

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Teradata Operations, Inc.
Petitioner

v.

Berkeley*IEOR
Patent Owner

Case Nos. CBM2019-00011, CBM2019-00014
Patent No. 8,612,316

DECLARATION OF JON SCARBROUGH

TABLE OF CONTENTS

I.	Introduction.....	1
II.	Background and Qualifications.....	1
II.	Materials Reviewed	4
III.	Legal Standards	4
A.	Claim Construction	4
B.	Patent Eligible Subject Matter.....	5
IV.	Summary of Opinions	7
V.	The Lepman Patents.....	7
A.	Relationships Among the Patents and Claims in This Three- Patent Family	7
B.	Improvements of the Lepman Patents	9
1.	Business Problem and Solution: More Detailed Profitability Measures	9
2.	Technical Problem and Solution: Processing Object Level Profitability on a Massive Scale.....	10
3.	Taking Advantage of an RDBMS's Calculation Capabilities	11
VI.	Pen and Paper	24
VII.	Claim 1 is Valid Under Both <i>Alice</i> Step One and <i>Alice</i> Step Two.....	25

I. Introduction

1. I have been retained by Patent Owner, Berkeley*IEOR, as an independent expert consultant to provide this declaration concerning the technical subject matter relevant to US 7,596,521, US 7,882,137, and US 8,612,316 (collectively, the “Lepman Patents”).

2. I am being compensated at my standard hourly rate of \$150 per hour for the time I spend on this matter. My compensation is not related in any way to the outcome of this proceeding, and I have no other interest in this proceeding.

II. Background and Qualifications

3. I have over 33 years of experience in computer science and information systems.

4. My experience extends from teaching college computer science to designing, developing and deploying, among many others, data warehouses, accounting and finance software (including, for example, general ledger, payroll and ERP systems), CRM systems, local and wide area networks, client/server ERP systems, and scalable Internet systems deployed at many multinational corporations, including Disney, Ben & Jerry’s, Moneygram, HP, and Safeway.

5. During the late 1990s, I was the CIO at Oakton Community College, a large community college in the Chicago suburbs, responsible for all IT functions. While there I oversaw the implementation of an Oracle-based enterprise finance

system, development of a new enterprise student registration system, as well as many IT infrastructure improvements.

6. From 2000 – 2002, I was the CIO of Group Arnold, a manufacturer of magnetic products, and responsible for all IT functions. While there I oversaw the design and deployment of several information systems responsible supporting the company's management accounting, including a new HR/payroll platform, Oracle Data Warehouse, general ledger system and ERP system.

7. From 2005 – 2016, I was the CTO of WHERE 2 GET IT, INC. (d/b/a Brandify), a digital marketing company, responsible for all technology functions for an enterprise serving over 500 national and multi-national brands. While there I oversaw the creation of several new software products and scaled the infrastructure to support millions of consumer's store location inquiries on a daily basis.

8. From late 2016 through early 2017, I was the Senior Director of Software Engineering at DocuTAP, a provider of electronic medical records software for the urgent care industry.

9. Presently I am the Director of IT for SDN Communications where I manage all corporate IT functions including application development, the corporate network and end user computers, and business systems support of commercial applications used in different departments such as accounting, network operations, engineering and sales.

10. Throughout my career, I have become proficient in a multitude of programming and query languages. These vary from Smalltalk, Perl, Javascript, NodeJS, PostgreSQL, and Oracle to analyzing company acquisition targets where the technology ranged from .Net, PHP, Java, Mysql and SQL Server.

11. As a result of my various roles throughout my career -- ranging from programmer and analyst to serving as the CTO / CIO at multiple corporations -- I have extensive experience studying and understanding the various architectural layers comprised in enterprise software solutions and the benefits and drawbacks of choosing among the different products and arrangements.

12. During my work years with Where2Getit, I was called upon to provide technology insight as part of Where2GetIt's patent defense in 3 different patent litigations. I was required to testify in one of the patent cases. While at Where2GetIt, I filed one patent application surrounding one of the unique technology processes we created.

13. In 1981 I received a Bachelor's of Arts in Mathematics from Augustana University. In 1987 I received a Master of Science in Computer Science from Indiana State University.

14. A copy of my curriculum vitae setting forth additional details on my background and qualifications is attached as Appendix A to this declaration.

II. Materials Reviewed

15. In forming my opinions expressed below, I considered each of the materials identified in the paragraphs and sections referenced in this Declaration. I have also reviewed the Exhibits identified on pages “ii” and “iii” of the CBM petitions challenging the claims of the ‘316 patent, which I understand corresponds to case nos. CBM2019-00011 and CBM2019-00014.

16. The exhibit numbers I refer to in this declaration are identified on pages “ii” and “iii” of both Petitions. In this declaration I also cite to Ex. 2002 being offered by Patent Owner. Ex. 2002 is the File History for U.S. Patent No. 7,596,521.

17. Further, in regards to my opinions herein, I rely on my own knowledge, training, and more than 33 years of experience of technology use in various industries.

III. Legal Standards

18. I am not a lawyer. However, in forming my opinions, I have been made aware of and understand the following legal standards.

A. Claim Construction

19. I understand the “broadest reasonable interpretation” (“BRI”) claim construction standard applies. I further understand this means that terms are to be given their plain meaning to a person of skill in the art (“POSITA”) unless the meaning is inconsistent with the (i) specification or (ii) for issued patents like the

Lepman Patents, the patent's prosecution history.

20. I understand Petitioner states a POSITA for the Lepman Patents would have been as follows:

A person of ordinary skill in the art would have at least a Bachelor's Degree in computer science, accounting, or management information systems, or a similar field with at least two years of experience in accounting or finance software, or a person with a master's degree in computer science, accounting, or management information systems, or a similar field with a specialization in accounting or finance software. A person with less education but more relevant practical experience may also meet this standard.

