Exhibit A

Non-Infringement of U.S. Patent No. 6,546,552
Asserted Claims: 8-12, 21-25, 28-34, 42-46, 55-60, and 62-67

The chart below demonstrates limitations of the asserted claims that are not met by the accused product, Courgette, based on Google's analysis to date, its current understanding of the claim language, and its current understanding of Red Bend's infringement contentions. This chart does not constitute a waiver of any argument arising under 35 U.S.C. §§ 101, 102, 103 or 112. A blank cell does not represent an admission by Google that the claim limitation is met by Courgette.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
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:	An unspecified portion of Courgette code; program processed by Courgette.	Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
8(a)	(a) generating a modified old program utilizing at least said old program;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
8(b)	(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:	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
8(b)(i)	(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;	An unspecified portion of Courgette code.	Courgette does not identify or process "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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result. Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			"modified old program."
8(c)	(c) generating said compact difference result utilizing at least said modified new program and modified old program.	An unspecified portion of Courgette code.	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed "compact difference result," using symbolic data rather than using the executable "modified old program" and "modified new program" as required by the claim. Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
9	The method of claim 8, further comprising the step of: (d) transmitting said compact difference result over a communication	An unspecified portion of Courgette code; Google's Update Process responsible for transmitting updates generated by Courgette.	Courgette does not infringe claim 8 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Courgette does not generate, receive, transmit, store or use a "compact

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	network.		difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
10	The method of claim 9, wherein said network includes the Internet.	An unspecified portion of Courgette code; Google's Update Process responsible for transmitting updates generated by Courgette over the Internet.	Courgette does not infringe claim 9 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
11	The method of claim 8, further comprising the step of: (d) storing said compact difference result on a storage medium.	An unspecified portion of Courgette code "that write an update package to a storage medium."	Courgette does not infringe claim 8 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
12	A method for performing an update in an old executable program so as to generate 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:	An unspecified portion of Courgette code; program processed by Courgette.	
12(a)	(a) receiving data that includes a compact difference result; said	An unspecified portion of Google code running on a users' computing device that	Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	compact difference result was generated utilizing a modified old program and a modified new program;	receives update package generated by Courgette.	references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed "compact difference result," using symbolic data rather than using the executable "modified old program" and "modified new program" as required by the claim. Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program." Google does not create a difference result that is used to update copies of the Chrome web browser using a modified old version of the Chrome web browser software and a modified new version of the Chrome web browser software.
12(b)	(b) generating a modified old program utilizing at least said old program;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program,"

Claim	‘552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			“modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed “modified new program” or the claimed “modified old program.” Google does not update copies of the Chrome web browser by generating a “modified old program” on the remote client computer.
12(c)	(c) reconstituting a modified new program utilizing directly or indirectly at least said modified old program and said compact difference result; said modified old program and modified new program have at least the following characteristics:	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed “modified old program,” or the claimed “modified new program.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed “modified new program” or the claimed “modified old program.” Courgette does not generate, receive, transmit, store or use the “compact difference result” claimed by the ‘552 patent. The ‘552

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting a modified new program utilizing directly or indirectly at least said modified old program and said compact difference result" on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new program" or the "modified old program" of the claims, and even if they were, neither are "reconstitute[ed]" on the remote client computer.
12(c)(i)	(i) substantially each reference in an entry in said old program that is different than corresponding entry in said new program due to delete/inset 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;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program." Courgette does not identify or process "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." Although

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result.
12(d)	(d) reconstituting said new program utilizing directly or indirectly at least said compact difference result and	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program,"

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	said modified new program.		"modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program." Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting said new program utilizing directly or indirectly at least said compact difference result and said modified new program" on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new program" or the "modified old program" of the claims. Even if they were, the "modified new program" still would not be present on the remote client computer, and is not used to "reconstitute[e]" an updated version of the Chrome web browser. Moreover, even if Courgette utilized a difference result to "reconstitute" a "modified new program" as required by step 12(c), it would not utilize the difference result to "reconstitute" a new program as required by step 12(d).
21	A system for generating a	An unspecified portion of	Courgette does not generate, receive, transmit, store or use the

