

EXHIBIT O

Claim Chart from Re-exam Petition

Anticipation Based On The Wetmore '713 Patent

Set 2: Server-Side Independent Claims 8, 21, 42, And 55:

Claim Element	Claim Element	The Wetmore '713 Patent
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:	21. A system 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 system comprising a processing device capable of:	As discussed in more detail in each step below, the Wetmore '713 Patent generates a compact difference result (vector patch resource) between an old executable program (old non-vectorized program) and a new executable program (new non-vectorized program). Each program (old and new non-vectorized programs) including reference entries ("jump" entry points) that contain reference that refer to other entries in the program.
(a) generating a modified old program utilizing at least said old program;	(a) generating a modified old program utilizing at least said old program;	During vectorization, the Wetmore '713 Patent generates a modified old program (vectorized old program) utilizing at least the old program (the old non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17, 18-31; FIG. 4.)
(b) generating a modified new program utilizing at least said new program,	(b) generating a modified new program utilizing at least said new program,	During vectorization, the Wetmore '713 Patent generates a modified new program (vectorized new program) utilizing at least the new program (the new non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17; col.10 ll.6-14, col.11 ll.2-12; FIG. 4.)
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	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	The modified old program (vectorized old program) and modified new program (vectorized new program) have at least the following characteristics: Substantially each reference (the location in the ROM code in the jump entry point) in an entry (in each "jump" entry point) in the old program (the old non-vectorized program) that is different than corresponding entry in said new program (the new non-vectorized program) due to delete/insert modifications that form part of the transition between the old program and new program (<i>e.g.</i> , routines in the Wetmore '713 Patent can be deleted or inserted), are reflected as invariant references (table pointers with an offset) in the corresponding entries in the modified old

Claim Element	Claim Element	The Wetmore '713 Patent
<p>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;</p>	<p>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;</p>	<p>and modified new programs. (Wetmore '713 Patent col.5 ll.18-56; col.6 l.45 – col.8 l.52; FIGS. 3-5.)</p> <p>Invariant references: Invariant references are discussed in the '552 Patent and an example is given of providing <i>invariant references</i> by generating modified old and new programs wherein the address references in entries are replaced by label marks. Namely, the '552 Patent states that: "the invention aims at generating a modified old program and a modified new program, wherein the difference in references in corresponding entries in said new and old programs as explained above, will be reflected as <i>invariant entries</i> in the modified old and new programs. The net effect is that the <i>invariant reference entries</i> (between the modified old program and the modified new program), will not appear in the difference result, thereby reducing its size as compared to a conventional difference result obtained by using hitherto known techniques." ('552 Patent. col.3 ll.36-46 (emphasis added)). The '552 Patent also states: "Those versed in the art will readily appreciate that according to the invention, it is desired to neutralize this change, since it has occurred solely due to the fact that other entries have been affected (<i>i.e.</i> entries 7 to 9). It is accordingly an object of the invention to give rise to a situation where modifications of this kind will be modified to <i>invariant references</i> with the obvious consequence that they are not reflected in the difference result, thereby keeping the latter relatively compact." ('552 Patent col.10 ll.7-15 (emphasis added).) Further, the '552 Patent notes that for the particular embodiment of FIGS. 1 and 2, "the desired <i>invariant references</i> are accomplished by generating modified old and new programs wherein address references in entries are replaced by label marks as follows" ('552 Patent col.10 ll.47-50 (emphasis added).) The Patentee also provided its claim construction for "invariant references" as "References that are the same. <i>See, e.g.</i>, '552 Patent 10:10-15." (Edwards Decl. Exh. A.)</p> <p>Thus, the vectorization process in the Wetmore '713 Patent that results in vectorized programs having table pointers with an offset are "invariant references" under the '552 Patent's label marks example and also under the Patent Owner's construction, since the table pointers with offsets will be "the same" in both the vectorized old program and vectorized new program.</p>
<p>(c) generating said compact difference result utilizing at least said modified new program and modified old</p>	<p>(c) generating said compact difference result utilizing at least said modified new program and</p>	<p>The Wetmore '713 Patent generates the compact difference result (the vector patch resource) utilizing at least the modified new program (vectorized new program) and modified old program (vectorized old program). (Wetmore '713 Patent col.10 l.62—col.11 l.12.)</p>

Claim Element	Claim Element	The Wetmore '713 Patent
program.	modified old program.	