CBM2019-000011 Petition at p. 15-16.

21. The opinions of this declaration apply claim terms according to their plain meaning to a person of skill in the art (POSITA) based on Petitioner's definition of what a POSITA would have been.

B. Patent Eligible Subject Matter

22. I understand that so called "abstract ideas" are not patent eligible.

23. I understand that the United States Supreme Court's decision in *Alice Corp. Pty. v. CLS Bank Int'l*, 137 S. Ct. 2347 (2014), provides a two-step test for determining if a claim is directed to eligible subject matter or an ineligible abstract idea.

24. In the first step, the Board must determine whether the claim is directed to an abstract idea. I understand that in this case Petitioner Teradata argues the claims of the Lepman Patents are directed to an abstract idea.

25. I understand that a specific implementation of a solution to a problem in the software arts is not an abstract idea. I also understand that software directed to improving the speed or efficiency of a database or computer is not an abstract idea.

26. I understand that is why Petitioner contends the Lepman Patents are merely “directed to solving a business problem, not improving the performance of a database or computer such as by increasing its speed or efficiency.” CBM2019-00011 Petition, p. 19.

27. I understand that, for *Alice* step one, courts favor claims that recite specific steps for achieving the desired result, as opposed to claims reciting the result alone.

28. If the Board agrees with Petitioner the claims are directed to an abstract idea, the second step is to determine whether the claim recites additional elements that amount to significantly more than the judicial exception, or, in other words, an “inventive concept.”

29. I understand a non-conventional and non-generic arrangement of known, conventional pieces qualifies as an “inventive concept” that is patent eligible.

30. I am told CBM review is unavailable for patents that are “for technological inventions.” I understand that whether a patent solves a technical

problem using a technical solution is relevant to the “technological invention” question.

IV. Summary of Opinions

31. Briefly, for reasons discussed in this declaration, it is my opinion claim 1 of each of the Lepman Patents satisfies *Alice* step one because it is not directed to an abstract idea, but instead to a specific implementation of a solution to a problem in the software arts.

32. It is also my opinion that, even if the Board were to find claim 1 comprises known, conventional pieces, claim 1 still satisfies *Alice* step two because claim 1 recites an “inventive step” for claiming a non-conventional and non-generic arrangement.

33. It is also my opinion that the Lepman Patents solve a technical problem using a technical solution.

V. The Lepman Patents

A. Relationships Among the Patents and Claims in This Three-Patent Family

34. I understand Petitioner, Teradata, filed six petitions challenging the claims of US 7,596,521, US 7,882,137, and US 8,612,316 (collectively, the “Lepman Patents”).

Case No.	Patent No.	Claim 1 or 2
CBM2019-00015	7,596,521	Claim 1
CBM2019-00016	7,596,521	Claim 2

CBM2019-00009	7,882,137	Claim 1
CBM2019-00013	7,882,137	Claim 2
CBM2019-00011	8,612,316	Claim 1
CBM2019-00014	8,612,316	Claim 2

35. While this declaration refers to and uses citations to the ‘316 patent (Ex. 1001 to CBM 2019-00011) I have reviewed each of the Lepman Patents and observed each shares a common specification, aside from minor differences.

36. Appendix B to this Declaration reproduces the language of claim 1 of each Lepman Patent. I note that, while there are some claim language differences among the claim 1s of each patent, each recites an identical limitation, which reads:

[e](i) using the relational database management system to (ii) independently calculate at least one marginal value of profit for each object being measured (iii) using the established rules as applied to a selected set of prepared information.

See Appendix B. These words are identified as claim limitation 1[d] in the ‘521 patent (i.e. step [d] of claim 1), and limitation 1[e] in the ‘137 and ‘316 patents. For purposes of this declaration, I refer to it as claim limitation 1[e], corresponding to how it appears in the ‘316 patent.

37. Note that in the paragraph above and Appendix B, I have broken down claim limitation [e] into three subparts.

- Limitation 1[e(i)] refers to the phrase “using the relational database

management system to ... calculate”

- Limitation 1[e(ii)] refers to the phrase “independently calculate”
- Limitation 1[e(iii)] refers to the phrase “independently calculate ...
using the established rules as applied to a selected set of prepared
information”

38. The remaining claim 1 limitations of each Lepman Patent are similar enough, such that the opinions of this declaration apply to all three.

B. Improvements of the Lepman Patents

1. Business Problem and Solution: More Detailed Profitability Measures

39. Before the Lepman Patents were filed many businesses turned to financial statements to make decisions. The Lepman Patents state that traditional financial statements failed to help management accurately determine, for example, which customers, accounts or transactions positively and negatively impact profitability. Ex. 1001, 1:24-27.

40. To address this problem, the Lepman Patents disclose specific methods for calculating profitability associated with the “smallest common component of profit measurement desired,” referred to as the profit “object.” Ex. 1001, 6:9-10.

41. Examples of profit objects referenced in the Lepman Patent specifications include an insurance company using a “policy” object, a bank using

an “account” object, and an airline using an airplane “seat” object. Ex. 1001, 7:33-37.

2. Technical Problem and Solution: Processing Object Level Profitability on a Massive Scale

42. Prior to the filing of the Lepman Patents, others tried to calculate detailed profitability measures on a limited scale. For example, Ex. 1009 (“Stuchfield”) describes an information system called “BEATRICE.” Ex. 1009, p. 14. BEATRICE calculated profit values for an investment firm’s individual financial transactions. Ex. 1009, p. 14. BEATRICE processed only 6,000 trade transactions each day on average and calculated profitability just once every three months. Ex. 1009, p. 14, 20.

43. Mr. Lepman worked to develop a profit calculator capable of processing far more than 6,000 transactions a day and calculating profitability just once per business day rather than every three months. In the file history Mr. Lepman mentions “allow[ing] the information to be analyzed in hours instead of days” and for large banks Ex. 1015, p. 263.

