UNITED STATES DISTRICT COURT
EASTERN DISTRICT OF TEXAS
TYLER DIVISION

BEDROCK COMPUTER TECHNOLOGIES LLC,
 	
        Plaintiff,
 	
    v.
 	
SOFTLAYER TECHNOLOGIES, INC., et al.,
 	
        Defendants.
	)
	)
	)
	)
	)
	)
	)
	)
	)
	)
	)
	)
	)
	)
		CASE NO. 6:09–CV–00269
		
		Hon. Leonard E. Davis
		
		JURY TRIAL DEMANDED

DEFENDANTS' JOINT RESPONSE BRIEF ON CLAIM CONSTRUCTION

TABLE OF CONTENTS

INTRODUCTION

Defendants submit this brief to address the construction of claim terms of U.S. Patent No. 5,893,120 ("the '120 patent") that Bedrock Computer Technologies ("Bedrock" or "Plaintiff") asserts. (*Ex.* 1)[1] The '120 patent is directed to a specific "on-the-fly" garbage collection technique used to free-up computer memory by removing expired, obsolete records occupying such memory. In particular, the invention opportunistically identifies and removes expired records when the list of records is being traversed during an insertion, deletion, or retrieval operation. According to the specification, "travers[ing the] entire list, deleting records as we search" is the "HEART OF THE TECHNIQUE" disclosed in the patent. Moreover, the goal of the '120 patent is "to remove these expired items to reclaim the storage and maintain fast access to the data."

As shown throughout this memorandum, the Defendants' constructions are in accord with the heart and goal of the '120 patent, the claim language, every embodiment in the specification, and the reasons for allowance stated by the Patent Office in allowing the claims to issue. In contrast, Bedrock's constructions and supporting arguments give a scope to the claims that is broader than what the inventor allegedly invented, drafted, and claimed in the '120 patent. Bedrock's proposed constructions and arguments disregard the claim language itself, the goals of the invention, the file history, the clear concessions in the patent specification as to the "present invention" and the "HEART OF THE TECHNIQUE," and the testimony of Dr. Nemes. For example, Bedrock's constructions (a) do not require the identification and removal of expired records to occur during the same traversal of the linked list of records and (b) do not require the expired records to be removed in a way that reclaims the storage occupied by those records for other uses. Bedrock's constructions ignore the canons of claim construction in a transparent attempt to improve its infringement

---

[1] References herein to "*Ex.* __" are the exhibits to the Declaration of Yar Chaikovsky.

arguments. Accordingly, Defendants respectfully request that the Court adopt their proposed

constructions for the sound reasons articulated in this memorandum.[2]

BACKGROUND

Bedrock is co-owned by three principals, one of whom is the sole named inventor of the '120

patent, Dr. Richard Nemes. Dr. Nemes assigned his rights to the patent to Bedrock prior to this

litigation. In this litigation, Bedrock alleges that a small portion of code in the well-known, open-

source Linux operating system infringes the '120 patent.

In the late 1980s, Dr. Nemes worked at Bell Communications Research, Inc. ("Bellcore"),

where he developed ideas relating to hashing and on-the-fly garbage collection. Bellcore filed a

number of patent applications based on these ideas that eventually matured into patents that Dr.

Nemes assigned to Bellcore. One such patent, U.S. Patent No. 5,121,495 ("the '495 patent"), related

to a system and method for "on the fly" garbage collection in a hash table utilizing linear probing to

resolve hash collisions. *See* Exh. 3. Another patent, U.S. Patent No. 5,287,499 ("the '499 patent"),

disclosed a hybrid hashing method utilizing both external chaining and linear probing to resolve hash

collisions. *See* Exh. 4. Both the '495 and '499 patents are cited on the face of the '120 patent.

In January 1997, eight years after Dr. Nemes left Bellcore, he filed a patent application that

matured into the '120 patent. Dr. Nemes prosecuted the '120 patent *pro se* and, in doing so, copied

verbatim many portions of the '495 and '499 patents. (Exh. 2, Nemes Tr. at 144:10-145:10).

Indeed, some claims of the '120 patent are virtually identical to those in the '495 patent. The

primary difference between Dr. Nemes' '495 patent and the '120 patent is that the '120 patent is

---

[2] Additional intrinsic evidence may result from the pending reexamination of the '120
patent. On July 23, 2010, the Patent Office rejected all '120 patent claims as being anticipated or
rendered obvious by numerous prior art combinations. (*See* Ex. 6). Bedrock's response to this
rejection is currently due Sept. 23, 2010.

directed at "on-the-fly" garbage collection of automatically expired records from a hash table utilizing external chaining, as opposed to a hash table utilizing linear probing. Dr. Nemes did not claim priority to the '495 patent nor assign the rights of the '120 patent to Bellcore.

An explanation of the technical details of hashing, external chaining, linear probing, and linked lists can be found in Defendants' Technical Tutorial, which will be provided to the Court on September 20, 2010.

OVERVIEW OF THE DISPUTED PATENT CLAIM TERMS

Bedrock asserts that Defendants infringe claims 1-8 of the '120 patent, entitled "Methods and Apparatus for Information Storage and Retrieval Using a Hashing Technique with External Chaining and On-the-fly Removal of Expired Data." The claims in the '120 patent are directed to means-plus-function elements and corresponding method steps.

Independent claim 1 is depicted below, including highlighting of disputed terms (note that the means-plus-function terms also are disputed but not highlighted below):

> 1. An information storage and retrieval system, the system comprising:
>
> a linked list to store and provide access to records stored in a memory of the system, at least some of the records automatically expiring,
>
> a record search means utilizing a search key to access the linked list,
>
> the record search means including a means for identifying and removing at least some of the expired ones of the records from the linked list when the linked list is accessed, and
>
> 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 the records in the linked list.

Independent method claim 3 is depicted below:

> **3.** A method for storing and retrieving information records using a ==linked list== to store and provide access to the records, at least some of the records ==automatically expiring==, the method comprising the steps of:
>
> accessing the linked list of records,
>
> ==identifying== at least some of ==the automatically expired ones== of the records, and
>
> ==removing at least some of the automatically expired records from the linked list when the linked list is accessed.==

The other two independent claims (5 and 7) contain similar elements and additional limitations.

Each independent claim has a single dependent claim which requires dynamically determining the maximum number of entries to be removed from the linked list (2, 4, 6, and 8). Dependent claim 4 is depicted below including highlighting of the disputed term:

> **4.** The method according to claim **3** further including the step of ==dynamically determining maximum number of expired ones of the records to remove when the linked list is accessed.==

CLAIM CONSTRUCTION PRINCIPLES

The Federal Circuit in its *en banc* decision in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) distilled the framework for claim construction. The "objective baseline" for construing patent claims is determining "how a person of ordinary skill in the art understands a claim term." *Id.* at 1313. The "claims themselves provide substantial guidance as to the meaning of particular claim terms." *Id.* at 1314 (citations omitted). "The claims, of course, do not stand alone. Rather, they are part of 'a fully integrated written instrument,' consisting principally of a specification that concludes with the claims. For that reason, claims 'must be read in view of the specification, of which they are a part.'" *Id.* at 1315 (citation omitted). "The specification 'is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.'" *Id.* (citation omitted).

