EXHIBIT A

	Term	Bedrock's Proposed Construction	Supporting Evidence
1.	"linked list to store and provide access to records" / "linked list of records" [Claims 1, 3, 5 and 7]	a list in which each record contains a pointer to the next record or information indicating that there is no next record	Intrinsic Evidence '120 patent, col. 5, lines 20-25. Definition of type list_element in the pseudo-code Appendix of the '120 patent. *See* statement "while p ≠ nil" in pseudo-code for Search Table Procedure for example of information indicating that there is no next record. Extrinsic Evidence IEEE Std. 610.5-1990, *IEEE Standard Glossary of Data Management Terminology*, at 43: **linked list.** A list in which each item contains a pointer to the next or preceding item in the list, making it unnecessary for the items to be physically sequential. *Note*: Unless the list is circular, the last item in the list contains a null link field. *Syn*: **chain; chained list; one-way chain; singly linked list.** *See also* **circularly linked list; doubly linked list; linked linear list.** IEEE Std. 100-1992, *The New IEEE Standard Dictionary of Electrical and Electronic Terms*, Fifth Edition, at 727: **linked list** [**1**]. A list in which each item contains a pointer to the next or preceding item in the list, making it unnecessary for the items to be physically sequential. *Note*: Unless the list is circular, the last item in the list contains a null link field. *Syn*: **chain; chained list; one-way chain; singly linked list**. *See also* **circularly linked list; doubly linked list; linked linear list.** 610.6-1990

	Term	Bedrock's Proposed Construction	Supporting Evidence
			[**2**] (**software**). *See*: **chained list**. 729-1983 Microsoft Press, a division of Microsoft Corporation, 1991, *Computer Dictionary, The Comprehensive Standard for Business, School, Library, and Home*, at 213: **linked list** In programming, a list of nodes or elements of a data structure connected by pointers. A singly linked list has one pointer in each node, pointing to the next node in the list; a doubly linked list has two pointers in each node, pointing to the next and previous nodes. In a circular list, the first and last nodes of the list are linked together. Both linked lists and arrays are often used to implement the list, stack, and queue data structures. Linked lists have the advantages of dynamic node allocations and no requirements that nodes all be of the same type; arrays have the advantage of direct access to a given node. Compare linear list; *see also* array, key, list, node.
2.	automatically expiring/expired [Claims 1, 3, 5, and 7]	after a limited period of time or after the occurrence of some event, becoming obsolete and therefore no longer needed or desired in the storage system / obsolete and therefore no longer needed or desired in the storage system	<u>Intrinsic Evidence</u> '120 patent, col. 2, lines 7-11. <u>Extrinsic Evidence</u> IEEE Std. 100-1992, *The New IEEE Standard Dictionary of Electrical and Electronic Terms*, Fifth Edition, at 67: **automatic** (**1**) (**computer applications**). Pertaining to a function, operation, process, or device that, under specified conditions, functions without intervention by a human operator. 610.2-1987 (**2**) (**NESC**). Self-acting, operating by its own mechanism when actuated by some impersonal influence—as, for example, a change in currrent strength; not manual; without

	Term	Bedrock's Proposed Construction	Supporting Evidence
			personal intervention. Remote control that requires personal intervention is not automatic, but manual. C2-1984
3.	a record search means utilizing a search key to access the linked list [Claim 1]	FUNCTION: record searching utilizing a search key to access the linked list. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software instructions as illustrated in Boxes 31 and 32 of FIG. 3, or as "index := hash (record_key); p:=table[index]" of the pseudo-code of Search Table Procedure (cols. 11 and 12) or Alternate Version of Search Table Procedure (cols. 11, 12, 13, and 14), and described in col. 5, lines 57-63, or the equivalents thereof.	*See* the citations to corresponding structure.
4.	the record search means including a means for identifying and	FUNCTION: record searching including identifying and removing at least some of the expired ones of the records from the linked list when	*See* the citations to corresponding structure.