44. I reviewed the Lepman Patent specifications and ‘316 patent file history [Ex. 1002]. I also reviewed the file histories of the ‘521 patent [Ex. 2002] and application No. 09/545,628 [Ex. 1015], which I understand are relevant to the ‘316 patent claims, which issued subsequently. I observed several technological improvements disclosed in these materials, two of which I address in detail in this

declaration: (1) moving the calculation engine from the software application server (where accounting calculations were traditionally performed) to the relational database management system (“RDBMS”) itself; and (2) decomposing profitability factors into “independent” calculations capable of being processed simultaneously on separate processors (i.e. in parallel).

3. Taking Advantage of an RDBMS’s Calculation Capabilities

45. The specification describes moving the calculation engine from the software application server (where accounting calculations were traditionally performed based on debit and credit transactions) to the RDBMS itself.

This approach allows for a **relational database management system implementation** (42). It is nearly impossible [to] develop and maintain **procedural based software** with as much flexibility and with the capability to simultaneously support the number of calculation permutations required

Ex. 1001, 10:42-47 (emphasis added).

46. The above passage mentions both a “relational database management system” (“RDBMS”) and “procedural based software.” I will discuss each below.

a. Relational Database Management System (RDBMS)

47. The textbook Petitioner marked as Ex. 1014, *Introduction to Database Systems* 6th ed, describes the RDBMS as including the “software that **handles all**

access to the database.” Ex. 1014, p. 39 (“[t]he database management system (DBMS) is the software that **handles all access to the database**”) (emphasis added).

48. I largely agree with that statement, but believe additional background on the RDBMS components and functions provides helpful context.

49. At its core, an RDBMS is about organizing data and supporting methods of retrieving the data. The data itself is typically stored in electronic form as files on a hard drive.

50. One component in the RDBMS provides the intelligence of how to access the files and what files contain what data. This data file access is normally hidden from users of the RDBMS as SQL (structured query language) is the technology facilitating user access to the data.

51. There are typically two means of using SQL to access the data. First, the RDBMS normally provides an administrative application that supports direct access to the underlying database files using SQL.

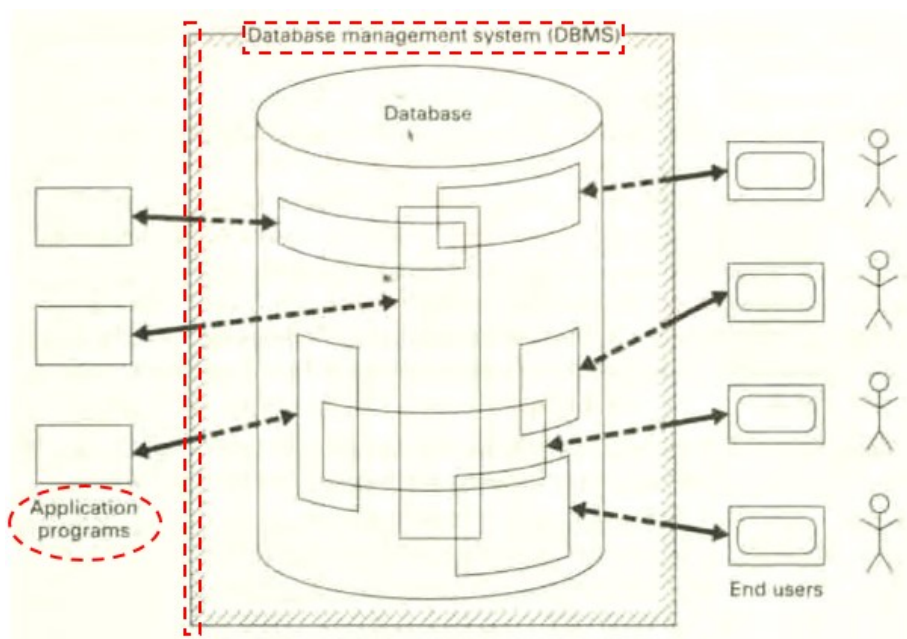
52. Second, software applications (such as SAP’s enterprise Cost Accounting application) may issue instructions that cause an RDBMS to access the database files. Ex. 1007, p. 13. When instructed to do so, the RDBMS can access the database files by executing a SQL command that may be provided by the software application. Software applications are a typical consumer of the data

stored in the database files.

53. A RDBMS will also contain a security component to require users and applications to authenticate themselves to gain access to the database. The security component applies authorization levels restricting what data the user or application may access once granted access to the RDBMS system.

54. The *Database Systems* textbook (Ex. 1014) illustrates these concepts.

Fig. 1.4 shows the DBMS as an *intermediate* layer disposed between the “[A]pplication programs” and “[D]atabase”:



Ex. 1014, FIG. 1.4 (p. 5) (annotated)

55. In other words, “applications programs” seeking access to database files rely on the [R]DBMS for such access. Ex. 1014, FIG. 1.4 (p. 5).

56. A POSITA would have considered the RDBMS to also encompass the database or database files themselves, but this distinction does not appear to be in

dispute.

57. RDBMS is an acronym for “relational database management system.” Some literature, such as the *Database Systems* textbook (Ex. 1014), use the term “DBMS,” an acronym for “database management system,” in referring to essentially the same thing. The only difference between RDBMS and DBMS is the “R,” where the “R” specifies that the system corresponds to a management system for a relational database. A relational database is a specific type of database.

b. Procedural Based Software

58. In the late 1990s, accounting platforms used by large enterprises relied on multi-tier architecture designs. For example, SAP-IS-B (Ex. 1007) discloses a three-tier architecture:

SAP-IS-B’s “three-tier” architecture

The standard R/3 System architecture consists of a three-tier hierarchy comprising:

- ❑ a host (database server) for carrying out all functions requiring access to the database.
- ❑ connected to this, several application servers for carrying out the actual dialog processing
- ❑ linked to this, in turn, several workstations (presentation servers) for the presentation functions.