Claim construction also may take into account extrinsic evidence that can assist a court in understanding the "way in which one of skill in the art might use the claim term." *Id.* at 1318. Inventor testimony is part of this extrinsic evidence. *Id.* at 1317.

The Federal Circuit definitively and routinely instructs that "[w]hen a patent thus describes the features of the 'present invention' as a whole, this description limits the scope of the invention." *Verizon Servs. Corp. v. Vonage Holdings Corp.*, 503 F.3d 1295, 1308 (Fed. Cir. 2007) (citations omitted). Where the written description discloses a single way to perform an aspect of the invention, refers to "this invention" or the "present invention" in the context of the description of this aspect, and does not indicate that the disclosed aspect is a "preferred embodiment," the "public is entitled to take the patentee at his word" and properly read the invention to be limited to the single embodiment of the aspect disclosed and described as the "present invention." *See Honeywell Int'l., Inc. v. ITT Indus., Inc.*, 452 F.3d 1312, 1318 (Fed. Cir. 2007). Furthermore, "common boilerplate language that does not specifically address the inventive features in any detail" does not provide additional embodiments that would preclude the proper narrowing of claim scope discussed above. *IP Innovation, L.L.C., v. eCollege.com*, 156 F. App'x 317, 321-22 (Fed. Cir. 2005); *see also ICHL, LLC v. NEC Corp. of Am.*, Nos. 5:08CV65, 5:08CV175, 5:08CV177, 2010 WL 1609232, at *17 (E.D. Tex. Apr. 20, 2010) (narrowly construing being in a "relative spaced relation" to mean in parallel because each embodiment disclosed by the specification disclosed a parallel relationship and giving no weight to boilerplate language stating that alternative shapes and dimensions may be used and modifications may be made).

ARGUMENT

I. "LINKED LIST TO STORE AND PROVIDE ACCESS TO RECORDS" / "LINKED LIST OF RECORDS" (CLAIMS 1, 3, 5, 7)

Defendants' Construction	*Bedrock's Construction*
Two or more records in which each record contains a pointer to the next record in the list or information indicating that there is no next record	A list in which each record contains a pointer to the next record or information indicating that there is no next record.

The parties dispute whether this phrase requires at least two records as Defendants contend, or whether a single record would be sufficient as Plaintiff contends.

The claim language itself uses the plural term "records" to define the contents of the linked list. The claim language does not use the term "one or more" records. Thus, the plain meaning of the phrase requires two or more records and there is no clear reason to depart from plain meaning. Indeed, the purpose of the invention is to traverse a linked list having multiple records and remove any that have expired. If a linked list contained only a single record, there would be no need to traverse the list and remove expired records.

The prosecution history also supports Defendants' construction. During the prosecution of the '120 patent, the Examiner rejected all claims as anticipated by the '495 patent. In traversing the rejection, the named inventor explicitly distinguished the '495 patent as "storing a <u>single</u> item called a record," and that "[t]his description excludes linked list implementations." (Exh. 5, '120 Patent File History, August 11, 1998 Response, p. 2.) (emphases added). Having avoided a prior art rejection on the basis that the claimed linked lists of records do not store a single record, Plaintiff cannot now argue that the claimed linked lists do, in fact, store a single record. "A patent may not, like a 'nose of wax' be twisted one way to avoid anticipation and another to find infringement." *Sterner Lighting, Inc. v. Allied Elec. Supply, Inc.*, 431 F.2d 539, 544 (5th Cir. 1970), *citing White v. Dunbar*, 119 U.S.

6

47 (1886). Bedrock's arguments completely ignore this definitional prosecution history statement by the applicant.

Bedrock also mischaracterizes the specification. The portion of the specification cited by Bedrock (7:34-38) only demonstrates that the deletion procedure may operate on the final element of the linked list, not that the linked list in its entirety may contain only one element. Furthermore, Bedrock concedes that the "linked list" "<u>ceased being a linked list when it had just one, final element</u>." (Br. 5) (emphasis added). Far from reading out the preferred embodiment, Defendants' construction is consistent with that embodiment, the rest of the specification, and the applicant's statement within the file history.

II. AUTOMATICALLY EXPIRING / AUTOMATICALLY EXPIRED OR EXPIRED (CLAIMS 1, 3, 5, 7)

Defendants' Construction	*Bedrock's Construction*
Becoming obsolete and no longer needed or desired in the storage system because of some external condition / obsolete and no longer needed or desired in the storage system because of some external condition.	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.

The parties dispute whether the automatically expiring record (or automatically expired record or expired record) becomes so due to some external condition, as proposed by Defendants, or due to any event, as proposed by Bedrock.[3] Although each possible triggering condition is not listed, the specification defines the condition that makes the record obsolete as an <u>external</u> condition:

---

[3] Bedrock contends that the phrases "automatically expiring" and "expired" have different meanings. (Br. 6-7.) However, the antecedent basis provided by the claim language requires that "expired" records be the "automatically expiring" records. (13:26-28; 13:44-45; 13:60-61; 14:23-24; 14:41-45.) Accordingly, these phrases should be construed consistently, with the only difference being the use of "becoming" for "automatically expiring" records to accord with the tense of this term.

> If the end of the list has not been reached as determined by decision box 33, decision box 38 is entered to determine if the record pointed to has expired. ==This is determined by comparing some portion of the contents of the record to some external condition.== A timestamp in the record, for example, could be compared with the current time-of-day value maintained by all computers. Alternatively, the occurrence of an event can be compared with a field identifying that event in the record.

(6:5-13). *See also* U.S. Pat. No. 5,121,495 at 5:22-26 ("[D]ecision box 40 is entered to determine if the record in that cell has expired. This is determined by *comparing some portion of the contents of the record to some external condition*.") (emphasis added).[4] According to the specification, whether a record has expired is "determined by comparing some portion of the contents of the record to some external condition." The specification then identifies a timestamp as an <u>example</u> of an external condition because it is based on a time source external to the system. Bedrock, however, ignores this definition.

Instead, Bedrock points to another part of the specification to support its construction. That portion of the specification, however, supports Defendants' construction, not Bedrock's. (Br. 6, citing 2:7-11.) The "limited period of time" and "scheduled event" described in the specification and relied on by Bedrock are more examples of external conditions because they occur at a specific point in time and are based on a time source external to the system. The named inventor's testimony is consistent. (Exh. 2, Nemes Tr. at 310:19-23.)

Bedrock's construction inexplicably changes the "scheduled event" recited in the specification to "some event." This effectively eliminates "automatic" from the claim language, as "some" event could encompass *manually*-set events as opposed to scheduled, automatic events. Bedrock's construction is so broad that it encompasses events *internal* to the system, and possibly

---

[4] The '495 patent issued to the named inventor of the '120 patent and is part of the intrinsic evidence as a cited reference in the patent-in-suit.

internal to the records. But, by its nature, each record contains data (which gets operated on but does not actually do anything) and will not execute conditions considered for record expiration purposes. With every aspect of the specification describing conditions external to the system for automatically expiring records, the intrinsic evidence simply does not contemplate internal events.