Claim Element	Claim Element	The Wetmore '713 Patent
42. A method for generating a compact difference result between an old data table and a new data table; each data table including reference entries that contain reference that refer to other entries in the data table; the method comprising the steps of:	55. A system for generating a compact difference result between an old data table and a new data table; each data table including reference entries that contain reference that refer to other entries in the data table; the system comprising a processing device capable of:	<p>As discussed in more detail in each step below, the Wetmore '713 Patent generates a compact difference result (vector patch resource) between an old data table (old non-vectorized program) and a new data table (new non-vectorized program). Each data table (old and new non-vectorized programs) includes reference entries ("jump" entry points) that contain references that refer to other entries in the data table.</p> <p>Data table: The '552 Patent defines a "data table" as "a table of entries, each may have a different size." ('552 Patent col.2 ll.33-34.) The '552 Patent also explains that "a data table can be an executable program either as a loaded program in machine-memory or as an executable-file. In this example, entries are individual machine instructions of the program or the individual data elements used by the program." ('552 Patent col.2 ll.61-65.) The Patentee also provided its claim construction for "data table" as "a table of entries, where an entry is an addressable unit within the data table. An executable program is one example of a data table. <i>See, e.g.,</i> '552 Patent 2:33-36; 2:61-63." (Edwards Decl. Exh. A.) Thus, the ROM based code of the Wetmore '713 Patent qualifies as a data table.</p> <p>Reference entry: The '552 Patent states in the Glossary that "Entries that include references are designated also as reference entries." ('552 Patent col.2 ll.46-47.) The Patentee also provided its claim construction for "reference entry" as "An addressable unit containing data that includes a reference. <i>See, e.g.,</i> '552 Patent 2:35-36; 2:46-47." (Edwards Decl. Exh. A.) In Wetmore, the entry points (304) may be "jump[s] to a label somewhere else in the ROM code." (Wetmore '713 Patent col.5 ll.38-39.) Under the '552 Patent's definition of "reference entries," the Wetmore '713 Patent's "jump" entry points (reference entries) include references to labels somewhere else in the ROM code.</p> <p>Reference: A "reference" is defined in the '552 Patent to be "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." ('552 Patent col.2 ll.42-45.) The Patentee also provided the identical claim construction for</p>

Claim Element	Claim Element	The Wetmore '713 Patent
		<p>"reference" as "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. <i>See, e.g., '552 Patent 2:42-45.</i>" (Edwards Decl. Exh. A.) Under this definition, in the Wetmore '713 Patent, all of the jump entry points in the old non-vectorized program would be all of the references.</p> <p>Entry: The '552 Patent states in the Glossary that "a data table includes entries, each of which is an addressable unit that contains data." ('552 Patent col.2 ll.35-36.)</p>
(a) generating a modified old data table utilizing at least said old data table;	(a) generating a modified old data table utilizing at least said old data table;	During vectorization, the Wetmore '713 Patent generates a modified old data table (vectorized old program) utilizing at least the old data table (the old non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17, 18-31; FIG. 4.)
(b) generating a modified new data table utilizing at least said new data table,	(b) generating a modified new data table utilizing at least said new data table,	During vectorization, the Wetmore '713 Patent generates a modified new data table (vectorized new program) utilizing at least the new data table (the new non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17; col.10 ll.6-14, col.11 ll.2-12; FIG. 4.)
<p>said modified old data table and modified new data table have at least the following characteristics:</p> <p>(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;</p>	<p>said modified old data table and modified new data table have at least the following characteristics:</p> <p>(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;</p>	<p>The modified old data table (vectorized old program) and modified new data table (vectorized new program) have at least the following characteristics:</p> <p>Substantially each reference (the location in the ROM code in the jump entry point) in an entry (in each "jump" entry point) in the old data table (the old non-vectorized program) that is different than corresponding entry in said new data table (the new non-vectorized program) due to delete/insert modifications that form part of the transition between the old data table and new data table (<i>e.g., routines in the Wetmore '713 Patent can be deleted or inserted</i>), are reflected as invariant references (table pointers with an offset) in the corresponding entries in the modified old and modified new data tables. (Wetmore '713 Patent col.5 ll.18-56; col.6 l.45 – col.8 l.52; FIGS. 3-5.)</p> <p>Invariant references: Invariant references are discussed in the '552 Patent and an example is given providing <i>invariant references</i> by generating modified old and new programs wherein the address references in entries are replaced by label marks. Namely, the '552 Patent states that: "the invention aims at generating a modified old program and a modified new program, wherein the difference in references in</p>