Ex. 1007, p. 19 (annotated).

59. According to SAP-IS-B, the “database server” tier hosted the RDBMS, which “carr[ied] out all functions requiring access to the database.” Ex.

1007, p. 19.

60. The “application servers” tier performed the costing and profitability calculations. Ex. 1007, p. 10, 19, 60. This is made clear by turning to page 19 of the SAP-IS-B, which states that the application servers hosted and carried out “the actual dialog processing,” of SAP’s software applications. Ex. 1007, p. 19.

61. Next, page 10 and 60 of SAP-IS-B state that Cost Accounting and Bank Profitability Analysis modules (responsible for calculating costing and profitability) are “dialog-based” and “dialog-oriented.” 1007, p. 10, 19, 60. This is the same “dialog processing” that page 19 states occurs on the “application server” tier. 1007, p. 19.

62. Thus, it is clear that SAP-IS-B’s “application servers” tier performed the cost accounting and profitability calculations, rather than the RDBMS. Ex. 1007, p. 10, 19, 60.

63. In the late 1990s, following the introduction of personal computers during the 1980s, there was a move away from mainframe based application to tiered applications where PCs could be used instead of a dumb terminal. As such, it was typical to rely on a “tiered” software application, separate and apart from the RDBMS, to perform core accounting or profitability calculations. Mr. Lepman sometimes refers to these types of software applications as “procedural based software” in the patent specification. Ex. 1001, 10:44.

64. For SAP_IS_B, the software applications that comprise the dialog processing are programmed using SAP's 4th generation programming language, namely ABAP (Advanced Business Application Programming Language). Ex. 1007, p. 86.

65. Traditional accounting and financial platforms relied on tiered software applications, such as SAP's Cost Accounting and Bank Profitability Analysis modules, to analyze and perform calculations and other types of sophisticated business logic, rather than relying on the mathematical expressions of structured query language ("SQL"), the language used for RDBMSs.

66. One reason for that is that software application programming languages traditionally used for enterprise business applications, such as SAP's ABAP or COBOL (Common Business-Oriented Language), made it easy for application programmers to program business oriented software to perform accounting or profitability-related functions, such as costing and profit calculations.

67. As the names ABAP and COBOL indicate – ABAP (Advanced **Business Application** Programming Language) or COBOL (Common **Business-Oriented** Language) – these languages were specifically designed for that purpose. Ex. 1007, p. 86.

68. Relying on traditional software applications carried a disadvantage,

however. Performing calculations within the application server required spending additional time transmitting data between the database and application servers. This is referred to as “marshalling” data between servers and environments. Thus, Mr. Lepman moved the profitability calculation engine from the application server to the database server where the data files reside. This improvement avoided inefficiencies associated with marshalling data between servers and environments and reduced the processing time needed to complete the same number of profit calculations.

69. SAP_IS_B says nothing about using its “database server” (i.e. RDBMS) tier to perform calculations.

c. Limitation 1[e(i)]: “using the [RDBMS] ... calculate”.

70. In a declaration to the patent office in the file history, Mr. Lepman explained that the invention requires an “[RDBMS] having ... **calculation capabilities**” with reference to “calculation capabilities” that can be “run using [SQL]”:

Claim 2 recites that the relational database comprises structured query language. The invention requires the use of an electronic **relational database management system having** parallel operation and **calculation capabilities, which can be run using structured query language.** There is no such discussion in Price of a relational

database management system or structured query language.

Ex. 2002, p. 119 (emphases added). Claim 2 contemplates that “SQL” commands in particular, comprised within the RDBMS, are used to perform the limitation 1[e] “calculating” step. Ex. 1001, claim 2 (“The process of claim 1, wherein the relational database comprises a structured query language (SQL)”).

71. This improvement is recited in claim limitation 1[e(i)] of the ‘316 patent, which requires “*using* the relational database management system to ... calculate”. Ex. 1001, 29:5-6 (emphasis added). Based on the words, “using the relational database management system to ... calculate” and for reasons discussed in this section V(B)(3), a POSITA having the level of skill Petitioner asserts (reproduced above) would have agreed limitation 1[e(i)] requires that the RDBMS performs the profit calculations.

4. Taking Advantage of Massively Parallel Computing Hardware

72. The Lepman Patents also mention “parallel” processing or “massively parallel” computing hardware. Ex. 1001, 4:6-7.

73. Parallel processing refers to executing computations simultaneously. This can be accomplished using separate processors.

74. “Massively parallel,” as used several times in the Lepman Patent specifications (Ex. 1001, 2:11-12, 4:6, 6:4, 8:38), contemplates adding more

processors to support more simultaneous calculations.

75. For example, suppose a computing platform needs to perform 1,000 calculations. Assuming all other factors are equal, processing 1,000 calculations on two processors uses roughly the same amount of time a single processor uses to process only 500 calculations. In other words, adding a second processor cuts the time in half because two sets of 500 calculations are performed simultaneously (i.e. in parallel).

76. The Lepman Patents contemplate using SQL queries, executing inside the RDBMS, to perform profitability calculations using massively parallel hardware. Ex. 1001, 10:42-43 (“this approach allows for a relational database management system implementation”). The file history mentions this as well:

The invention requires the use of an electronic **relational database management system** having parallel operation and **calculation capabilities**, which can be run using structured query language. There is no such discussion in Price of a relational database management system or structured query language.

Ex. 2002, p. 204 (emphasis added).

77. Mr. Lepman recognized that, in order to extend the processing to multiple processors, those RDMBS calculations needed to be independent of each other.