III.   IDENTIFYING AND REMOVING AT LEAST SOME OF THE EXPIRED RECORDS FROM THE LINKED LIST WHEN THE LINKED LIST IS ACCESSED (CLAIMS 1, 3, 5, AND 7)

There are three disputes regarding this phrase, each discussed in turn below.

A.   Identification and Removal Occur During the Same Traversal of the Linked List

Disputed phrase: identifying and removing at least some of the expired records from the linked list "when the linked list is accessed" (Claims 1, 3, 5, and 7)	
Defendants' Construction	*Bedrock's Construction*
Both identification and removal of the automatically expired record(s) occurs during the same traversal of the linked list	No separate construction offered: For "removing at least some of the expired ones of the records from the linked list when the linked list is accessed", Bedrock states: No construction necessary; however, should the Court construe this term: "removing at least some of the automatically expiring 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."

The parties dispute whether "when the linked list is accessed" modifies both the identifying and removing elements of the claim language, as proposed by Defendants, or only the removing element, as proposed by Bedrock. In other words, the dispute is whether the "identification" and "removal" elements must occur in the same traversal of the linked list.

The claim language itself shows that Defendants' construction is correct. In the means-plus-function limitations of claims 1 and 5 (system claims), the recited function in the claim language

9

says "<u>identifying **and** removing</u> at least some [of the] expired ones of the records from the linked list <u>when the linked list is accessed</u>." (13:31-34; 14:28-31) (emphases added). The conjunctive "and" indicates that each of the "identifying" and "removing" is tied to the phrase "at least some of the expired ones of the records from the linked list when the linked list is accessed." The corresponding functional language in the method claims (claims 3 and 7) in the '120 patent should be construed consistently.

The order of steps in method claims 3 and 7 similarly dictate that both the identifying and removing occur "when the linked list is accessed." Claim 3 recites "accessing the linked list of records," and then "identifying at least some of the automatically expired ones of the records, and removing at least some of the automatically expired records from the linked list when the linked list is accessed." Similarly, claim 7 recites "accessing a linked list of records having same hash address," and then "identifying at least some of the automatically expired ones of the records" and "removing at least some of the automatically expired records from the linked list when the linked list is accessed." In each of claims 3 and 7, the linked list is accessed, and then expired records are both identified and removed. *See also*, Section VI below.

The file history also supports Defendants' construction. In overcoming a prior art rejection of all the claims, the named inventor argued that:

> Neither '499 nor Shackelford suggest what is recited in claims 1, 3, 5, and 7, for example, ==means and methods for identifying and removing== "at least some expired ones of the records" from the linked list =="when the linked list is accessed."== In addition, for the reasons explained in detail in the previous discussion, rejection of claims 2, 4, 6, and 8, which are directed to "dynamically determining maximum number," has already been discussed above. Thus, this rejection should be withdrawn.

10

(Exh. 5, '120 Patent File History, August 10, 1998 Response at page 6) (emphasis added). The

applicant thus treated the pertinent functional language in both the means and method claims

consistently, as does Defendants' construction.

The specification supports Defendants' construction.[5] The Abstract states that "each

insertion, retrieval, or deletion of a record is *an occasion to search an entire linked-list chain of*

records for expired items and then remove them." (emphasis added). The Brief Summary of the

Invention further provides that

> In accordance with the illustrative embodiment of the invention,
> these and other problems are overcome by using a garbage
> collection procedure 'on-the-fly' while other types of access to the
> storage space are taking place. In particular, during normal data
> insertion or retrieval probes into the data store, the expired,
> obsolete records are identified and removed from the external
> chain linked list. Specifically, expired or obsolete records in the
> linked list including the record to be accessed are removed as part
> of the normal search procedure.

(2:54-63) (emphasis added). The Summary continues: "The method *accesses the linked list of*

records and identifies at least some automatically expired ones of the records. *It also removes* at

least some automatically expired ones of the records from the linked list *when the linked list is*

accessed." (*Id.* at 3:7-11) (emphasis added).

Furthermore, the specification twice states in connection with the only embodiments

disclosed that the "HEART OF THE TECHNIQUE" is to "[t]raverse [the] entire list, deleting

expired records as we search" through "ON-THE-FLY REMOVAL OF EXPIRED RECORD[S]!":

---

[5] There is no ordinary meaning for accessing a linked list. The inventor's testimony is
consistent. (Exh. 2, Nemes Tr. at 245:15-246:9.) The intrinsic evidence defines what it means
when the claim language says identifying and removing expired records "when the linked list is
accessed."

```
previous_position := nil;                                                    /* Ditto */
while p ≠ nil                        /* HEART OF THE TECHNIQUE: Traverse entire list, deleting */
                                     /* expired records as we search. */

   begin
     if p↑ .record_contents is expired
        then remove (p, previous_p, index)      /* ON-THE-FLY REMOVAL OF EXPIRED RECORD! */
```

(Appendix, columns 11-12 (same pseudocode and comments repeated twice).) The algorithms disclosed explicitly remove expired records as they are identified ("as we search") during the same traversal of the linked list. The specification goes on to repeatedly describe, in *every* description of the claimed invention, that the identifying and removing elements occur during the same traversal of the linked list. (5:57-6:27; 6:35-38; 6:46-53; 6:56-59; FIG. 3; Appendix) For example, in Figure 3 the removal of an expired record (box 42) occurs immediately after the record is identified as expired (box 38) and before moving to the next record in the linked list.

FIG. 3

12

The goal of the "on-the-fly" garbage collection of the '120 patent is to efficiently and quickly remove expired records to reclaim storage. (1:23-33; 2:19-21.) In view of the alleged invention of the '120 patent, it simply would not make sense to identify expired records during one traversal and remove the expired records during yet another traversal of the linked list. This is less efficient than identifying and removing expired records during the same traversal of the linked list. Repeatedly accessing a linked list to identify expired records during a traversal and then remove the expired records during another traversal would be contrary to the goal of the alleged invention. *Id.*

Moreover, "when the linked list is accessed" means "during the same traversal of the linked list." The Federal Circuit has interpreted "when" to mean "at the time of, and not some appreciable time thereafter." *Renishaw PLC v. Marposs Societa' Per Azioni*, 158 F.3d 1243, 1252 (Fed. Cir. 1998). There, the patentee argued that the term "when said sensing tip contacts an object" could include any time after the tip contacted the object. The court noted that, though the term "when" was "imprecise as used in the [patent]," it was not ambiguous because "the written description provides overwhelming evidence to guide a proper interpretation of the term." *Id.* at 1251. The court rejected the patentee's construction because "the entire patent document exhibits [the inventor's] intent to make the delay between contact and signaling as small as possible." *Id.* at 1253. Similarly here, as already discussed, the patent makes clear that the "heart of the technique" is to identify and remove expired records during the same traversal of the linked list. (5:57-6:27; 6:35-38; 6:46-53; 6:56-59; Fig. 3; Appendix). Permitting the identification or removal to occur after "the linked list is accessed" or after the "accessing [the / a] linked list of records" (*i.e.*, after the same traversal of the linked list) would undermine the purpose of the patent.

