

**IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION**

**BEDROCK COMPUTER
TECHNOLOGIES LLC,**

Plaintiff,

v.

**SOFTLAYER TECHNOLOGIES,
INC.,
et al.**

Defendants.

§
§
§
§
§
§
§
§
§
§
§

CASE NO. 6:09-cv-269-LED

Jury Trial Demanded

DECLARATION OF MARK T. JONES, PH.D

1. I, Mark Jones, declare under penalty of perjury that the following is true and correct.

2. I have been retained by McKool Smith, P.C., counsel for Bedrock Computer Technologies, LLC, as an expert on the lawsuit as captioned above.

3. I am a Professor of Electrical and Computer Engineering at Virginia Tech in Blacksburg Virginia. I graduated *summa cum laude* from Clemson University in 1986 with a B.S. in Computer Science and a minor in Computer Engineering while holding a National Merit Scholarship and the R. F. Poole Scholarship. I then graduated from Duke University in 1990 with a PhD in Computer Science while holding the Von Neumann Fellowship. A detailed record of my professional qualifications is set forth in the attached Appendix B to this Declaration, which is a curriculum vitae, including a list of publications, awards, research grants, and professional activities.

4. I understand that Bedrock Computer Technologies (“Bedrock”) has sued the defendants for infringement of United States Patent No. 5,893,120 (the “’120 patent”).

5. I have reviewed the ’120 patent and the Patent Office file history for this patent. In addition, I have reviewed each party’s Preliminary Claim Construction filings and the parties’ Joint Claim Construction and Pre-Hearing Statement. In forming the opinions given below, I have considered these documents from the perspective of one of ordinary skill-in-the-art at the time of the invention.

6. I understand that the specification is to be interpreted from the point of view of a person of ordinary skill in the art. I further understand that the level of ordinary skill in the art is a function of many factors, including (1) the educational level of the inventor; (2) type of problems encountered in the art; (3) prior art solutions to those problems; (4) rapidity with which innovations are made; (5) sophistication of the technology; and (6) educational level of active workers in the field.

7. I find the pertinent art to lie generally in the field of computer-based information storage and retrieval systems. Considering all of the factors in the context of the technology of the ’120 patent, I believe that a person of ordinary skill in the art would have a Bachelor of Science degree in computer science or computer engineering, including practical experience writing computer programs.

8. I have been asked to give my opinions regarding several of the terms in the ’120 patent. In particular, I have been asked to give opinions regarding several means-plus-function claim terms. I understand that a patent may claim a means for performing a specified function without claiming the structure that performs the function.

I understand that this type of claim term is referred to as a “means-plus-function” term. I further understand that the specification must recite some structure corresponding to the claimed means. I understand that the corresponding structure of a means-plus-function limitation must be disclosed in the written description in such a manner that one skilled in the art will know and understand what structure corresponds to the means limitation.

9. The patent includes flow charts in Figures 3-7. One of ordinary skill in the art would view these flowcharts (and corresponding descriptions in the specification) as detailed descriptions of algorithms and understand how to implement them as computer software. The patent includes an Appendix with pseudocode in columns 9-14. One of ordinary skill in the art would view the pseudocode as detailed descriptions of algorithms and understand how to implement them as computer software. Both flowcharts and pseudocode are commonly used in computing to describe algorithms.

Term 3: “a record search means utilizing a search key to access the linked list”
[Claim 1]

10. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as “record searching utilizing a search key to access the linked list” and the corresponding structure as “(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 31-36 and Boxes 39-41 of FIG. 3, or as portions 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, line 57-col. 6 line 4 and col. 6 lines 15-20, or the equivalents thereof.” The Defendants contend that this term is indefinite. For the reasons given below, I agree that Bedrock’s construction is proper and disagree that the term is indefinite.