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	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:	Courgette code; computer or device capable of executing Courgette; program processed by Courgette; Computer or device capable or executing Courgette.	"compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
21(a)	(a) generating a modified old program utilizing at least said old program;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
21(b)	(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:	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
21(b)(i)	(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;	An unspecified portion of Courgette code.	Courgette does not identify or process "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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result. Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
21(c)	(c) generating said compact difference result utilizing at least said modified new program and modified old program.	An unspecified portion of Courgette code.	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed "compact difference result," using symbolic data rather than using the executable "modified old program" and "modified new program" as

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			required by the claim. Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."
22	The system of claim 21, wherein said processor is further capable of transmitting said compact difference result over a communication network.	An unspecified portion of Courgette code; computer or device capable of communication over a network.	Courgette does not infringe claim 21 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
23	The system of claim 22, wherein said network includes the Internet.	An unspecified portion of Courgette code; computer or device capable of communication over the Internet.	Courgette does not infringe claim 22 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
24	The system of claim 21,	An unspecified portion of	Courgette does not infringe claim 21 for at least the reasons stated in

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	wherein said processor is further capable of storing said compact difference result on a storage medium.	Courgette code; computer or device capable of storing data to a storage medium.	these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
25	A system for performing an update in an old executable program so as to generate 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:	Courgette; computer or processing device capable of executing Courgette; program processed by Courgette.	
25(a)	(a) receiving data that includes a compact difference result; said compact difference result was generated utilizing a modified old program and a modified new program;	An unspecified portion of Google code running on a users' computing device that receives update package generated by Courgette.	Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed "compact difference result," using symbolic data rather than using the executable "modified old program" and "modified new program" as required by the claim. Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program,"

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			"old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program." Google does not create a difference result that is used to update copies of the Chrome web browser using a modified old version of the Chrome web browser software and a modified new version of the Chrome web browser software.
25(b)	(b) generating a modified old program utilizing at least said old program;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."

Claim	‘552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Google does not update copies of the Chrome web browser by generating a “modified old program” on the remote client computer.
25(c)	(c) reconstituting a modified new program utilizing directly or indirectly at least said modified old program and said compact difference result; said modified old program and modified new programs have at least the following characteristics:	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed “modified old program,” or the claimed “modified new program.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed “modified new program” or the claimed “modified old program.” Courgette does not generate, receive, transmit, store or use the “compact difference result” claimed by the ‘552 patent. The ‘552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by “reconstituting a modified new program utilizing directly or indirectly at least said modified old program and said compact difference result” on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the “modified new program” or the “modified old program” of the claims, and even if they

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			were, neither are "reconstitute[ed]" on the remote client computer.
25(c)(i)	(i) substantially each reference in an entry in said old program that is different than corresponding entry in said new program due to delete/inset 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;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program." Courgette does not identify or process "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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result.
25(d)	(d) reconstituting said new program utilizing directly or indirectly at least said compact difference result and said modified new program.	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old program," or the claimed "modified new program." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new program" or the claimed "modified old program."

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting said new program utilizing directly or indirectly at least said compact difference result and said modified new program" on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new program" or the "modified old program" of the claims. Even if they were, the "modified new program" still would not be present on the remote client computer, and is not used to "reconstitute[e]" an updated version of the Chrome web browser. Moreover, even if Courgette utilized a difference result to "reconstitute" a "modified new program" as required by step 25(c), it would not utilize the difference result to "reconstitute" a new program as required by step 25(d).
28	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.	Computer or other processing device; computer readable storage medium associated with the computer or processing device; update generated by Courgette.	Courgette does not infringe claims 8-11 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
29	The method of claim 12, wherein said data is received in step (a) from a remote site	An unspecified portion of Courgette code; Courgette generated update received	Courgette does not infringe claim 12 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	through a network.	through a network.	
30	The method of claim 29, wherein said network includes the Internet.	An unspecified portion of Courgette code; Courgette generated update received through a network or the Internet.	Courgette does not infringe claim 29 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
31	The method of claim 12, wherein said data is received in step (a) from a storage medium.	An unspecified portion of Courgette code; Courgette generated update accessed from computer readable storage medium.	Courgette does not infringe claim 12 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Google uses Courgette as part of the process of generating updates for the Google Chrome web browser on computers running Microsoft Windows. The data received by a client computer does not include the claimed "compact difference result" (*i.e.*, is not the "said data") and is not received from a storage medium.
32	The system of claim 25, wherein said data is received in step (a) from a remote site through a network.	An unspecified portion of Courgette code; computer or processor capable of executing the Courgette code; Courgette generated update received through a network.	Courgette does not infringe claim 25 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
33	The system of claim 32, wherein said network includes the Internet.	An unspecified portion of Courgette code; computer or processor capable of executing the Courgette code; Courgette generated update received through a network or the Internet.	Courgette does not infringe claim 32 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
34	The system of claim 25, wherein said data is received in step (a) from a storage medium.	An unspecified portion of Courgette code; computer or processor capable of executing the Courgette code; computer readable	Courgette does not infringe claim 25 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Google uses Courgette as part of the process of generating updates for

