Date of Download: Nov 10, 2001
TP-ALL (Texts & Periodicals - All Law Reviews, Texts & Bar Journals)
453 PLI/Pat 239
Copr. © West 2001 No Claim to Orig. U.S. Govt. Works

(Cite as: 453 PLI/Pat 239)

Practising Law Institute
Patents, Copyrights, Trademarks, and Literary Property Course Handbook Series
PLI Order No. G4-3961
September 10-11, 1996

PLI's Second Annual Institute for Intellectual Property Law

*239 PATENTING CONTENT: THE EXPANDING ROLE OF PATENT PROTECTION FOR INTERNET-BASED INFORMATION PRODUCTS

Ronald S. Laurie [FNa1]
Joseph Yang [FNa2]

Copyright (c) 1996, Practising Law Institute

Table of Contents

ABSTRACT

I. INTRODUCTION

A. The Coming Internet Market Transition

B. Intellectual Property Protection

II. PATENTING CONTENT

A. Executable Content

B. The New PTO Guidelines for Examination of Software Based Inventions

1. Non-Statutory Subject Matter

(a) Descriptive Material

(b) Product Claims

(c) Process Claims

2. Obviousness determination

3. Summary of the New Guidelines

C. Application of the PTO Guidelines to Executable Content

D. The Respective Roles of Copyright and Patent Protection for

Software Products

1. The Traditional View

2. The Modern View

*243 ABSTRACT

Over the short history of the Internet, users have come to regard the system as essentially free. This perception does not reflect the reality that most of the basic building materials used to construct the information infrastructure have been, or are in the process of being, patented. Beyond patents on the infrastructure, in the very near future market forces will cause the Internet to transition from a construction phase to a commercialization phase in which the information products distributed over the Net will be increasingly subject to patent protection. This result will flow from the combined effect of technological advances and broadened concepts of software patentability.

The distinction between functionality (computer programs) and content (data) which began to blur with the advent of object-oriented programming, will be further obscured by the emergence of "executable content" distributed across the Internet. With the widespread adoption of architecture-neutral, networked computing protocols, the Internet will evolve into a global computing environment in which any application program necessary in order for *244 particular content to be utilized (e.g., displayed, performed, printed, etc.) will be invoked by the content irrespective of where in the world (literally) the application may reside.

The coincidence of this significant technological shift, the new rules on software patentability, and the huge financial investment in the coming wave of electronic commerce will dramatically expand the number and importance of patents on subject matter which has traditionally been regarded as protectable only by copyright.

I. INTRODUCTION

A. The Coming Internet Market Transition

Investment banker Hambrecht & Quist ("H&Q"), which focuses on high-growth emerging industries, has predicted that 1996 will mark a transition in the Internet industry. [FN1] In the terminology of high tech marketing consultant Geoffrey Moore, this transition represents crossing the chasm between the visionary and the pragmatist phases of the market. [FN2] Other authors have described the two phases as emergence and rapid growth, respectively. [FN3] In each case, the first phase is characterized by: (a) high prices; (b) limited or specialized distribution channels; (c) few product offerings; and (d) poor standardization, all of which limit the potential *245 audience to early adopters of the new technologies. The second phase is characterized by: (a) lower but still fairly high prices; (b) recently opened mass distribution channels; (c) competitive products battling for presence and market share; and (d) development of enduring standards, all of which lead to the adoption of the technology by the more conservative mainstream consumers. This paper addresses the interaction between the anticipated transition in the Internet market, advances in distributed object software technology and recent legal developments concerning the patentability of computer-implemented inventions.

The two phases of the Internet market transition will be referred to herein as the Infrastructure and Productization phases respectively. Each phase is dominated by the activity suggested by its name -- deployment of information delivery mechanisms in the Infrastructure phase and development of information products and services for commercial distribution via those delivery systems in the Productization phase -- but also reflects some activity of the other type. The first phase is a build-out period in which companies are constructing technology infrastructure, rather than developing useful commercial products, so the Internet is still a novelty for the majority of users. Although some commercial products are available, they are often rudimentary, poorly supported, and unstable. Thus, approximately 96% of the market ($1.15 billion of the $1.2 billion in sales) in the current Infrastructure phase relates to technology, with only 4% ($50 million) in products. [FN4] During the Productization phase, although companies will continue to improve the delivery technology, the primary focus will be on developing "real *246 applications" that are "useful, safe and fun." [FN5] In this phase, mainstream consumer access to the Internet will result in a several orders of magnitude increase in net sites, net traffic, net sales, and almost any other measure of growth. H&Q predicts that the overall Internet industry, characterized by approximately $1.2 billion in combined sales of technology and products in 1995 will grow to approximately $23 billion in combined sales by the year 2000. [FN6] This growth will also be accompanied by a dramatic increase in the relative value of products to technology: only 56% ($13 billion of the projected $23 billion in sales) of the Productization phase market will relate to technology, while 44% ($10 billion) will relate to products. [FN7] In absolute terms, the projected increase in product sales is even more dramatic: an increase of two hundred-fold in five years, or an equivalent growth rate of 189% per year, compared to 61% per year for technology. As a precursor (or confirmation) of the expected commercial activity in products, the number of commercial domains (e.g., "AOL" in the email address USERXXX@AOL.COM) registered with InterNIC, the primary Internet name registration authority, grew by 160% in the seven months ending on July 31, 1995. [FN8] Presumably this reflects a corresponding increase in the number of companies wishing to do business on the Internet. But the increase is not limited to businesses alone -- individual consumers, too, are rushing to the Internet. For example, the *247 subscriber base of NETCOM, a popular Internet access provider, grew by 174% in a roughly comparable period. [FN9]

B. Intellectual Property Protection

During the Infrastructure phase, the few "products" that have been available have generally been academic, non-commercial, or experimental, and unencumbered by intellectual property protection. This has led many users to mistakenly assume that the Internet is essentially open and free. This perception does not reflect the reality that virtually every component of the infrastructure, from the fiber backbone to the user interface, has been or is being patented. To extend an overused, and technically inapt, metaphor, the information superhighway is not a freeway -- it is a toll road -- and the collection booths are just now being erected. For example, infrastructure components that are or will be subject to patent protection involve:

1) Information Access -- how users access the Internet;

2) Marketing -- how information is marketed to users;

3) Information Management -- how users select and organize information;

4) Digital Contracting -- how buyers and sellers consummate a commercial transaction;

5) Security -- how the authenticity, integrity and privacy of the transaction or information are assured;

*248 6) Information Delivery -- how information moves across the Internet;

7) Usage Metering -- how information access and use are measured, priced and billed;

8) Payment Systems -- how purchases (or rentals) are paid for;

9) Order Fulfillment -- how tangible goods orders are executed;

10) Enterprise Integration -- how the various elements are interconnected and managed in a business organization; and

11) User Mobility -- how Internet access and security become personal and portable.