11. To one of ordinary skill in the art, the ’120 patent contains multiple disclosures of a structure that is “a record search means utilizing a search key to access the linked list.” The specification discloses that the invention can be implemented in application software packages, user access software, or operating system software that is part of a computer system as illustrated in Figures 1 and 2. [e.g., 3:53-56, 4:22-24, 4:45-48] The specification describes these types of software. [e.g., Figure 2, 4:24-44]. Further, the specification describes this software as running on a computer system such as the one illustrated in Figure 1. [e.g., 4:22-24] The specification makes it clear that the two components of the computer system that are required to run the software are a Central Processing Unit (CPU) and a Random Access Memory (RAM) unit. The specification indicates that programs are stored in RAM, they are accessed and executed by the CPU, and that data is also stored in RAM and is operated upon by the program instructions. [e.g., 3:53-61] This is consistent with the understanding of one of ordinary skill in the art.

12. Note that portion (1) of the structure is common to Bedrock’s constructions for each of the disputed “means-plus-function” terms. One of ordinary skill in the art would understand that this because the specification describes the present invention as being implemented in such a structure as described above.

13. The specification discloses accessing an external chain (a linked list) using a search key, where a hash function is computed upon the search key to arrive at a hash

address and access the proper external chain. See, for example, Figure 3, boxes 31 and 32; the first two lines after the first “begin” in the “Search Table Procedure” and “Alternate Version of the Search Table Procedure” in columns 11, 12, 13, and 14; and described in col. 5:57-63. In each of these examples, one of ordinary skill in the art would understand that a design for executable software is being described, in the form of text, flowcharts, and detailed pseudocode. Hash functions were well-known to one of ordinary skill in the art. One of ordinary skill in the art would have been familiar with the term, concept, implementation, and use of a hash function, as well as the fact that multiple types of hash functions were available to practitioners. The Background of the Invention makes this clear and cites to specific pages of textbooks. [1:23-2:6] One such textbook, “The Art of Computer Programming” by Knuth is considered a classic work in computer science.¹ Another reference is a textbook that was used (in an earlier edition) when I took a data structures course early in my undergraduate curriculum (such a course was often offered at the sophomore level in a computer science curriculum). When describing (or implementing) an algorithm that uses a hash function, it was (and remains) standard practice to embody the hash function as a separate function because, for example, (a) a range of hash functions are available, (b) the module using the hash function can be described (and implemented) in a way that is independent of the particular hash function chosen, and (c) one would want the convenience of quickly substituting one hash function for another. Figure 3, as well as the “Search Table Procedure” and “Alternate Version of the Search Table Procedure,” disclose loops in

¹ *American Scientist* (1999) named this one of the “100 or so Books that shaped a Century of Science.” See <http://www.americanscientist.org/bookshelf/pub/100-or-so-books-that-shaped-a-century-of-science>

which the linked list indicated by the hash function is searched for a record that matches the search key. In sum, it is my opinion that these structures, together with the structures for the rest of the claim terms, would render the bounds of the claim understandable to an ordinary artisan.

Term 4: “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 [of records] when the linked list is accessed” [Claims 1 and 5]

14. This term provides a further limitation on the record search means. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as “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” and the corresponding structure as “(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) including the lines “while ... /*HEART OF THE TECHNIQUE...”, and/or as described in col. 5, line 63 - col. 6, line 34, or the equivalents thereof.” The Defendants construe the function to be similar to Bedrock’s but excluding the “record searching including” that is part of the function as described in the claim. The Defendants construe the means as “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.”² For the reasons given below, I agree that Bedrock’s construction is proper and disagree that the Defendants’ construction is proper.

15. I have already given the reasons above for why I agree with part (1) of the structure that Bedrock has put forward in its construction. Further, it appears that the Defendants agree at least with the portion of Bedrock’s construction that requires a CPU and RAM.

16. Bedrock’s construction properly includes in the structure the portion of the disclosed algorithms that includes the traversal of the linked list and the comparison to the search key. [e.g., Boxes 33-42 of Figure 3] Included in record searching is the identification of expired records and the removal of those records. [e.g., Boxes 38 and 42 of Figure 3]

17. To the extent that the Defendants argue that the “record search means” in claims 1 and 5 does not require a means to search for records, I disagree. My discussion of these terms makes it clear that the specification discloses a “record search means” and that this record search means includes searching for records using a search key to (a) locate the linked list and (b) to search for records while traversing the linked list. Construing the claims to read this out would be inconsistent with the disclosed algorithms in the specification as well as the language of the claims.