Claim	‘552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
		storage medium; Courgette generated patch accessed from a computer readable storage medium.	the Google Chrome web browser on computers running Microsoft Windows. The data received by a client computer does not include the claimed “compact difference result” (*i.e.*, is not the “said data”) and is not received from a storage medium.
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:	An unspecified portion of Courgette code; program processed by Courgette.	Courgette does not generate, receive, transmit, store or use the “compact difference result” claimed by the ‘552 patent. The ‘552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
42(a)	(a) generating a modified old data table utilizing at least said old data table;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed “modified old data table.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed “modified new data table” or the claimed “modified old data table.”
42(b)	(b) generating a modified new data table utilizing at least said new data table, said modified old data table and modified new data table have at least the following	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed “modified old data table,” or the claimed “modified new data table.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	characteristics:		Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table."
42(b)(i)	(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;	An unspecified portion of Courgette code.	Courgette does not identify or process "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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result. Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table."
42(c)	(c) generating said compact difference result utilizing at least said modified new data table and modified old data table.	An unspecified portion of Courgette code.	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette generates a difference result, which is not the claimed "compact difference result," using symbolic data rather than using the executable "modified old data table" and "modified new data table" as required by the claim. Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table."
43	The method of claim 42, further comprising the step of: (d) transmitting said compact difference result over a communication network.	An unspecified portion of Courgette code; Google's update process responsible for transmitting updates.	Courgette does not infringe claim 42 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.
44	The method of claim 43, wherein said network	An unspecified portion of Courgette code; Google's	Courgette does not infringe claim 43 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	includes the Internet.	Update Process responsible for transmitting updates over the Internet.	infringe any claim(s) depending therefrom.
45	The method of claim 42, further comprising the step of: (d) storing said compact difference result on a storage medium.	An unspecified portion of Courgette code "that write an update package to a storage medium."	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette does not infringe claim 42 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
46	A method for performing an update in an old data table so as to generate 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:	An unspecified portion of Courgette code; program processed by Courgette.	
46(a)	(a) receiving data that includes a compact difference result; said compact difference result was generated utilizing a modified old data table and a modified new data table;	An unspecified portion of Google code running on a users' computing device that receives update package generated by Courgette.	Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed "compact difference result," using symbolic data rather than using the executable "modified old data table" and "modified new data table" as required by the claim.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Google does not create a difference result that is used to update copies of the Chrome web browser using a modified old version of the Chrome web browser software and a modified new version of the Chrome web browser software.
46(b)	(b) generating a modified old data table utilizing at least said old data table;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Google does not update copies of the Chrome web browser by generating a "modified old data table" on the remote client computer.
46(c)	(c) reconstituting a modified new data table utilizing directly or indirectly at least said modified old data table and said compact difference result; said modified old data table and modified new data table have at least the following characteristics:	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting a modified new data table utilizing directly or indirectly at least said modified old data table and said compact difference result" on the remote client computer. Neither of the executable programs that

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new data table" or the "modified old data table" of the claims, and even if they were, neither are "reconstitute[ed]" on the remote client computer.
46(c)(i)	(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/inset 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;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Courgette does not identify or process "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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "substantially each reference in an entry in said old data table that is different than corresponding entry in said new data table due to

