Link: 126

UNITED STATES DISTRICT COURT

CENTRAL DISTRICT OF CALIFORNIA

WESTERN DIVISION

The CALIFORNIA INSTITUTE OF TECHNOLOGY,

Plaintiff,

v.

HUGHES COMMUNICATIONS INC., HUGHES NETWORK SYSTEMS LLC, DISH NETWORK CORPORATION, DISH NETWORK L.L.C., and DISHNET SATELLITE BROADBAND L.L.C.,

Defendants.

Case No. 2:13-cv-07245-MRP-JEM

ORDER DENYING DEFENDANTS' MOTION FOR SUMMARY JUDGMENT ON 35 U.S.C. § 101 INELIGIBILITY

I. Introduction

Plaintiff California Institute of Technology ("Caltech") has asserted U.S. Patent Nos. 7,116,710 ("the '710 patent"), 7,421,032 ("the '032 patent"), 7,916,781 ("the '781 patent"), and 8,284,833 ("the '833 patent,") against Defendants Hughes Communications, Inc., Hughes Network Systems, LLC, DISH Network Corporation, DISH Network L.L.C., and dishNET Satellite Broadband L.L.C. (collectively, "Hughes"). The Court issued a claim construction order on August 6, 2014. *See Cal. Inst. of Tech. v. Hughes Commc'ns Inc.*, No. 2:13-cv-07245, 2014 WL 3866129 (C.D. Cal. Aug. 6, 2014).

Hughes moves for summary judgment on the grounds that the asserted claims are not patentable under 35 U.S.C. § 101.[1] The asserted claims focus on a particular form of error correction code, but the concerns underlying the patentability of these claims are the same concerns underlying the patentability of software generally. Having considered the parties' briefs and the papers filed therewith, the Court concludes that all asserted claims are patentable. Therefore, the Court denies Hughes' motion for summary judgment.

II. Background

The asserted claims are method and apparatus claims relating to error correction.[2] In modern electronic systems, data are stored in the form of bits having the value "1" or "0." During data transmission, a random or irregular fluctuation (known as noise) can occur in the signal and corrupt data. For example, a transmitter may send a bit with the value "1," but noise may corrupt this bit and cause the receiver to read the value as "0." To mitigate this problem,

[1] In this order, the Court uses the term "patentable" to refer to subject matter eligibility under § 101.

[2] All four patents share a common specification and claim priority to the same patent application U.S. Serial Application No. 09/861,102. The parties briefed this motion before Caltech's final election of asserted claims on Oct. 31, 2014. *See* Final Election of Asserted Claims, Dkt. No. 153. This order addresses the election of asserted claims filed Sept. 12, 2014, which includes all claims in the final election of asserted claims. *See* Election of Asserted Claims, Dkt. No. 125.

electronic systems use error correction. Error correction depends on redundancy, which refers to "extra" bits that may be duplicates of original information bits[3] and are transmitted along with the original bits. These extra bits are not necessary, in the sense that the original information exists without them, but they serve an important purpose. Using these extra bits, the receiver can ensure that the original information bits were not corrupted during transmission.

Caltech's patents are directed to a form of error correction code called an irregular repeat and accumulate ("IRA") code. An IRA code operates as follows: the code can introduce redundancy by repeating (i.e., duplicating) different original bits irregularly (i.e., a different number of times). These information bits may then be randomly permuted and combined to form intermediate bits, which are accumulated to form parity bits. Parity bits reflect the values of a selection of original information bits. These parity bits are transmitted along with the original information bits. The receiver ensures that the received original information bits were not corrupted during transmission. It can do this by modulo-2 ("mod-2") adding the original information bits and parity bits.[4] The receiver knows whether this sum is supposed to be odd or even. If the sum is supposed to be odd but is instead even, the receiver will know that an error occurred and can perhaps correct the error using other information it has received.

The benefit of an IRA code is that not all bits are repeated the same number of times. The repetition of certain bits provides redundancy. Although greater repetition of every bit would allow for better error correction, it would also force

[3] The '032 patent uses the term "message bits" rather than "information bits." This Court will generally use the term "information bits" when discussing error correction.

[4] For an explanation of mod-2 arithmetic, see *Modular Arithmetic – An Introduction*, Rutgers University, http://www.math.rutgers.edu/~erowland/modulararithmetic.html.

the transmitter to send more bits, decreasing the coding rate and increasing data transfer time.[5] IRA codes balance competing goals: data accuracy and efficiency. The asserted claims recite generally encoding and decoding bits in accordance with an IRA code.