Claim Element	Claim Element	The Wetmore '713 Patent
		<p>corresponding entries in said new and old programs as explained above, will be reflected as <i>invariant entries</i> in the modified old and new programs. The net effect is that the <i>invariant reference entries</i> (between the modified old program and the modified new program), will not appear in the difference result, thereby reducing its size as compared to a conventional difference result obtained by using hitherto known techniques." ('552 Patent. col.3 ll.36-46 (emphasis added)). The '552 Patent also states: "Those versed in the art will readily appreciate that according to the invention, it is desired to neutralize this change, since it has occurred solely due to the fact that other entries have been affected (<i>i.e.</i> entries 7 to 9). It is accordingly an object of the invention to give rise to a situation where modifications of this kind will be modified to <i>invariant references</i> with the obvious consequence that they are not reflected in the difference result, thereby keeping the latter relatively compact." ('552 Patent. col.10 ll.7-15.) Further, the '552 Patent notes that for the particular embodiment of FIGS. 1 and 2, "the desired <i>invariant references</i> are accomplished by generating modified old and new programs wherein address references in entries are replaced by label marks as follows" ('552 Patent. col.10 ll.47-50.) The Patentee also provided its claim construction for "invariant references" as "References that are the same. <i>See, e.g.</i>, '552 Patent 10:10-15." (Edwards Decl. Exh. A.)</p> <p>Thus, the vectorization process in the Wetmore '713 Patent that results in vectorized programs (data table) having table pointers with offsets are "invariant references" under the '552 Patent's label marks example and also under the Patent Owner's construction, since the table pointers with offset will be "the same" in both the vectorized old program and vectorized new program.</p>
(c) generating said compact difference result utilizing at least said modified new data table and modified old data table.	(c) generating said compact difference result utilizing at least said modified new data table and modified old data table.	<p>The Wetmore '713 Patent generates the compact difference result (the vector patch resource) utilizing at least the modified new data table (vectorized new program) and modified old data table (vectorized old program). (Wetmore '713 Patent col.10 l.62—col.11 l.12.)</p>

Obviousness Based On The Wetmore '713 Patent Alone

Set 2: Server-Side Independent Claims 8, 21, 42, And 55:

Claim Element	Claim Element	The Wetmore '713 Patent
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:	21. A system 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 system comprising a processing device capable of:	As discussed in more detail in each step below, the Wetmore '713 Patent generates a compact difference result (vector patch resource) between an old executable program (old non-vectorized program) and a new executable program (new non-vectorized program). Each program (old and new non-vectorized programs) includes reference entries ("jump" entry points) that contain references that refer to other entries in the program.
(a) generating a modified old program utilizing at least said old program;	(a) generating a modified old program utilizing at least said old program;	During vectorization, the Wetmore '713 Patent generates a modified old program (vectorized old program) utilizing at least the old program (the old non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17, 18-31; FIG. 4.)
(b) generating a modified new program utilizing at least said new program,	(b) generating a modified new program utilizing at least said new program,	During vectorization, the Wetmore '713 Patent generates a modified new program (vectorized new program) utilizing at least the new program (the new non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17; col.10 ll.6-14, col.11 ll.2-12; FIG. 4.)
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 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	The modified old program (vectorized old program) and modified new program (vectorized new program) have at least the following characteristics: To the extent that it is asserted or determined that the Wetmore '713 Patent does not disclose reflecting "substantially each reference in an entry" as invariant references, it would have been obvious to one of ordinary skill in the art at the time of the alleged invention of the '552 Patent to have done so. In the Wetmore '713 Patent, when the entry points (304) are jumps to labels somewhere else in the ROM code, they will be replaced by references to a table (table pointers) in RAM with an offset. However, reference to entry points (303), which themselves point to entry points (304), are not expressly disclosed as