78. Conventional profitability solutions used dependent (as opposed to

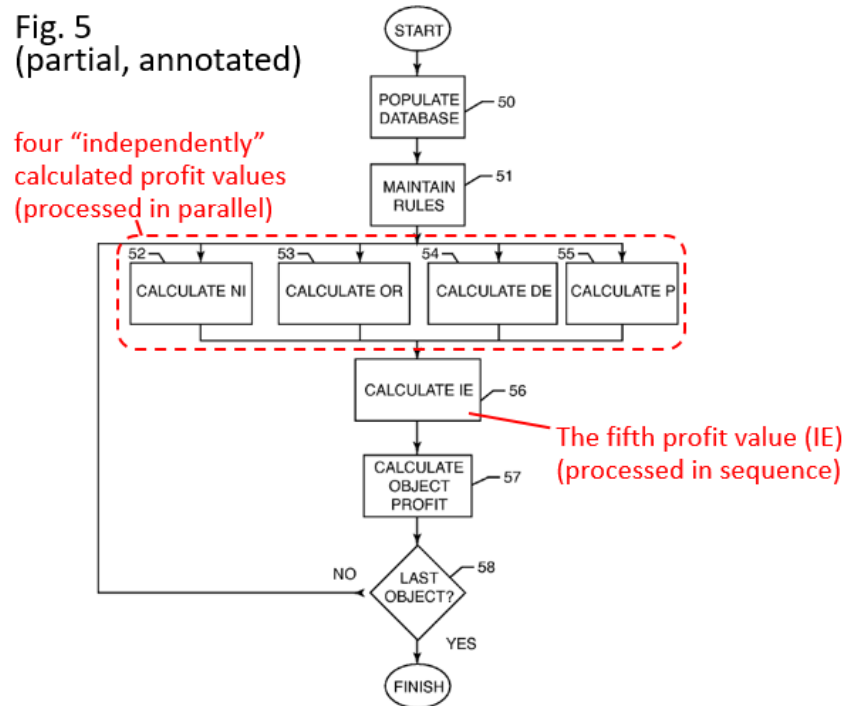
independent) calculations that would need to be executed in sequence. Ex. 2002, p. 203 (“Price teaches a sequential process, first classification and then aggregation . . .”).

79. As explained above, claim 2 contemplates that “SQL” commands in particular, comprised within the RDBMS, are used to perform the “calculating” step. Ex. 1001, claim 2 (“The process of claim 1, wherein the relational database comprises a structured query language (SQL)”).

80. A dependent calculation performed inside a SQL query, that relies on the result of a separate calculation, must wait for the separate calculation to complete first. If there is a dependency, it was not possible to process both calculation queries simultaneously (e.g. in parallel) in an RDBMS on separate processors. See e.g. Ex. 1001, 11:36-40 (“step 7 requires values derived in step 3, 4, 5, and 6 and therefore occurs sequentially”).

81. That is why Mr. Lepman arranged his method to have four profitability values - (1) net interest (“NI”); (2) other revenue (“OR”); (3) direct expense (“DE”); and (4) provisioning (“P”) - calculated “*independently*” of each other. This made it possible for the RDBMS to process multiple profit value calculations simultaneously and in parallel - rather than dependently, where calculations depending on prior calculations would execute in sequence. Ex. 1001, Fig. 5, 11:36-40.

82. Fig. 5 provides one example, illustrating that at least NI, OR, DE and P are calculated “independently” of each other (in parallel):



Ex. 1001, FIG. 5 (partial, annotated).

83. Elsewhere Mr. Lepman explains in the specification:

Steps 3, 4, 5, and 6 [referring to the NI, OR, DE and P calculations] can be processed **independently**; step 7 requires values derived in step 3, 4, 5, and 6 and therefore occurs **sequentially**.

Ex. 1001, 11:36-40 (emphases added);

In sum [sic] implementations this calculation is **dependent** on object level calculations and therefore **processed sequentially** (e.g. ratios 3 & 6 below). Otherwise, when the IE rules are **independent** of object

level calculations IE can be processed **in parallel** with the other profit calculations (e.g. ratios 1, 2, 4, 5, & 7).

Ex. 1001, 19:55-61.

84. Mr. Lepman explained this concept to the examiner during prosecution:

The invention allows for four of the five factors to be processed in parallel by a relational database management system (RDBMS) using formulas using selected object level data directly. Price teaches a sequential process, first classification and then aggregation, calculations are additions, without subsets.

Ex. 2002, p. 118.

85. This improvement is recited in claim limitation 1[e(ii)] of the '316 patent, which requires “**independently** calculat[ing] at least one marginal value of profit for each object being measured.” Ex. 1001, 29:5-6 (emphasis added).

5. Rules For Processing In the RDBMS

86. The Lepman Patents disclose incorporating the “rules for processing” into the RDBMS as well. The Lepman Patents refer to this as combining:

rules and data in a mathematical set theoretic framework (41). This approach allows for a relational database management system implementation (42).

Ex. 1001, 10:39-43.

87. In other words, rules defining, for example, how the profitability calculations are processed and selected, are established in the database itself. Placing the rules in the database together with the data facilitates using the RDBMS (via SQL commands, for example) to perform the core profitability calculations.

88. Similarly, the Lepman Patents disclose applying “rules based on sets” so that rules of similar calculation types, but for separate objects (e.g. separate bank accounts or airline seats), can be processed efficiently on a “massively parallel” hardware platform:

DPM is designed to allow the user the freedom to **associate a group of objects with a rule** and to use object-level information in combination with rule parameters to calculate profit values. The DPM invention uses profit measurement rules separate from, but applied to, object data and the use of relational database concepts, giving the user flexibility in both the assignment and depth of definition of measurement rules and measurement resolution. Use of this method is especially suited for massively parallel computing technology where linear scalable capital investment in processing technology is possible vis-a-vis object and event count and rule complexity.

Id.,8:30-41 (emphasis added).

89. The above passage refers to “scalable capital investment in processing technology,” which raises the concept of “scalability.” “Scalability” refers to the ability of a system to maintain performance as the number of users and volume of data grows.