III. Standard for Summary Judgment

The Court shall grant summary judgment if there is no genuine dispute as to any material fact, as supported by facts on the record that would be admissible in evidence, and if the moving party is entitled to judgment as a matter of law. Fed. R. Civ. P. 56; *see Celotex Corp. v. Catrett*, 477 U.S. 317, 322 (1986); *Anderson v. Liberty Lobby, Inc.*, 477 U.S. 242, 250 (1986). Ineligibility under § 101 is a question of law.[6] *In re Comiskey*, 554 F.3d 967, 975 (Fed. Cir. 2009). The Court may appropriately decide this issue at the summary judgment stage.

---

[5] Coding rate is calculated through the following equation: Coding Rate = (Original information bits) / (Original information bits + Extra bits). The closer the coding rate is to 1, the more efficient it is.

[6] The Federal Circuit has noted that § 101 analysis is "rife with underlying factual issues." *Ultramercial, Inc. v. Hulu, LLC*, 722 F.3d 1335, 1339 (Fed. Cir. 2013), *vacated sub nom. WildTangent, Inc. v. Ultramercial, LLC*, 134 S. Ct. 2870 (2014). If the § 101 inquiry involves asking "whether genuine human contribution is required, and that requires more than a trivial appendix to the underlying abstract idea, . . . [which was] not at the time of filing routine, well-understood, or conventional, factual inquiries likely abound." *Id.* at 1339 (internal quotation marks omitted). Therefore, the Federal Circuit has held that a challenger must prove ineligibility under § 101 by "clear and convincing evidence," even though § 101 eligibility is a question of law. *Id.* at 1338–39.

This Court believes that the clear and convincing evidence standard does not apply to § 101 analysis, because § 101 eligibility is a question of law. Courts frequently make findings when deciding purely legal questions. *See, e.g.*, *District of Columbia v. Heller*, 554 U.S. 570, 581–92 (2008) (determining meaning of "keep and bear Arms" during the founding era by analyzing dictionary definitions and then-prevailing usage). Eligibility questions mostly involve general historical observations, the sort of findings routinely made by courts deciding legal questions. *Compare ABC, Inc. v. Aereo, Inc.*, 134 S. Ct. 2498, 2505–06 (2014) (relying on legislative history and context of 1976 Copyright Act to justify finding copyright liability for online television streaming service), *with Alice Corp. Pty. Ltd. v. CLS Bank Int'l*, 134 S. Ct. 2347, 2356 (2014) (citing to historical evidence showing intermediated settlement is a longstanding practice). Moreover, eligibility frequently depends on a court's interpretation of § 101. *Cf. Parker v. Flook*, 437 U.S. 584, 588 (1978) (noting eligibility for a claimed algorithm "turn[ed] entirely on the proper construction of § 101"). As stated by Justice Breyer in his *i4i* concurrence,

IV. Ineligibility Under 35 U.S.C. § 101

Section 101 of the Patent Act defines patentable subject matter: "Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title." 35 U.S.C. § 101. Section 101 defines four broad categories of patentable inventions: processes, machines, manufactures, and compositions of matter. "Congress took this permissive approach to patent eligibility to ensure that ingenuity should receive a liberal encouragement." *Bilski v. Kappos*, 561 U.S. 593, 601 (2010) (internal quotation marks omitted). But § 101 does not encompass all products of human effort and discovery. Laws of nature, physical phenomena, and abstract ideas are not patentable. *Diamond v. Chakrabarty*, 447 U.S. 303, 309 (1980). These exceptions are well established. *See, e.g.*, *Chakrabarty*, 447 U.S. at 309; *Diamond v. Diehr*, 450 U.S. 175, 185 (1981); *Parker v. Flook*, 437 U.S. 584, 599 (1978) (Stewart, J., dissenting); *Gottschalk v. Benson*, 409 U.S. 63, 67 (1972); *Funk Bros. Seed Co. v. Kalo Inoculant Co.*, 333 U.S. 127, 130 (1948); *Le Roy v. Tatham*, 55 U.S. 156, 175 (1853).

On occasion, the Federal Circuit has described § 101 as a "coarse eligibility filter," barring only "manifestly abstract" inventions and leaving §§ 102, 103, and 112 as the finer sieves. *See Ultramercial, Inc. v. Hulu, LLC*, 722 F.3d 1335, 1341, 1354 (Fed. Cir. 2013), *vacated sub nom.*, *WildTangent, Inc. v. Ultramercial, LLC*,

---

the clear and convincing evidence standard "applies to questions of fact and not to questions of law." *Microsoft Corp. v. i4i Ltd. P'ship*, 131 S. Ct. 2238, 2253 (2011) (Breyer, J., concurring). Tellingly, the Supreme Court has never mentioned the clear and convincing evidence standard in its post-*i4i* § 101 decisions.