Claim Element	Claim Element	The Wetmore '713 Patent
<p>said old program and new program are reflected as invariant references in the corresponding entries in said modified old and modified new programs;</p>	<p>said old program and new program are reflected as invariant references in the corresponding entries in said modified old and modified new programs;</p>	<p>being replaced or modified because there is no need to change reference to entry points (303) as they would not change when updating the ROM program and thus do not need to be accounted for as would normal lines of code with internal references to other lines of code. Rather, any code that may need to be changed is already moved out into RAM and referenced by the entry point (304) replaced by the table pointer and offset. These reference to entry points (303) are therefore akin to invariant references as discussed in the '552 Patent wherein the "net effect is that the invariant reference entries ... will not appear in the difference result." ('552 Patent col.3 ll.40-43.)</p> <p>Moreover, the Wetmore '713 Patent also discloses "modification of existing vector table entries (replacing a routine), adding new vector table entries (adding new routines to an existing function) or adding a new vector table (adding new routines for new functions). (Wetmore '713 Patent col. 3, lines 9-13; FIGS.7A, 7B). Thus, the Wetmore '713 Patent's handling of vectorization and replacement of references is all done with respect to insert/delete type modifications like the '552 Patent, and is analogous to only replacing substantially all references when there are insert/delete modifications like in the '552 Patent.</p> <p>Thus, in view of the Wetmore '713 Patent, it would have been obvious for substantially each reference (the location in the ROM code in the jump entry point) in an entry (in each "jump" entry point) in the old program (the old non-vectorized program) that is different than corresponding entry in said new program (the new non-vectorized program) due to delete/insert modifications that form part of the transition between the old program and new program (e.g., routines in the Wetmore '713 Patent can be deleted or inserted), to be reflected as invariant references (table pointers with an offset) in the corresponding entries in the modified old and modified new programs. (Wetmore '713 Patent col.5 ll.18-56; col.6 l.45 – col.8 l.52; FIGS. 3-5.)</p> <p>Invariant references: Invariant references are discussed in the '552 Patent and an example is given of providing <i>invariant references</i> by generating modified old and new programs wherein the address references in entries are replaced by label marks. Namely, the '552 Patent states that: "the invention aims at generating a modified old program and a modified new program, wherein the difference in references in corresponding entries in said new and old programs as explained above, will be reflected as <i>invariant entries</i> in the modified old and new programs. The net effect is that the <i>invariant reference entries</i> (between the modified old program and the modified new program),</p>

Claim Element	Claim Element	The Wetmore '713 Patent
		<p>will not appear in the difference result, thereby reducing its size as compared to a conventional difference result obtained by using hitherto known techniques." ('552 Patent. col.3 ll.36-46 (emphasis added)). The '552 Patent also states: "Those versed in the art will readily appreciate that according to the invention, it is desired to neutralize this change, since it has occurred solely due to the fact that other entries have been affected (<i>i.e.</i> entries 7 to 9). It is accordingly an object of the invention to give rise to a situation where modifications of this kind will be modified to <i>invariant references</i> with the obvious consequence that they are not reflected in the difference result, thereby keeping the latter relatively compact." ('552 Patent col.10 ll.7-15 (emphasis added).) Further, the '552 Patent notes that for the particular embodiment of FIGS. 1 and 2, "the desired <i>invariant references</i> are accomplished by generating modified old and new programs wherein address references in entries are replaced by label marks as follows" ('552 Patent col.10 ll.47-50 (emphasis added).) The Patentee also provided its claim construction for "invariant references" as "References that are the same. <i>See, e.g.</i>, '552 Patent 10:10-15." (Edwards Decl. Exh. A.)</p> <p>Thus, the vectorization process in the Wetmore '713 Patent that results in vectorized programs having table pointers with an offset are "invariant references" under the '552 Patent's label marks example and also under the Patent Owner's construction, since the table pointers with offsets will be "the same" in both the vectorized old program and vectorized new program.</p>
(c) generating said compact difference result utilizing at least said modified new program and modified old program.	(c) generating said compact difference result utilizing at least said modified new program and modified old program.	<p>The Wetmore '713 Patent generates the compact difference result (the vector patch resource) utilizing at least the modified new program (vectorized new program) and modified old program (vectorized old program). (Wetmore '713 Patent col.10 l.62—col.11 l.12.)</p>