	Term	Bedrock's Proposed Construction	Supporting Evidence
	removing at least some [of the] expired ones of the records from the linked list when the linked list is accessed [Claims 1 and 5]	the linked list is accessed. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software as described in Boxes 33 - 42 of FIG. 3, and/or as pseudo-code in the Search Table Procedure (cols. 11 and 12) or Alternate Version of Search Table Procedure (cols. 11, 12, 13, and 14) starting at the line "while … /*HEART OF THE TECHNIQE…" and ending at the end of each procedure, and/or as described in col. 5, line 63 - col. 6, line 34, or the equivalents thereof.	
5.	means, utilizing the record search means, for accessing the linked list and, at the same time, removing at least some of the expired ones of	FUNCTION: utilizing the record search means, accessing the linked list and, at the same time, removing at least some of the expired ones of the records in the linked list. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or	*See* the citations to corresponding structure.

	Term	Bedrock's Proposed Construction	Supporting Evidence
	the records in the linked list [Claim 1]	operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software which provides the insert, retrieve, or delete record capability illustrated in the flowchart of FIG. 5, FIG. 6, or FIG. 7, respectively, and/or as pseudo-code of Insert Procedure (cols. 9 and 10), Retrieve Procedure (cols. 9, 10, 11, and 12), or Delete Procedure (cols. 11 and 12), respectively, and/or described in col. 7, line 65 - col. 8, line 32, col. 8, lines 33-44, or col. 8 lines 45-59, or the equivalents thereof.	
6.	means for dynamically determining maximum number for the record search means to remove in the accessed linked list of records [Claims 2 and 6]	FUNCTION: dynamically determining maximum number of records for the record search means to remove in the accessed linked list of records. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer	*See* the citations to corresponding structure.

	Term	Bedrock's Proposed Construction	Supporting Evidence
		system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software, as described in col. 6, line 56 - col. 7, line 15, that dynamically chooses among removal strategies (e.g., chooses whether to execute Search Table Procedure [cols. 11-12] or Alternate Version of Search Table Procedure [cols. 11-14]) "at the time the record search means is invoked by the caller, thus sometimes removing all expired records, at other times removing some but not all of them, and yet at other times choosing to remove none of them. Such a dynamic decision can be based on factors such as, for example, how much memory is available in the system storage pool, general system load, time of day, the number of records currently residing in the information system, and other factors both internal and external to the information storage and retrieval system itself" (col. 7, lines 1-10), or the equivalent thereof.	
7.	identifying at least some of the automatically expired ones of	identifying at least some of the automatically expired ones of the records when the linked list is accessed for a purpose other than	<u>Intrinsic Evidence</u> '120 patent, title; abstract; col. 2, lines 54-63; FIG. 3 and related description; pseudo-code for Search Table Procedure and

	Term	Bedrock's Proposed Construction	Supporting Evidence
	the records [Claim 3]	garbage collection, using the same linked list traversal performed for the purpose other than garbage collection.	Alternate Version of Search Table Procedure.
8.	removing at least some of the automatically expired records from the linked list when the linked list is accessed [Claims 3 and 7]	removing at least some of the automatically expired records from the linked list when the linked list is accessed for a purpose other than garbage collection, using the same linked list traversal performed for the purpose other than garbage collection.	Intrinsic Evidence '120 patent, title; abstract; col. 2, lines 54-63; FIGs. 3-4 and related description; pseudo-code for Search Table Procedure, Alternate Version of Search Table Procedure, and Remove Procedure.
9.	dynamically determining maximum number for the record search means to remove in the accessed linked list of records / dynamically determining maximum number of expired ones of the records to remove when the linked list is	The means-plus-function limitations that contain this language (claims 2 and 6) are addressed above, and not considered here. Bedrock believes that no construction of this language is needed. To the extent the Court chooses to construe it, Bedrock believes that the following construction is appropriate: "determining, during program execution, maximum number of expired ones of the records to remove when the linked list is accessed for a purpose other than garbage collection."	Intrinsic Evidence '120 patent, col. 6, line 56 - col. 7, line 15. Extrinsic Evidence IEEE Std. 100-1992, *The New IEEE Standard Dictionary of Electrical and Electronic Terms*, Fifth Edition, at 396: (**2**) (**software**). Pertaining to an event or process that occurs during computer program execution; for example, dynamic analysis, dynamic binding. *Contrast with*: **static**. 610.12-1990 Microsoft Press, a division of Microsoft Corporation, 1991, *Computer Dictionary, The Comprehensive Standard for Business, School, Library, and Home*, at 120: **dynamic** An adjective used to describe events or processes

