences field 208 (Fig. 9) of visitor database 70 for visitor specified parameter set 212. Although, fourth exemplary entry 212 is shown having a predetermined service response 106, it should be readily understood that the service response may be a conditional response 176 (Fig. 7) in which the Web page designer has examined metaphor 111 (Fig. 4) during registration subroutine 132 (Fig. 6).

With reference back to process 122 (Fig. 5) following task 230, when query task 214 determines that entry 210 (Fig. 9) of visitor database 70 does not include visitor specified parameter set 212, process 122 proceeds to a task 238.

Task 238 causes processor 62 to execute code assembler instructions 86 (Fig. 1) to assemble second code module 90. Second code module 90 is assembled by accessing the predetermined one of denial of service response 162 (Fig. 7), conditional service response 176 (Fig. 7), and predetermined service response 186 (Fig. 7) from Web address database 68. In addition, second code module 90 is assembled in response to browser information 56 and platform information 58. In other words, second code module 90 is assembled to include the service response and to work with any combinations of browser/platform systems.

This feature eliminates the need for an adverse program to be hard coded, installed onto Web page 34, then tested and debugged by programmer. In addition, since second code module 90 is assembled in response to browser information 56, second code module 90 is compatible with Web browser 52 (Fig. 1) used by second processor platform 24 (Fig. 1).

Second code module 90 may also include another Web address 240, represented in parameter set field 154 of second entry 175 of Web address database 68 (Fig. 7). In this example, server 246 returns to and in response to the media source (not shown) connected through Internet 20 (Fig. 1) whose location is specified by Web address 240.

Following assembly of second code module 90 in task 238, a task 242 is performed by server system 26. Task 242 causes processor 62 through the execution of CGI program 84 (Fig. 1) to communicate second code module 90 to second processor platform 24 via network connection 96. In addition, through the execution of communication instructions 88 (Fig. 1) and the execution of appropriate control and control protocols, processor 62 manages servers 72 (Fig. 1) in order to direct information content from the media source having Web address 240 to second processor platform 24.

Referring to Web page display process 110 (Fig. 3), display process 110 performs a task 244. Task 244 is complementary to task 242 of provision process 122. The process 244 causes processor 62 to execute code module 90 as server system 26 communicates second code module 90 to second processor platform 24, task 244 causes platform 24 to receive and process requests for second code module 90. Secondly, second code module 90 is subsequently stored in temporary memory 54 (Fig. 1) of second processor platform 24.

Following receipt of second code module 90, process 110 proceeds to a task 246. Task 246 causes Web browser 52 (Fig. 1) to execute short constrained line 100 (Fig. 2) of first code module 56 containing command tag 102. In addition, task 246 causes Web browser 52 to execute fourth constrained line 116 (Fig. 2) of first code module 26 issuing second command 110 to initiate (Fig. 7) execution of second code module 90.

In response to issuing second command 104 in task 248, a task 248 is performed. Task 248 causes Web browser 52 to execute second code module 90.

In response to task 248, a task 250 is performed. Task 250 causes media appliance metaphor 111 (Fig. 4) to be applied to Web page 34 for display for display device 48 (Fig. 1). Of course, as discussed previously, if the service response is denial of service response 162, media appliance metaphor 111 may be presented with a stash through it or may be absent from Web page 34.

Referring to Fig. 4, the service response is media appliance metaphor 111 possessing a radio image. Through radio appliance metaphor 111, streaming media in the form of a radio channel 252 (playing country music) is provided and transmitted through speaker 89 (Fig. 1). Country radio channel 252 enhances the appeal of Web page 24 through an audio experience that complements Web page 34 whose information content involves "Jackie, Cooking."

In connection with music provided through radio channel 252, commercials may be aired that are related to the information content
of Web page 34. Such commercials may include content relevant to Texas cooking, for example, food items, spices, barbecues, and so forth. Thus, metaphor III is able to deliver targeted advertising to a visitor accessing Web page 34.

Metaphor III also includes additional controls. For example, a drop-down menu 254 is provided for selection of a different radio channel. In addition, a control button 256 allows a user to forward and reverse radio channel 252. Another control button 258 allows a user to play or pause radio channel 252, and a volume slider 260 allows a user to adjust the volume of radio channel 252. An arrow image 262 included in metaphor III moves a portable mode (discussed below).