90. In the context of this invention of the Lepman Patents, as the number of objects, rules and/or required calculations grew, more processors could be added to enable the calculation processing to conclude in the same amount of time.

91. The concept of placing the rules in the database together with the data to facilitate using the RDBMS to perform the core profitability calculations is recited in claim limitation 1[e(iii)] of the ‘316 patent, which requires “independently calculat[ing] ... using the established rules as applied to a selected set of prepared information.” Claim limitation (d) specifies that the “rules” are established “in the relational database” and “includ[e] the steps of providing information necessary to select objects and performing a profit calculus.” Ex. 1001, 29:1-4 (emphasis added).

VI. Pen and Paper

92. I understand Petitioner believes claim 1 is simply a mental process that can be performed by a human on pen and paper:

the Challenged Claims simply collect information (financial and non-financial information) and analyze that information to calculate profitability. The steps of the Challenged Claims represent a mental process, as

calculating object level profitability for an organization
can be calculated by a human on pen and paper.

CBM2019-00011 Petition, p. 21.

93. I disagree. There may be ways to use a pen and paper to perform an object level profitability calculation, but that is not claim 1.

94. There were also different ways to use a computer to perform object level profitability calculations using a traditional procedural software application in communication with an RDBMS, as I discuss above. That also is not claim 1.

95. Claim 1 specifically recites how object level profitability is calculated in a very particularized way – namely performing “independent[]” profitability calculations within the RDBMS *itself* using rules written to the database. See claim limitations 1[e(i)], 1[e(ii)] and 1[e(iii)]. These improvements made it possible to perform simultaneous RDBMS calculations in a massively parallel computing environment to achieve a faster, more efficient and more scalable profit calculator. These technological concepts recited in the claim do not have a “pen and paper” or “mental process” equivalent.

VII. Claim 1 is Valid Under Both *Alice* Step One and *Alice* Step Two.

96. Briefly, for reasons discussed below, it is my opinion claim 1 of the Lepman Patents satisfies *Alice* step one because it is not directed to an abstract idea, but instead to a specific implementation of a solution to a problem in the

software arts.

97. As I explain above claim element 1[e(i)] – reciting “using the [RDBMS] to calculate” – promoted speed and efficiency. By moving the profitability calculation engine from the application server to the database server where the data files reside, inefficiencies associated with marshalling data between servers and environments were avoided. This substantially reduced the processing time needed to complete the same number of calculations compared to traditional procedural software solutions.

98. Further, claim element 1[e(ii)] – reciting “independently calculate” – also promoted speed and efficiency. Using “independent” calculations, and avoiding dependencies, made it possible to take full advantage of improved RDBMS hardware platforms having multiple processors capable of parallel processing. This improvement further reduced the processing time needed to complete the same number of calculations.

99. These improvements also made the invention more scalable. As the number of objects, rules and/or required calculations grew, more processors could be added to enable the calculation processing to conclude in the same amount of time.

100. It is also my opinion that claim 1 recites more than the result of an object level profitability calculation. Limitations 1[e(i)] and 1[e(ii)] specify how

the speed, efficiency and scalability is achieved. Also, Limitation 1[e(iii)] recites a particularized way for the RDBMS calculation and “independently calculate” features of 1[e(i)] and 1[e(ii)] to be carried out.

101. I have also been asked to consider if claim 1 recites a non-conventional and non-generic arrangement of known, conventional pieces because, based on what I understand of *Alice* step two, such an arrangement qualifies as an “inventive concept” that is patent eligible.

102. It is my opinion that claim 1 recites an “inventive step” for claiming a non-conventional and non-generic arrangement. For example, using the RDBMS to perform calculations was non-conventional and non-generic in 1999 for reasons I explained above. The same applies to using the RDBMS to “independently” calculate. The same also applies to placing the rules themselves into the database, which differed from placing rules in traditional procedural software. Finally, combining each of these concepts into a single method was an unconventional change compared to prior attempts.

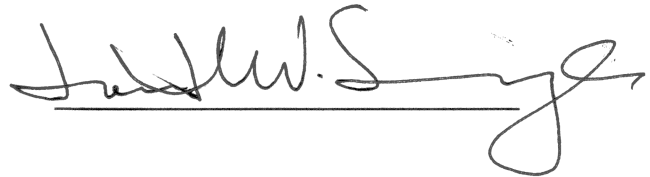
103. For the same reasons explained above, it is also my opinion that the Lepman Patents solve a technical problem using a technical solution.

Declaration of Jon Scarbrough

I declare under penalty of perjury of the laws of the United States that all statements made of my own knowledge are true and that all statements made on information and belief are believed to be true.

Dated: April 9, 2019

Respectfully submitted,

A handwritten signature in black ink, appearing to read "Jon Scarbrough", written over a horizontal line.

Jon Scarbrough

7105 E Sierra Trail • Sioux Falls, SD 57110 • (224) 623-1880 • jonscarbrough@gmail.com

IT Executive – CIO/CTO

- **Leadership** – active mentor creating an environment which encourages accountability, fosters innovation and drives results; earn credibility with staff by demonstrating hands-on technical competence; integrated off shore technical team members; provided technical defense during patent troll litigation, led technology discovery as part of two acquisitions.
- **Communication skills** – collaborate with executive management, customers, agencies, marketing, sales and IT staff to make technology a business partner; translate client/user objectives into tangible solutions; articulate technology vision, business and technical concepts clearly and effectively to both technical and non-technical audiences.
- **Project management** – track record of implementing new systems as project manager and hands-on leader of key technical tasks; led initiative to replace Y2K plagued mainframe with new Web based application architecture; created and managed development of web platform handling 1.2B+ annual transactions; utilize Agile development methodologies.
- **Innovation** – early adopter of AJAX technology to create new Web 2.0 applications; created social data acquisition patent; early adopter of iSCSI SAN technology to consolidate storage and reduce costs; active use of open source technology to create cost effective architectures and solutions.
- **Broad application knowledge** – web, mobile, HR/Payroll, ERP, CRM, general ledger, data warehouse, manufacturing, digital marketing.
- **MS** • Mathematics / Computer Science • Indiana State University
- **BA** • Mathematics (Cum Laude) • Augustana University

PROFESSIONAL EXPERIENCE

SDN Communications (private) • 2017-present

Director of IT

Provide executive leadership for application development, corporate IT, business system analysts and support. The company is a leader in regional broadband fiber optic networks and cybersecurity. Coordinate systems and reporting activities within the organization including Business Continuity and Disaster Recovery (BCDR) activities.