² The Defendants conclude that due to the inclusion of “the record search means” the claim is indefinite because they have concluded that this term is indefinite. I disagree for the reasons given in my discussion of Terms 3 and 12.

Term 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 the records in the linked list” [Claim 1]

18. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as “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” and the corresponding structure as “(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 reasons given below, I agree that Bedrock’s construction is proper.

19. I have already given the reasons above for why I agree with part (1) of the structure that Bedrock has put forward in its construction. Further, it appears that the Defendants agree at least with the portion of Bedrock’s construction that requires a CPU and RAM.

20. Bedrock’s construction properly includes in the structure the algorithms that use the search means to perform inserting, retrieving, or deleting records. [e.g., the insert(), retrieve(), or delete() functions in columns 9-12] These algorithms make use of

the search_table() function (portions of which are identified as the record search means in Bedrock's construction) to access the linked list for the purpose of finding a record and, when the linked list is accessed, remove expired records from the linked list (as described above). Note that the search_table() function may be the one in "Search Table Procedure" in columns 11 and 12 or the one in "Alternate Version of Search Table Procedure" in columns 11-14.

21. The Defendants' construction is not proper because it suffers from several flaws.³ Most notably, the Defendants' seem to ignore a bulk of structure from the specification as alternative structures to each other. One of ordinary skill in the art would understand that claim 1 merely requires that one of these be present, rather than all three, particularly in light of the language in claim 5.

Term 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]

22. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as "dynamically determining maximum number of records for the record search means to remove in the accessed linked list of records" and the corresponding structure as "(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

³ The Defendants conclude that due to the inclusion of "the record search means" the claim is indefinite because they have concluded that this term is indefinite. I disagree for the reasons given in my discussion of Terms 3 and 12.

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." The Defendants' argue that the term is indefinite. For the reasons given below, I agree that Bedrock's construction is proper and disagree that the term is indefinite.

23. I have already given the reasons above for why I agree with part (1) of the structure that Bedrock has put forward in its construction.

24. Bedrock cites to col. 6, line 56 – col.7, line 15 for the description of the algorithm for the executable software in part (2) of the means in its construction. The Defendants cite to the same section, presumably arguing that no algorithm is disclosed in this text. Note that one of ordinary skill in the art does not require pseudocode and/or flowcharts to teach (or understand) an algorithm, an algorithm may be described in text form as is the case in this section of the specification. The result of the function for this term is a maximum number of expired records to be removed. The algorithm for arriving at this result can be quite simple. The specification teaches several ways to find this

result. For example, the specification teaches at col. 6, line 66 – col. 7, line 4⁴ that a dynamic decision among several strategies can be made regarding the number of expired records to remove (all, some, or none). It then explains that the way in which this decision can be made at col. 7 lines 4-10. The way the decision is made is to base it upon a factor(s) internal and/or external to the information storage and retrieval system. Examples of such factors are given, including available memory, system load, and the number of records.⁵ One of ordinary skill in the art would understand that there is a tradeoff to be made between “shortening the linked list traversal time and speeding up the search” and “the expense of perhaps leaving some expired records in the list.” [col. 6, lines 61-63] The patent discusses search times and system memory elsewhere in the specification. [e.g., col. 5, lines 41-52] One of ordinary skill in the art would have understood that the algorithm disclosed here chooses a limitation on the record deletion based on a tradeoff between using processor time to traverse more of the list (and spend time deleting records) versus (potentially) deleting more records to free up more space. One of ordinary skill in the art would have understood that the specific limitation would depend on the details of the particular information storage and retrieval system. The specific combination of factors and weights accorded to each factor would be very system dependent. A simple example of pseudocode for such an algorithm is as follows:

if (available system memory is greater than 25MB) **then** max_to_delete = 0

⁴ As well as col. 7 lines 10-15, where it is explained that all records, no records, or a specific number of records may be deleted.

⁵ Note that “time of day” is also included. One of ordinary skill in the art would understand that for many computer systems, the system processing load and the memory usage is highly correlated with the time of day.