In response to the display of metaphor III in task 250, a query task 256 is performed. Query task 264 causes second processor platform 24, operating through Web browser 52, to determine if a command is detected to detach metaphor III from Web page 34 in order to activate a portable mode. A portable mode may be selected when a user clicks on arrow image 262. When task 252 determines that the portable mode has been selected process 118 proceeds to a task 266.

Task 264 causes second processor platform 24 to display metaphor III, in a portable mode, on a reflected display. FIG. 11 shows electronic display 40 presenting media appliance metaphor III detached from the Web page 34 and appearing on a portable mode 266. In an exemplary embodiment, when arrow image 262 is clicked, metaphor III changes in appearance to portable mode 266. This change of appearance may reflect a predetermined response by server system 26 or visitor specified preferences set in visitor pre-registration process 216 (FIG. 16).

FIG. 12 shows electronic display 40 presenting a new Web page 275 downloaded on second processor platform 24 and including media appliance metaphor III in portable mode 266. Thus, although Web page 34 (FIG. 11) is no longer displayed on electronic display 40, a user of second processor platform is still able to enjoy the information content supplied by metaphor III.

Following task 266 and when query task 264 determines that metaphor III is not to be detached from Web page 34, a query task 272 is performed. Query task 272 determines if display of metaphor III is to be terminated. Metaphor III may be terminated when a user of second processor platform 24 does not detach metaphor III from Web page 34 and downloads a subsequent Web page. In another exemplary scenario, second processor platform 24 may be voluntarily or involuntarily disconnected from server system 26 through the execution of fifth command line 369 (FIG. 2) of first code module 36 terminating second command 106 (FIG. 2). In yet another exemplary scenario, metaphor III may be terminated when in portable mode 266 by clicking on the close window control, such as an X symbol 274 (FIG. 12).

When query task 272 determines that metaphor III is not to be terminated, iteration control loops back to task 250 to continue display of metaphor III. However, when query task 272 determines that metaphor III is to be terminated process 118 proceeds to a task 276.

Task 276 causes second processor platform 24 to disconnect the display of metaphor III on display device 48. Following task 276, process 118 exits.

Referring to service response provision process 122 (FIG. 5), processes 62 (FIG. 1) of server system 26 performs query task 278. Query task 278 is complementary to query task 272 of display process 118. That is, processes 62 execute for the termination of metaphor III in query task 272 and determination of query task 278 whether service should continue.

Communication instructions 88 (FIG. 1) executed by processor 62 includes a timing parameter, or clock, that is shown that is started to allow for a continuous periodic check for continuation of service. In query task 278, when service is to continue, process 122 proceeds to a task 280. Task 280 causes server system 26, through the continual execution of communication instructions 88 as processor 62, to continue streaming media associated with metaphor III 111 to second processor platform 24. Following task 280, process 122 repeats task 278 to continue the periodic check for continuation of service.

When query task 278 determines that service is to be discontinued, process 122 proceeds to a task 282. Task 282 causes server system 26 to terminate service. That is, task 282 causes server system 26 to discontinue directing streaming media associated with metaphor III 111 to second processor platform 24. Following task 282, process 122 exits.

In summary, the present invention teaches a method and system for adding function, such as streaming audio or other media services to a Web page, through the implementation of a simple code module embedded in the HTML of the Web page. The code module is compatible with Web browsers which adhere to the standards for HyperText Transfer Protocol (HTTP) because it is implemented using a common subset of the current HTML standard command set. In addition, the code module is easily distributed through the Internet, and is readily copied and pasted into a Web page during Web page development activities, and undergoes automatic execution and registration with minimal effort by the Web page developer. The present invention is able to tailor the added function based on information about the Web page in which it is embedded and based on visitor specified preferences.

Although the preferred embodiments of the invention have been illustrated and described in detail, it will be readily apparent to those skilled in the art that various modifications may be made therein without departing from the spirit of the invention or from the scope of the appended claims. The specifications and drawings are, accordingly, to be regarded as illustrative rather than restrictive in nature. Furthermore, although the present invention is described in connection with a media appliance metaphor for providing streaming audio, it is not intended to be limited to just this form. For example, the metaphor may provide streaming video and other multimedia communication services.