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			delete/insert modifications that form part of the transition between said old data table and new data table," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result.
46(d)	(d) reconstituting said new data table utilizing directly or indirectly at least said compact difference result and said modified new data table.	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting said new data table utilizing directly or indirectly at least said compact difference result and said modified new data table" on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new data table" or the "modified old data table" of the claims. Even if they were, the "modified new data table" is not used to "reconstitute[e]" an updated version of the Chrome web browser as claimed. Moreover, even if Courgette utilized a difference result to "reconstitute" a "modified new data table" as required by step 46(c), it would not utilize the difference result to "reconstitute" a new data table as required by step 46(d).
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	An unspecified portion of Courgette code; computer or device capable of executing Courgette; program processed by Courgette; Computer or device capable or executing Courgette.	Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	device capable of:		
55(a)	(a) generating a modified old data table utilizing at least said old data table;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table."
55(b)	(b) generating a modified new data table utilizing at least said new data table, said modified old data table and modified new data table have at least the following characteristics:	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table."
55(b)(i)	(i) substantially each reference in an entry in said	An unspecified portion of Courgette code.	Courgette does not identify or process "substantially each reference in an entry in said old data table that is different than corresponding entry

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	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;		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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result.

Claim	‘552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette does not generate or use the claimed “modified old data table,” or the claimed “modified new data table.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed “modified new data table” or the claimed “modified old data table.”
55(c)	(c) generating said compact difference result utilizing at least said modified new data table and modified old data table.	An unspecified portion of Courgette code.	Courgette does not generate, receive, transmit, store or use a “compact difference result” within the meaning of the ‘552 patent. The ‘552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed “compact difference result,” using symbolic data rather than using the executable “modified old data table” and “modified new data table” as required by the claim. Courgette does not generate or use the claimed “modified old data table,” or the claimed “modified new data table.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table."
56	The system of claim 55, wherein said processor is further capable of transmitting said compact difference result over a communication network.	An unspecified portion of Courgette code; computer or device capable of communication over a network.	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette does not infringe claim 55 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
57	The system of claim 56, wherein said network includes the Internet.	An unspecified portion of Courgette code; computer or device capable of communication over the Internet.	Courgette does not infringe claim 56 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
58	The system of claim 55, wherein said processor is further capable of storing said compact difference result on a storage medium.	An unspecified portion of Courgette code; computer or device capable of storing data to a storage medium.	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette does not infringe claim 55 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore

Claim	‘552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			infringe any claim(s) depending therefrom.
59	A system for performing an update in an old data table so as to generate 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:	Courgette; computer or processing device capable of executing Courgette; program processed by Courgette.	
59(a)	(a) receiving data that includes a compact difference result; said compact difference result was generated utilizing a modified old data table and a modified new data table;	An unspecified portion of Google code running on a users’ computing device that receives update package generated by Courgette.	Courgette does not generate, receive, transmit, store or use the “compact difference result” claimed by the ‘552 patent. The ‘552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette generates a difference result, which is not the claimed “compact difference result,” using symbolic data rather than using the executable “modified old data table” and “modified new data table” as required by the claim. Courgette does not generate or use the claimed “modified old data table,” or the claimed “modified new data table.” In the ‘552 prosecution history, the patentee made it clear that the “old program,” “old data table,” “modified old program,” “modified new program,” “modified old data table,” and “modified new data table” are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Google does not create a difference result that is used to update copies of the Chrome web browser using a modified old version of the Chrome web browser software and a modified new version of the Chrome web browser software.
59(b)	(b) generating a modified old data table utilizing at least said old data table;	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Google does not update copies of the Chrome web browser by generating a "modified old data table" on the remote client computer.
59(c)	(c) reconstituting a modified new data table utilizing directly or indirectly at least said modified old data table and said compact difference	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	result; said modified old data table and modified new data table have at least the following characteristics:		to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting a modified new data table utilizing directly or indirectly at least said modified old data table and said compact difference result" on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new data table" or the "modified old data table" of the claims, and even if they were, neither are "reconstitute[ed]" on the remote client computer.
59(c)(i)	(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/inset modifications that form part of the	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
	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;		reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Courgette does not identify or process "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." Although Courgette recognizes all absolute addresses, it recognizes only certain relative addresses that are used as references. Courgette therefore does not recognize a majority of the instructions that contain references, much less "substantially each reference" or address. Where Courgette does not recognize an address, it treats the address data as raw bytes of instructions or data that are reflected in the instruction list and the byte stream. Because Courgette does not identify or process "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," such references cannot be and are not "reflected as invariant references." As a result, the unrecognized altered addresses also appear in the difference result along with references that are different due to delete/insert modifications that are recognized. Courgette does not find reference entries that change due to delete/insert modifications and replace them with invariant references, as required by the claim. Courgette does not distinguish for special treatment references that change due to delete/insert modifications