In sum, it is my opinion that these structures, together with the structures for the rest of the claim terms, would render the bounds of the claim understandable to an ordinary artisan.

Term 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]

25. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as “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” and the corresponding structure as “(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.*/” which point to records of type “list_element” in cols. 9-10 that allocates in memory an external chaining hash table, and/or as described in col. 5, lines 16-41, or the equivalents thereof.” The Defendants contend that this term is indefinite. For the reasons given below, I agree that Bedrock’s construction is proper and disagree that the term is indefinite.

26. I have already given the reasons above for why I agree with part (1) of the structure that Bedrock has put forward in its construction.

27. To one of ordinary skill in the art, this function identified for this term is performed by the hash table data structure pseudocode⁶ identified in Bedrock's construction. The pseudocode and the description provide a specific description of a hash table data structure that uses external chaining. One of ordinary skill in the art could readily implement this algorithmic description using a programming language. In sum, it is my opinion that these structures, together with the structures for the rest of the claim terms, would render the bounds of the claim understandable to an ordinary artisan.

Term 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]

28. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as “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” and the corresponding structure as “(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), respectively, and/or described in col. 7, line 65 -

⁶ Or the description in col. 5, lines 16-41.

col. 8, line 32, col. 8, lines 33-44, and col. 8, lines 45-59, or the equivalents thereof.” For the reasons given below, I agree that Bedrock’s construction is proper.

29. I have already given the reasons above for why I agree with part (1) of the structure that Bedrock has put forward in its construction. Further, it appears that the Defendants agree at least with the portion of Bedrock’s construction that requires a CPU and RAM.

30. Bedrock’s construction properly includes in the structure the algorithms that use the search means to perform inserting, retrieving, and deleting records. [e.g., the insert(), retrieve(), and delete() functions in columns 9-12] These algorithms make use of the search_table() function (portions of which are identified as the record search means in Bedrock’s construction) to access the linked list for the purpose of finding a record and, when the linked list is accessed, remove expired records from the linked list (as described above). Note that the search_table() function may be the one in “Search Table Procedure” in columns 11 and 12 or the one in “Alternate Version of Search Table Procedure” in columns 11-14.

Term 12: “a record search means utilizing a search key to access a linked list of records having the same hash address” [Claim 5]

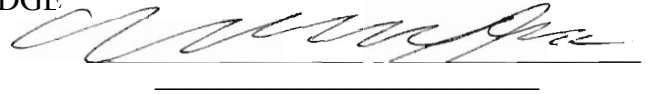
31. The parties appear to agree that this claim element is subject to interpretation under 35 U.S.C. §112, ¶6. Bedrock identifies the function as “record searching utilizing a search key to access a linked list of records having the same hash address” and the corresponding structure as “(1) Portions 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 31-36 and Boxes 39-41 of FIG. 3, or as portions 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, line 57-col. 6 line 4 and col. 6 lines 15-20, or the equivalents thereof.” The Defendants contend that this term is indefinite. For the reasons given below, I agree that Bedrock’s construction is proper and disagree that the term is indefinite.

32. The primary difference between Term 3 (in claim 1) and this term is that the records in the linked list have the same hash address. The specification describes the hash address as the index that results from computing the hash function. [e.g., 2:58-62] The structure given in Bedrock’s construction for this term is the same structure given for Term 3. Each of the elements in (2) of Bedrock’s structure describes a technique in which all records in the linked list have the same address. Note that each record in the list can have a different search key, but every record in the same linked list will have the same hash address. See the discussion regarding Term 3 for further reasons why Bedrock’s structure is proper. In sum, it is my opinion that these structures, together with the structures for the rest of the claim terms, would render the bounds of the claim understandable to an ordinary artisan.

I DECLARE UNDER THE PENALTY OF PERJURY UNDER THE LAWS OF THE
UNITED STATES OF AMERICA THAT THE FOREGOING IS TRUE AND
CORRECT TO THE BEST OF MY KNOWLEDGE

Date: August 27, 2010

A handwritten signature in black ink, appearing to read 'Mark T. Jones', is written over a horizontal line.

Mark T. Jones, Ph.D