Claim Element	Claim Element	The Wetmore '713 Patent
42. A method for generating a compact difference result between an old data table and a new data table; each data table including reference entries that contain	55. A system for generating a compact difference result between an old data table and a new data table; each data table including reference entries that contain	<p>As discussed in more detail in each step below, the Wetmore '713 Patent generates a compact difference result (vector patch resource) between an old data table (old non-vectorized program) and a new data table (new non-vectorized program). Each data table (old and new non-vectorized programs) includes reference entries ("jump" entry points) that contain references that refer to other entries in the data table.</p> <p>Data table: The '552 Patent defines a "data</p>

Claim Element	Claim Element	The Wetmore '713 Patent
reference that refer to other entries in the data table; the method comprising the steps of:	reference that refer to other entries in the data table; the system comprising a processing device capable of:	<p>table" as "a table of entries, each may have a different size." ('552 Patent col.2 ll.33-34.) The '552 Patent also explains that "a data table can be an executable program either as a loaded program in machine-memory or as an executable-file. In this example, entries are individual machine instructions of the program or the individual data elements used by the program." ('552 Patent col.2 ll.61-65.) The Patentee also provided its claim construction for "data table" as "a table of entries, where an entry is an addressable unit within the data table. An executable program is one example of a data table. <i>See, e.g.,</i> '552 Patent 2:33-36; 2:61-63." (Edwards Decl. Exh. A.) Thus, the ROM based code of the Wetmore '713 Patent qualifies as a data table.</p> <p>Reference entry: The '552 Patent states in the Glossary that "Entries that include references are designated also as reference entries." ('552 Patent col.2 ll.46-47.) The Patentee also provided its claim construction for "reference entry" as "An addressable unit containing data that includes a reference. <i>See, e.g.,</i> '552 Patent 2:35-36; 2:46-47." (Edwards Decl. Exh. A.) In Wetmore, the entry points (304) may be "jump[s] to a label somewhere else in the ROM code." (Wetmore '713 Patent col.5 ll.38-39.) Under the '552 Patent's definition of "reference entries," the Wetmore '713 Patent's "jump" entry points (reference entries) include references to labels somewhere else in the ROM code.</p> <p>Reference: A "reference" is defined in the '552 Patent to be "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." ('552 Patent col.2 ll.42-45.) The Patentee also provided the identical claim construction for "reference" as "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. <i>See, e.g.,</i> '552 Patent 2:42-45." (Edwards Decl. Exh. A.) Under this definition, in the Wetmore '713 Patent, all of the jump entry points in the old non-vectorized program would be all of the references.</p> <p>Entry: The '552 Patent states in the Glossary that "a data table includes entries, each of which is an addressable unit that contains data." ('552 Patent col.2 ll.35-36.)</p>
(a) generating a modified old data table utilizing at	(a) generating a modified old data table utilizing at	During vectorization, the Wetmore '713 Patent generates a modified old data table (vectorized old program) utilizing at least the old data table (the old