In comparison, Bedrock's apparent position that the phrase "when the linked list is accessed" only modifies the "removing" element is arbitrary and inconsistent with the intrinsic evidence. There is no support for this in the patent, and Bedrock offers none.

B.   Removal Requires Deallocating Memory.

Disputed phrase: identifying and "removing at least some [of the] expired records from the linked list" when the linked list is accessed (Claims 1, 3, 5, and 7)	
Defendants' Construction	*Bedrock's Construction*
While traversing the linked list, both adjusting the pointers in the linked list to bypass the previously identified expired records and deallocating the memory occupied by those records.	No separate construction offered: For "removing at least some of the expired ones of the records from the linked list when the linked list is accessed", Bedrock states: No construction necessary; however, should the Court construe this term: "removing at least some of the automatically expiring 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."

The parties dispute whether "removing . . . from the linked list" requires that the memory occupied by the record be deallocated. Bedrock agrees in its alternative construction that "removing" occurs during traversal of the linked list. (Br. at 10.)

The objective of the '120 patent is clear. As Bedrock points out, expired records burden the performance of a system because "expired records occupy dynamically allocated memory storage that could be returned to the system memory pool for useful allocation." (Br. 2, citing 5:47-49.) Accordingly, "**[t]he goal is to remove these expired items to reclaim the storage and maintain fast access to the data**." (2:19-21) (emphasis added). The patent repeatedly states that the objective is to reclaim the storage occupied by expired records. (2:12-16; 2:22-27; 2:64-67.) If a system does

not deallocate memory occupied by records to reclaim their storage space, it has not accomplished the objective of the alleged invention.

The patent defines removing and accordingly describes each and every embodiment of removing records from the system as adjusting a pointer to bypass the record and deallocating the memory occupied by the record. Further, the patent states that "[t]he attached APPENDIX contains PASCAL-like pseudocode listings for all of the programmed components **necessary to implement** an information storage and retrieval system operating **in accordance with the present invention**." (8:60-64) (emphasis added). The pseudocode explicitly requires both pointer adjustment and deallocation during a remove operation, and nothing else:

Remove Procedure

```
procedure remove (var elem_to_del: list_element_pointer;
                      previous_elem: list_element_pointer;
                      index: 0 . . . table_size − 1);
/*    Delete elem_to_del↑ from list, advancing elem_to_del to next element. previous_elem points to
      elem_to_del's predecessor, or nil if elem_to_del↑ is 1ˢᵗ element in list.*/
var p: list_element_pointer;                              /* Save pointer to elem_to_del for disposal. */
begin
   p : = elem_to_del;                         /* Save so we can dispose when finished adjusting pointers. */
   elem_to_del : = elem_to_del↑ .next;
   if previous_elem = nil                               /* Deleting 1ˢᵗ element requires changing */
      then table[index] := elem_to_del                  /* head pointer, as opposed to */
      else previous_elem↑ .next := elem_to_del;         /* predecessor's next pointer. */
   dispose (p)                                                   /* Dynamically de-allocate node. */
end                                             /* remove*/
```

Pointer Adjustment

Deallocation

Every time that the Remove Procedure is called, the pointer in the linked list is adjusted and the "dispose(p)" function is called to "Dynamically de-allocate node." Because the algorithm described in the code is **necessary** to implement the **present invention**, deallocation must be required by the claim language. *Verizon*, 503 F.3d at 1308; *Honeywell*, 452 F.3d at 1318; *IP Innovation*, 156 F. App'x. at 321-22.

15

The specification of the '120 patent discloses the "present invention" with regard to the removing term, rather than disclosing illustrative, exemplary, or preferred embodiments thereof.[6] The specification states that the "information and storage and retrieval system provided by the **present invention** are herein disclosed as flowcharts in FIGS. 3 through 7, and shown in PASCAL-like pseudocode in this APPENDIX." (4:48-52; *see also* 3:18-45; 5:53-56; 7:65-67) (emphasis added). In describing Figure 3, the specification describes "returning storage to the storage pool with each removal" (6:35-45). Figure 4 is the sole figure depicting the remove procedure. (3:30-45).
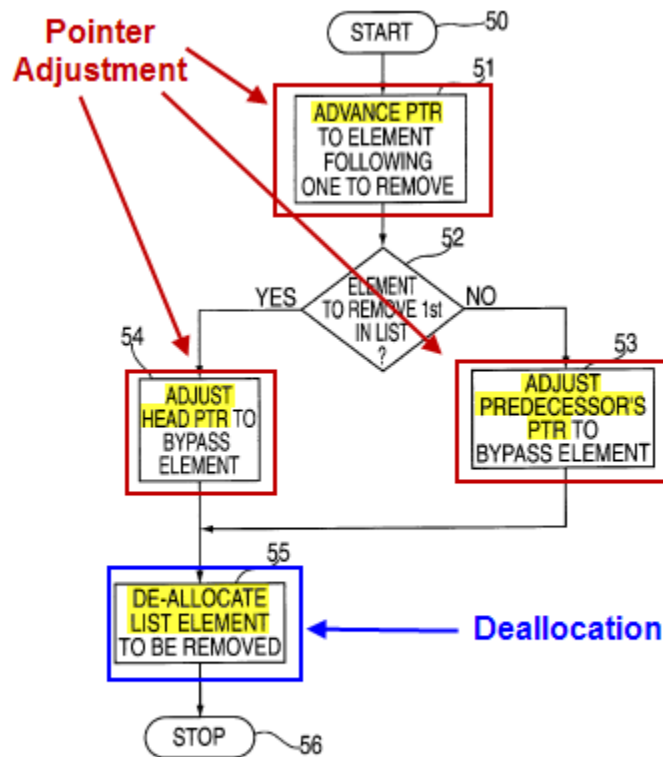

FIG. 4

_____

16

As is illustrated above, each time that the remove function of Figure 4 is called, pointers are adjusted at box 51 and box 53 or 54, and then the memory occupied by the record "<u>to be removed</u>" is deallocated at box 55. Thus, pointer adjustment alone cannot be "removal" since pointer adjustment occurs in boxes 51, 53, and 54 and the list element remains one "to be removed" until box 55 (deallocation). The specification repeatedly states that "removing" requires deallocation of the record from memory. (5:25-33; 6:35-45; 7:43-51; 7:56-64.) Accordingly, deallocation is a required step to remove a record. *Verizon*, 503 F.3d at 1308; *Honeywell*, 452 F.3d at 1318; *IP Innovation*, 156 F. App'x at 321-22.