What is claimed is:

1. A method of operating a computer network to add function to a Web page comprising:
downloading said Web page at a processor platform, said downloading step being performed by a Web browser;
when said Web page is downloaded, automatically executing a first code module embedded in said Web page;
said first code module issuing a first command to retrieve a second code module;
linking, in response to said issuing operation, said second code module having a service response;
said first code module issuing a second command to initiate execution of said second code module; and
initiating execution of said second code module at said processor platform in response to said second command.

2. A method of operating a computer network to add function to a Web page comprising:
downloading said Web page at a processor platform, said downloading step being performed by a Web browser;
when said Web page is downloaded, automatically executing a first code module embedded in said Web page;
said first code module issuing a first command to retrieve a second code module;
linking, in response to said issuing operation, said second code module having a service response;
said first code module issuing a second command to initiate execution of said second code module; and
initiating execution of said second code module at said processor platform in response to said second command.

3. A method of operating a computer network to add function to a Web page comprising:
downloading said Web page at a processor platform, said downloading step being performed by a Web browser;
when said Web page is downloaded, automatically executing a first code module embedded in said Web page;
said first code module issuing a first command to retrieve a second code module;
linking, in response to said issuing operation, said second code module having a service response;
said first code module issuing a second command to initiate execution of said second code module; and
initiating execution of said second code module at said processor platform in response to said second command.
A method as claimed in claim 1 wherein said first code module issues said first command to retrieve said second code module from a server system via a network connection.

A method as claimed in claim 1 wherein said assembly operation of claim 1 is performed by a server system and said method further comprises downloading said second code module to said processor platform.

A method as claimed in claim 1 wherein said Web browser employs HyperText Transfer Protocol (HTTP) said first code module and said Web page are generated in a HyperText Markup Language (HTML), and said first code module includes an element tag informing said Web browser to ignore said second command.

A method as claimed in claim 1 wherein said method further comprises:

- manner, at a server system, of said Web page;
- determining if said Web page is registered with said server system; and
- when said Web page is not registered, performing a registration of said Web page.

A method as claimed in claim 5 wherein said performing operation comprises:

- receiving said Web page at said processor platform;
- retrieving informational content of said Web page;
- archiving said informational content of said Web page; and
- producing a profile of said Web page in response to said extracting and archiving steps.

A method as claimed in claim 8 wherein said service response is related to said profile of said Web page, and said method further comprises:

- storing said service response in association with said Web address, and
- accessing said service response when said first code module issues said command so that said service response is included in said second code module.

A method as claimed in claim 1 wherein said service response is one of a denial of service indication, a provisional service indication, and a moderated service indication.

A method as claimed in claim 1 wherein said service response is a metaphor, and said method further comprises:

- the step of displaying said metaphor in connection with said Web page on said processor platform.

A method as claimed in claim 11 wherein said method further comprises:

- the step of customizing said metaphor to include a parameter set relevant to said Web page, said customized metaphor describing a conditional service response presented upon execution of said second code module.

A method as claimed in claim 1 wherein said method further comprises the steps of:

- executing said second code module in response to said initiating operation, said second code module including:
- a Web address for a second Web page;
- downloading information content from said second Web page at said processor platform; and
- presenting said information content in said service response at said processor platform.

A method of operating a computer network to said function to a Web page comprising:

- downloading said Web page at a processor platform, said downloading step being performed by a Web browser;
- when said Web page is downloaded, automatically executing a first code module embedded in said Web page;
- said first code module issuing a command to retrieve a second code module;
- receiving, at a server system, information characterizing at least one of said processor platform and said Web browser;
- assembling, in response to said issuing operation, said second code module having a service response, said assembling operation being performed by a server system, and said assembling operation assembling said second code module in response to said information; and
- downloading said second code module to said processor platform;

A method as claimed in claim 14 further comprising:

- using said tracking index to said processor platform in response to said information; and
- using said tracking index at said server system to track and identify said processor platform.

A method of operating a computer network to said function to a Web page comprising:

- downloading said Web page at a processor platform, said downloading step being performed by a Web browser;
- when said Web page is downloaded, automatically executing a first code module embedded in said Web page;
- said first code module issuing a command to retrieve a second code module;
- assembling, in response to said issuing operation, said second code module having a service response, said service response is a metaphor;
- initiating execution of said second code module at said processor platform;
- displaying said metaphor in connection with said Web page on said processor platform;
- detecting said metaphor from said Web page; and
- displaying said metaphor disassociated from said Web page.