Claim Element	Claim Element	The Wetmore '713 Patent
least said old data table;	least said old data table;	non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17, 18-31; FIG. 4.)
(b) generating a modified new data table utilizing at least said new data table,	(b) generating a modified new data table utilizing at least said new data table,	During vectorization, the Wetmore '713 Patent generates a modified new data table (vectorized new program) utilizing at least the new data table (the new non-vectorized program). (Wetmore '713 Patent col.4 ll.38-39; col.5 ll.1-3, 10-17; col.10 ll.6-14, col.11 ll.2-12; FIG. 4.)
<p>said modified old data table and modified new data table have at least the following characteristics:</p> <p>(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;</p>	<p>said modified old data table and modified new data table have at least the following characteristics:</p> <p>(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;</p>	<p>The modified old data table (vectorized old program) and modified new data table (vectorized new program) have at least the following characteristics:</p> <p>To the extent that it is asserted or determined that the Wetmore '713 Patent does not disclose reflecting "substantially each reference in an entry" as invariant references, it would have been obvious to one of ordinary skill in the art at the time of the alleged invention of the '552 Patent to have done so.</p> <p>In the Wetmore '713 Patent, when the entry points (304) are jumps to labels somewhere else in the ROM code, they will be replaced by references to a table (table pointers) in RAM with an offset. However, reference to entry points (303), which themselves point to entry points (304), are not expressly disclosed as being replaced or modified because there is no need to change reference to entry points (303) as they would not change when updating the ROM program and thus do not need to be accounted for as would normal lines of code with internal references to other lines of code. Rather, any code that may need to be changed is already moved out into RAM and referenced by the entry point (304) replaced by the table pointer and offset. These reference to entry points (303) are therefore akin to invariant references as discussed in the '552 Patent wherein the "net effect is that the invariant reference entries ... will not appear in the difference result." ('552 Patent col.3 ll.40 43.)</p> <p>Moreover, the Wetmore '713 Patent also discloses "modification of existing vector table entries (replacing a routine), adding new vector table entries (adding new routines to an existing function) or adding a new vector table (adding new routines for new functions). (Wetmore '713 Patent col. 3, lines 9 13; FIGS.7A, 7B). Thus, the Wetmore '713 Patent's handling of vectorization and replacement of references is all done with respect to insert/delete type modifications like the '552 Patent, and is analogous to only replacing substantially all references when there are insert/delete modifications like in the '552 Patent.</p> <p>Thus, in view of the Wetmore '713 Patent, it</p>

Claim Element	Claim Element	The Wetmore '713 Patent
		<p>would have been obvious for substantially each reference (the location in the ROM code in the jump entry point) in an entry (in each "jump" entry point) in the old program (the old non-vectorized program) that is different than corresponding entry in said new data table (the new non-vectorized program) due to delete/insert modifications that form part of the transition between the old data table and new data table (e.g., routines in the Wetmore '713 Patent can be deleted or inserted), to be reflected as invariant references (table pointers with an offset) in the corresponding entries in the modified old and modified new data tables. (Wetmore '713 Patent col.5 ll.18-56; col.6 l.45 – col.8 l.52; FIGS. 3-5.)</p> <p>Invariant references: Invariant references are discussed in the '552 Patent and an example is given providing <i>invariant references</i> by generating modified old and new programs wherein the address references in entries are replaced by label marks. Namely, the '552 Patent states that: "the invention aims at generating a modified old program and a modified new program, wherein the difference in references in corresponding entries in said new and old programs as explained above, will be reflected as <i>invariant entries</i> in the modified old and new programs. The net effect is that the <i>invariant reference entries</i> (between the modified old program and the modified new program), will not appear in the difference result, thereby reducing its size as compared to a conventional difference result obtained by using hitherto known techniques." ('552 Patent. col.3 ll.36-46 (emphasis added)). The '552 Patent also states: "Those versed in the art will readily appreciate that according to the invention, it is desired to neutralize this change, since it has occurred solely due to the fact that other entries have been affected (<i>i.e.</i> entries 7 to 9). It is accordingly an object of the invention to give rise to a situation where modifications of this kind will be modified to <i>invariant references</i> with the obvious consequence that they are not reflected in the difference result, thereby keeping the latter relatively compact." ('552 Patent. col.10 ll.7-15.) Further, the '552 Patent notes that for the particular embodiment of FIGS. 1 and 2, "the desired <i>invariant references</i> are accomplished by generating modified old and new programs wherein address references in entries are replaced by label marks as follows" ('552 Patent. col.10 ll.47-50.) The Patentee also provided its claim construction for "invariant references" as "References that are the same. <i>See, e.g.</i>, '552 Patent 10:10-15." (Edwards Decl. Exh. A.)</p> <p>Thus, the vectorization process in the Wetmore</p>

Claim Element	Claim Element	The Wetmore '713 Patent
		'713 Patent that results in vectorized programs (data table) having table pointers with offsets are "invariant references" under the '552 Patent's label marks example and also under the Patent Owner's construction, since the table pointers with offset will be "the same" in both the vectorized old program and vectorized new program.
(c) generating said compact difference result utilizing at least said modified new data table and modified old data table.	(c) generating said compact difference result utilizing at least said modified new data table and modified old data table.	The Wetmore '713 Patent generates the compact difference result (the vector patch resource) utilizing at least the modified new data table (vectorized new program) and modified old data table (vectorized old program). (Wetmore '713 Patent col.10 l.62—col.11 l.12.)