The file history also supports the Defendants' construction. Consistent with the patentee equating "ON-THE-FLY REMOVAL OF AN EXPIRED RECORD!" and deallocation of memory records while describing the "THE HEART OF THE TECHNIQUE" in the pseudocode as was discussed above, the Examiner's Reasons for Allowance equated removing expired records with "deletion of records" (again, for both the claimed systems and methods):

1. The following is an Examiner's Statement of Reasons for Allowance:

   The prior art does not teach or fairly suggest a method and apparatus for on-the-fly deletion of records in linked lists based on automatic expiration of data as claimed. In other words, the prior art of record does not teach or fairly suggest the means (or an equivalent step in the method claim) of "means for ..accessing a linked list, at the same time, removing ....some of the expired ones of the records in the linked list," as recited in lines 7-8 of claim 1. Although the prior art of record (Nemes, '495 reference) teaches the use of chains of records and the deletion of records, the Applicant, in the Response dated August 11, 1998, Paper No. 5, provided arguments as to why the chain of records as taught in the '495 reference is not the same as the linked list as claimed. The Applicant also distinguishes the claimed invention over the teachings of the '499 references, see page 3, Paper no. 5.

(Exh. 5, '120 Patent File History, Paper No. 6 at 2.) The specification then defines "deletion of records" as deallocating memory (returning storage to storage pool):

17

> Deletion of records involves merely adjusting the pointers to bypass the deleted record and returning the storage it occupied to the available storage pool maintained by the system.

(5:31-33). Deallocation of memory is a key feature of the alleged invention of the '120 patent, according to both the specification and the file history. *See Alloc, Inc. v. ITC*, 342 F.3d 1361, 1371 (Fed. Cir. 2003) ("Although the specification alone is sufficiently clear, the prosecution history of this patent family confirms the description in the specification of each patent, namely, that play is a key feature of the claimed invention."). Defendants' construction is further confirmed by the testimony of the named inventor. (Exh. 2, Nemes Tr. at 195:22-202:23.)

Plaintiff offers no construction for "removal," nor does it offer *any* support for its position. (Br. 9.) Rather, Bedrock claims that Defendants' are trying to limit the claims to a preferred embodiment. The intrinsic evidence discussed above shows that deallocation of memory is not just an embodiment; it is fundamental to the "present invention" as indicated in the specification, recognized by the Examiner, and admitted by the named inventor. Bedrock has not and cannot point to any support from the patent that Figure 4 or the pseudocode are only the "preferred" embodiment,[7] or that removal can include a definition that does not deallocate the memory occupied by the record. The patent provides no written description for any alternatives and any alternative examples.[8]

---

[7] Bedrock concedes that Figure 4 and the Remove Procedure in the pseudocode are intrinsic evidence supporting the construction of this claim element. (Dkt. 251 at 8.)

[8] Bedrock may point to the standard throw-in clause at the end of the patent to state that it contemplated alternative embodiments. However, this clause will not save Bedrock for two reasons: (1) the clause refers to "other embodiments of the present invention" may be made by those skilled in the art "without departing from the teachings of the present invention" and (2) the case law makes clear that these boilerplate clauses do not in themselves provide alternatives. *See IP Innovation*, 156 F. App'x at 321-22; *Wireless Agents LLC v. Sony Ericsson Mobile Commc'ns (USA) Inc.*, 189 F. App'x 965, 967-68 (Fed. Cir. 2006); *ICHL*, 2010 WL 1609232, at *17. The pseudocode necessary to implement the present invention requires both delinking and deallocation. Thus, any alternative embodiments must include both delinking and deallocation.

Accordingly, "removal" must include both adjustment of the pointer and deallocation of the record from memory.

C. Identify[ing] Requires Determin[ing] Whether a Record is Expired by Comparing Some Portion of the Contents of the Record to Some External Condition

Disputed phrase: "identifying" and removing "at least some of the [automatically] expired [ones of the] records" from the linked list when the linked list is accessed	
Defendants' Construction	*Bedrock's Construction*
determining whether a record is expired by comparing some portion of the contents of the record to some external condition	no construction necessary; however, should the Court construe this term: "identifying at least some of the automatically expired ones of the records 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.

The issue in the construction of "identify" is whether it requires a "determination" that the record has automatically expired. From the express disclosure in the specification, the identification of the expired records is an active determination into the status of the record. The patent discloses only one embodiment for the act of "identifying" expired records stating that "[i]f the end of the list has not been reached as determined by decision box 33, decision box 38 is entered to determine if the record pointed to has expired. **This is determined by comparing some portion of the contents of the record to some external condition**." (6:5-9 (emphasis added); see FIG. 3.)

The specification does not qualify this disclosure as an illustrative or exemplary embodiment – it **is** the disclosure of the present invention. (4:48-52.) Dr. Nemes' testimony is consistent. (Exh. 2, Nemes Tr. at 307:8-309:1.) Likewise, the '495 patent cited by Dr. Nemes in the '120 patent discloses that the identification of an automatically expired record is determined by comparing the record to an external condition. (Exh. 3, '495 patent, 5:22-26.)

19

Indeed, Bedrock did not challenge the requirement of such determination of whether a record has automatically expired, and instead limited its dispute to whether an automatically expired record is "based on an external condition," which is discussed in Section II herein. Given the express language in the specification and the inventor's own testimony, it is clear that the term "identify" requires "determining whether a record is expired by comparing some portion of the contents of the record to some external condition," and should be construed accordingly.

IV. DYNAMICALLY DETERMINING MAXIMUM NUMBER OF EXPIRED ONES OF THE RECORDS TO REMOVE [WHEN THE LINKED LIST IS ACCESSED] (CLAIMS 2, 4, 6, AND 8)

Defendants' Construction	*Bedrock's Construction*
Immediately before the linked list is traversed, determining a single number that serves as an upper limit on the number of records to remove while the linked list is traversed	Determining, during program execution, maximum number of expired ones of the records to remove when the linked list is accessed

A. Dynamically Determining

The parties dispute whether the dynamic determination occurs "immediately before the linked list is traversed" as proposed by Defendants or "during program execution" as proposed by Bedrock.

Each of claims 2, 4, 6, and 8 recite "dynamically determining maximum number.…" Defendants propose a construction consistent with this functional claim language in the claims.[9]

First, the Brief Summary of the Invention and the specification disclose making a dynamic determination after the linked list is accessed. (3:11-14.) More importantly, the "dynamically" claim language in dispute answers the question in dispute between Bedrock and Defendants: What

---

[9] Defendants note that this is not a construction of the corresponding structure for the means-plus-function elements in claims 2 and 6. As further discussed in this brief and Defendants' indefiniteness brief, the '120 patent fails to disclose corresponding structure for these claims.

event triggers the dynamic determination of the maximum number of expired records to remove? The claim language itself answers this question: the event occurs "when the linked list is accessed" (claims 4 and 8) or "in the accessed linked list of records (claims 2 and 6). And thus, indicating the immediacy of the determination (the dynamic determination) is made with respect to accessing the linked list. Thus, Defendants propose that the determination is made "immediately before the linked list is traversed."