As shown in Table 1, each of these infrastructure elements includes one or more technology enablers, particular aspects of which have been patented or on which patent applications have been filed.

*249 Table 1 -- Patented Elements of the Internet Techology Infrastructure

Infrastructure Element	Technology Enabler
Networking	Protocols/topologies . LAN, WAN Switches/routers . ATM
Information Access	Browsers
	Servers
	. Subscriber services
	Modems . Cable modems
Networked (Web-Based Computing)	Application program distribution along with content . Executable content Integrating existing (e.g., legacy or database) application into Web servers . Server-side distributed processing
Marketing	Advertising Customer database Distribution channels

Information Management	Directories/searchers
Digital Contracting	Digital signatures . Alogorithms . Secure cryptoprocessors
	Certification authorities . binding cryptographic key to user/device
	Electronic notaries . Personal identification . Time stamping
	Electronic data interchange
Security	Cryptography . Public key algorithms . Key escrow . Key management
	Hardware access control . Portable, tamper-resistant smart cards, tokens, etc. . Lock-and-key systems . Dongles . Bometric readers
	Software access control . Demonstration copies (use restrictions, expiration dates)
	Network security
Information Delivery	Video compression . JPEG . MPEG
Usage Metering	Metering . Pay-per-item . Pay-per-bit . Pay-per-time
Payment Systems	Digital cash

	Digital accounts . Credit . Debit
	Electronic funds transfer
Order Fulfillment	Product distribution (goods) . Interfaces to back office systems
Enterprise Integration	Distributed computing models . Object-oriented programming systems
User Mobility	Electronic wallet . Digital Case . Credit . Debit . Identification . Medical History
	Hardware devices . Portable smart cards, tokes, etc.

*251 The above table illustrates that, contrary to popular opinion about the Internet being essentially free, much, if not all, of the technology infrastructure has been patented. The combined effect of these enabling technologies will allow widespread, low-cost, and interactive systems for delivery of virtually any kind of digital information. Such systems will also include the financial transaction capabilities needed to deploy a universal electronic commerce infrastructure having all of the functionality of the conventional (non-electronic) commerce infrastructure with the exception of the actual delivery of "hard goods" to the consumer.

Of course, systems for electronic commerce, such as Electronic Data Interchange (EDI), are not new. Existing systems, however, suffer from three primary limitations:

(1) they generally require users to be previously known to the system (e.g., a verified account), for reasons of security or payment processing;

(2) they require a predetermined relationship between the user's computing facilities and the type of information that can be used thereby. [FN10] In the extreme case, they may be architecture-specific to the point of requiring dedicated (e.g., hardwired) terminals running proprietary protocols. In a more common case, the architecture may be open (e.g., standard PCs running commercially available software) but still only accommodate data of a predetermined type for which the client has a corresponding application program (e.g., requiring a MPEG viewer to display MPEG images of products offered for sale); and

*252 (3) processing is either server-based with clients being little more than dumb terminals, or client-based with servers being little more than passive information warehouses.

In contrast to these "first generation" electronic commerce paradigms, the emerging Web-based [FN11] commerce systems will:

(1) require no preexisting relationship between the buyer (or client) and the seller (or server); and

(2) require no predetermined relationship between data and supporting application programs, which may be delivered, separately from the data, across the network from any location.

The first element will be enabled by the use of digital certificates and other cryptographic techniques to authenticate user identities and information integrity. As suggested by Table 1, cryptographic technologies have been widely patented, and will not be discussed in detail here. [FN12] [FN13] The second element, executable content, will be enabled by the universal deployment *253 of object-oriented, networked computing environments, such as Sun Microsystems' Java, into Web browsers and other Internet access tools. Java, and the implications for patent protection of Java "Applets" under the new Patent and Trademark Office ("PTO") Examination Guidelines for Computer-Related Inventions (the "Guidelines"), will be examined in the following sections of this paper.

Besides patent, other forms of intellectual property protection, namely, copyright and trademark, have attached to even the limited amount of content-based material that is currently available. For example, the recent Scientology case [FN14] examined the question of when an Internet service provider may be liable for copyright infringement due to the unauthorized posting of a copyrighted work by a subscriber to the service. The even more recent Juris case [FN15] has focused attention on trademark infringement resulting from the use of a domain name. In most past technological advances, the transition from technology deployment to product commercialization has resulted in a relative shift in intellectual property protection away from *254 patents and toward copyrights and trademarks. It is submitted that this familiar pattern will be substantially altered in the case of the Internet, and that patents will remain a significant intellectual property guarantor in the Productization phase of the Internet, due to the temporal coincidence of the above-identified factors, i.e., (1) exponential growth of the Internet industry during the Productization phase, (2) development, standardization and widespread availability of platform-independent, networked-computing programming tools for creating and distributing executable content (e.g., Java), and (3) broadened notions of software patentability under the new PTO Guidelines. Finally, this paper will conclude with some thoughts on the evolving interrelationship between patent and copyright for software-based inventions in the new global computing environment.

II. PATENTING CONTENT

A. Executable Content

As mentioned previously, a key enabling technology for the Productization phase of the Internet industry is object-oriented, networked computing tools such as Sun's Java. Java includes two primary capabilities that make it ideally suited for Internet products:

(1) it allows the deployment of architecture-neutral code (called a Java "Applet") to any computer running a Java interpreter, by making that computer behave as an architecture-neutral Java Virtual Machine; and

*255 (2) Java-enabled Web browsers allow a Web client to download content from anywhere on the Web, and interactively view or otherwise use

such content, without necessarily having the application program [FN16] associated with the content. Instead, the client can download such application programs, at run time, from remote Web sites as needed. These two capabilities together enable the universal distribution of executable content -- content that can be utilized by any computer on the Web regardless of its architecture or lack of preexisting application programs. By providing this capability, Java transforms the Web from a passive content (text, graphics, audio and video) distribution system connected to browsers acting as dumb viewers into a software distribution system connected to browsers acting as fully functional computers.

An Applet consists of selected data of interest ("Content") as well as calls to any application programs ("Functionality") needed to process the data. In object-oriented terminology, a particular Applet comprises one or more objects, each of which in turn comprises a data structure (Content) and methods [FN17] (Functionality) for manipulating the data structures. A particular Applet may also be said to be an instance (a particular object within a class) of the general Applet class, where a class may be considered a template for creating an object of a *256 specified type. Classes, in turn, may contain various levels of subclasses, so that each object in a subclass inherits the characteristics of objects in its parent class.

Typically, data structures contain Content and methods specify Functionality for accessing that Content. However, functional modules could also take object form, with the data structures and methods together defining ways of receiving and processing the data in the data structures. Thus, a Java Applet can represent Content, Functionality, or any combination thereof.

Applets are authored in Java Source Code, a C++ like object-oriented programming language. The source is then compiled to form Java Bytecode, a platform-independent instruction set that runs on any Java-compliant Web browser (e.g., Netscape's Navigator 2.0/3.0 or Sun's HotJava). Because of this universality, the Java-enabled client is said to be a Java Virtual Machine. Although the Java Virtual Machine presents a platform-independent interface from the Applet perspective, it is really a platform-specific implementation of a Java run-time interpreter. It is this run-time interpreter that allows the dynamic linking of Content and Functionality.

The Applet Bytecode is transmitted, across the Web, from a server (e.g., the accessed Web page) anywhere on the Web to a client (e.g., the accessing Web page) desiring to run the Applet. More specifically, an Applet is imbedded within the Web page by a simple HTML tag of the form APP Class="AppletName", where "AppletName.class" is the compiled Java Bytecode file located on the Web page server. Furthermore, since Java-enabled browsers support remote files at any URL, the above Applet could even be on a different Web page server specified by a HTML tag that also specifies a source (URL) for the desired Applet.

*257 The above example illustrates that an entire Applet can be located anywhere in a network for access by any other location. The same distributive property allows different portions of a Applet to be located at different locations on the Web. Thus, the Applet is an ideal delivery vehicle for executable content -- i.e., compiled Bytecode containing Content and calls to the application programs needed to execute that Content -- downloaded from a Web page to a client that does not possess the application programs. The Content carried in the Applet is loaded from the server to the client, and then the application programs, themselves Java Bytecode objects, are obtained as needed either locally or from remote Web sites as specified by a tag in the web page. The client browser's Java Virtual Machine then dynamically binds the Java Applet and the associated Bytecode application programs during interpreted, real-time execution on the client computer. This dynamic binding allows the necessary Functionality to be loaded on-the-fly, without

recompiling the entire Applet. This process is summarized in Figure 1, which illustrates the use of a Java Applet for distributing executable content to a client PC from one or more remote servers.

The foregoing shows that Java is a networked-computing technology directed to client-side computing wherein the Content and/or Functionality is accessible from anywhere on the Web. Other networked-computing technologies are directed at sever-side computing. For example, EOlas' FastApp Internet Tools Family (for creating "Weblets") and NeXT's NeXT Object Model (for creating "WebObjects") are Java-compatible networked-computing technologies directed to server-side computing wherein application programs are served from anywhere on the Web. By combining this distributed processing capability with the capability to allow distinct computer processors (e.g., legacy systems, database servers, or LAN servers) to be *258 wrapped in a compatible API, the server-side computing technologies allow existing or specialized processors to become part of a distributed Web server. Thus, existing non-Web applications can be upgraded into the Web environment -- vastly expanding the functionality available for Web-based applications. Together, the client- and server-side networked- computing technologies turn the Web into a "delivery platform for mainstream enterprise-wide mission-critical applications. These applications can take advantage of the uniform graphical interface and ease of use of the most popular Web browsers to greatly simplify user training and minimize support costs. These applications can [also] take advantage of Internet-standard tools for file transfer (FTP and MIME), asynchronous communications (email and Usenet), real-time communications (IRC, video teleconferencing, digital whiteboard, etc.), secure transactions and access control (SHTTP, PGP, etc.), as well as any new software technology that may become popular." [FN18]

Applets [FN19] thus enable a wide balance of Content and Functionality. At the Content extreme, Applets might be simple data display applications differentiated from today's Web browsers only in that the application software for displaying the data is automatically downloaded from the Web rather than being preloaded on the browsing computer. At the Functionality extreme, Applets might be highly interactive, fully-functional software systems such as electronically distributed business tools and financial services.

*259 As will be discussed below, software inventions involving executable content should qualify as patentable subject matter under the PTO's new software examination Guidelines. The following two sections of this paper discuss the Guidelines and their application to executable content.

B. The New PTO Guidelines for Examination of Software Based Inventions

1. Non-Statutory Subject Matter

During the past thirty years, the patentability of inventions implemented wholly or partially in software has been frequently challenged -- by the PTO and in the courts -- on the basis that such inventions constituted non-statutory subject matter under 35 USC 101. [FN20] In 1995 the Federal Circuit decided several cases which collectively represented a significant expansion of the scope of software patentability. On February 28, 1996, in response to these cases, the PTO released the final version of its Examination Guidelines for Computer-Related Inventions [FN21] (the "Guidelines") which deal at length with statutory subject matter issues. The history of the case *260 law and the Guidelines, respectively, is set forth in the appended two-part article which appeared in the September and October, 1995 issues of The Computer Lawyer. [FN22]

(a) Descriptive Material

As shown in Figure 2, the Guidelines instruct PTO examiners to begin the examination of a patent claim involving software by asking whether the claimed subject matter consists of descriptive material such as music, a literary work, a compilation of data, a data structure, or a computer program. This is because descriptive material per se (which is the proper domain of copyright law) constitutes non-statutory subject matter. [FN23] The Guidelines divide descriptive material into two categories: non-functional descriptive material (e.g., music, literary works, and data compilations) and functional descriptive material (e.g., data structures and computer programs). The former is always non-statutory; the latter may or may not be statutory depending on how it is claimed.

In the computer context, non-functional descriptive material is always non-statutory because it consists of passive data which is merely stored so as to be read or output by a computer without creating any functional interrelationship between the data and the specific operations performed by the computer in processing the data. Similarly, functional descriptive material per se is non-statutory because it does not functionally interact with the computer. On *261 the other hand, data structures and computer programs have the potential to cause the computer to operate in a particular manner if they are converted to a form which can be processed by the computer. Thus, functional descriptive material is statutory when claimed in terms of its structural and functional interrelationship with other claimed aspects of the invention (e.g., a computer-readable memory) which permit the descriptive material's potential functionality to be realized. [FN24]

In the case of a claim that includes both non-functional descriptive material and functional descriptive material, the presence of properly claimed functional descriptive material is sufficient to avoid a finding of non- statutory subject matter as described above. For example, a computer that recognizes a specific grouping of musical notes read from memory and upon detecting that particular sequence, causes another pre-defined series of notes to be played, defines a functional interrelationship among the data and the computing processes performed when utilizing that data, and, as such, is statutory because it implements a statutory process. [FN25]

Even if a patent claim survives the descriptive material inquiry, it must still satisfy additional statutory subject matter criteria. These criteria depend on whether the claim is to a product, i.e., a "machine" (aka system or apparatus) or an "article of manufacture" (aka manufacture) on the one hand or a process on the other.

*262 (b) Product Claims

The Guidelines state that a product claim may be one of two types: (a) one that defines a specific product ("specific product claims"), or (b) one that, while cast in terms of a machine or manufacture actually covers any physical embodiment of the underlying process ("non-specific product claims"). The latter category would include: (i) a claim that encompasses each and every machine for performing the process, and (ii) a claim that encompass each and every manufacture that can cause a computer to perform the process.

A specific product claim requires the recitation of hardware. More particularly, claims that define a computer-related invention as a specific machine or specific article of manufacture must define the physical structure of the machine or manufacture in terms of: (1) specific hardware, or (2) generic hardware plus specific software. For example, to adequately define a specific (special purpose) computer, e.g., an FFT (Fast Fourier Transform) processor, the claim could identify: (1) a custom logic chip; or (2) a

generic memory and general-purpose CPU, plus specific software (e.g., the FFT algorithm) which imparts the claimed functionality to the processor.

Conversely, a non-specific product claim encompasses any and every computer implementation of a process when read in light of the specification. The Guidelines state that such claims have two characteristics:

(a) they define the physical characteristics of a computer or computer component exclusively as functions or steps to be performed on or by a computer, and

(b) they encompass any and every product in the stated class (e.g., a computer or computer-readable memory) configured in any manner to perform that process.

*263 For example, the patent specification might describe a particular algorithm (process) to be executed on a general purpose digital computer, without any disclosure of program code or logic. The specification will often also state that it is a matter of "routine skill in the art" to select an appropriate conventional computer system and write a program to implement the claimed invention. If the claim is in product (rather than process) form it will be found to be a non-specific machine or manufacture.

Non-specific product claims will be examined on the basis of the underlying process, i.e., the patentability of non-specific product claims depends on whether the underlying process is statutory.

(c) Process Claims

The foregoing shows that, for purposes of analysis, the term "process claim" should be understood to include both bona fide processes and non-specific products. To be statutory, a computer-related process must either:

(a) belong to one of two safe harbor categories involving some physical activity or transformation outside the computer for which a practical technological application is (explicitly or implicitly) disclosed in the specification, or

(b) be limited by the language of the claim to a practical technological application.

The latter possibility is the key to patenting computer-related processes that might otherwise have been unpatentable prior to the 1995 Federal Circuit cases and the Guidelines.

The Guidelines define two "safe harbors," for process claims involving physical transformations, that constitute statutory subject matter per se: (i) post-computer physical *264 process activity, and (ii) pre-computer physical process activity. Safe harbor post-computer physical process activity claims require:

. performing physical acts outside the computer,

. independently of and after any steps to be performed by the computer, and

. such that the physical acts involve manipulating a tangible physical object to change its physical attribute or structure.

Conversely, safe harbor pre-computer physical process activity claims require:

. transforming measurements or characteristics of physical objects or activities external to the computer into electrical signals or data, and

. processing those signals or data within the computer.

A claim that fails to meet either of the above safe harbors can nevertheless qualify as statutory subject matter by explicitly reciting a practical technological application (in addition to disclosing the practical application in the specification). Thus, a claimed process that merely manipulates an abstract idea or performs a purely mathematical algorithm is non-statutory. However, the claimed process would be statutory if explicitly

limited to a practical technological application of the abstract idea or mathematical algorithm. Note that claim language merely specifying an intended use or field of use for the invention will generally not limit the scope of a claim, particularly when only stated in the claim preamble. Examples of other claim language that has been held insufficient to limit the claim to a practical application include: [FN26]

*265 . "collecting" or "selecting" data for use in a process consisting of one or more mathematical operations;

. magnetically recording the result of a calculation;

. displaying the result of a calculation as a shade of gray rather than as simply a number where the data were numerical values that did not represent anything; and

. transmitting electrical signals representing the result of calculations.

Conversely, if a claim requires that the direct result of a mathematical operation be evaluated and transformed into something else, the claim is not a non-statutory mathematical algorithm. For example, converting numbers representing values of a wavefunction for a chemical compound into values representing an image that conveys information about the three-dimensional structure of the compound and the displaying of the three-dimensional structure is a statutory practical application limitation.

2. Obviousness Determination

In addition to the statutory subject matter determination, descriptive material also plays a role in obviousness determinations. Under the Guidelines, if the difference between the prior art and the claimed invention is limited to descriptive material stored on an article of manufacture or employed by a machine, the PTO will determine whether the descriptive material is functional or non-functional. Functional descriptive material will preclude an obviousness rejection unless the functional descriptive material would have been suggested by the prior art. On the other hand, non-functional descriptive material cannot render non-obvious (and thus unpatentable) an *266 invention that would have been obvious without the presence of the non-functional descriptive material.

3. Summary of the New Guidelines

As summarized in Figure 2, an initial determination is made as to whether a claim is directed to clearly non-statutory subject matter. Functional descriptive material per se and non-functional descriptive material -- whether claimed per se or as recorded on a computer-readable medium -- are both non- statutory subject matter. If the claim survives this test, it is further scrutinized as follows. Claims are classified into product or process types, and product claims are subdivided into specific and non-specific types. A specific product claim will always be statutory subject matter. A non-specific product claim will be further evaluated on the basis of the underlying process. Process claims (and non-specific product claims) may qualify as statutory subject matter under two safe harbors (pre- or post-computer processing connection to external physical elements or activity) or, alternatively, the practical technological application must be explicitly claimed. The safe harbors represent an attempt by the PTO to preserve the possibility of qualifying as statutory subject matter under the pre 1995 cases (e.g., the "Freeman-Walter-Abele Test") while the alternate "technological application" test represents the PTO's interpretation of the 1995 Federal Circuit cases.

*267 C. Application of the PTO Guidelines to Executable Content

The first question is whether a claim to executable content is non-statutory because it is non-functional descriptive material. Consider an article of manufacture claim to executable content (e.g., a Java Applet in the form of Java Bytecode) on a computer-readable medium. As stated previously, such executable content can be heavily Content-based, highly Functional, or any combination thereof. The overall claim should not constitute non-functional descriptive material because, as an object, the Applet contains not only the data structures holding the Content or data on which the Functionality acts, but also the methods [FN27] used to manipulate the data structure. The methods therefore arguably provide the structural and functional interrelationships between the data structure and the computing processes that will be performed on the Applet when the computer-readable medium is accessed. Similarly, a machine claim for executing the Java Applet should not be functional descriptive material per se because the methods provide the structural and functional interrelationships between the data structure and the computing processes performed. Therefore, a product claim to executable content should survive the Guidelines' descriptive material examination.

The next step in the analysis is whether the product as claimed is specific or non-specific. The product is non-specific if the claim would read on any product "configured in any manner" or "each and every computer implementation of an underlying process." This, in turn, depends on the amount of detail included in the claim. A broadly written claim without details of the data *268 structures' architecture or the methods for accessing the data structures could satisfy these criteria and thus be non-specific. Conversely, a claim including details of the data structures and the methods for accessing the data structures should be specific, and thus, statutory subject matter. However, it could be argued that even a narrow claim containing such details might be construed as non-specific because of the architectural neutrality of Java Bytecode and because any computer capable of running Java Bytecode is, at a certain level of abstraction, an architecture-neutral Java Virtual Machine. This, of course, requires that the phrase "each and every computer implementation" be limited to a Java Virtual Machine -- an unreasonable interpretation. Assuming, arguendo, that such an interpretation is correct, the patentability of any but the narrowest product claim would then depend on the patentability of the underlying process.

Thus, consider a process claim to receiving a Java Applet and running it on a Java Virtual Machine by interpreting the specific methods contained in the Applet. For the same reasons given above, the presence of the methods in the Applet would avoid the descriptive material bar. Assuming the Applet is for purely executable content, i.e., there is no pre- or post-computer physical process activity, the process will not come within either of the safe harbors set forth in the Guidelines. However, the claim will still be statutory under the alternative test as long as some practical technological application is claimed, e.g., conveying airfare information for travelers to use in making reservations, educating children by presenting geography information, *269 etc. Basically, the practical technological application requirement forces the applicant to narrow the scope of the claim so that not every possible application of the Applet is foreclosed. [FN28]

The foregoing examples suggest that, because of the Functionality provided by the methods of an Applet, executable content should constitute statutory subject matter under the Guidelines. [FN29] As discussed above, executable content covers a spectrum ranging from highly Functional (e.g., electronically distributed business tools and financial management systems) to highly Content oriented (e.g., passive data plus downloaded application

programs needed for data display). Especially at the latter extreme, where executable content turns what previously would have been purely passive data into a functional application, the convergence of technology and the new patentability rules suggest a blurring of the traditional distinctions between copyrights (for Content) and patents (for Functionality). To more fully understand this trend, it is instructive to review briefly the recent history of copyright and patent protection for computer software products.

*270 D. The Respective Roles of Copyright and Patent Protection for Software Products

1. The Traditional View

Historically, patent protection has been available for the functional aspects of a computer program, while copyright has been available for its expressive elements. The functional aspects are generally embodied in the program's "internals," e.g., architecture and logic, while the non-functional aspects are generally embodied in the program's "externals," e.g., output or user interface. In the 1980's, courts gave broad copyright protection to both internal, non-code aspects of program design, e.g., "structure, sequence and organization," [FN30] and external visual aspects of program design, e.g., "look and feel." At the same time, courts gave limited patent protection to the functional aspects of software-based inventions provided the claim did not "wholly preempt a mathematical algorithm," constitute a "method of doing business" or comprise "printed matter."

2. The Modern View

In the first half of the 1990s, the courts significantly reduced the scope of copyright protection for software, while enlarging the scope of patent protection. The first step in the narrowing of copyright protection was Altai, [FN31] where the Second Circuit Court of Appeals refused to find copyright infringement based on the use of "externally constrained" elements, *271 conventional programming practice or standard techniques, (the computer version of scenes-a-faire) public domain material, etc. The Altai court adopted a three-part "abstraction-filtration-comparison" test to determine copyright infringement: (1) construct levels of abstractions ranging from detailed expression to abstract idea; (2) within each level of abstraction, perform the filtration as described above; and (3) within each level, compare the remaining elements of the copyrighted and allegedly infringing work. If the two works are similar at the expressive levels, there is infringement. Conversely, if the two works are similar only at the abstract, but not the expressive, levels there is no infringement. The filtration step greatly reduced the scope of copyright protection for functional elements in computer software, and set the stage for two later copyright cases that further reduced the scope of copyright protection for functional elements of computer software: Apple v. Microsoft/Hewlett-Packard [FN32] and Lotus v. Borland. [FN33]

In Apple, the Ninth Circuit Court of Appeals analyzed the copyrightability of the "look and feel" of the Macintosh graphical user interface (GUI) by rejecting a claim that it was protectable as a whole, and dissecting it into a large number of graphical sub-elements for analysis on an element-by-element basis. The court then found that almost all of the elements were unprotectable under criteria similar to those of Altai (commonly used, functionally dictated, not original, etc.) In so doing, the court significantly narrowed the scope of protection for the GUI, particularly its

functional aspects comprised of the sum of its largely unprotectable elements.

*272 In Lotus, the First Circuit considered the functional interface to the Lotus 1-2-3 spreadsheet as embodied in its command hierarchy. The court (later affirmed 4-4 by the Supreme Court) held that the command hierarchy was unprotectable functional subject matter in its entirety because, like the buttons on a VCR, it was the "only way" to operate a spreadsheet. While all spreadsheets must have a command hierarchy, the particular expressions thereof can differ greatly. In ignoring this fact, the court seems to suggest that where a particular software display is primarily functional, no amount of expressivity will render it copyrightable.

The line of cases starting with Altai and ending with Lotus shows that the courts have considerably narrowed the scope of copyright protection for functional aspects of computer software. At the same time, as discussed above, a series of 1995 Federal Circuit cases expanded the scope of patent protection for computer software. As discussed in the previously mentioned Computer Lawyer [FN21] article, the views of the various judges on the Federal Circuit are far from uniform, with the court dividing into roughly three factions. However, the general trend toward increased software patentability is well established. Furthermore, the views of the two most prosoftware of the three factions appears to be reflected in the new Guidelines, and the Federal Circuit sitting en banc has recently suggested that the Guidelines may be persuasive authority. [FN34]

Unfortunately, achieving uniformity in judicial application of the Guidelines may be a difficult process, as district courts struggle to understand the Guidelines. The first case to interpret the Guidelines, State Street, [FN35] illustrates that the road may be a bit rocky.

*273 The statutory subject matter issue in State Street arose on a motion for summary judgment of invalidity of a patent on a computerized accounting system for a complex hub-and-spoke mutual fund partnership. In particular, independent mutual funds (the spokes) would pool their funds in an investment portfolio (the hub) organized as a partnership. The system assessed gains and losses of the spoke funds on a pro rata basis, including accounting for individual investors in the spoke funds adding or withdrawing assets on a periodic basis. The claim in question, the only independent claim, is reproduced below:

> 1. A data processing system for managing a financial services configuration of a portfolio established as a partnership, each partner being one of a plurality of funds, comprising:
>
> (a) computer processor means for processing data;
>
> (b) storage means for storing data on a storage medium;
>
> (c) first means for initializing the storage medium;
>
> (d) second means for processing data regarding assets in the portfolio and each of the funds from a previous day and data regarding increases or decreases in each of the funds, assets and for allocating the percentage share that each fund holds in the portfolio;
>
> (e) third means for processing data regarding daily incremental income, expenses, and net realized gain or loss for the portfolio and for allocating such data among each fund;
>
> (f) fourth means for processing data regarding daily net unrealized gain or loss for the portfolio and for allocating such data among each fund; and
>
> (g) fifth means for processing data regarding aggregate year-end income, expenses, and capital gain or loss for the portfolio and each of the funds.

According to the State Street opinion,

*274 the specification discloses the specific structure, circuitry or other devices by which the invention operates as follows: The portfolio/fund accountant makes use of a personal computer 44 programmed with software 50. One example of software 50 is the "HandS" (a service mark of Signature Financial Group, Inc.) computer program. The personal computer 44 used by portfolio/fund accountant is capable of producing printed output 46 and storing data on data disk 52, which preferably is a floppy disk, although other types of storage media may be used. State Street at 4.

The district court held that, under the Guidelines, computer software performing mathematical operations on data, without any physical transformation of such data, was non-statutory subject matter. More particularly, the court first found the claim to be a non-specific product claim whose patentability depended on the underlying process. The court then found the underlying process constituted non-statutory subject matter because it failed to perform a physical transformation. As discussed in more detail below, it is submitted that both findings represent a flawed application of the Guidelines.

With regard to the first finding, the claim was written in means-plus-function ("MPF") form. Under 35 USC 112, the scope of MPF claim elements is limited to the structure disclosed in the specification "and equivalents." Assuming that the specification discloses at least a minimal amount of structure (hardware and/or software) most MPF claims will not be sufficiently broad to encompass each and every computer implementation of the functionality set forth in the claim, as required for a finding of non-specificity. Thus, the court should have performed a detailed examination of the limits placed on the claim by the technology disclosed in the specification and equivalents thereof. Instead, the court's claim construction was superficial, consisting only of a verbatim recitation of a single paragraph from the specification describing operation of the accounting system on a personal computer in very general terms. However, *275 even this paragraph clearly indicates the presence of hardware (elements (a) - (c) of the above claim) and specific software (elements (d) - (f) of the claim in view of the specific disclosure of the "HandS" computer program) sufficient to satisfy the Guidelines' requirement for a specific product claim.

Having found product nonspecificity, the court then turned to the underlying process. As discussed previously, the Guidelines explicitly provide two distinct alternatives by which a process claim can be found to be statutory: (1) falling within one of the two safe harbors involving pre- or post-computer physical process activity, or (2) explicitly claiming a practical technological application. The latter is especially crucial for processes that otherwise would consist of mathematical operations or business methods performed on a computer. Unfortunately, the court confused the safe harbor and technological utility alternatives and combined them into a single requirement:

> ... a process will receive statutory protection if it is limited to a practical application of the abstract idea or mathematical algorithm in the technological arts (i.e., involve some species of physical transformation of input data). State Street at 22.

The parenthetical portion of the above sentence finds support in the Guidelines as one of the physical activity safe harbors guaranteeing statutory subject matter. The first part of the sentence, ending just before the parenthesis, embodies the Guidelines' savings clause for processes that fail to meet the physical activity safe harbors. Thus, the court fundamentally *276 misunderstood the Guidelines' alternative tests for patentability of process (and nonspecific product) claims. [FN36]

In spite of such missteps as State Street, the authors expect that most practitioners and courts will soon come to understand and correctly apply the Guidelines. It is further anticipated that the legal/technological/business convergence of (1) the Guidelines, (2) the proliferation of executable content and (3) the productization of the Internet market, will facilitate significant patent protection for technology that would traditionally have fallen under the aegis of copyright.

Finally, the authors also expect that widespread use of the Guidelines' broad "claimed practical application within the technological arts" savings clause for process (and nonspecific product) claims will bring an end to most Section 101 rejections of software-related claims on statutory subject matter grounds. Rather, the authors expect that, apart from the usual questions of novelty and nonobviousness over the prior art, in the future the primary inquiry concerning software-based claims will relate to the scope of the claim under 35 U.S.C. 112. The Section 112 inquiry is particularly important in light of: (1) the Guidelines' position that a specific product claim (e.g., one that does not encompass each and every computer-implementation of the invention) will constitute statutory subject matter without further scrutiny; and (2) the Supreme *277 Court's recent pronouncement on claims construction in Markman [FN37] and its anticipated pronouncement on the doctrine of equivalents in Hilton-Davis. [FN38]

The Hilton-Davis case provides an example of the introduction of copyright concepts into patent infringement analysis involving the doctrine of equivalents, which is used to find infringement in the absence of literal infringement. The Federal Circuit's per curiam en banc opinion clearly held that substantial similarity of function way and result does not represent the only route to a finding of equivalents. Rather, the essence of equivalents is "insubstantial differences" between the claimed subject matter and the accused product or process. Similarity of function, way and result is one way of demonstrating that the differences are insubstantial. However, a court must consider all evidence relevant to the substantiality of the differences and, when appropriate, evidence other than similarities in function, way and result may be relied upon to determine equivalence. The court then introduced the notion that copying of the patent invention during development of the allegedly infringing product could provide an alternative basis for a finding of equivalence:

> When an attempt to copy occurs, the fact finder may infer that the copyist, presumably one of some skill in the art, has made a fair copy with only insubstantial changes. (Hilton-Davis at 1519)

*278 The trier of fact is therefore allowed to assume, in the absence of countervailing evidence, that the accused infringer's attempts to copy were successful.

On the other hand, where the accused infringer presents credible evidence of attempts to "design around" the patent claims in order to avoid infringement liability, the trier of fact is similarly entitled to conclude that it was successful in doing so:

> When a competitor becomes aware of a patent, and attempts to design around its claims, the fact-finder may infer that the competitor, presumably one of skill in the art, has designed substantial changes into the new product to avoid infringement. (Hilton-Davis at 1520)

Thus, evidence of attempted copying presented by the patent owner, supports an inference of equivalence, i.e., insubstantial differences, and thus, infringement, while evidence of attempted designing around, presented by the accused infringer, supports an inference of non-equivalence, i.e., substantial differences, and thus, noninfringement.

Hilton Davis represents an interesting importation of copyright principles into patent law. A plaintiff may seek to establish equivalence by

demonstrating that the accused infringer "copied" the patented invention. An inference of copying may be drawn from evidence of access (knowledge of the patented invention) plus substantial similarity -- just as in a copyright case. The accused infringer, on the other hand may seek to rebut the copying evidence, and the resulting inference of equivalence, with evidence of independent development -- just as in a copyright case. Or, an accused infringer may seek to establish nonequivalence by presenting evidence of good faith -- and reasonable -- attempts to avoid infringement by "designing around." This suggests an analogy to the "clean-room" design methodology used by developers of compatible software products to establish lack of copyright infringement by demonstrating good faith attempts to avoid access to the copyrighted expression (e.g., source code) in the product with which the accused *279 infringer wishes to be compatible. In the case of the clean-room defense, evidence of good faith and reasonable efforts to avoid infringement can be persuasive, though not by any means determinative, that the attempt was successful.

The Guidelines provide another example of the merger of patent and copyright principles. As previously discussed, claims directed to functional descriptive material such as a computer program or a data structure are statutory subject matter if the claim sets forth a functional interrelationship between the program or data structure and the other (hardware) elements of the claimed combination. According to the Guidelines, the requirement of a functional interrelationship is satisfied if the program or data structure is claimed as residing in, or on, a "computer readable memory."

Computer-readable memories include not only electronic or magnetic storage media such as Read-Only Memories, floppy disks, and the like, but also various forms of optically readable media such as CD-ROMs. By logical extension, a computer-readable memory should also include a piece of paper on which a computer program, in either source or object form, is printed, or even handwritten. This is because: (1) the image on the paper can be scanned and the writing digitized; and (2) the digital image signals can be processed by OCR or handwriting recognition software to generate program instructions in binary (machine-readable) form which can be used to direct a computer to function in a particular manner and thereby impart the functionality represented by the program or data structure to the computer. The only real difference between a floppy disk on which a program has been magnetically recorded and a piece of paper on which a program has been printed or written relates to the apparatus used for transforming the program into a form that can be directly used by the computer.

*280 Thus, a generic memory-plus-program article of manufacture claim would cover the writing-on-paper embodiment. (Alternatively, a narrower claim could be written specifically covering this embodiment.) The unauthorized "making" of the article, i.e., the paper bearing the written program code, would infringe the article claim, i.e., reproduction (copying) of the writing would constitute patent infringement. Clearly, the very same act could also constitute copyright infringement. In such a case, one may ask, what is the difference between the scope of the respective rights afforded by the patent and the copyright? To be sure, in some situations the patent right would be broader as one might expect. Thus, if the accused program were independently developed, i.e., without access to the protected program, it would infringe the patent but not the copyright. Similarly, if an accused infringer examined or reverse engineered the protected program and created a "new" program which incorporated the patented functionality but not the copyrighted expression therein, the new program would infringe the patent but not the copyright. On the other hand, where the protected program is copied, the patent and copyright rights seem to converge to the point of congruency. One might

question whether this is an "appropriate" result in light of the different policy underpinnings of the patent and copyright laws.

What about Section 117 of the Copyright Act? Does the owner of a copy of a patented and copyrighted computer program have the right, under the patent law, to make a back-up copy? Ordinarily, neither the patent nor the copyright version of the first sale doctrine permits the owner of a purchased product to "make" (i.e., replicate) the product. In 1980, 17 USC 117 was amended to create a specific exception for computer programs. Is a similar amendment to Title 35 needed?

*281 It would seem that the only basis on which to deny statutory subject matter status to a program or data structure written on paper is the so-called "printed matter" doctrine. However, the Federal Circuit cast doubt on the continued viability of the doctrine, holding that extension of the printed matter rule to a new field, namely information stored in a computer memory, was unwise in view of the "questionable legal and logical footing" on which the rule stands. [FN39] In any case, the program-on-paper hypothetical meets the criteria set forth in the Guidelines for a statutory article.

The above examples illustrate that the modern trend is toward merging copyright and patent concepts and enlarging the scope of patent protection for software-based inventions. It is submitted that application of the Guidelines to the new executable content technologies which will be proliferated with the coming wave of electronic commerce is likely to result in a further blurring of the relationship between patent and copyright for internet-based information products.

FNa1. Chair, Computers and Software Industry Group

FNa2. Ph.D., Technology Consultant McCutchen Doyle Brown & Enersen, LLP San Francisco & Palo Alto

FN1. J. Neil Weintraut, "Presentation on the Internet to the Participants of the RSA Data Security Conference," January 19, 1996, p. 4.
FN2. Geoffrey Moore, Crossing the Chasm, Harper Business, 1991.
FN3. James C. Collins & William C. Lazier, Beyond Entrepreneurship, Prentice Hall, 1992, p. 102.
FN4. Weintraut, ibid.
FN5. Ibid.
FN6. J. Neil Weintraut & Daniel H. Rimer, "The Internet -- Webbing the Digital Economy," Presented at the 1996 RSA Data Security Conference, January 19, 1996, p. 3.
FN7. Ibid.
FN8. Weintraut & Rimer at 20.
FN9. Ibid.
FN10. Accessed, sent, received, transmitted, or otherwise manipulated
FN11. The Internet is not the same as the World Wide Web: the Internet is a particular (although large and somewhat ill-defined) network, while the Web is a collection of applications that happen to be running primarily on the Internet. Thus, Web applications can also be deployed on other networks besides the Internet (e.g., stand-alone LANs or WANs sometimes called Intranets). Java, as a Web programming language, can therefore also be deployed in non-Internet contexts. For example, Mitsubishi has been working to use Java technology in embedded systems such as consumer electronics, handheld devices, telephones and VCRs. John December, Presenting Java, Sams.net Publishing, 1995, p. 8. However, the differences between the Internet and the World Wide Web are not important for this discussion and, as a matter of convenience, the terms will be used interchangeably throughout.
FN12. For more information, see Ronald S. Laurie, "Commercial Cryptography: Mapping the Patent Minefield," Electronic Cash Conference, International