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			from references that change for other reasons. Rather, Courgette treats all references that it recognizes the same way—by reflecting them in symbolic tables and encoded streams. Courgette does not create the required "invariant references." Courgette preserves all unique references that it can recognize in a symbol table for the old and the new programs. The unique references that Courgette identifies retain their original values and are not changed to invariant. Additionally, unique references within Courgette are both sent to the difference generator and are reflected in the difference result.
59(d)	(d) reconstituting said new data table utilizing directly or indirectly at least said compact difference result and said modified new data table.	An unspecified portion of Courgette code.	Courgette does not generate or use the claimed "modified old data table," or the claimed "modified new data table." In the '552 prosecution history, the patentee made it clear that the "old program," "old data table," "modified old program," "modified new program," "modified old data table," and "modified new data table" are required to be executable and not symbolic with certain references replaced. Courgette generates ancillary data tables and streams that symbolically reflect information from the old and new programs. These ancillary data tables and streams are not an executable program. Courgette first derives the symbolic data structures by disassembling the old and the new programs and then further abstracts them into separate symbolic streams that represent related information, such as addresses, index bytes, size data and other control information in a format. This symbolic information is not in executable format and cannot, therefore, constitute either the claimed "modified new data table" or the claimed "modified old data table." Courgette does not generate, receive, transmit, store or use the "compact difference result" claimed by the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
			modifications from appearing in the difference result. Google does not update copies of the Chrome web browser by "reconstituting said new data table utilizing directly or indirectly at least said compact difference result and said modified new data table" on the remote client computer. Neither of the executable programs that are disassembled by Courgette into streams of symbolic data so that Courgette can create a difference result are the "modified new data table" or the "modified old data table" of the claims. Even if they were, the "modified new data table" is not used to "reconstitute[e]" an updated version of the Chrome web browser as claimed. Moreover, even if Courgette utilized a difference result to "reconstitute" a "modified new data table" as required by step 59(c), it would not utilize the difference result to "reconstitute" a new data table as required by step 59(d).
60	The system of claim 59, wherein said data is received in step (a) from a storage medium.	An unspecified portion of Courgette code; Courgette generated update accessed from computer readable storage medium.	Courgette does not infringe claim 59 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Google uses Courgette as part of the process of generating updates for the Google Chrome web browser on computers running Microsoft Windows. The data received by a client computer does not include the claimed "compact difference result" (*i.e.*, is not the "said data") and is not received from a storage medium.
62	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.	Computer or other processing device at a user location and/or a server; computer readable storage medium associated with the computer or processing device; update generated by Courgette.	Courgette does not generate, receive, transmit, store or use a "compact difference result" within the meaning of the '552 patent. The '552 patent requires elimination from the compact difference result of references that change due to delete/insert modifications. Courgette does not prevent references that change due to delete/insert modifications from appearing in the difference result. Courgette does not infringe claims 42 to 45 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.

Claim	'552 Patent Claim Language	Accused Component	Exemplary Basis of Non-Infringement Contention
63	The method of claim 46, wherein said data is received in step (a) from a remote site through a network.	An unspecified portion of Courgette code; Courgette generated update received through a network.	Courgette does not infringe claim 46 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
64	The method of claim 63, wherein said network includes the Internet.	An unspecified portion of Courgette code; Courgette generated update received through a network or the Internet.	Courgette does not infringe claim 63 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
65	The method of claim 46, wherein said data is received in step (a) from a storage medium.	An unspecified portion of Courgette code; Courgette generated update accessed from computer readable storage medium.	Courgette does not infringe claim 46 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom. Google uses Courgette as part of the process of generating updates for the Google Chrome web browser on computers running Microsoft Windows. The data received by a client computer does not include the claimed "compact difference result" (*i.e.*, is not the "said data") and is not received from a storage medium.
66	The system of claim 59, wherein said data is received in step (a) from a remote site through a network.	An unspecified portion of Courgette code; computer or processor capable of executing the Courgette code; Courgette generated update received through a network.	Courgette does not infringe claim 59 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.
67	The system of claim 66, wherein said network includes the Internet.	An unspecified portion of Courgette code; computer or processor capable of executing the Courgette code; Courgette generated update received through a network or the Internet.	Courgette does not infringe claim 66 for at least the reasons stated in these Preliminary Non-Infringement Contentions, and cannot therefore infringe any claim(s) depending therefrom.