Bedrock's proposed construction of "during program execution" is hopelessly vague and runs counter to the intrinsic evidence. Under Bedrock's proposed construction, the "dynamic decision" could occur at any time – immediately before the linked list is accessed; hours, days, or years before the linked list is accessed; or after searching the linked list has already begun. Indeed, Bedrock's construction permits the dynamic determination to occur long after the program has finished accessing the linked list, so long as that program is still executing. It would be nonsensical to determine the maximum number of records to remove when accessing the list if that determination was made after the list had been accessed. Consequently, Bedrock's construction, which relies on extrinsic evidence that is at odds with the intrinsic evidence, should be rejected. *Playtex Prods., Inc. v. Procter & Gamble Co.*, 400 F.3d 901, 908 n.1 (Fed. Cir. 2005) (holding that a trial court "may not use extrinsic evidence to arrive at a claim construction that is at odds with the intrinsic evidence"). (Br. 10-11.)

B.   Maximum Number of Expired Records

The parties also dispute whether "maximum number of expired ones of the records to remove when the linked list is accessed" should be construed at all, and if so, whether it should mean "a single number that serves as an upper limit of the number of records to remove while the linked list is traversed" as proposed by Defendants. Bedrock offers no construction.

Defendants' proposed construction is taken nearly verbatim from the named inventor's arguments made to distinguish prior art during the prosecution of the '120 patent:

> The "maximum number of records" (in the instant application) and "threshold" (in '499) serve different purposes and are structured and determined differently. In the instant application, the number is a single quantity that serves as an upper limit on the number of records removed from the linked list whenever the linked list is accessed (see claims 6 and 8), whereas in '499 the threshold is a pair of coupled quantities, an upper threshold and a lower threshold, that serve as two-way signals indicating when the system should automatically reorganize a group of records that reside in cells of the hash table into a linked list, and vice versa (col. 6, lines 44–54 and 61–65; APPENDIX).

(Exh. 5, '120 Patent File History, August 10, 1998 Response, at page 4.) In an attempt to explain away this clear definition, Bedrock mistakenly relies on clear disavowal case law. The issue is not one of disavowal of the prior art "threshold." Rather, the inventor, who prosecuted the '120 patent, clearly defined the term "maximum number" of records during prosecution. *See C.R. Bard., Inc. v. U.S. Surgical Corp.*, 388 F.3d 858, 869 (Fed. Cir. 2004). Further, that definition is consistent with the named inventor's deposition testimony. (Exh. 2, Nemes Tr. at 326:9-327:8.)

Instead of offering a construction, Bedrock opposes Defendants' construction of maximum number. In doing so, Bedrock appears to be suggesting that the "maximum number" is an indeterminate amount, so that dynamically determining the maximum number of expired records to remove would encompass removing none, some or all of the records, but not any particular number. To the extent Bedrock is suggesting this by not construing "maximum number," Bedrock's position is directly at odds with the claim language and the named inventor's definitional statement during prosecution.

22

V.   **MEANS PLUS FUNCTION LIMITATIONS (CLAIMS 1, 2, 5, 6)**

In a means-plus-function claim "in which the disclosed structure is a computer, or microprocessor, programmed to carry out an algorithm, the disclosed structure is not the general purpose computer, but rather the special purpose computer programmed to perform the disclosed algorithm." *WMS Gaming, Inc. v. Int'l Game Tech.*, 184 F.3d 1339, 1349 (Fed. Cir. 1999). This is because disclosing only "a computer as the structure designed perform a particular function does not limit the scope of the claim to the 'the corresponding structure, material, or acts' that perform the function, as required by section 112 paragraph 6." *Aristocrat Techs Austl. Pty Ltd. v. Int'l Game Tech.*, 521 F.3d 1328, 1333 (Fed. Cir. 2008).

To avoid pure functional claiming, the applicant must disclose a particular algorithm that performs the claimed function. Thus, the corresponding structure for a computer-implemented means-plus-function term is "the algorithm disclosed in the specification." *Harris Corp. v. Ericsson Inc.,* 417 F.3d 1241, 1249 (Fed. Cir. 2005). Absent any such algorithm, the claim lacks sufficient disclosure of structure under 35 U.S.C. § 112 ¶ 6 and is therefore indefinite under 35 U.S.C. § 112 ¶ 2; *Aristocrat,* 521 F.3d at 1331.

A.   **Indefinite Terms**

The terms "a record search means utilizing a search key to access the linked list" (claims 1 and 5) and "a hashing means to provide access to records stored in a memory of the system" (claim 5) are indefinite because there is no corresponding algorithm disclosed in the specification. Similarly, the term "means for dynamically determining maximum number for the record search means to remove in the accessed linked list of records" (claims 2 and 6) is indefinite because there is

no corresponding algorithm.[10]   Consequently, claims 1, 2, 5 and 6 are indefinite and therefore

invalid. *See* Defendants' Motion for Summary Judgment of Indefiniteness, filed concurrently.

B.   **Means for Identifying and Removing at Least Some of the Expired Ones of the Records from the Linked List when the Linked List is Accessed (Claims 1, 5)**

Defendants' Construction	*Bedrock's Construction*
Function: identifying and removing at least some [of the] expired ones of the records from the linked list [of records] when the linked list is accessed.	**Function:** record searching including identifying and removing at least some of the expired ones of the records from the linked list when the linked list is accessed.
For the construction of this function, see Argument, Section III above, as further described in Joint Claim Construction Statement [Dkt. 251]. **Structure**: Boxes 10 and 11 of Fig. 1, Boxes 38 and 42 of Fig. 3, Fig 4, pseudocode in the Search Procedure (cols. 11-14) and Remove Procedure (cols. 13-14), and corresponding portions of the specification.[11]	**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. 11-14) starting at the line "while … /*HEART OF THE TECHNIQUE…" 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.

There are two primary disputes regarding this means-plus-function term. First, the parties

dispute the construction of the function. The parties agree that the function includes "identifying and

removing at least some of the expired ones of the records from the linked list when the linked list is

accessed." The Defendants contend, however, that this language should be construed as set forth in

---

[10] During his deposition, the named inventor admitted the patent does not disclose the algorithm for performing the recited function. (Exh. 2, Nemes Tr. 288:61-289:21.)

[11] Defendants agree that a CPU 10 and RAM 11 are part of the structure of the claim limitation.

Section III.A and III.B above. Defendants' arguments above apply with equal force here and will

not be repeated. Accordingly, this language requires:

- "both identification and removal of an automatically expired record occurs during the same traversal of the linked list."
- the phrase "when the linked list is accessed" refers to the time during which the "utilizing a search key to access the linked list" function in the limitation is carried out;
- "removing requires, while traversing the linked list, both adjusting the pointers in the linked list to bypass the previously identified expired records and deallocating the memory occupied by those records."

Second, while the parties largely agree on the structure corresponding to the function, the

parties dispute whether Figure 4 (the "remove procedure") and the accompanying removal

pseudocode is part of that structure. The claim language requires "means for identifying <u>and</u>

<u>removing</u>" at least some of the expired records when the linked list is accessed (emphasis added).