A computer readable code module for said function to a Web page, said code module configured to be embedded said Web page produced in a HyperText Markup Language (HTML) and configured for automatic execution when said Web page is downloaded to a client machine supporting a graphical user interface and a Web browser, said computer readable code module including:

- methods for communicating a Web address of said Web page to a server system via a network connection to initiate a downloading of said second computer readable code module to said client machine;
- methods for communicating an assembly at said server system, of said second computer readable code module, to said client machine;
- methods for initiating execution of said second computer readable code module; and
- methods for downloading said second computer readable code module.
A method as claimed in claim 20 further comprising:

17. A method as claimed in claim 20 further comprising:

18. A computer readable code module as claimed in claim

19. A computer readable code module as claimed in claim

20. A computer readable code module as claimed in claim

21. A method as claimed in claim 20 wherein said

22. A method as claimed in claim 20 wherein said

23. A method as claimed in claim 20 further comprising:

24. A method as claimed in claim 20 wherein said

25. A method as claimed in claim 20 further comprising:

26. A method as claimed in claim 25 further comprising:

27. A method as claimed in claim 27 further comprising:

28. A method as claimed in claim 27 further comprising:

29. A method as claimed in claim 20 wherein said second

30. A method as claimed in claim 20 wherein said second

31. A method as claimed in claim 20 wherein said second

32. A method as claimed in claim 20 further comprising:

33. A method as claimed in claim 20 further comprising:

34. A method as claimed in claim 20 further comprising:

35. A method as claimed in claim 20 further comprising:

36. A method as claimed in claim 20 further comprising:

37. A method as claimed in claim 20 further comprising:

38. A method as claimed in claim 20 further comprising:

39. A method as claimed in claim 20 further comprising:

40. A method as claimed in claim 20 further comprising:

41. A method as claimed in claim 20 further comprising:

42. A method as claimed in claim 20 further comprising:

43. A method as claimed in claim 20 further comprising:

44. A method as claimed in claim 20 further comprising:

45. A method as claimed in claim 20 further comprising:

46. A method as claimed in claim 20 further comprising:

47. A method as claimed in claim 20 further comprising:

48. A method as claimed in claim 20 further comprising:

49. A method as claimed in claim 20 further comprising:

50. A method as claimed in claim 20 further comprising:

51. A method as claimed in claim 20 further comprising:

52. A method as claimed in claim 20 further comprising:

53. A method as claimed in claim 20 further comprising:

54. A method as claimed in claim 20 further comprising:

55. A method as claimed in claim 20 further comprising:

56. A method as claimed in claim 20 further comprising:

57. A method as claimed in claim 20 further comprising:

58. A method as claimed in claim 20 further comprising:

59. A method as claimed in claim 20 further comprising:

60. A method as claimed in claim 20 further comprising:

61. A method as claimed in claim 20 further comprising:

62. A method as claimed in claim 20 further comprising:

63. A method as claimed in claim 20 further comprising:

64. A method as claimed in claim 20 further comprising:

65. A method as claimed in claim 20 further comprising:

66. A method as claimed in claim 20 further comprising:

67. A method as claimed in claim 20 further comprising:

68. A method as claimed in claim 20 further comprising:

69. A method as claimed in claim 20 further comprising:

70. A method as claimed in claim 20 further comprising:

71. A method as claimed in claim 20 further comprising:

72. A method as claimed in claim 20 further comprising:

73. A method as claimed in claim 20 further comprising:

74. A method as claimed in claim 20 further comprising:

75. A method as claimed in claim 20 further comprising:

76. A method as claimed in claim 20 further comprising:

77. A method as claimed in claim 20 further comprising:

78. A method as claimed in claim 20 further comprising:

79. A method as claimed in claim 20 further comprising:

80. A method as claimed in claim 20 further comprising:

81. A method as claimed in claim 20 further comprising:

82. A method as claimed in claim 20 further comprising:

83. A method as claimed in claim 20 further comprising:

84. A method as claimed in claim 20 further comprising:

85. A method as claimed in claim 20 further comprising:

86. A method as claimed in claim 20 further comprising:

87. A method as claimed in claim 20 further comprising:

88. A method as claimed in claim 20 further comprising:

89. A method as claimed in claim 20 further comprising:

90. A method as claimed in claim 20 further comprising: