

EXHIBIT D

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Reexamination of)	MAIL STOP: <i>Ex Parte Reexamination</i>
)	
U.S. Patent No.: 6,546,552)	Group Art Unit: 3992
)	
Sharon Peleg)	Examiner: Andrew L. Nalven
)	
Issued: April 8, 2003)	Confirmation No.: 4316
)	
Reexam Control No.: 90/009,670)	
)	
For: DIFFERENCE EXTRACTION BETWEEN)	
TWO VERSIONS OF DATA-TABLES)	
CONTAINING INTRA-REFERENCES)	

SUPPLEMENTAL RESPONSE

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to a telephone call from the Reexamination Unit at the U.S. Patent and Trademark Office, the Patent Owner ("Owner") files a Supplemental Response relating to the Response filed in the U.S. Patent and Trademark Office on June 21, 2010, in the above-identified reexamination proceeding. Specifically, after filing the initial Response, three oversights were noted in a Letter Regarding Response filed on June 22, 2010. However, the U.S. Patent and Trademark Office Reexamination Unit has requested that this Supplemental Response be filed incorporating the changes mentioned in the Letter Regarding Response. Other than the changes noted in the Letter Regarding Response, no other substantive changes have been made to the Response of June 21, 2010, reproduced below.

This document is in response to the Office Action dated May 28, 2010.
Reconsideration is respectfully requested.

The Owner thanks the Examiner for courtesies extended to the undersigned during the telephone call of June 17, 2010. It is the undersigned's understanding from this telephone call that if the Examiner is not fully persuaded by this paper to confirm all the remaining rejected claims of the above-captioned patent, he will contact the undersigned within a few weeks of this submission to arrange an interview. On this basis, the Owner will forego the interview previously scheduled for June 25, 2010.

The Owner also thanks the Examiner for the indication that claims 5-7, 12-13, 18-20, 25-26, 29-34, 39-41, 46-47, 52-54, 59-60 and 63-68 (the "Confirmed Claims") are patentable and/or confirmed. The Owner submits the following remarks and arguments regarding the rejections of the remaining claims.

TABLE OF CONTENTS

1. REMARKS/ARGUMENT	3
A. The Pending Rejections.....	3
B. The Wetmore Reference	3
C. The '552 Patent.....	7
D. Responses to Rejections.....	7
 i. The Invariant Reference Claims	7
(1) The Inputs to Wetmore Are Not Executable.....	8
(2) No "References" In Wetmore Are Reflected As Invariant.....	10
(3) Wetmore Fails To Disclose Reflecting "Substantially Each" Reference Invariant.....	12
(4) Wetmore Fails To Disclose Generation of A "Compact Difference Result"	13
 ii. The Distinct Label Mark Claims	
(1) The Inputs to Wetmore Are Not Executable.....	15
(2) No "References" In Wetmore Are Replaced By a Distinct Label Mark...	15
(3) Wetmore Fails To Disclose "For Substantially Each Reference Entry . . . Replacing The Reference . . . By a Distinct Label Mark"	16
(4) Wetmore Fails To Disclose Generation of A "Compact Difference Result"	17
 iii. The Rejections Under 35 U.S.C. § 103.....	17
E. Comments on Statement of Reasons For Patentability And/Or Confirmation.....	19
F. Conclusion	19

1. REMARKS/ARGUMENT

A. The Pending Rejections

Claims 1-4, 8-11, 14-17, 21-24, 27, 28, 35-38, 42-45, 48-51, 55-58, 61 and 62 stand rejected (the "Rejected Claims") under 35 U.S.C. §102(b) as allegedly being anticipated by U.S. Patent No. 5,481,713 issued to Wetmore et al. (hereinafter "Wetmore"). This rejection is respectfully traversed.

As an initial matter, the Office Action is slightly ambiguous regarding the basis for the rejection of claims 2, 3, 9, 10, 15, 16, 22, 23, 36, 37, 43, 44, 49, 50, 56 and 57. On page 5 of the Office Action, those claims are listed as being rejected over Wetmore. However, the Office has not identified any disclosure of Wetmore that teaches the relevant limitations. Instead, it appears that the Office intended to reject those claims only under 35 U.S.C. § 103, for the reasons stated on page 18 of the Office Action. Accordingly, this response treats claims 2, 3, 9, 10, 15, 16, 22, 23, 36, 37, 43, 44, 49, 50, 56, and 57 as rejected only under § 103.

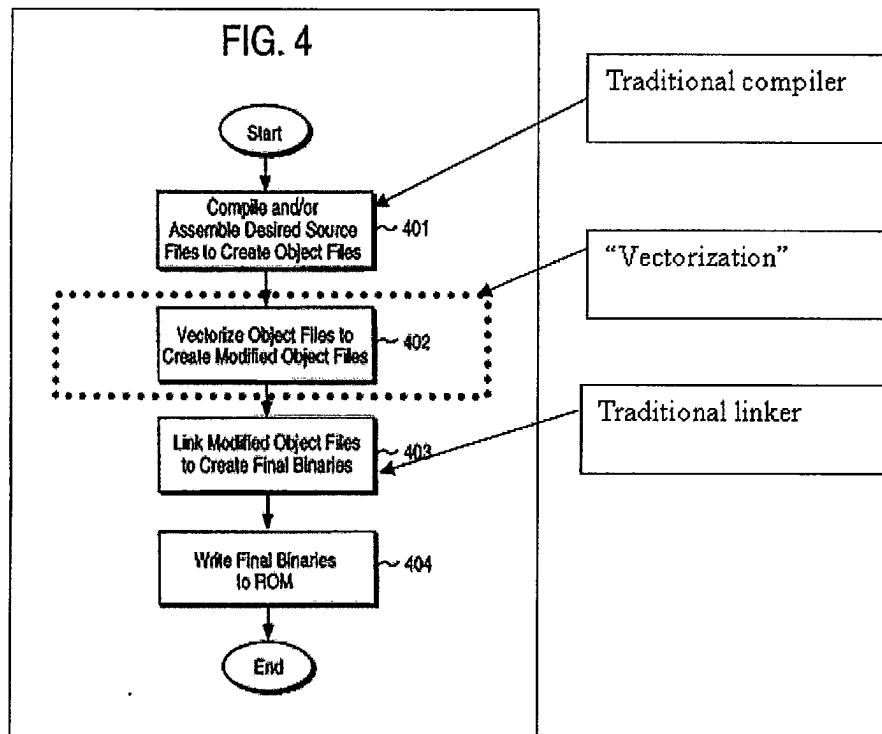
B. The Wetmore Reference

The Office has applied Wetmore either as an anticipatory or primary reference against all Rejected Claims. Wetmore, however, is directed to distribution of "Vectorized" software that is to be installed into a Read Only Memory ("ROM") at an end-user's computer, as was commonly done with the Operating System of early Macintosh computers sold by Apple -- Wetmore's assignee. (Wetmore 1:29-2:45). Vectorization of software is complex, and involves modifying the developer's software development tool-chain (e.g. compiler and linker) by adding a vectorization step in the middle of software development to create specialized binary executables for ROM:

... First, the source files are compiled ... to create object files, step 401. The object files are then vectorized to create vectorized object files, step 402. It is significant that only the object files are modified. ... Object files contain a series of defined records, each one containing specific items such as the

object code for a routine, the name of a routine, external reference from one routine to another, or comments. In object files the references to other routines have not been resolved. Therefore object files are an ideal place to alter the code without modifying the source code files. ... *The object files are then linked together to create the final binary values* which will be written to ROM, step 403. This is performed through a traditional linkage editing step. *Finally, after the object files have been "linked" together to create the final binaries, the ROM image is created, step 404.*

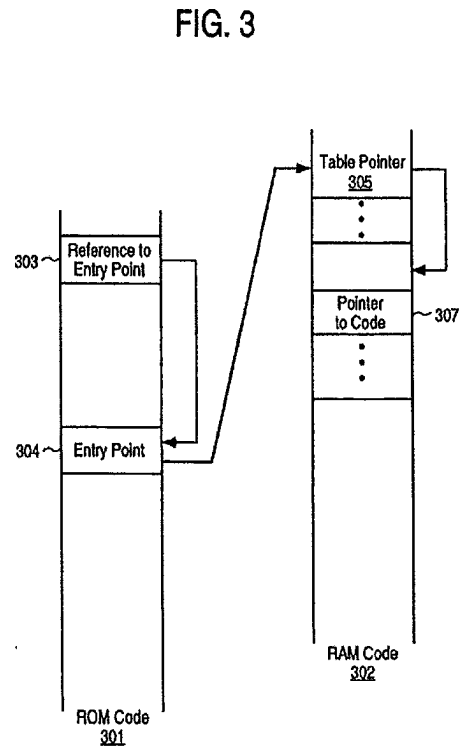
(Wetmore 6:47-67) (emphasis supplied). In other words, Wetmore requires introduction of a vectorization step (402) in the middle of the compilation chain, as shown in Wetmore's Fig. 4 (annotated below):



By vectorizing the software, the resulting binary file takes on different characteristics than it would have had in the absence of vectorizing. In particular, as described in Wetmore, "more entry points into the ROM are created, thus providing more locations at which the ROM may be accessed and code fixed." (Wetmore 6:8-11). Wetmore describes the "effect of vectorization" with reference to Figure 3:

"illustrated is ROM based code 301 and RAM code 302 . . . ROM code 301 will contain a reference to an entry point 303. The entry point may be a sub-routine, function, macro or a jump to a label somewhere else in the ROM code 301. . . . The reference to entry point 303 will effectively point to entry point 304. *Without vectorization, the executable code would be immediately following the entry point 304. With vectorization, the location of entry point 304 has been modified to be a reference to a table pointer 305 that resides in RAM.* The table pointer 305 is the vector in this example. The *table pointer 305 will point to vector table 306 which resides in RAM code 302* (specifically the system heap area).

As noted above, the reference to the table would in most cases include an offset into the vector table 306. Assuming the offset, *the entry 307 will contain a pointer to the location where the code to be executed would reside.* While the entry 307 may simply point back into the ROM, in the instance of a patch, the entry 307 may contain a pointer to an updated routine located somewhere in the RAM 302." (Wetmore 5:32-57) (emphasis supplied).



Thus, Wetmore describes vectorization as a process of modifying the compilation and linking process in order to introduce a level of indirection to the resulting binary file so that when entering an external routine such as a subroutine or function call (5:27-32), program execution does not flow sequentially -- but is first redirected to one of several vector tables in RAM which will contain the address of the next line of code that should be executed.

Also noteworthy in Wetmore, the vectorization process operates only on "entry points" (e.g., 304) which it describes as locations in code that may be entered via a symbolic

reference. (Wetmore 7:1-8). Accordingly, the "entry points" that are vectorized are names or labels in the object files. *Id.* Indeed, Wetmore operates only on object files, which it describes as the "ideal place to alter the code" because in "object files the references to other routines have not been resolved" (Wetmore 6:57-60), meaning that they have not yet been linked, are symbolic (rather than numeric references to memory addresses), and are not yet executable.

Using Wetmore's technique, the binary code that performs the subroutines or function calls can be replaced (or new subroutines or function calls can be added) in a subsequent version of the binary file by sending to the user in a system disk the new code for those subroutines or function calls. (Wetmore 10:24-53; 11:45-48). Significantly, therefore, Wetmore does not utilize a binary difference result algorithm, such as those prior art techniques discussed in the specification of the '552 Patent, to prepare a difference result. ('552 Patent at 1:50-2-12. See also Coppieters, K. "A Cross-Platform Binary Diff" of record in the '552 patent). Instead, Wetmore describes a far less granular comparison of the "object files of two versions of the vectorized ROM code to *identify routines* which are different or new" (Wetmore 11:1-4) (emphasis supplied), which new routines are then wholly transferred to the client. So, if even one line of code were changed in a routine, the entire routine would be identified for transfer to the client, where the new version of the routine would be loaded into RAM for subsequent execution in place of the obsolete version of the routine in ROM (Wetmore 2:4-5; 5:51-57; 10:35-53), and the vector tables in RAM would be modified to point to the updated location of the new version. (Wetmore 11:35-67).

In other words, Wetmore permits updates of software by transmitting entire routines to the client, which are executed instead of obsolete routines in ROM. Notably, this would result in the transmission of *large* difference results, because if even a single line of binary code were added to a routine, the *entire* routine would need to be included in Wetmore's difference result. This would potentially result in a larger difference result than even the prior art difference techniques, such as those described in Coppieters.

C. The '552 Patent

In marked contrast, the techniques of the '552 Patent are substantially different and are directed towards a substantially different problem than Wetmore. The '552 Patent is directed to a process for creating or applying a compact difference result as between two executable programs (claim 1-34) or two data tables (claims 35-68), involving the use of modifications performed on the old and new programs/data tables where the modifications include the use of distinct label marks (claims 1-7, 13-20, 26-27, 35-41, 47-54, 61) and/or invariant references (8-12, 21-25, 28-34, 42-46, 55-60, 62-68) for the purpose of distributing a *compact* difference result. The '552 Patent is not concerned with updating programs previously stored in a Read-Only Memory, and does not require changes to the developer's compilation tool-chain in order to function. Indeed, the techniques of the '552 Patent can be applied to an arbitrary executable program or data table, and it is not necessary to have access to the source code or pre-linked object files of the old or new programs/data tables being updated. Further, the difference result generated by the techniques of the '552 Patent is compact, *i.e.* smaller than traditional binary difference results, and unlike Wetmore, would not re-transmit an entire routine to the client when only a single instruction in that routine was changed.

D. Responses to Rejections

As discussed herein with respect to the claims under reexamination, these important differences between the disclosure of Wetmore and the '552 Patent show that Wetmore does not anticipate or render obvious any of the Rejected Claims.

i. The Invariant Reference Claims

The Office has rejected Claims 8, 11, 21, 24, and 28 (hereinafter the "Executable Program Invariant Reference Claims") and Claims 42, 45, 55, 58, and 62 (hereinafter the "Data Table Invariant Reference Claims") (collectively, the "Executable Program Invariant Reference Claims" and the "Data Table Invariant Reference Claims" are referred to herein

as the "Invariant Reference Claims") under 35 U.S.C. § 102(b) over Wetmore. The Owner respectfully traverses.

Exemplary claim 8 is set forth below:

8. A method for generating a compact difference result between an *old executable program and a new executable program*; each program including reference entries that contain *reference that refer to other entries in the program*; the method comprising the steps of:

(a) generating a modified old program *utilizing at least said old program*;

(b) generating a modified new program *utilizing at least said new program*,

said modified old program and modified new program have at least the following characteristics:

(i) substantially each reference in an entry in said old program that is different than corresponding entry in said new program due to delete/insert modifications that form part of the transition between said old program and new program are reflected as invariant references in the corresponding entries in said modified old and modified new programs;

(c) generating said compact difference result utilizing at least said modified new program and modified old program.

(1) The Inputs to Wetmore Are Not Executable

Regarding the Executable Program Invariant Reference Claims, Wetmore fails to disclose several claim elements. In particular, these claims require that old and new *executable* programs be input into the method (claims 8, 11) or system (21, 24) and that those old and new executable programs be used to generate a "modified old program" and a "modified new program" such that certain references (*i.e.* substantially each reference in an entry in one program that is different than a corresponding entry in the other program due to

insert/deletes that occurred in the transition between the two programs) are reflected as invariant references in the modified forms. (This, of course, does not mean that the modified programs need to themselves be in executable form, but only that the inputs or starting old and new programs that are to be modified, are executable. The preamble and claim text make this clear.)

Significantly, the claimed executable programs used as inputs to the difference result generation process are not disclosed in Wetmore. It appears that the Office has considered the "object files" of Wetmore to correspond to the claimed old and new *executable* programs. (Office Action at 7)¹. Owner respectfully disagrees. The object files of Wetmore that are vectorized are not executable. (Wetmore Fig. 4; 6:45-67). See *supra* at Section 1(B), discussing Fig. 4 and Wetmore's vectorization process. Instead, those files include unresolved, symbolic references. (Wetmore 6:53-60) ("In object files the references to other routines have not been resolved"). Such a file would not become an executable binary until *after* linking, which is the next step in the Wetmore process, after vectorization. (Wetmore 6:63-67) ("The object files are *then* linked together to create the final binary values which will be written to ROM") (emphasis supplied).² Note that the process of Wetmore *only* works when the unlinked object files (containing symbolic references) are accessible. Wetmore repeatedly makes this point clear:

- "It is significant that only the object files are modified" (Wetmore 6:52-53)
- "object files are an ideal place to alter the code" (Wetmore 6:59-60)
- "An entry point may be the *name* of a routine or a *label* in the file. Generally, an entry point is merely a location in the code which may be entered via a *symbolic reference*. It is these entry points which become the code access points which are vectorized" (Wetmore 7:4-8) (emphasis supplied)

¹ To the extent the old and new linked, vectorized ROM images are viewed as the old or new executable programs, respectively, Wetmore would not anticipate the claims because (under that reading) it fails to disclose generations of a modified old or modified new program utilizing those old (or new) executable programs.

² This is not meant to suggest that the claimed executable files must exclude *all* symbolic information. To the contrary, as is well-known in the art, executable files often contain header and other symbolic information to assist the loader and operating system with loading the program image into machine memory and commencing execution, as well as for other purposes. However, the Executable Program Invariant Reference Claims cover program files that are ready to be run.

Exemplary techniques of the Executable Program Invariant Reference Claims, by contrast, can be designed to work on input files that are already executable, and thus have the advantage of being able to be employed without access to source code or unlinked object files and, as discussed previously, can be employed without modifying the developer's development tool-chain (*i.e.* the compilation and linking process used by the developer to translate source code into an executable binary file).

Therefore, Wetmore does not disclose this element of the Executable Program Invariant Reference Claims, and, consequently, at least the Executable Program Invariant Reference Claims (Claims 8, 11, 21, 24, and 28) and the claims dependent therefrom (claims 9, 10, 22 and 23) are not anticipated by or rendered obvious over Wetmore.

(2) No "References" In Wetmore Are Reflected As Invariant

Regarding both the Executable Program Invariant Reference Claims (Claims 8, 11, 21, 24, and 28) and the Data Table Invariant Reference Claims (Claims 42, 45, 55, 58, and 62) -- because Wetmore only operates on object files, it also fails to disclose in its inputs or starting program any "references," as defined by the '552 Patent and as recited in this claim limitation:

substantially each *reference in an entry in said old program* that is different than corresponding entry in said new program due to delete/insert modifications that form part of the transition between said old program and new program are reflected as invariant references in the corresponding entries in said modified old and modified new programs... (emphasis added).

It appears that the Office treated Wetmore's "external references to other routines" in the object files as the claimed "references" of the '552 Patent. (Office Action at 7-8). The Owner respectfully disagrees with this characterization. Importantly, the Glossary of '552 Patent defines "references" as follows:

Reference -- a part of the data appearing in an entry in the data table which is used to refer to some other entry from the same data table. A

reference can be either an address or a number used to compute an address.

Address -- a *number* which is uniquely assigned to a single entry by which that entry is accessed...

('552 Patent 2:37-45)(emphasis supplied). Accordingly, to qualify as a "reference" in the Invariant Reference Claims, the reference must be either an "address" (which is further defined to be a "number") or a "number used to compute an address" (such as an offset). In either case, the reference must be a "number."³ Such references are not described as being vectorized in Wetmore.

The references to external routines in Wetmore, which the Office has identified as allegedly corresponding to the claimed references (Office Action at 7), are not numeric. Instead, they are purely symbolic. See *supra* at Section 1(D)(i)(1), (citing Wetmore 6:52-53; 6:59-60; 7:4-8)("An entry point may be the *name* of a routine or a *label* in the file. Generally, an entry point is merely a location in the code which may be entered via a *symbolic* reference. It is these entry points which become the code access points which are vectorized"). Wetmore describes creating numeric (as opposed to symbolic) references in the vector table object file, which is the *output of vectorization*. (Wetmore 8:21-52) (showing introduction of the entry: "jmp ([$\$0584$])" *after* vectorization took place). Accordingly, there are no "references" as claimed in the '552 Patent that are disclosed as being *input* to a vectorization process in Wetmore, because the numeric addresses that Wetmore describes are not present in the files input to Wetmore's vectorization process. Therefore, Wetmore fails to disclose that "substantially each reference in an entry in said old program ... are reflected as invariant."

This significant difference between Wetmore and the '552 Patent is a result of Wetmore's being limited to operating on object files, which is not a restriction on the claimed

³ This is not meant to suggest that "invariant references" must be numeric. To the contrary, the specification makes clear that invariant references may be numeric or symbolic label marks. ('552 Patent 10:47-60). It is only the references in the old and new executable programs or data tables that must be numeric.

techniques. Instead, the claimed techniques are intended to operate on files *after* references have been resolved to become numeric, as opposed to symbolic -- thereby permitting the techniques of the '552 Patent to be applied to executable files and data tables. In this manner, the techniques of the '552 Patent have utility in many contexts where Wetmore's techniques could not be used. Accordingly, for this reason alone, none of the Invariant Reference Claims (Claims 8, 11, 21, 24, 28, 42, 45, 55, 58 and 62, and therefore dependent claims 9, 10, 22, 23, 43, 44, 56 and 57) are anticipated by or rendered obvious over Wetmore.

**(3) Wetmore Fails To Disclose Reflecting
"Substantially Each" Reference Invariant**

Regarding both the Executable Program Invariant Reference Claims and the Data Table Invariant Reference Claims, Wetmore fails to disclose reflecting as invariant "substantially each reference" that has been altered as a result of delete/insert modifications between the old and new programs/data tables. Instead, Wetmore describes vectorizing *only* "entry points" to routines, rather than substantially all references that are likely to have changed due to delete/insert modifications. (Wetmore 7:1-9; *supra* at Section 1(D)(i)(2)). Assuming for the sake of argument that non-numeric labels in object files could qualify as "references," Wetmore still fails to teach, suggest or provide reason for a modification that would meet the claimed invention.

In particular, Wetmore does not teach, disclose or even suggest reflecting as invariant those references *within* routines that have changed due to delete/insert modifications (*i.e.* the non-"entry point" references). This missing disclosure is not surprising, because Wetmore's updates operate at the granularity of the routine itself. (Wetmore 5:22-31; 7:5-9) ("entry points for *external routines* are replaced"; "upon encountering a *reference to an external routine*, e.g. a subroutine or function call, the actual entry point will reference the vector table..."). As previously discussed, if any part of a routine is changed, Wetmore transmits the *entire routine* to the client as part of the patch. See *supra* at Section 1(D)(i)(4). Therefore, Wetmore does not bother making modifications

that would cause references *within* the changed routines to be reflected as invariant. This is a significant departure from the Invariant Reference Claims of the '552 Patent, which attempt to generate modified old and new programs such that "substantially each reference" that has been changed as a result of a delete/insert modification -- including the numeric references within routines that have changed due to delete/insert modifications -- are reflected as invariant. Accordingly, the Owner respectfully submits that at least the Invariant Reference Claims are not anticipated by or obvious over Wetmore.

**(4) Wetmore Fails To Disclose Generation of A
"Compact Difference Result"**

Regarding both the Executable Program Invariant Reference Claims and the Data Table Invariant Reference Claims (Claims 8, 11, 21, 24, 28, 42, 45, 55, 58 and 62, and therefore dependent claims 9, 10, 22, 23, 43, 44, 56 and 57), Wetmore fails to disclose generation of a "compact difference result." As the specification makes clear, the techniques of the '552 Patent are directed toward improving prior art "difference results" so as to generate a smaller (i.e. *compact*) difference result" compared to prior binary comparison tools, such as that described in the Coppieters reference. ('552 Patent, 3:30-46; 14:5-14). Those prior techniques looked for all "matches" between two binary files, and represented the matches as a series of references to portions of the old file where the matching portions were located. When there were portions in the new file that did not match a portion of the old file, the new portion would be included in the diff file itself. (See Coppieters at 35 (describing diff file format)). Accordingly, to minimize the size of the diff file, the prior art techniques focused on finding the largest number of matches between the old and new file as possible, because the failure to find a match meant that the size of the diff file would be increased as a result of having to include the non-matching code and/or data within the diff file.

Notably, to reduce the diff size, Coppieters recommended treating as a match "chunks" of "six or more bytes" that are found in both the old and new files. (Coppieters at 32). The result of this technique, therefore, would not include in the diff any chunks of six or

more bytes that were the same in both the new file and the old file, such as chunks of code and/or data that did not change between the two versions. Instead, as mentioned above, those chunks would be identified in the diff by specifying their location and size in the old file, so that the client could locate those chunks in the old file and make use of them when applying the "diff" to create the updated program. In addition, the diff file would include chunks that did not match (*i.e.* new code/data as well as matching chunks of less than six bytes).

These prior art techniques, however, were not well-suited for comparing program files or data tables containing references that are numbers. In particular, those prior techniques would fail to detect a match between portions of code in the old and new files when those portions included references that had changed between versions solely due to delete/insert modifications made to other portions of the file. As explained in the '552 Patent: "insertion of only one new entry may result in [a] plurality of altered reference entries which will naturally be reflected in the difference result and obviously will inflate its volume." ('552 Patent 2:6-9).

Wetmore, in contrast, generates difference results that would be significantly larger than those created by even the Coppieters-type binary difference techniques. In particular, Wetmore describes a far less granular comparison of the "object files of two versions of the vectorized ROM code *to identify routines* which are different or new." (Wetmore 11:1-4) (Emphasis supplied). *If any part of the routine does not match, the entire new routine is then transferred* to the client, where the new version would be loaded into RAM for subsequent execution in place of the obsolete version in ROM (Wetmore 2:4-5; 5:51-57; 10:35-53; 11:35-67). In other words, Wetmore results in the transmission of large difference results, because if even a single line of binary code were added to a routine, the *entire* routine would need to be included in Wetmore's difference result. Accordingly, Wetmore does not teach, suggest or provide reason for a modification that would meet a *compact* difference result,

which would entail generation of a difference result even *smaller* than those computed by prior art techniques, such as Coppieters.

For at least the foregoing reasons, Owner respectfully traverses the rejections of the Executable Program Invariant Reference Claims (*i.e.* claims 8, 11, 21, 24, and 28) and the Data Table Invariant Reference Claims (*i.e.* claims 42, 45, 55, 58, and 62 and therefore dependent claims 9, 10, 22, 23, 43, 44, 56 and 57) under 35 U.S.C. § 102(b).

ii. The Distinct Label Mark Claims

The Office has rejected Claims 1, 4, 14, 17, and 26 (hereinafter the "Executable Program Distinct Label Mark Claims") and Claims 35, 38, 48, 51, and 61 (hereinafter the "Data Table Distinct Label Mark Claims") (collectively, the "Executable Program Distinct Label Mark Claims" and the "Data Table Distinct Label Mark Claims" are referred to herein as the "Distinct Label Mark Claims") under 35 U.S.C. § 102(b) over Wetmore. The Owner respectfully traverses.

(1) The Inputs to Wetmore Are Not Executable

As to the Executable Program Distinct Label Mark Claims, each of these claims requires scanning an old and new program, both of which are *executable*. (See *e.g.* '552 Patent claim 1, preamble). As discussed above with respect to the Executable Program Invariant Reference Claims, Wetmore fails to disclose this limitation because the files that Wetmore operates on as part of its vectorization process are object files that contain unresolved references, prior to linking, and are not executable. See, *supra* pp. 8-10.

(2) No "References" In Wetmore Are Replaced By a Distinct Label Mark

Regarding both the Executable Program Distinct Label Mark Claims and the Data Table Distinct Label Mark Claims, Wetmore also fails to disclose a modification that replaces any "reference," as defined in the Glossary of the '552 Patent and recited in this claim limitation, for example:

(a) scanning the old program and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark...

As discussed previously (see *supra* at Section 1(D)(i)(2)), the "references" of the '552 Patent that are initially processed or input as part of the claimed techniques are numeric. In contrast, the "references" of Wetmore that are vectorized are purely symbolic, because they are unresolved references in object files. Accordingly, Wetmore fails to teach, suggest or provide reason for a modification that would meet all elements of these claims.

(3) Wetmore Fails To Disclose "For Substantially Each Reference Entry . . . Replacing The Reference . . . By a Distinct Label Mark"

Regarding both the Executable Program Distinct Label Mark Claims and the Data Table Distinct Label Mark Claims, Wetmore fails to disclose replacing the reference by a distinct label mark for "substantially each reference entry." (See *e.g.* 552 Patent, claim 1). Instead, Wetmore describes vectorizing *only* "entry points" to routines (Wetmore 7:1-9; *supra* at Section 1(D)(i)(2)), rather than the *references*. ('552 Patent, claim 1). Assuming for the sake of argument that non-numeric labels in object files could qualify as "references," Wetmore still fails to teach, suggest or provide reason for a modification that would meet the claimed invention.

First, Wetmore causes *entry points* to become invariant through vectorization of the old and new object files. Wetmore does not describe replacing any *reference* to an entry point with a distinct label mark -- an enormous deficiency in Wetmore's disclosure that the Requester seems to concede. (Request at 63 ("in the Wetmore '713 Patent . . . reference to entry points (303) . . . are not expressly disclosed as being replaced or modified")). Accordingly, Wetmore fails to disclose or suggest "replacing the reference" in "substantially each reference entry," which the Distinct Label Mark Claims (*i.e.*, Claims 1, 4, 14, 17, 27, 35, 38, 48, 51, and 61 and therefore dependent claims 2, 3, 15, 16, 36, 37, 49 and 50) require.

Second, Wetmore does not teach, disclose or even suggest replacing references that are not themselves "entry points," such as references within the routines. Thus, as

discussed above with respect to the Invariant Reference Claims (Claims 8, 11, 21, 24, 28, 42, 45, 55, 58 and 62, and therefore dependent claims 9, 10, 22, 23, 43, 44, 56 and 57) (see supra pp. 12-13), Wetmore fails to teach or suggest this limitation.

**(4) Wetmore Fails To Disclose Generation of A
"Compact Difference Result"**

Regarding both the Executable Program Distinct Label Mark Claims and the Data Table Distinct Label Mark Claims (Claims 1, 4, 14, 17, 27, 35, 38, 48, 51 and 61, and therefore dependent claims 2, 3, 15, 16, 36, 37, 49 and 50), Wetmore fails to disclose generation of a "compact difference result." As discussed above with respect to the Invariant Reference claims, Wetmore's difference result simply includes the entirety of the code of a modified routine if any change is made within that routine. (See supra pp. 13-14). This level of granularity in generation of a difference result does not constitute a "compact difference result" as recited in these claims, as explained above.

iii. The Rejections Under 35 U.S.C. § 103

The Office has rejected the claims 2, 3, 9, 10, 15, 16, 22, 23, 36, 37, 43, 44, 49, 50, 56, and 57 under 35 U.S.C. § 103 over Wetmore in view of U.S. Patent No. 5,790,796 to Sadowsky (hereinafter "Sadowsky"). Owner respectfully traverses this rejection.

As discussed above, Wetmore does not disclose or suggest the features of independent claims 1, 8, 14, 21, 35, 42, 48, or 55 from which these claims depend. Accordingly, the Office has failed to make out a prima facie case of obviousness with regard to claims 2, 3, 9, 10, 15, 16, 22, 23, 36, 37, 43, 44, 49, 50, 56, and 57.

Moreover, the Owner traverses the Office's suggestion that it would be obvious to a person of ordinary skill in the art to utilize Sadowsky's method of transmitting a difference result over a network with Wetmore. In particular, since Wetmore is concerned with updating the operating system of a computer running in ROM and RAM, where the updated software modules to be distributed to the user must be loaded and the "patch" applied "at boot time," the only way to implement the techniques of Wetmore would be to provide the patch on a "system disk." (Wetmore 10:29-34; 11:35-39). The Office does not explain, and

Wetmore fails to suggest, how a patch formed using the techniques of Wetmore could be provided to a user over a network such that the patch could be subsequently applied to update the operating system (which in Wetmore is presumed to be stored in a Read-Only Memory), without making use of a "system disk" that is read "at boot time" to load the appropriate patch code into RAM. (Wetmore 11:35-39).

Additionally, the Owner respectfully suggests that the Office has employed improper hindsight reasoning in suggesting that the transmission of the Wetmore update over a network would "reduc[e] the costs associated with transferring the disks via normal transportation channels." (Office Action at 18). In particular, as of the priority date of the present invention, bandwidth was far more expensive than today, and the Owner respectfully disagrees with the Office's contention that transmitting the "patch" files of Wetmore (which, as discussed above, would contain far more data than a compact difference result of the presently claimed invention) via a network would reduce the costs associated with transferring disks, or would even be practical given the data transfer rates to end-users that were typical as of the Owner's priority date. Even if this were possible and practical, there still would remain the problem of how the user would apply the patch to his or her machine. Presumably, the user would need to first burn the received file to a CD, so that it could be used as Wetmore's "system disk" during the boot process. This would require each user wishing to update his or her system to purchase or make use of a CD recorder, in order to copy the patch files received via the network onto a "system disk." However, CD recorder technology was in its early stages as of the Owner's priority date, and was still expensive, with blank media costing approximately \$10/disk. (See Chris O'Malley, *A New Spin*, Time, Aug. 24, 1998 <<http://www.time.com/time/magazine/article/0,9171,988955,00.html>>). Thus, in view of the foregoing, Owner respectfully traverses the Office's rejections under § 103.

Owner reserves the right to submit substantial evidence of secondary considerations of non-obviousness in the unlikely event that the Office lodges further rejections under § 103.

E. Comments on Statement of Reasons For Patentability And/Or Confirmation

In addition to the reasons identified by the Examiner in the Statement of Reasons for Patentability of the confirmed claims, the Owner notes that those claims would also be allowable for many of the same reasons discussed above with regard to the rejected claims. Also, the Owner notes that the claims were somewhat inaccurately paraphrased (e.g., the statement "modified old program is reconstituted" more accurately would have said "reconstituting the modified *new* program" as per claims 5 and 12, for instance), but it is understood that the Office is relying on the claim language, rather than its paraphrasing of the claims, in indicating the reasons for allowance. The Owner also respectfully disagrees with the Examiner's characterization of the teachings of Wetmore for the reasons given above.

F. Conclusion

For the reasons set forth above, Owner respectfully traverses and requests withdrawal of the outstanding rejections, and requests prompt issuance of a Notice of Intent to Issue a Reexamination Certificate (NIRC) confirming the patentability of all claims. Should any residual issues exist or arise, or the Examiner believes an interview would be helpful noting the conversation above, the Examiner is invited to contact the undersigned below.


The Director is hereby authorized to charge any appropriate fees that may be required by this paper, and to credit any overpayment, to Deposit Account No. 02-4800.

Respectfully submitted,

BUCHANAN INGERSOLL & ROONEY PC

Date: June 29, 2010

By:



Charles F. Wieland III
Registration No. 33096

Customer No. 21839
703 836 6620

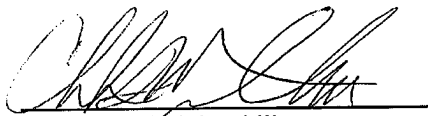
CERTIFICATE OF SERVICE

It is hereby certified by the undersigned that a true copy of the Supplemental Response filed on June 29, 2010 was transmitted via e-mail to:

Jonathan A. David, Esq.
Lerner, David, Littenberg, Krumholz & Mentlik, LLP
600 South Avenue West
Westfield, New Jersey 07090

on this 29th day of June, 2010.

By:



Charles F. Wieland III
Registration No. 33096

Customer No. 21839
703 836 6620



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

DO NOT USE IN PALM PRINTER

(THIRD PARTY REQUESTER'S CORRESPONDENCE ADDRESS)

GOOGLE

LERNER, DAVID, LITTENBERG, KRUMHOLZ & MENTLIK, LLP

600 SOUTH AVENUE WEST

WESTFIELD, NJ 07090

MAILED

MAY 28 2010

CENTRAL REEXAMINATION UNIT

EX PARTE REEXAMINATION COMMUNICATION TRANSMITTAL FORM

REEXAMINATION CONTROL NO. 90/009,670.

PATENT NO. 6546552

ART UNIT 3992.

Enclosed is a copy of the latest communication from the United States Patent and Trademark Office in the above identified *ex parte* reexamination proceeding (37 CFR 1.550(f)).

Where this copy is supplied after the reply by requester, 37 CFR 1.535, or the time for filing a reply has passed, no submission on behalf of the *ex parte* reexamination requester will be acknowledged or considered (37 CFR 1.550(g)).

LDLK & M
JUN 01 2010
RECEIVED

PTOL-465 (Rev.07-04)



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

DO NOT USE IN PALM PRINTER

(THIRD PARTY REQUESTER'S CORRESPONDENCE ADDRESS)

GOOGLE

LERNER, DAVID, LITTENBERG, KRUMHOLZ & MENTLIK, LLP

600 SOUTH AVENUE WEST

WESTFIELD, NJ 07090

MAILED

MAY 28 2010

CENTRAL REEXAMINATION UNIT

EX PARTE REEXAMINATION COMMUNICATION TRANSMITTAL FORM

REEXAMINATION CONTROL NO. 90/009,670.

PATENT NO. 6546552.

ART UNIT 3992.

Enclosed is a copy of the latest communication from the United States Patent and Trademark Office in the above identified *ex parte* reexamination proceeding (37 CFR 1.550(f)).

Where this copy is supplied after the reply by requester, 37 CFR 1.535, or the time for filing a reply has passed, no submission on behalf of the *ex parte* reexamination requester will be acknowledged or considered (37 CFR 1.550(g)).

PTOL-465 (Rev.07-04)



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/009,670	01/22/2010	6546552	GOOGLE 3.6-141	4316
71839	7590	05/28/2010	EXAMINER	
BUCHANAN, INGERSOLL & ROONEY PC POST OFFICE BOX 1404 ALEXANDRIA, VA 22313-1404			ART UNIT	PAPER NUMBER

DATE MAILED: 05/28/2010

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

Office Action in Ex Parte Reexamination	Control No. 90/009,670	Patent Under Reexamination 6546552	
	Examiner ANDREW L. NALVEN	Art Unit 3992	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

- a Responsive to the communication(s) filed on _____ . b This action is made FINAL.
c A statement under 37 CFR 1.530 has not been received from the patent owner.

A shortened statutory period for response to this action is set to expire 2 month(s) from the mailing date of this letter. Failure to respond within the period for response will result in termination of the proceeding and issuance of an *ex parte* reexamination certificate in accordance with this action. 37 CFR 1.550(d). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c).** If the period for response specified above is less than thirty (30) days, a response within the statutory minimum of thirty (30) days will be considered timely.

Part I THE FOLLOWING ATTACHMENT(S) ARE PART OF THIS ACTION:

1. Notice of References Cited by Examiner, PTO-892. 3. Interview Summary, PTO-474.
2. Information Disclosure Statement, PTO/SB/08. 4. _____.

Part II SUMMARY OF ACTION

- 1a. Claims 1-68 are subject to reexamination.
1b. Claims _____ are not subject to reexamination.
2. Claims _____ have been canceled in the present reexamination proceeding.
3. Claims 5-7, 12-13, 18-20, 25-26, 29-34, 39-41, 46-47, 52-54, 59-60, 63-68 are patentable and/or confirmed.
4. Claims 1-4, 8-11, 14-17, 21-24, 27, 28, 35-38, 42-45, 48-51, 55-58, 61 and 62 are rejected.
5. Claims _____ are objected to.
6. The drawings, filed on _____ are acceptable.
7. The proposed drawing correction, filed on _____ has been (7a) approved (7b) disapproved.
8. Acknowledgment is made of the priority claim under 35 U.S.C. § 119(a) (d) or (f).
a) All b) Some* c) None of the certified copies have
1 been received.
2 not been received.
3 been filed in Application No. _____
4 been filed in reexamination Control No. _____
5 been received by the International Bureau in PCT application No. _____
* See the attached detailed Office action for a list of the certified copies not received.
9. Since the proceeding appears to be in condition for issuance of an *ex parte* reexamination certificate except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte* Quayle, 1935 C.D. 11, 453 O.G. 213.
10. Other: _____

cc: Requester (if third party requester)

DETAILED ACTION

I. Procedures Governing Reexamination

Proposed Amendments, Affidavits, or Declarations

In order to ensure full consideration of any amendments, affidavits or declarations, or other documents as evidence of patentability, such documents must be submitted in response to this Office action. Submissions after the next Office action, which is intended to be a final action, will be governed by the requirements of 37 CFR 1.116, after final rejection and 37 CFR 41.33 after appeal, which will be strictly enforced.

Patent owner is notified that any proposed amendment to the specification and/or claims in this reexamination proceeding must comply with 37 CFR 1.530(d)-(j), must be formally presented pursuant to 37 CFR 1.52(a) and (b), and must contain any fees required by 37 CFR 1.20(c).

Extensions of Time

Extensions of time under 37 CFR 1.136(a) will not be permitted in these proceedings because the provisions of 37 CFR 1.136 apply only to "an applicant" and not to parties in a reexamination proceeding. Additionally, 35 U.S.C. 305 requires that reexamination proceedings "will be conducted with special dispatch" (37 CFR 1.550(a)). Extension of time in *ex parte* reexamination proceedings are provided for in 37 CFR 1.550(c).

Concurrent Litigation

The patent owner is reminded of the continuing responsibility under 37 CFR 1.565(a) to apprise the Office of any litigation activity, or other prior or concurrent proceeding, involving the patent at issue in this reexamination proceeding throughout the course of this reexamination proceeding. The third party requester is also reminded of the ability to similarly apprise the Office of any such activity or proceeding throughout the course of this reexamination proceeding. See MPEP §§ 2207, 2282 and 2286.

II. Summary of the Prosecution and Reexamination Proceeding

US Patent No. 6,546,552 ("the '552 patent") was issued on April 8, 2003 from an application filed August 18, 1999. During the prosecution of the '552 patent, the claims were rejected in view of the Okuzumi and Kenji references. Following claim amendments, the claims were allowed in a notice of allowance mailed on 8/27/2002. In that notice of allowance, the Examiner stated that the reasons for allowance was the failure of the prior art to teach or suggest "generating a modified new file and using the modified new file and the modified old file to generate a difference result" (*see '552 Patent, Notice of Allowance mailed 8/27/2002, Page 2*). The limitation that most closely relates to these reasons for allowance states: "generating said difference result utilizing directly or indirectly at least said modified old program and modified new program" (*see Claim 1*).

On January 22, 2010, Third Party Requester ("Requester") submitted a request for reexamination of claims 1-68 of the '552 patent in view of the following prior art patents and publications:

1. U.S. Patent No. 5,481,713 to Wetmore et al entitled "Method And Apparatus For Patching Code Residing On A Read Only Memory Device," issued on January 2, 1996 (hereafter "Wetmore"). Wetmore was not cited in a previous examination. Wetmore qualifies as prior art under 102(b).
2. IBM Technical Disclosure Bulletin, Batalden, G.D., et al., "Maintainable ROS Code Through The Combination of ROM And EEPROM." Vol.32 No. 9A, p.273-76. published in February, 1990 (hereafter "Batalden"). Batalden was not cited in a previous examination. Batalden qualifies as prior art under 102(b).
3. U.S. Patent No. 4,111,853 to Dummermuth entitled "Jump Structure For A Digital Control System," filed on December 21, 1976, and issued on September 19, 1978 (hereafter "Dummermuth"). Dummermuth was not cited in a previous examination. Dummermuth qualifies as prior art under 102(b).
4. U.S. Patent No. 5,790,796 to Sadowsky entitled "Polymorphic Package Files To Update Software Components," filed on June 14, 1996, and issued on August 4, 1998 (hereafter "Sadowsky"). Sadowsky was not cited in a previous examination. Sadowsky qualifies as prior art under 102(b).
5. Coppieters, K., "A Cross-Platform Binary Diff," Dr. Dobb's Journal, US, San Mateo, California, pp. 32, XP 000610668, was published in May 1995 (hereafter

Art Unit: 3992

"Coppieters"). Coppieters was cited in, but not discussed or applied in an earlier examination. Coppieters qualifies as prior art under 102(b).

Reexamination was granted for claims 1-68 in the order mailed March 16, 2010.

III. Grounds of Rejection

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1-4, 8-11, 14-17, 21-24, 27-28, 35-38, 42-45, 48-51, 55-58, and 61-62 are rejected under 35 U.S.C. 102(b) as being anticipated by Wetmore. This rejection for claims 1-4, 8-11, 14-17, 21-24, 27-28, 35-38, 42-45, 48-51, 55-58, and 61-62 appears below. Further, the proposed rejection of claims 1-4, 8-11, 14-17, 21-24, 27-28, 35-38, 42-45, 48-51, 55-58, and 61-

Art Unit: 3992

62 set forth in the January 22, 2010 request for reexamination on pages 50-163 is incorporated by reference.

With regards to claim 1, Wetmore teaches a method for generating a compact difference result between an old executable program and a new executable program (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version*);

each program including reference entries that contain reference that refer to other entries in the program (*Wetmore, column 5 lines 18-56; column 6 lines 47-67 – source code is compiled into a object file where the object file includes external references to other routines. the object files are linked into a final ROM image*);

the method comprising the steps of: (a) scanning the old program and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified old program is generated (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) scanning the new program and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified new program is generated (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*);

Art Unit: 3992

(c) generating said difference result utilizing directly or indirectly at least said modified old program and modified new program (*Wetmore, column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines*).

With regards to claim 4, *Wetmore* teaches the step of: (d) storing said compact difference result on a storage medium (*Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums*).

With regards to claim 8, *Wetmore* teaches a method for generating a compact difference result between an old executable program and a new executable program (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version and installing the patch*);

each program including reference entries that contain reference that refer to other entries in the program (*Wetmore, column 5 lines 18-56; column 6 lines 47-67 – source code is compiled into a object file where the object file includes external references to other routines, the object files are linked into a final ROM image*);

the method comprising the steps of: (a) generating a modified old program utilizing at least said old program (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) generating a modified new program utilizing at least said new program (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*),

said modified old program and modified new program have at least the following characteristics: (i) substantially each reference in an entry in said old program that is different than corresponding entry in said new program due to delete/insert modifications that form part of the transition between said old program and new program are reflected as invariant references in the corresponding entries in said modified old and modified new programs (*Wetmore, Figures 3-5; column 5 lines 18-56, column 6 line 45 – column 8 line 52; column 8 lines 1-16- the invariant references are reflected as the table pointers with offsets that are included in the modified old and new programs*);

(c) generating said compact difference result utilizing at least said modified new program and modified old program (*Wetmore, column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines*).

With regards to claim 11, *Wetmore* teaches the step of: (d) storing said compact difference result on a storage medium (*Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums*).

With regards to claim 14, *Wetmore* teaches a system for generating a compact difference result between an old executable program and a new executable program;

Art Unit: 3992

each program including reference entries that contain reference that refer to other entries in the program (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version*);

the system comprising a processing device capable of: (a) scanning the old program and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified old program is generated (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) scanning the new program and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified new program is generated (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*);

(c) generating said difference result utilizing directly or indirectly at least said modified old program and modified new program (*Wetmore, column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines*).

With regards to claim 17, Wetmore teaches said processor device is further capable of storing said compact difference result on a storage medium (*Wetmore, column 11 lines 34-57, vector patch resource is loaded: column 2 lines 49-60, ROM and RAM storage mediums*).

With regards to claim 21, Wetmore teaches a system for generating a compact difference result between an old executable program and a new executable program (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version and installing the patch*);

each program including reference entries that contain reference that refer to other entries in the program (*Wetmore, column 5 lines 18-56: column 6 lines 47-67 – source code is compiled into a object file where the object file includes external references to other routines, the object files are linked into a final ROM image*);

the system comprising a processing device capable of: (a) generating a modified old program utilizing at least said old program (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) generating a modified new program utilizing at least said new program (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*),

said modified old program and modified new program have at least the following characteristics: (i) substantially each reference in an entry in said old program that is different than corresponding entry in said new program due to delete/insert modifications that form part of the transition between said old program and new program are reflected as invariant references in the corresponding entries in said modified old and modified new programs (*Wetmore, Figures 3-*

Art Unit: 3992

5; column 5 lines 18-56, column 6 line 45 – column 8 line 52; column 8 lines 1-16- the invariant references are reflected as the table pointers with offsets that are included in the modified old and new programs);

(c) generating said compact difference result utilizing at least said modified new program and modified old program (*Wetmore*, column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines).

With regards to claim 24, *Wetmore* teaches said processor is further capable of storing said compact difference result on a storage medium (*Wetmore*, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums).

With regards to claim 27, *Wetmore* teaches a processing device having associated therewith a storage medium which holds compact difference result data that was generated by the method of anyone of claims 1 to 4 (*Wetmore*, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums).

With regards to claim 28, *Wetmore* teaches a processing device having associated therewith a storage medium which holds compact difference result data that was generated by the method of anyone of claims 8 to 11 (*Wetmore*, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums).

With regards to claim 35, Wetmore teaches a method for generating a compact difference result between an old data table and a new data table (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version*);

each data table including reference entries that contain reference that refer to other entries in the data table (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version*);

the method comprising the steps of: (a) scanning the old data table and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified old data table is generated (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) scanning the new data table and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified new data table is generated (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*);

(c) generating said difference result utilizing directly or indirectly at least said modified old data table and modified new data table (*Wetmore, column 10 line 65 – column 11 line 12 – the object files of two versions of the vectorized code are compared to identify new or different routines*).

With regards to claim 38, Wetmore teaches the step of: (d) storing said compact difference result on a storage medium (*Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums*).

With regards to claim 42, Wetmore teaches a method for generating a compact difference result between an old data table and a new data table (*Wetmore, Abstract patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version and installing the patch*);

each data table including reference entries that contain reference that refer to other entries in the data table (*Wetmore, column 5 lines 18-56; column 6 lines 47-67 – source code is compiled into a object file where the object file includes external references to other routines, the object files are linked into a final ROM image*);

the method comprising the steps of: (a) generating a modified old data table utilizing at least said old data table (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) generating a modified new data table utilizing at least said new data table (*Wetmore, column 10 line 65 column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*);

said modified old data table and modified new data table have at least the following characteristics: (i) substantially each reference in an entry in said old data table that is different than corresponding entry in said new data table due to delete/insert modifications that form part of the transition between said old data table and new data table are reflected as invariant references in the corresponding entries in said modified old and modified new data tables (*Wetmore, Figures 3-5: column 5 lines 18-56, column 6 line 45 – column 8 line 52; column 8 lines 1-16- the invariant references are reflected as the table pointers with offsets that are included in the modified old and new programs*);

(c) generating said compact difference result utilizing at least said modified new data table and modified old data table (*Wetmore. column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines*).

With regards to claim 45, Wetmore teaches the step of: (d) storing said compact difference result on a storage medium (*Wetmore. column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums*).

With regards to claim 48, Wetmore teaches a system for generating a compact difference result between an old data table and a new data table (*Wetmore. Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version*);

Art Unit: 3992

each data table including reference entries that contain reference that refer to other entries in the data table (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version*);

the system comprising a processing device capable of: (a) scanning the old data table and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified old data table is generated (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) scanning the new data table and for substantially each reference entry perform steps that include: (i) replacing the reference of said entry by a distinct label mark, whereby a modified new data table is generated (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*);

(c) generating said difference result utilizing directly or indirectly at least said modified old data table and modified new data table (*Wetmore, column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines*).

With regards to **claim 51**, *Wetmore* teaches said processor device is further capable of storing said compact difference result on a storage medium (*Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums*).

With regards to claim 55, Wetmore teaches a system for generating a compact difference result between an old data table and a new data table (*Wetmore, Abstract – patch resources are generated for each ROM version by comparing previous ROM versions to the new ROM version and installing the patch*);

each data table including reference entries that contain reference that refer to other entries in the data table (*Wetmore, column 5 lines 18-56; column 6 lines 47-67 – source code is compiled into a object file where the object file includes external references to other routines, the object files are linked into a final ROM image*);

the system comprising a processing device capable of: (a) generating a modified old data table utilizing at least said old data table (*Wetmore, column 8 lines 1-20, the object files are used to generate a vector table object file where the entry point references are replaced with appropriate vector code*);

(b) generating a modified new data table utilizing at least said new data table (*Wetmore, column 10 line 65 – column 11 line 12 – when creating a Vector Patch Resource, two versions of Vectorized ROM code are compared. Hence, a new and an old program are vectorized; for a discussion of the vectorization of code see Wetmore, column 8 lines 1-20*),

said modified old data table and modified new data table have at least the following characteristics: (i) substantially each reference in an entry in said old data table that is different than corresponding entry in said new data table due to delete/insert modifications that form part of the transition between said old data table and new data table are reflected as invariant references in the corresponding entries in said modified old and modified new data tables

Art Unit: 3992

(Wetmore, Figures 3-5; column 5 lines 18-56, column 6 line 45 – column 8 line 52; column 8 lines 1-16- the invariant references are reflected as the table pointers with offsets that are included in the modified old and new programs);

(c) generating said compact difference result utilizing at least said modified new data table and modified old data table *(Wetmore, column 10 line 65 – column 11 line 12 - the object files of two versions of the vectorized code are compared to identify new or different routines).*

With regards to claim 58, Wetmore teaches said processor device is further capable of storing said compact difference result on a storage medium *(Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums).*

With regards to claim 61, Wetmore teaches a processing device having associated therewith a storage medium which holds compact difference result data that was generated by the method of anyone of claims 35 to 38 *(Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums).*

With regards to claim 62, Wetmore teaches a processing device having associated therewith a storage medium which holds compact difference result data that was generated by the method of anyone of claims 42 to 45 *(Wetmore, column 11 lines 34-57, vector patch resource is loaded; column 2 lines 49-60, ROM and RAM storage mediums).*

Claims 2, 3, 9, 10, 15, 16, 22, 23, 36, 37, 43, 44, 49, 50, 56, and 57 are rejected under 35 U.S.C. 103(a) as being unpatentable over Wetmore in view of Sadowsky.

With regards to claims 2, 9, 15, 22, 36, 43, 49, and 56, Wetmore fails to teach transmitting said compact difference result over a communication network. However, Sadowsky teaches transmitting said compact difference result over a communication network (*Sadowsky, column 4 lines 49-55, communication channel may be the Internet; Figure 5 – determine appropriate update package in step 614 and then download the appropriate update package in steps 606 and 608; column 5 lines 18-54*). At the time the invention was made, it would have been obvious to one of ordinary skill in the art to utilize Sadowsky's method of transmitting a difference result over a network because it offers the advantage of allowing simplifying the acquiring of a software update by removing the need to received disks or CD-ROMs and reducing the costs associated with transferring the disks via normal transportation channels (*Sadowsky, column 1 lines 12-23*).

With regards to claims 3, 10, 16, 23, 37, 44, 50, and 57, Wetmore as modified teaches the network includes the Internet (*Sadowsky, column 4 lines 49-55, communication channel may be the Internet*).

STATEMENT OF REASONS FOR PATENTABILITY AND/OR CONFIRMATION

The following is an examiner's statement of reasons for patentability and/or confirmation of the claims found patentable in this reexamination proceeding:

Regarding claims 5-7, 12-13, 18-20, 25-26, 29-34, 39-41, 46-47, 52-54, 59-60, 63-68, the prior art of record fails to teach or suggest the step of or structural element for reconstituting said new program utilizing directly or indirectly at least said compact difference result and said modified new program. The closest prior art, Wetmore, teaches that an old and a new program are vectorized to create modified old and modified new programs. The differences between the modified programs are determined to generate a compact difference result (*Wetmore, Figure 7b; column 11*). Next, the modified old program is reconstituted using NewVector loader to match the modified new program utilizing the compact different result and the modified old program (*see Wetmore, Figure 7b; column 11 lines 35-67*). However, Wetmore does not teach the reconstituting of the original, non-vectorized new program utilizing directly or indirectly at least said compact difference result and said modified new program. Accordingly, the prior art of record fails to anticipate or render obvious the above noted claims.

Any comments considered necessary by PATENT OWNER regarding the above statement must be submitted promptly to avoid processing delays. Such submission by the patent owner should be labeled: "Comments on Statement of Reasons for Patentability and/or Confirmation" and will be placed in the reexamination file.

Information Disclosure Statement

The information disclosure statements (IDS) submitted after the mailing date of the Order Granting Ex Parte Reexamination on 16 March 2010. The submission is in compliance with the

provisions of 37 CFR 1.97. Accordingly, the IDSs are being considered by the examiner to the extent that the references have been explained by the Patent Owner.

CORRESPONDENCE

All correspondence relating to this ex parte reexamination proceeding should be directed:

By EFS: Registered users may submit via the electronic filing system EFS-Web, at <https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html>.

By Mail to: Mail Stop *Ex Parte* Reexam
Central Reexamination Unit
Commissioner for Patents
United States Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

By FAX to: (571) 273-9900
Central Reexamination Unit

By hand: Customer Service Window
Randolph Building
401 Dulany Street
Alexandria, VA 22314

For EFS-Web transmissions, 37 CFR 1.8(a)(1)(i) (C) and (ii) states that correspondence (except for a request for reexamination and a corrected or replacement request for reexamination) will be considered timely filed if (a) it is transmitted via the Office's electronic filing system in accordance with 37 CFR 1.6(a)(4), and (b) includes a certificate of transmission for each piece of correspondence stating the date of transmission, which is prior to the expiration of the set period of time in the Office action.

Art Unit: 3992

Any inquiry concerning this communication or earlier communications from the Examiner, or as to the status of this proceeding, should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

Signed:

/Andrew Nalven/

Andrew Nalven
CRU Examiner
GAIJ 3992
(571) 272-3839

Conferee: ESK

Conferee: 707

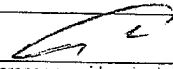
FIRST
INFORMATION DISCLOSURE
(use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.:	0077898-000001

Sheet 1 of 5

U.S. PATENT DOCUMENTS

Examiner Initials	Document Number-Kind Code	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines Where Relevant Passages or Figures Appear
AK	US-3,969,723	07-13-1976	KENNICOTT	
	US-4,278,837	07-14-1981	BEST	
	US-4,319,079	03-09-1982	BEST	
	US-4,464,650	08-07-1984	EASTMAN et al.	
	US-4,465,901	08-14-1984	BEST	
	US-4,542,453	09-17-1985	PATRICK et al.	
	US-4,683,549	07-28-1987	TAKAKI	
	US-4,807,182	02-21-1989	QUEEN	
	US-4,831,517	05-16-1989	CROUSE et al.	
	US-4,972,504	11-20-1990	DANIEL, Jr. et al.	
	US-4,987,550	01-22-1991	LEONARD et al.	
	US-5,146,221	09-08-1992	WHITING et al.	
	US-5,051,745	09-24-1991	KATZ	
	US-5,146,221	09-08-1992	WHITING et al.	
	US-5,155,847	10-13-1992	KIROUAC et al.	
	US-5,170,465	12-08-1992	McKEEMAN et al.	
	US-5,204,960	04-20-1993	SMITH et al.	
	US-5,210,854	05-11-1993	BEAVERTON et al.	
	US-5,260,693	11-09-1993	HORSLEY	
	US-5,307,492	04-26-1994	BENSON	
	US-5,319,645	06-07-1994	BASSI et al.	
	US-5,347,653	09-13-1994	FLYNN et al.	
	US-5,357,629	10-18-1994	DINNIS et al.	
	US-5,392,390	02-21-1995	CROZIER	
	US-5,410,703	04-25-1995	NILSSON et al.	
	US-5,450,589	09-12-1995	MAEBAYASHI et al.	
	US-5,465,258	11-07-1995	ADAMS	
	US-5,473,772	12-05-1995	HALLIWELL et al.	
	US-5,479,654	12-26-1995	SQUIBB	
	US-5,491,821	02-13-1996	KILIS	
	US-5,493,674	02-20-1996	MIZUTANI et al.	
	US-5,502,439	03-26-1996	BERLIN	
US-5,546,586	08-13-1996	WETMORE et al.		
US-5,574,573	11-12-1996	RAY et al.		
US-5,574,898	11-12-1996	LEBLANG et al.		
US-5,574,906	11-12-1996	MORRIS		
US-5,581,697	12-03-1996	GRAMLICH et al.		
US-5,581,768	12-03-1996	GARNEY et al.		
US-5,588,143	12-24-1996	STUPEK, Jr. et al.		
US-5,600,834	02-04-1997	HOWARD		

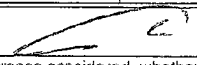
Examiner Signature		Date Considered	5/27/00
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FIRST
INFORMATION DISCLOSURE
(use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.	0077898-000001

Sheet 2 of 5

AP	US-5,619,698	04-08-1997	LILLICH et al.	
	US-5,630,072	05-13-1997	DOBBINS	
	US-5,636,366	06-03-1997	ROBINSON et al.	
	US-5,659,755	08-19-1997	STROHACKER	
	US-5,675,802	10-07-1997	ALLEN et al.	
	US-5,675,803	10-07-1997	PREISLER et al.	
	US-5,694,566	12-02-1997	NAGAE	
	US-5,732,275	03-24-1998	KULLICK et al.	
	US-5,745,058	04-28-1998	AUERBACH et al.	
	US-5,745,906	04-28-1998	SQUIBB	
	US-5,752,039	05-12-1998	TANIMURA	
	US-5,757,690	05-26-1998	McMAHON	
	US-5,764,987	06-09-1998	EIDT et al.	
	US-5,764,994	06-09-1998	CRAFT	
	US-5,790,856	08-04-1998	LILLICH	
	US-5,790,860	08-04-1998	WETMORE et al.	
	US-5,794,254	08-11-1998	McCLAIN	
	US-5,799,144	08-25-1998	MIO	
	US-5,802,549	09-01-1998	GOYAL et al.	
	US-5,815,714	09-29-1998	SHRIDHAR et al.	
	US-5,829,001	10-27-1998	LI et al.	
	US-5,832,520	11-03-1998	MILLER	
	US-5,835,701	11-10-1998	HASTINGS	
	US-5,837,986	11-17-1998	BARILE et al.	
	US-5,838,264	11-17-1998	COOPER	
	US-5,844,508	12-01-1998	MURASHITA et al.	
	US-5,848,274	12-08-1998	HAMBY et al.	
	US-5,873,097	02-16-1999	HARRIS et al.	
	US-5,884,094	03-16-1999	BEGUN et al.	
	US-5,886,991	03-23-1999	GUARNERI et al.	
US-5,889,995	03-30-1999	SEGNAN		
US-5,893,113	04-06-1999	McGRATH et al.		
US-5,897,633	04-27-1999	NOLAN		
US-5,901,225	05-04-1999	IRETON et al.		
US-5,938,766	08-17-1999	ANDERSON et al.		
US-5,953,534	09-14-1999	ROMER et al.		
US-5,956,479	09-21-1999	McINERNEY et al.		
US-5,958,048	09-28-1999	BABAIAN et al.		
US-5,960,189	09-28-1999	STUPEK, Jr. et al.		
US-5,964,873	10-12-1999	CHOI		
US-5,966,541	10-12-1999	AGARWAL		
US-5,974,254	10-26-1999	HSU		
US-5,978,791	11-02-1999	FARBER et al.		
US-5,982,937	11-09-1999	ACCAD		
US-5,983,000	11-09-1999	PERRON		
US-5,990,810	11-23-1999	WILLIAMS		
Examiner Signature		Date Considered	5/27/10	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FIRST
INFORMATION DISCLOSURE
(use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group/Art Unit:	Andrew L. Naiven/3992
Confirmation No.:	4316
Attorney Docket No.	0077898-000001

Sheet 3 of 5

AN	US-5,995,754	11-30-1999	HOLZLE et al.	
	US-6,006,328	12-21-1999	DRAKE	
	US-6,009,261	12-28-1999	SCALZI et al.	
	US-6,018,349	01-25-2000	SZELISKI et al.	
	US-6,018,747	01-25-2000	BURNS et al.	
	US-6,026,235	02-15-2000	SHAUGHNESSY	
	US-6,052,531	04-18-2000	WALDIN, Jr. et al.	
	US-6,071,004	06-06-2000	Le GALL et al.	
	US-6,071,317	06-06-2000	NAGEL	
	US-6,076,134	06-13-2000	NAGAC	
	US-6,077,311	06-20-2000	LICHTENSTEIN et al.	
	US-6,078,931	06-20-2000	MOTOYAMA	
	US-6,112,025	08-29-2000	MULCHANDANI et al.	
	US-6,115,550	09-05-2000	HUNTER et al.	
	US-6,122,731	09-19-2000	SON	
	US-6,141,698	10-31-2000	KRISHNAN et al.	
	US-6,163,780	12-19-2000	ROSS	
	US-6,163,811	12-19-2000	PORTER	
	US-6,167,407	12-26-2000	NACHENBERG et al.	
	US-6,175,663	01-16-2001	HUANG	
	US-6,189,145	02-13-2001	BELLIN, Jr. et al.	
	US-6,192,157	02-20-2001	PREBBLE	
	US-6,192,475	02-20-2001	WALLACE	
	US-6,202,208	03-13-2001	HOLIDAY, JR.	
	US-6,216,175	04-10-2001	SLIGER et al.	
	US-6,216,213	04-10-2001	BRETERNITZ, JR. et al.	
	US-6,202,208	03-13-2001	HOLIDAY, JR.	
	US-6,230,316	05-08-2001	NACHENBERG	
	US-6,240,550	05-29-2001	NATHAN et al.	
	US-6,243,766	06-05-2001	SLIGER et al.	
US-6,253,317	06-26-2001	KNAPP, III et al.		
US-6,260,157	07-10-2001	SCHURECHT et al.		
US-6,263,497	07-17-2001	MAEDA et al.		
US-6,282,698	08-28-2001	BAKER et al.		
US-6,282,709	08-28-2001	REHA et al.		
US-6,289,358	09-11-2001	MATTIS et al.		
US-6,289,509	09-11-2001	KRYLOFF		
US-6,292,880	09-18-2001	MATTIS et al.		
US-6,295,644	09-25-2001	HSU et al.		
US-6,298,481	10-02-2001	KOSAKA et al.		
US-6,305,010	10-16-2001	AGARWAL		
US-6,330,574	12-11-2001	MURASHITA		
US-6,330,712	12-11-2001	IWAYA		
US-6,334,213	12-25-2001	LI		
US-6,349,407	02-19-2002	TOWFIQ		
US-6,360,363	03-19-2002	MOSEER et al.		
Examiner Signature			Date Considered	5/27/10

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FIRST
INFORMATION DISCLOSURE
(use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.:	0077898-000001

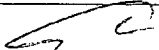
Sheet 4 of 5

Ar	US-6,374,250	04-16-2002	AJTAI et al.	
	US-6,389,592	05-14-2002	AYRES et al.	
	US-6,397,385	05-28-2002	KRAVITZ	
	US-6,404,923	06-11-2002	CHADDA	
	US-6,421,679	07-16-2002	CHANG et al.	
	US-6,425,125	07-23-2002	FRIES et al.	
	US-6,430,685	08-06-2002	YU et al.	
	US-6,434,695	08-13-2002	ESFAHANI et al.	
	US-6,438,621	08-20-2002	KANAMORI et al.	
	US-6,438,748	08-20-2002	GARD et al.	
	US-6,463,582	10-08-2002	LETHIN et al.	
	US-6,466,999	10-15-2002	SLIGER et al.	
	US-6,526,574	02-25-2003	JONES	
	US-6,532,588	03-11-2003	PORTER	
	US-6,594,822	07-15-2003	SCHWEITZ et al.	
	US-6,610,103	08-26-2003	EHRMAN et al.	
	US-6,618,747	09-09-2003	FLYNN et al.	
	US-6,629,123	09-30-2003	HUNT	
	US-6,643,506	11-04-2003	CRISS et al.	
	US-6,643,775	11-04-2003	GRANGER et al.	
	US-6,651,249	11-18-2003	WALDIN et al.	
	US-6,691,305	02-10-2004	HENKEL et al.	
	US-6,738,799	05-18-2004	DICKENSON	
	US-6,757,893	06-29-2004	HAIKIN	
	US-6,760,907	07-06-2004	SHAYLOR	
	US-6,804,663	10-12-2004	DELO	
	US-6,952,823	10-04-2005	KHYLOFF et al.	
	US-7,203,708	04-10-2007	LIU et al.	
	US-7,542,758	06-02-2009	RAJARAM et al.	
	US-7,587,433	09-08-2009	PELEG et al.	

Enter Office that issued the document, by the two-letter code.

FOREIGN PATENT DOCUMENTS

Examiner Initials	Foreign Patent Document		Name of Patentee or Applicant of Cited Document	STATUS							
	Country Code ¹ , Number, Kind Code	Publication Date (MM-DD-YYYY)		Translation	Partial Translation	Eng. Lang. Summary	Search Report	IPER	Abstract	Cited in Spec. Pg. No(s)	
Ar	JP 59-201150	11-14-1984	OMRON TATEISI ELECTRONICS CO.						X		
	JP 63-208941	08-30-1988	FUJITSU LTD.						X		
	EP 0 411 232	02-06-1991	INTERNATIONAL BUSINESS MACHINES CORPORATION								
	JP 05-120067	05-18-1993	NEC SOFTWARE LTD.						X		

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FIRST
INFORMATION DISCLOSURE
(use as many sheets as necessary)

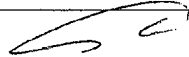
Sheet 5 of 5

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.:	0077898-00000:

FOREIGN PATENT DOCUMENTS

Office	Patent No.	Date	Applicant	1	2	3	4	5	6	7	8	9	10	11	12
AN	EP 0 546 684	06-16-1993	INTERNATIONAL BUSINESS MACHINES CORPORATION												
	JP 06-314194	11-08-1994	NEC CORP.										X		
	EP 0 665 496	08-02-1995	SUN MICROSYSTEMS, INC.												
	EP 0 695 040	01-31-1996	FUJITSU LIMITED												
	EP 0 702 299	03-20-1995	AT&T CORP.												
	WO 96/32679	10-17-1996	HIGHWAYMASTER COMMUNICATIONS, INC.												
	JP 09-069042	03-11-1997	HITACHI LTD.											X	
	WO 97/43711	11-20-1997	ASYMETRIX CORPORATION												
	JP 10-002383	01-06-1998	NEC CORP.											X	
	GB 2 330 428	04-21-1999	WINBOND ELECTRONICS CORPORATION												
	JP 11-126161	05-11-1999	HITACHI LTD.											X	
	WO 99/54816	10-28-1999	SYMANTEC CORPORATION												
	WO 00/34861	06-15-2000	INCERT SOFTWARE CORPORATION												
	EP 1 014 263	06-28-2000	APPLIED MICROSYSTEMS CORPORATION												
	EP 1 087 533	03-28-2001	TADAIRO OHMI												
	EP 0 813 167	10-01-2003	AGFA MONOTYPE CORPORATION												

*Enter Office that issued the document, by the two-letter code.

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

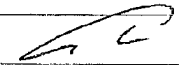
THIRD
INFORMATION DISCLOSURE
(use as many sheets as necessary)

Sheet 1 of 2

In Re Reexamination Application of:	SHARON PELEG
Patent No./issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Naiven/3992
Confirmation No.:	4316
Attorney Docket No.	0077898-000001

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials	Include name of the author: (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.
AP	AJILA, Samuel, "Software Maintenance: An Approach to Impact Analysis of Objects Change", Software - Practice and Experience, October 1995, Vol. 25, pp. 1155-1181, John Wiley & Sons, Ltd., United Kingdom
	AJMANI, Sameer, "A Review of Software Upgrade Techniques for Distributed Systems", MIT Computer Science and Artificial Intelligence Laboratory, August 7, 2002, pp. 1-19, Cambridge, MA
	BAKER, Brenda, "A Program for Identifying Duplicated Code", Computing Science and Statistics, 24th Symposium, March 18-21, 1992, Vol. 24, 9 pages, College Station, TX
	BAKER et al., "Compressing Differences of Executable Code", ACM SIGPLAN Workshop on Compiler Support for System Software (WCSS), April 22, 1999, pp. 1-10, Atlanta, GA
	BARTOLETTI et al., "Secure Software Distribution System", Proceedings of the 20th National Information Systems Security Conference, June 18, 1997, 11 pages, Baltimore, MD
	BASHAR et al., "Low-Threat Security Patches and Tools", Proceedings of the International Conference on Software Maintenance, November 30, 1996, 23 pages, IEEE Computer Society, Washington, DC
	BURNS, Randal C., "Differential Compression: A Generalized Solution for Binary Files", December 1996, pp. 1-69, University of California, Santa Cruz, CA
	BURNS et al., "In Place Reconstruction of Delta Compressed Files", Proceedings of the Seventeenth Annual Symposium on Principles of Distributed Computing, 1998, pp. 267-275, Puerto Vallarta, Mexico
	CHAMBERS et al., "A User Designed Software System for Electron Microprobes - Basic Premises and the Control Program", Journal de Physique, February, 1984, pp. 223-226, Vol. 45, No. 2, Les Editions de Physique, Les Ulis, France
	CHEUNG et al., "Optimal Routing Table Design for IP Address Lookups Under Memory Constraints", IEEE Proceedings of INFOCOM, March, 1999, 8 pages, New York, NY
	CHURCH et al., "Dotplot: a Program for Exploring Self-Similarity in Millions of Lines of Text and Code", June, 1993), Journal of Computational and Graphical Statistics, Vol. 2, No. 2, 12 pages, American Statistical Association, Alexandria, VA
	CIFUENTES et al., "Decompilation of Binary Programs", Software - Practice and Experience, July, 1995, Vol. 25 (7), pp. 811-829, John Wiley & Sons, Ltd., United Kingdom
	CIFUENTES, Cristina, "Reverse Compilation Techniques", Queensland University of Technology, July 1994, pp. 1-342
	CLODFELTER et al., "Technological Profile of Shopping Centers: Present and Future Use", Journal of Shopping Center Research, Spring/Summer 1996, pp. 59-93, International Council of Shopping Centers Educational Foundation, New York, NY
	COOK et al., "A Methodology for Cost-Effective Analysis of In-Place Software Processes", IEEE Transaction on Software Engineering, January 1997, pp. 1-25
	COPPIETERS, Kris, "A Cross-Platform Binary Diff", Dr. Dobb's Journal, May 1, 1995, 5 pages, Miller Freeman, Inc., San Mateo, CA
	DUNTEMANN, Jeff, "Assembly Language: Step-by-Step, pp. 1-469, 1992, John Wiley & Sons, Inc., New York, NY
	EASTWOOD, David B., "Information Technology and Fresh Produce: A Case Study Using Store Level Scan Data to Analyze Sales", July 1997, 20 pages, University of Minnesota, St. Paul, MN
	ENGLER, Dawson R., "VCODE: A Retargetable, Extensible, Very Fast Dynamic Code Generation System", May, 1996, SIGPLAN Conference on Programming Language Design and Implementation, 11 pages, ACM Press, Philadelphia, PA
	FLAKE, Halvar, "Structural Comparison of Executable Objects", Proceedings of the IEEE Conference on Detection of Intrusions and Malware & Vulnerability Assessment, 2004, pp. 161-173, Darmund, Germany
	FRASER et al., "A Machine-Independent Linker", Software - Practice and Experience, Department of Computer Science, The University of Arizona, 1982, vol. 12, pp. 351-366, United Kingdom

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

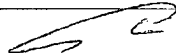
THIRD
INFORMATION DISCLOSURE
 (use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Naiven/3992
Confirmation No.:	4316
Attorney Docket No.	0077898-000001

Sheet 2 of 2

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials	Include name of the author (in CAPITAL LETTERS); title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.
AT	GOSLING et al., "The Java Language Environment", May 1996, pp. 1-98, Sun Microsystems, Mountain View, CA
AT	GRAY, Robert M., "Fundamentals of Data Compression", IEEE Proceedings of International Conference on Information, Communications, and Signal Processing, September 9, 1997, pp. 1-186, Singapore

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

SIXTH
 INFORMATION DISCLOSURE
 (use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.	0077898-000001

Sheet 1 of 1

U.S. PATENT DOCUMENTS

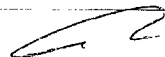
Examiner Initials	Document Number Kind Code	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines Where Relevant Passages or Figures Appear
Ar	6,754,848 B1	06-22-2004	FROELICH et al.	
	6,510,552 B1	01-21-2003	BENAYOUN et al.	
	4,425,618	01-10-1984	BISHOP et al.	
	2007/0255764 A1	11-01-2007	SONNIER et al.	
	2006/0004756 A1	01-05-2006	PELEG et al.	
	2005/0132359 A1	06-16-2005	MCGUIRE et al.	
	2005/0257206 A1	11-17-2005	SEMERDZHIEV	
	2004/0078793 A1	04-22-2004	BRAGULLA et al.	
	2004/0063899 A	07-29-2004	GUSTAFSON et al.	
	2003/0217257 A1	11-20-2003	EBSEN et al.	
	2003/0163805 A1	08-28-2003	HATA et al.	
	2003/0163508 A1	08-28-2003	GOODMAN	
	2002/0100036 A1	07-25-2002	MOSHIR et al.	

FOREIGN PATENT DOCUMENTS

Examiner Initials	Foreign Patent Document Country Code ¹ , Number, Kind Code	Publication Date (MM-DD-YYYY)	Name of Patentee or Applicant of Cited Document	STATUS						
				Translation	Partial Translation	Eng. Lang. Summary	Search Report	IPER	Abstract	Cited in Spec. / Pg. No(s).
Ar	WO 2005/024628 A2	03-17-2005	BITFONE CORP.				X			
	WO 2005/024628 A3	03-17-2005	BITFONE CORP.				X			
	WO 2004/063899 A2	07-29-2004	BITFONE CORP.				X			
	WO 2004/063899 A3	07-29-2004	BITFONE CORP.				X			

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.
Ar	Copy of International Search Report mailed March 25, 2010, International Application No. PCT/IL2009/000762
	Copy of International Search Report mailed February 15, 2010, International Application No. PCT/IL2009/000754
	Copy of International Search Report mailed February 15, 2010, International Application No. PCT/IL2009/000755
	INTERNATIONAL BUSINESS MACHINES CORPORATION, "Method for permitting several versions of a module to coexist within a virtual machine", March 1, 2001, 8 pages, Vol. 443, No. 177, Mason Publications, Hampshire, Great Britain

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

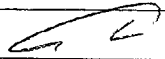
FIFTH
INFORMATION DISCLOSURE
(use as many sheets as necessary)

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.:	0077898-000001

Sheet 1 of 1

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.
AN	MOGUL et al., "Potential benefit of delta encoding and data compression for H1 IP", Proceedings of SIGCOMM '97, September, 1997, pp. 14 pages, ACM, New York, NY
	MUNCH, Bjorn, "Versioning in a Software Engineering Database - the Change Oriented Way", Division of Computer Systems and Telematics, The Norwegian Institute of Technology, September 17, 1993, pp. 1-284, Germany
	MYCROFT et al., "Optimising compilation, Part I: classical imperative languages", Proceedings of SOFSEM, 1992, pp. 1-17, Czechoslovakia
	NILSEN, Kelvin, "Issues in the Design and Implementation of Real-Time Java", July 19, 1996, pp. 1-28, SYS-CON Media, Montvale, NJ
	OPLINGER et al., "Software and Hardware for Exploiting Speculative Parallelism with a Multiprocessor", Stanford University Computer Systems Laboratory, February, 1997, pp. 1-23, Stanford, CA
	OREIZY, Peyman, "Decentralized Software Evolution", Proceedings of the International Conference on the Principles of Software Evolution, April 20-21, 1998, pp. 1-5, ACM, New York, NY
	"PKZIP 6.0 for Windows Getting Started Manual", Copyright 2002 PKWARE, Inc., 13 pages, Milwaukee, WI
	PANDEY et al., "Providing Fine-Grained Access Control for Mobile Programs Through Binary Editing", pp. 1-22, Technical Report TR98-08, 1998, University of California, Davis, CA
	PELLEGRINI et al., "Efficient IP Table Lookup via Adaptive Stratified Trees with Selective Reconstruction", Lecture Notes in Computer Science, September, 2004, pp. 1-26, Springer Berlin, Germany
	ROTH et al., "Database Compression", Sigmod Record, September 1993, Vol. 22, No. 3, pp. 31-39, ACM, New York, NY
	SAHA et al., "Symbolic Support Graph: A Space Efficient Data Structure for Incremental Tabled Evaluation", Department of Computer Science, Lectures Notes in Computer Science, October 10, 2005, Vol. 3668, pp. 1-15, Springer, Germany
	STEVENS, Curt, "Knowledge-Based Assistant for Accessing Large, Poorly Structured Information Spaces", Thesis Submitted to the Faculty of the Graduate School of the University of Colorado, 1993, 125 pages, Boulder, CO
	TICHY, Walter F., "RCS - A System for Version Control", Department of Computer Sciences, Purdue University, 1985, Vol. 15, pp. 1-19, Wiley, United Kingdom
	TICHY, Walter F., "The String-to-String Correction Problem with Block Moves", Purdue University, October 26, 1983, 16 pages, ACM, New York, NY
	TSDU et al., "Client-Server Components and Metadata Objects for Distributed Geographic Information Services", Proceedings of the GIS/LIS, 1998, pp. 590-599, Association of American Geography, Washington, DC
	WANG et al., "BMAT -- A Binary Matching Tool, Microsoft Research Technical Report, November 15, 1999, pp. 1-11, Microsoft Corporation, Redmond, WA
	WOLFE et al., "Executing Compressed Programs on an Embedded RISC Architecture", 1992, pp. 81-91, IEEE, Computer Society Press, Los Alamitos, CA
	ZELLER, Andreas, "Configuration Management with Version Sets", A Unified Software Versioning Model and its Applications, April 1, 1997, pp. 1-320, Technische Universitat Braunschweig, Germany

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

FOURTH
INFORMATION DISCLOSURE
(use as many sheets as necessary)

Sheet 1 of 2

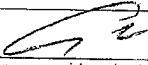
In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Hexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4315
Attorney Docket No.	0077898-000001

FOREIGN PATENT DOCUMENTS

Examiner Initials	Foreign Patent Document		Name of Patentee or Applicant of Cited Document	STATUS							
	Country Code ¹ , Number, Kind Code	Publication Date (MM-DD-YYYY)		Translation	Partial Translation	Eng. Lang. Summary	Search Report	IPER	Abstract	Cited in Soc. / Pg. Nots.	
AP	WO 98/54639	12-03-1981	CIRRUS LOGIC, INC.								

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published
AP	GRYDER et al., "Survey of Data Compression Techniques", Martin Marietta Energy Systems, Inc. for the United States of Energy, September 1991, 45 pages, Oak Ridge, TN
	HARDY et al., "Essence: A Resource Discovery System Based on Semantic File Indexing", Proceedings of the USENIX Winter Conference, January 25-29, 1993, pp. 361-374
	HIRSCHBERG, Daniel S., "Algorithms for the Longest Common Subsequence Problem", Journal of the ACM, October, 1977, Vol. 24, No. 4, pp. 664-675, ACM, New York, NY
	HORWITZ, Susan, "Identifying the Semantic and Textual Differences Between Two Versions of a Program", Proceedings of the ACM SIGPLAN 1990 Conference on Programming Language Design and Implementation, 1990, pp. 234-246, ACM, New York, NY
	HOSHI et al., "Software Updated System Using Wireless Communication", NTT DoCoMo Technical Journal, 2004, Vol. 5, No. 4, pp. 36-43, Japan
	HUNG et al., "UNIX Code Management and Distribution", September 1992, pp. 1-4, Stanford Linear Accelerator Center, Stanford, CA
	HUNT et al., "Delta Algorithms: An Empirical Analysis", ACM Transactions on Software Engineering and Methodology, April, 1998, Vol. 7, No. 2, pp. 192-214, ACM, New York, NY, ACM, New York, NY
	HUNT et al., "An Algorithm for Differential File Comparison", Department of Electrical Engineering, Stanford University, Computing Sciences Technology Report No. 41, June, 1976, pp. 1-9, Bell Laboratories, Murray Hill, NJ
	KELLER et al., "Binary Component Adaptation", Computing Sciences Technology Report No. ECOOP '98 Proceedings, Lecture Notes on Computer Science, Proceedings of the 12th European Conference on Object-Oriented Programming, 1998, Vol. 1445, pp. 1-17
	KIFER, Michael, "Ediff User's Manual", Version 2.70, March 1998, pp. 1-44
	KISHAN et al., "Dynamic Kernel Modification and Extensibility", Technical Report of the SUIF Group, Department of Computer Science, 2002, 26 pages, Stanford University, Stanford, CA
	LARUS et al., "Rewriting Executable Files to Measure Program Behavior", Software, University of Wisconsin - Madison, March 25, 1992, Vol. 24, pp. 1-17, Wiley Interscience, United Kingdom
	LARUS et al., "Rewriting Executable Files to Measure Program Behavior", University of Wisconsin - Madison, October 24, 1994, pp. 1-23
	LEFURGY, Charles Robert, "Efficient Execution of Compressed Programs", Dissertation, 2000, pp. 1-212, University of Michigan, Ann Arbor, MI
	LEFURGY et al., "Improving Code Density Using Compression Techniques", Proceedings of the 30th Annual International Symposium on Microarchitecture, Micro-30, December 1-3, 1997, pp. 10 pages, ACM, New York, NY
	LEFURGY, Charles Robert, "Space-efficient Executable Program Representations for Embedded Microprocessors", Thesis, 1998, pp. 1-48, Ann Arbor, MI

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 605. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

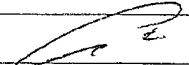
FOURTH
INFORMATION DISCLOSURE
(use as many sheets as necessary)

Sheet 2 of 2

In Re Reexamination Application of:	SHARON PELEG
Patent No./Issued:	6,546,552; April 8, 2003
Reexam Control No.:	90/009,670
Examiner/Group Art Unit:	Andrew L. Nalven/3992
Confirmation No.:	4316
Attorney Docket No.:	0077898-00000:

NON-PATENT LITERATURE DOCUMENTS

Examiner Initials	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.
AL	LEKATSAS et al., "Code Compression for Embedded Systems", Proceedings of the 35th Design Automation Conference, 1998, June 1998, 6 pages, San Francisco, CA
	LYNCH et al., "PATCH Graphs: an Efficient Data Structure for Completion of Finitely Presented Groups", Proceedings of the 3rd International AISMC Conference, Lecture Notes in Computer Science, 1996, vol. 1138, 15 pages, Springer Verlag, United Kingdom
	MacDONALD et al., "PRCS: The Project Revision Control System", Lecture Notes in Computer Science, 1998, Vol. 1439, 14 pages, University of California at Berkeley, National Energy Research Scientific Computing Center, Springer Verlag, United Kingdom
	MacKENZIE et al., "Comparing and Merging Files", diff, diff3, sdiff, cmp, and patch, Edition 2.8, for diffutils 2.8 and patch 2.5.4, March 23, 2002, pp. 1-100, Free Software foundation Ltd., Boston, MA
	MICROSOFT PressPass, News Press Release, May 14, 1997, 2 pages, Microsoft Corp., Redmond, WA 2 pages
	MILLER et al., "A File Comparison Program", Software - Practice and Experience, November 1985, Vol. 15 (11), pp. 1025-1040, Wiley Interscience, United Kingdom

Examiner Signature: 	Date Considered: 5/27/10
---	--------------------------

*EXAMINER Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 509. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.

Substitute for form 1449/PTO & 1449B/PTO Complete if Known

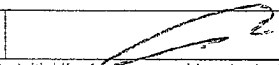
SECOND INFORMATION DISCLOSURE (use as many sheets as necessary)	in Re Reexamination Application of:	SHARON PELEG
	Patent No./Issued:	6,546,552; April 8, 2003
	Reexam Control No.:	90/009,670
	Examiner/Group Art Unit:	Andrew L. Nalven/3992
	Confirmation No.:	4316
Attorney Docket No.:		0077898-000001

Sheet 1 of 1

U.S. PATENT DOCUMENTS				
Examiner Initials	Document Number-Kind Code	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines Where Relevant Passages or Figures Appear
AN	US-5,043,871	06-27-1991	NISHIGAKI et al.	
	US-5,247,660	09-21-1993	ASHCRAFT et al.	
	US-5,534,052	05-27-1997	MORRIS	
	US-5,805,899	09-08-1998	EVANS et al.	
	US-5,813,017	09-22-1998	MORRIS	
	US-6,018,747	01-25-2000	BURNS et al.	
	US-6,317,754	11-13-2001	PENG	
	US-6,374,250	04-16-2000	AJTAI et al.	
	US-2002/0087500	07-04-2002	BERKOWITZ et al.	

FOREIGN PATENT DOCUMENTS											
Examiner Initials	Foreign Patent Document		Publication Date (MM-DD-YYYY)	Name of Patentee or Applicant of Cited Document	STATUS						
	Country Code ¹ , Number, Kind Code				Translation	Partial Translation	Eng. Lang. Summary	Office Action	IPER	Abstract	Cited in Spec. / Pg. No(s).
AN	JP 63-208941 A		08-30-1988	FUJITSU LTD.				X			
	EP 0 752 794 A3		12-29-1999	FUJITSU LIMITED				X			
	EP 0 723 226 B1		05-30-2001	HEWLETT-PACKARD COMPANY				X			

NON-PATENT LITERATURE DOCUMENTS	
Examiner Initials	Include name of the author (in CAPITAL LETTERS), title of the article (when appropriate), title of the item (book, magazine, journal, serial, symposium, catalog, etc.), date, page(s), volume-issue number(s), publisher, city and/or country where published.
AN	Copy of Japanese Office Action dated November 28, 2006, with translation
↓	HOULDER et al., "XOR Commands on SCSI Disk Drives", Seagate Technology, pp. 1-30
	HUNT et al., "An Empirical Study of Delta Algorithms", University of Karlsruhe and AT&T Research, pp. 49-65
	IBM Corp., "Logging Status Information While Installing New Software", June 1991, Vol. 34, No. 1, one page

Examiner Signature		Date Considered	5/27/10
--------------------	---	-----------------	---------

*EXAMINER Initial if reference considered, whether or not citation is in conformance with M.P.E.P. § 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to Applicant.