Plaintiff concedes that Box 42 of Figure 3 is part of the required structure. (Br. 18-19.) However,

Box 42 does not disclose any *structure*: it merely identifies the *function* of removing. Language that

"simply describes the function to be performed, not the algorithm by which it is performed" does not

provide adequate structure for a computer-implemented means-plus-function claim. *Aristocrat*, 521

F. 3d at 1334.

As Box 42 explicitly states, the corresponding removal algorithm is disclosed in Figure 4:
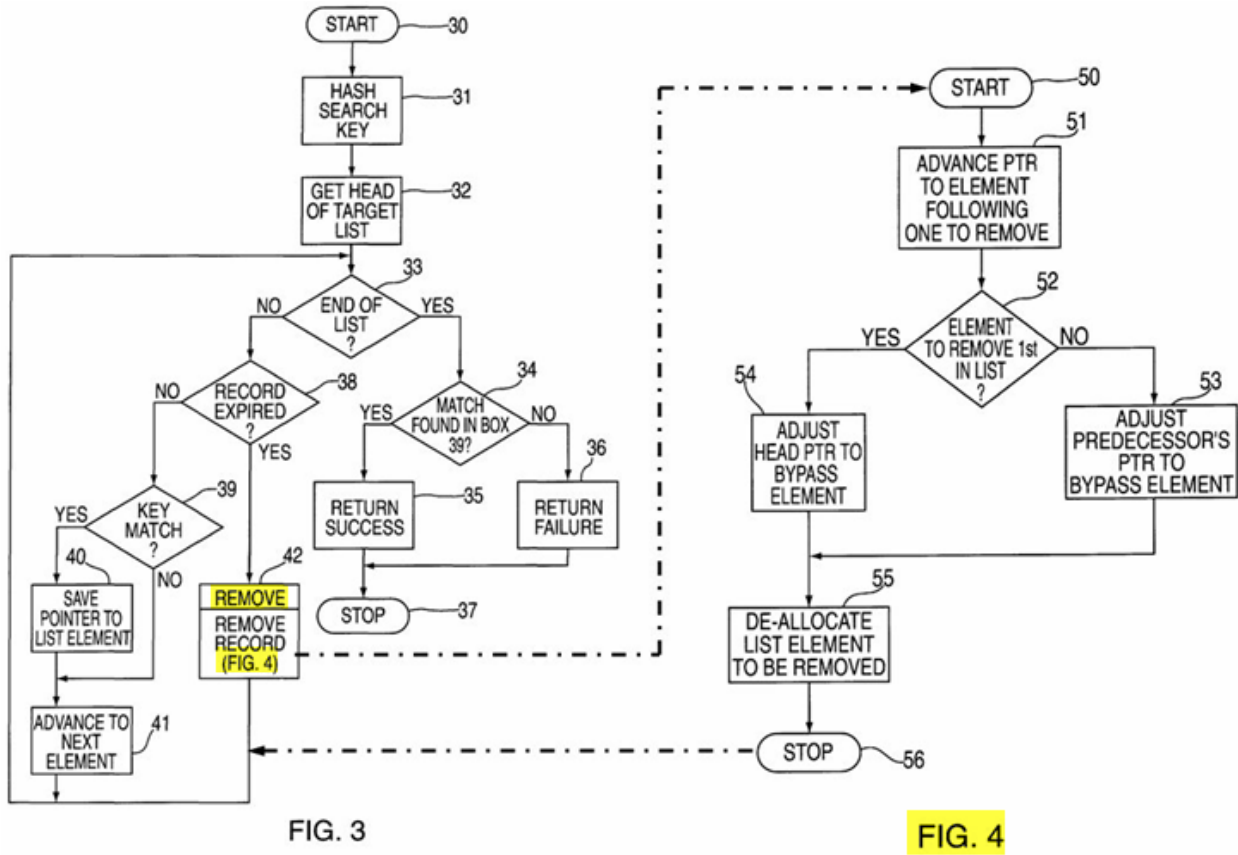
FIG. 3   FIG. 4

According to the specification, Figure 4 is part of the algorithm depicted in Figure 3. *See* 3:33-35 ("FIG. 4 shows a general flow chart for a linked-list element remove procedure that forms part of the table searching operation of FIG. 3); 6:21-25 ("If decision box 38 determines that the record under question has expired, box 42 is entered to perform the on-the-fly removal of the expired record from the linked list and the return of the storage it occupies to the system storage pool, as will be described in connection with FIG. 4." (emphases added). Accordingly, the algorithm in Figure 4, which requires adjusting the pointers in the linked list to bypass the previously identified expired records and deallocating the memory occupied by those records while traversing the linked list, is part of the means-plus-function claim limitation. For similar reasons, the "Remove Procedure" in the pseudocode, which also adjusts pointers and de-allocates memory, is also part of the corresponding algorithm. (13:7-20; 14:7-20.)

26

C. **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 the Records in the Linked List (Claim 1)/Means, 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)**

Defendants' Construction	*Bedrock's Construction*
Function: utilizing the record search means, [accessing the linked list / inserting, retrieving, and deleting from the system] and, at the same time, removing at least some of the expired ones of the records in the linked list.	**Function**: utilizing the record search means, [accessing the linked list / inserting, retrieving, and deleting from the system] and, at the same time, removing at least some of the expired ones of the records in the linked list.
"At the same time" means during the same traversal of the linked list as [accessing the linked list / inserting, retrieving, and deleting records from the system].	**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, 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.
For the construction of "removing …", see Argument, Section III.C above, as further described in Joint Claim Construction Statement [Dkt. 251].	
Structure: Boxes 10 and 11 of Fig. 1; Figs. 4-7, pseudocode in the Search Procedure (cols. 11-14), Insert Procedure (cols. 9 and 10), Retrieve Procedure (cols. 9 and 10), Delete Procedure (cols. 11-12), and Remove Procedure (cols. 13-14), and corresponding portions of the specification.	
Inserting, retrieving, and deleting are all required.	

These claim terms, which appear in claims 1 and 5, are closely related. Claim 1 recites a "means . . . for accessing the linked list and, at the same time, removing some of the expired ones of the records in the linked list," and claim 5 recites a "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." The parties essentially agree on the functions of these means-plus-function limitations, but have several disagreements regarding the corresponding algorithms.

First, the parties dispute whether the "means . . . for accessing" in claim 1 requires corresponding algorithms for inserting, retrieving *and* deleting or whether it only requires an algorithm for inserting, retrieving *or* deleting.[12] Bedrock asserts that "any one of the insert, retrieve, or delete structures can serve as the corresponding structure" for the means "for accessing the linked list." Br. 23. However, the specification explicitly states that all three algorithms/structures are "<u>necessary</u> to implement an information storage and retrieval system operating in accordance with <u>the present invention</u>." (8:60-63) (emphases added). Furthermore, the specification, drawings and pseudocode repeatedly and consistently refer to accessing as including inserting, retrieving *and* deleting. Nothing in the specification states or even suggests that accessing can somehow be limited to just inserting, retrieving or deleting. Accordingly, all three are part of claim 1.