- Successfully contracted and held BCDR tabletop exercises for both internal company operations as well as statewide local telephone companies.
- Wrote the company's first BCDR plan document.
- Successfully led the development of sales automation applications resulting in real time quotes reducing the average quote time from 8 days to 5 minutes. This was achieved by leveraging geospatial capabilities in the fiber optic network software and other geospatial data.

DocuTAP (private) • 2016-2017

Senior Director of Software Engineering

Provided executive leadership over software engineering functions including software developers, QA and business analysts. The company provides electronic medical records management and patient management software for the urgent care industry.

- Successfully led the development of a minimum viable product (MVP) for consumer use in medical appointments and registration. The project was delivered ahead of schedule and under budget. The MVP was developed using Web Components, NodeJS, PostgreSQL all running on AWS. CI/CD processes were utilized to automate QA tasks and code delivery using Jenkins and AWS Code Deploy and Cloud Formation.
- Successfully introduced the use of data stream processing (Kafka) for integrating the legacy EMR system with the new technology stack and for automating data synchronization from the legacy EMR system to the BI platform.
- Successfully introduced JIRA for project management.

WHERE 2 GET IT, INC. (private – dba Brandify) • 2005 – 2016**Vice President of Technology (CTO)**

Provided executive leadership for all technology functions in a Local Digital Marketing company. The company provides location based services and solutions (web/mobile/social/search) to over 500 national and multi-national brands such as Safeway, Applebees, Chick-fil-A, Disney, Ben & Jerry, Office Depot, Moneygram, Cheesecake Factory, HP, Loreal and The Hartford representing over 4.5M world wide locations.

- **Web 2.0** - Designed and created a new Web 2.0 SaaS platform delivering omni-channel marketing technology solutions. Single code base reduced infrastructure requirements by 50% over legacy platform. Scalable web architecture handles 1.2B+ annual transactions with available capacity up to 5B transactions. Multi-terabyte PostgreSQL database.
- **Product Innovations** – Created or increased revenue streams with a 2-way SMS system, a VXML based IVR system, local search landing pages, native mobile SDK, Facebook locators, a rule based leads system and a thematic analytics solution to visualize large data sets on a map. Created patent pending social data acquisition technology. Introduced real time data streaming (Kafka/Storm) for big data solutions. Consolidated invoicing, prospect database and account management into a single internal CRM solution. Products used in delivering company's 25% YOY revenue growth.
- **Oracle Geospatial** - Implemented Oracle based geospatial platform (J2EE) reducing data processing requirements by 50% and increasing transaction throughput by 500%. Received 2007 Oracle Spatial Innovator of the year award.
- **Enterprise Security Policies** – Created company security policies including adoption of external vulnerability testing, intrusion detection and automated SDLC processes to meet Fortune 500 client requirements.
- **Data Center Operations** – Plan and implemented a clustered, high availability infrastructure in dual 24x7 co-location data centers (Chicago/Seattle). Implemented KVM virtualization resulting in 50% reduction of data center expenses while achieving 99.999% uptime. Reduced support expenses through adoption of Google Enterprise Apps (email/shared documents/calendar). Created Amazon, Rackspace and Google Cloud instances in support of failover capacity and queuing system for social data acquisition.
- **Geo-diverse team** – Manage local and off shore developers/technical staff encompassing 5 countries.
- **Cybersecurity Policies** – Implemented enterprise information asset policies to secure data, desktops, servers and infrastructure from cybersecurity threats.

CAREER EDUCATION CORPORATION (NASDAQ:CECO) • 2003 – 2005**Director of Operational Technology**

Hired to drive and articulate CEC's technology strategy and integrate technology into the business unit operations. Led large change efforts by working with the business units and the corporate IT staff. Defined and infused best practices, tools and techniques to assure the quality of enterprise applications. 78 operating units in the US, Canada, and Europe with annual revenue base exceeding \$2B.

- **Automation** – Led implementation of two large scale enterprise systems, web based attendance and online financial aid process. Daily attendance is the core attribute driving revenue. Financial aid accounted for 60% of total revenue.
- **Sales forecasting** – Created database driven sales forecasting tool producing 95% accurate forecasts. Report was emailed weekly to business unit leaders and company management.

GROUP ARNOLD, division of SPS (NYSE:ST), manufacturer of magnetic products • 2000 – 2002**Vice President, Information Technology (CIO)**

Directed all IT functions and resources for 12 plants with \$140 million revenue; IT budget \$1.1 million and staff of 7. Hired to implement Oracle-based HR/Payroll system, upgrade core ERP capabilities, and reduce IT costs. Transformed IT staff, infrastructure, and applications to become highly efficient and cost-effective corporate asset.

- **Oracle based HR/Payroll** – Proposed and migrated mainframe based system to new web based system (Humanic HR/Payroll) for 50% of the budgeted cost (saving \$300,000 capital expense). System improved reporting and processing, reduced payroll process time by 75%, and cut staff cost by \$300,000 annually. Controlled system issue and enhancement tracking by obtaining vendor's source code.