Business Communications, San Francisco, Nov. 16-17, 1995 (also available at http://www.mccutchen.com).
FN13. The US government classifies cryptographic technology as a munition subject to export control under the Arms Control Export Act ("ACEA") or the International Traffic in Arms Regulations ("ITAR"). As the Web recognizes no national boundaries, it has been said that interoperability considerations may indirectly coerce manufacturers of cryptographic technology to adopt the same restrictions for domestic products. Even a purely domestic market is not safe from government restrictions; under current law, cryptographic software running on the laptop computer of an US citizen temporarily traveling overseas is subject to the export regulations. Thus, many observers view the government's restrictions as an impediment to reaching the full potential of state-of-the- art cryptographic technology. However, the tide may be turning. The Ninth Circuit has recently held that computer programs are a form of speech and, therefore, that the government's prohibitions on exporting cryptographic software "come with a heavy presumption against their validity when they act as a prior restraint on speech." Daniel J. Bernstein v. United States Department of State, et al., No. C-95-0582 MHP (N.D. California), April 15, 1996 (Denying defendants' motion to dismiss because plaintiff has colorable and, therefore, justiciable constitutional claim that export regulations violate his First Amendment free speech rights).
FN14. Religious Technology Center v. Netcom On-Line Communication Services, Inc., 907 F. Supp. 1361 (N.D. Cal.), November 21, 1995 (services provider not liable for infringing material posted to its network without having knowledge of the infringement or having an opportunity to address it).
FN15. The Comp Examiner Agency Inc. v. Juris Inc., No. 96-0213 (C.D. California), April 24, 1996 (Defendant ordered to cease using domain name "juris.com" where "juris" is competitor's registered trademark).
FN16. For example, Protocol Handlers for communicating to a Java-enabled browser how to handle a new method of processing information or Content Handlers to give Java-enabled browsers the capability to interpret new data formats. See December at 77.
FN17. The terms "methods" and "processes" are often used interchangeably in a patent claim sense. For reasons of clarity, "methods" shall be used herein exclusively in reference to its object-oriented programming meaning, with "processes" reserved for the patent context.
FN18. "http://www.eolas.com/overview.htm".
FN19. either alone or in conjunction with such Java extensions as EOlas' Weblets or NeXt's WebObjects.
FN20. 35 USC 101 provides: "Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title."
FN21. 61 Fed. Reg. 7478 (also available at the PTO website, http://www.uspto.gov).
FN22. Ronald S. Laurie & Joseph K. Siino, "A Bridge Over Troubled Waters? Software Patentability and the PTO's Proposed Guidelines", The Computer Lawyer, vol. 12, nos. 9 & 10, Sept. & Oct. 1995 (also available as "The Future of Software Patents," at http://www.mccutchen.com/ip/fut/fut_swp 1.htm).
FN23. Although descriptive material per se is ineligible for patent protection, copyright protection is often appropriate. See Ronald S. Laurie, "A Comparison of Patent and Copyright Protection for Computer Software Under U.S. Law," 36th AIPPI World Congress, Montreal, June 25, 1995 (also available at http://www.mccutchen.com/ip/ip_2102.htm).
FN24. Guidelines.
FN25. Ibid.

FN26. Guidelines.
FN27. In an object-oriented programming sense.
FN28. The claiming of practical technological application may, in some cases, suggest going so far as to include some non-functional Content of the data structure in the claim. This raises the question of what effect such claiming would have on the ability of the claim to read on allegedly infringing devices or to be prior art against future patent applications on similar technologies.
FN29. The patentability of such claims having been established under the Guidelines, an interesting issue arises as to how they should be claimed. Consider an extreme case involving both remotely accessed content and remotely distributed processing. Such a claim, particularly when drawn to a system, will clearly meet the Guidelines' tests for statutory subject matter. However, the system claim may be difficult to enforce due to difficulties associated with finding a single party infringer. In contrast, a claim drafted to require only a single party infringer might present deficiencies with respect to the requisite functionality under the new Guidelines.
FN30. See Jaslow Dental Lab v. Whelan Associates, 797 F.2d 1222 (3d Cir. 1986); cert. denied, 107 S.Ct. 877 (1987).
FN31. Computer Associates, Intl. v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992).
FN32. 35 F.3d 1435 (9th Cir. 1994); cert. denied, 115 S.Ct. 1176 (1995).
FN33. Lotus Development Corp. v. Borland Intl., Inc., 49 F.3d 807 (1st Cir. 1995); affd. 4-4 116 S.Ct. 804 (1996).
FN34. In re Trovato, 60 F.3d 807 (Fed. Cir. 1995).
FN35. State Street Bank and Trust Company v. Signature Financial Group, Inc., No. 94-11344-PBS (D.Mass), March 26, 1996.
FN36. In addition to rejecting the claim for failing to recite a physical transformation, the court also rejected the claim as an unpatentable "method of doing business" -- an old and disfavored doctrine today. Indeed, the Guidelines explicitly state that claims should not be rejected under the method of doing business doctrine. In reaching this conclusion the court cited but ignored (with a cavalier "But see..." reference) the most relevant opinion in this area, Paine Webber v. Merrill Lynch, 564 F.Supp. 1358, 1369 (D. Del. 1983). The State Street court thus made a third fundamental error in applying the Guidelines.
FN37. Herbert Markman and Positek, Inc. v. Westview Instruments, Inc. and Althon Enterprises, Inc., 52 F.3d 967 (Fed. Cir. 1995) (claims construction is a question of law to be determined exclusively by the court); affirmed, 116 S.Ct. 1384 (1996).
FN38. Hilton Davis Chemical Co. v. Warner-Jenkinson Co., Inc., 62 F.3d 1512 (Fed. Cir. 1995); cert. granted, 116 S.Ct. 1014 (1996).
FN39. In re Lowry, 32 U.S.P.Q.2d. 1031 (Fed. Cir. 1994).

*283 Exhibit A