	Term	Bedrock's Proposed Construction	Supporting Evidence
	accessed [Claims 4 and 8]		that occur immediately and concurrently as opposed to those planned for in advance or reacted to after the fact. *Dynamic* is used in reference to both hardware and software; in each case it describes some action or event that occurs when and as needed. In nondynamic memory management, a program is given a certain amount of memory when the program is first run and must run within that constraint. In dynamic memory management, a program is able to negotiate with the operating system when it needs more memory.
10.	a hashing means to provide access to records stored in a memory of the system and using an external chaining technique to store the records with same hash address, at least some of the records automatically expiring [Claim 5]	FUNCTION: using hashing to provide access to records stored in a memory of the system and using an external chaining technique to store the records with same hash address, at least some of the records automatically expiring. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software instructions corresponding to pseudo-code "var table: array [0 . . . table_size - 1] of list_element_pointer /* Hash table. */" in cols. 9-10 that allocates in	*See* the citations to corresponding structure.

	Term	Bedrock's Proposed Construction	Supporting Evidence
		memory an external chaining hash table, and/or as described in col. 5, lines 16-41, or the equivalents thereof.	
11.	mea[n]s, utilizing the record search means, for inserting, retrieving, and deleting records from the system and, at the same time, removing at least some expired ones of the records in the accessed linked list of records [Claim 5]	FUNCTION: utilizing the record search means, inserting, retrieving, and deleting records from the system and, at the same time, removing at least some expired ones of the records in the accessed linked list of records. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software which provides the insert, retrieve, and delete record capabilities illustrated in the flowcharts of FIG. 5, FIG. 6, and FIG.7, respectively, and/or as pseudo-code of Insert Procedure (cols. 9 and 10), Retrieve Procedure (cols. 9, 10, 11, and 12), and Delete Procedure (cols. 11 and 12),	*See* the citations to corresponding structure.

	Term	Bedrock's Proposed Construction	Supporting Evidence
		respectively, and/or described in col. 7, line 65 - col. 8, line 32, col. 8, lines 33-44, and col. 8, lines 45-59, or the equivalents thereof.	
12.	a record search means utilizing a search key to access a linked list of records having the same hash address [Claim 5]	FUNCTION: record searching, utilizing a search key to access a linked list of records having the same hash address. CORRESPONDING STRUCTURE: (1) Portions of the application software, user access software or operating system software, as described at col. 4, lines 30-48 and illustrated in FIG. 2, of a computer system that includes at least a CPU 10 and RAM 11, see FIG. 1 and col. 3 lines 52-56. (2) Executable software instructions as illustrated in Boxes 31 and 32 of FIG. 3, or as "index := hash (record_key); p:=table[index]" of the pseudo-code of Search Table Procedure (cols. 11 and 12) or Alternate Version of Search Table Procedure (cols. 11, 12, 13, and 14), and described in col. 5, lines 57-63, or the equivalents thereof.	*See* the citations to corresponding structure.
13.	Ordering of limitations of claim 3	No construction needed. If the Court is inclined to address	

	Term	Bedrock's Proposed Construction	Supporting Evidence
	[Claim 3]	this issue, then it should hold that the steps of claim 3 may be performed in a consecutive manner, in an overlapping manner, or a combination of the two.	
14.	Ordering of limitations of claim 7 [Claim 7]	No construction needed. If the Court is inclined to address this issue, then it should hold that the steps of claim 7 may be performed in a consecutive manner, in an overlapping manner, or a combination of the two, except that the ultimate step must follow or at least partially follow the penultimate step.	