- **Oracle Data Warehouse** – Developed and implemented ad-hoc query and management reporting encompassing Inventory, WIP, Job Cost, Scrap, Productivity, Efficiency, Sales, AP/AR/GL. Replaced outmoded report printing system (75 reports), reducing staff cost by \$150,000 annually. Standardized, simplified, and enhanced reporting to vastly improve management control.
- **General Ledger System** – Migrated GL from mainframe to local ERP system (on HP/UX). Cut financial closing time by 90% and reduced staff cost by \$140,000. Simplified GL account structure by consolidating 10,000 GL accounts down to 1,300.
- **Customer Relationship Management System** – Designed, developed and implemented a comprehensive Web-based CRM. System supports quick response to sales opportunities and enhanced customer service.
- **ERP System** – Re-implemented system at one plant where earlier system failed. System supported job cost accounting for manufacturing producing accurate product pricing and improved margins.
- **Cost Reduction** – Greatly improved IT service and impact while cutting annual IT operating costs by 43%. Used techniques such as hardware leasing; voice/frame relay contract re-negotiation; outsourcing EDI; staff reduction; and automating WAN (frame relay) monitoring and nightly tape backup operations.

OAKTON COMMUNITY COLLEGE • 1990 – 2000

Associate Vice President of Information Technology (CIO) (1996 – 2000)

Promoted to direct and integrate all IT functions and resources for college with 11,000 students and 2,500 employees; IT budget \$4.5 million and staff of 42 plus 75 student employees. Directed custom system development project to replace mainframe-based legacy system. Planned, developed, and implemented new Web-based ERP platform and upgraded infrastructure to support academic and administrative needs.

- **College Information Management System** – Led initiative to replace Y2K plagued mainframe system with new Web-enabled application for registration, records, degree audit, etc. System used object-oriented development tools and Gemstone object database to accommodate exception based business processes, LDAP security, and integrated IVR. Converted 30 years of historical data from mainframe system. Implemented comprehensive security policies to abide by federal FERPA rules.
- **Oracle Financials and Oracle HR/Payroll** – Implemented new client/server applications to replace legacy mainframe systems. Oracle database ran on HP/UX.
- **Enterprise Directory Services** – Designed and implemented integrated email, sign-on, network access, and desktop software configuration capability to improve security and user productivity.
- **IT Infrastructure Improvement:**
 - Replaced mainframe computer with client-server platform.
 - Retrained and upgraded staff skills from mainframe to object oriented client/server development.
 - Standardized 1400+ PCs to Windows NT and Novell Zenworks to support 300+ desktop applications.
 - Installed leading-edge high-speed network combining voice, video and data. Created remote access solution for 400+ employees.
 - Consolidated 3 proprietary email systems into one Internet standards based system (13,500 mailboxes).
 - Designed IT infrastructure as part of 400,000 sq ft new building construction enabling network access in every classroom.
 - Created college's first Help Desk.
- **Web Based Applications** – Designed, developed and implemented Web-accessed applications for easy access and query, including online class schedule with open seat availability, and a prospective student database.
- **Regional Broadband Network** – Conceived and managed installation of regional broadband (ATM) network integrating data and video for member institutions.

Director of Instructional Technology (1990 – 1996)

Appendix B

US 7,596,521	US 7,882,137	US 8,612,316
<p>1. A process for determining object level profitability in a computer, comprising the steps of:</p> <p>[a] providing a relational database management system operable in association with a computer;</p> <p>[b] preparing information to be accessed electronically through the relational database management system;</p> <p>[c] establishing, in the relational database, rules for processing the prepared information;</p> <p>[d](i) using the relational database management system to (ii) independently calculate at least one marginal value of profit for each object being measured (iii) using the established rules as applied to a selected set of prepared information;</p> <p>[e] using the relational database management system to calculate a fully absorbed profit adjustment value for each object being measured; and</p> <p>[f] combining the at least one marginal value of profit and the fully absorbed profit adjustment value to create a measure for object level profitability.</p>	<p>1. A method for transforming a computerized profit database, comprising the <u>computer implemented</u> steps of:</p> <p>[a] providing a computerized profit database having profit information;</p> <p>[b] providing a relational database management system operable in association with the computerized profit database;</p> <p>[c] preparing the profit information to be accessed electronically through the relational database management system, including the step of calculating opportunity values for funds used or supplied by each object being measured;</p> <p>[d] establishing, in the relational database, rules for processing the prepared information, including the steps of providing information necessary to select objects and performing a profit calculus;</p> <p>[e](i) using the relational database management system to (ii) independently calculate at least one marginal value of profit for each object being measured (iii) using the established rules as applied to a selected set of prepared information;</p>	<p>1. A process for determining object level profitability in a relational database management system, comprising the computer implemented steps of:</p> <p>[a] providing a computerized profit database having profit information;</p> <p>[b] providing a relational database management system operable in association with the computerized profit database;</p> <p>[c] preparing the profit information to be accessed electronically through the relational database management system, including the step of calculating opportunity values for funds used or supplied by each object being measured;</p> <p>[d] establishing, in the relational database, rules for processing the prepared information, including the steps of providing information necessary to select objects and performing a profit calculus;</p> <p>[e](i) using the relational database management system to (ii) independently calculate at least one marginal value of profit for each object being measured (iii) using the established rules as applied to a selected set of prepared information; and</p>

Appendix B

	<p>[f] using the relational database management system to calculate a fully absorbed profit adjustment value for each object being measured; and</p> <p>[g] combining the at least one marginal value of profit and the fully absorbed profit adjustment value to create a measure for object level profitability.</p>	<p>[f] combining the at least one marginal value of profit to create a measure for object level profitability.</p>
<p>2. The process of claim 1, wherein the relational database comprises a structured query language (SQL).</p>	<p>2. The process of claim 1, wherein the relational database comprises a structured query language (SQL).</p>	<p>2. The process of claim 1, wherein the relational database comprises a structured query language (SQL).</p>