Second, Bedrock again asserts that Figure 4 and the corresponding pseudocode are not part of the required structure. However, the claim language requires a means for "accessing the linked list <u>and</u>, at the same time, <u>removing</u> at least some of the expired ones of the records" in the linked list. (emphases added). Bedrock inexplicably points to figures and pseudocode for the first function—accessing—while ignoring the corresponding figures and pseudocode for the second function—removing. The '120 patent only discloses one algorithm for removing records: Figure 4 and the accompanying pseudocode. "Remove Procedure" at 13:7-20; 14:7-20. That algorithm, which includes adjusting the pointers and deallocation of memory, is accordingly corresponding structure for these means-plus-function terms. *Harris Corp.*, 417 F.3d at 1249.

Third, Bedrock completely ignores the algorithmic structure that corresponds to "at the same time" language in the function. Each and every algorithm in the specification requires that the removal of expired records occur *during the same traversal* of the linked list as the access/insertion,

---

[12]   The parties agree that claim 5 requires all three. (Br. 23.)

28

retrieval or deletion of the record from the system. Indeed, this is described as the "HEART OF THE TECHNIQUE" in the pseudocode. (11:25-40; 12:25-40 ("Traverse entire list, deleting . . . expired records as we search.")). Accordingly, this algorithmic structure corresponds and is integral to the function recited in these terms.

VI. ORDERING OF METHOD CLAIM STEPS (CLAIMS 3, 7)

Defendants' Construction	Bedrock's Construction
The elements of claims 3 and 7 must be executed in order. Moreover, "when the linked list is accessed" in the removing step refers to the accessing step, and the identifying and removing steps must occur during the same traversal of the linked list of records.	No construction needed. If the Court is inclined to address this issue, then it should hold that the steps of claims 3 and 7 may be performed in a consecutive manner, in an overlapping manner, or a combination of the two, except that the ultimate step of claim 7 must follow or at least partially follow the penultimate step of claim 7.

"Unless the steps of a method actually recite an order, the steps are not ordinarily construed to require one. However, such a result can ensue when the method steps implicitly require that they be performed in the order written." *Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363, 1369 (Fed. Cir. 2003) (citation omitted). The first step is to look at "the claim language to determine if, as a matter of logic or grammar, they must be performed in the order written." *Id.* If not, the court looks to the rest of the specification. *Id.* at 1369-1370. Here, both the claim and the specification confirm Defendants' construction.

Claim 3	Claim 7
The three steps of claim 3 are:	The four steps of claim 7 are:
[a] accessing the linked list of records,	[a] accessing a linked list of records having same hash address,
[b] identifying at least some of the automatically expired ones of the records, and	
	[b] identifying at least some of the automatically expired ones of the records,
[c] removing at least some of the automatically expired records from the linked list when the linked list is accessed.	[c] removing at least some of the automatically expired records from the linked list when the linked list is accessed, and

	[d] inserting, retrieving or deleting one of the records from the system following the step of removing.

As a matter of logic and grammar, steps [a] through [c] must be performed in the recited order.

Logically, the linked list of records must be accessed in step [a] prior to determining whether any of

those records in the linked list have automatically expired in step [b]. *See* Argument, Section III.A

and III.C. Likewise, step [c] cannot remove automatically expired records until such records have

been identified in step [b]. *See* Argument, Section III.B. Accordingly, step [b] must begin prior to

the commencement of step [c]. Furthermore, step [c] expressly claims removal "when the linked list

is accessed." Based on this plain language, the steps of identifying and removing must occur while

continuing to access the linked list of records set forth in step [a]. Bedrock's unsupported contrary

arguments do not address the ordering of the steps imposed by the logic and grammar of the claim

language and the clear bounds of the alleged invention set forth in the specification.

CONCLUSION

For the foregoing reasons, Defendants respectfully request that their proposed claim constructions be adopted.

DATED: September 10, 2010   Respectfully submitted,

 /s/ Yar R. Chaikovsky
Yar R. Chaikovsky
California State Bar No. 175421
MCDERMOTT WILL & EMERY LLP
275 Middlefield Road, Suite 100
Menlo Park, CA 94025
Tel: 650.815.7400
Fax: 650.815.7401
E-mail: ychaikovsky@mwe.com
ATTORNEY FOR DEFENDANT
Yahoo! Inc.

 /s/ Todd M. Briggs (with permission)
Claude M. Stern
claudestern@quinnemanuel.com
Todd M. Briggs
toddbriggs@quinnemanuel.com
QUINN EMANUEL URQUHART
OLIVER & HEDGES, LLP
555 Twin Dolphin Dr., Suite 560
Redwood Shores, CA 94065
Tel: 650.801.500
Fax: 650.801.5100

Michael E. Jones
State Bar No. 10929400
mikejones@potterminton.com
POTTER MINTON
110 N. College
Tyler, TX 75702
Telephone: 903.597.8311
Fax: 903.593.0846

ATTORNEYS FOR DEFENDANTS
Google, Inc. and Match.com, LLC

 /s/ E. Danielle T. Williams (with permission)
Steven Gardner
E. Danielle T. Williams
John C. Alemanni
Alton Absher II
KILPATRICK STOCKTON LLP
1001 West 4th Street
Winston-Salem, NC 21701
Tel: 336.607.7300
Fax: 336.607.7500

William H. Boice
Russell A. Korn
KILPATRICK STOCKTON LLP
Suite 2800
1100 Peachtree Street
Atlanta, GA 30309-4530
Telephone: 404.815-.6500
Fax: 404.815.6555

J. Thad Heartfield
Texas Bar No. 09346800
thad@jth-law.com
M. Dru Montgomery
Texas Bar No. 24010800
dru@jth-law.com
THE HEARTFIELD LAW FIRM
2195 Dowlen Road
Beaumont, TX 77706
Telephone: 409.866.2800
Fax: 409.866.5789

ATTORNEYS FOR DEFENDANTS
Softlayer Technologies, Inc. and Amazon.com, Inc.

  /s/ Marissa S. Ducca (with permission)
Alan L. Whitehurst
Alan.whitehurst@alston.com
Marissa R.Ducca
Marissa.ducca@alston.com
ALSTON& BIRD LLP
The Atlantic Building
950 F. Street, N.W.
Washington, D.C. 2004
Tel: 202.756.3300
Fax: 202.756.3333

Frank G. Smith
Frank.smith@alston.com
ALSTON & BIRD LLP
The Atlantic Center
1201 West Peachtree Street
Atlanta, GA 30309
Telephone: 404.881-7240
Fax: 404.256.8184

Michael J. Newton (SBN 24003844)
mike.netwon@alston.com
ALSTON & BIRD LLP
Chase Tower
2200 Ross Avenue, Suite 3601
Dallas, TX 75201
Telephone: 214.922.3423
Fax: 214.922.3839

Louis A. Karasik
lou.karasik@alston.com
ALSTON & BIRD LLP
333 South Hope Street
16th Floor
Los Angeles, CA 90071
Telephone: 213.576.1148
Fax: 213.576.1100

ATTORNEYS FOR DEFENDANTS
MySpace, Inc. and AOL LLC.

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a copy of the foregoing document was filed and served electronically on all counsel of record in compliance with Local Rule CV-5(a) on September 10, 2010.

By: _/s/ Yar R. Chaikovsky_____
Yar R. Chaikovsky