Regardless, the Court must follow binding precedent. The Court notes that the parties have identified no material disputed facts. The parties primarily dispute legal conclusions drawn from undisputed facts, such as the conventionality of claim elements or the relevance of certain claim elements to the § 101 issue. Inasmuch as the parties dispute the characterization of certain elements of the technology, the Court is unconvinced that these are factual questions, and in any case, the Court's analysis does not turn on the characterization of these elements.

134 S. Ct. 2870 (2014). But in its last few terms, the Supreme Court has indicated that patentability is a higher bar. *See Alice Corp. Pty. Ltd. v. CLS Bank Int'l*, 134 S. Ct. 2347, 2354–55 (2014); *Ass'n for Molecular Pathology v. Myriad Genetics, Inc.*, 133 S. Ct. 2107, 2116 (2013); *Mayo Collaborative Servs. v. Prometheus Labs., Inc.*, 132 S. Ct. 1289, 1293–94 (2012); *Bilski*, 561 U.S. 609–13. As noted by Judge Mayer of the Federal Circuit, a "robust application" of § 101 ensures "that patent protection promotes, rather than impedes, scientific progress and technological innovation." *I/P Engine, Inc. v. AOL Inc.*, 576 F. App'x 982, 996 (Fed. Cir. 2014) (nonprecedential) (Mayer, J., concurring).

Courts must evaluate patent eligibility using a two-part test. First, a court must ask if the claim is "directed to one of those patent-ineligible concepts"—a law of nature, physical phenomenon, or abstract idea. *Alice*, 134 S. Ct. at 2355. Second, if the claim is directed to one of these concepts, the court must ask "[w]hat else is there in the claims before us?" *Mayo*, 132 S. Ct. at 1297. This second step determines whether there is an "inventive concept" that "ensure[s] that the patent in practice amounts to significantly more than a patent upon the [ineligible concept] itself." *Alice*, 134 S. Ct. at 2355.

These steps are broadly stated and, without more, would be difficult to apply. Fortunately, although the two-part test was created in *Mayo*, pre-*Mayo* precedents offer some guidance in applying these two steps. Briefly, these precedents suggest the following methodology: **(1)** At step one, the court ascertains the purpose of the claimed invention. The court then analyzes whether this purpose is abstract. If the purpose is abstract, the court moves to the second step. **(2)(A)** At step two, the court tries to identify an inventive concept by considering the claim elements both individually and as an ordered combination. **(2)(B)** When viewing claim elements individually, the court must remember that recitation of conventional, routine, or well-understood activity will not save an abstract claim. *See, e.g.*, *Alice*, 134 S. Ct. at 2358 (reciting generic computer does not save an abstract idea because

computers are ubiquitous). But a claim element is not conventional just because it appears in prior art. **(2)(C)** When viewing claim elements as an ordered combination, the court should not ignore the presence of any element, even if the element, viewed separately, is abstract. If the ordered combination of elements constitutes conventional activity, the claim is not patentable, but courts should remember that a series of conventional elements may together form an unconventional, patentable combination.

A. Supreme Court Decisions on § 101

The Supreme Court decisions on § 101 often confuse more than they clarify. The cases appear to contradict each other on important issues, such as the role of prior art in § 101 analysis. Although these cases provide some clues to applying § 101, they leave open the question of when, if ever, computer software is patentable. A basic principle about computer technology is that algorithms comprise computer software and computer codes. *See* J. Glenn Brookshear, *Computer Science: An Overview* 2 (6th ed. 2000) ("A machine-compatible representation of an algorithm is called a **program**. Programs, and the algorithms they represent, are collectively referred to as **software**."); *see also id*. at 168–77 (discussing further algorithms and their form). Supreme Court cases show skepticism toward patenting algorithms, though not an outright rejection of patentability.

Given the state of § 101 case law, this Court finds it useful to trace the evolution of the Supreme Court's views through the six most relevant cases: *Gottschalk v. Benson*, *Parker v. Flook*, *Diamond v. Diehr*, *Bilski v. Kappos*, *Mayo Collaborative Services v. Prometheus Laboratories, Inc.*, and *Alice Corp. Pty. Ltd. v. CLS Bank International*.

i. *Gottschalk v. Benson*: Mathematical Formula Is Abstract

In *Gottschalk v. Benson*, 409 U.S. 63 (1972), the Supreme Court invalidated method claims for converting binary-coded decimal ("BCD") numerals into pure

binary numerals.[7] *Benson*, 409 U.S. at 71–72. The Supreme Court reaffirmed the principle that "while a scientific truth, or the mathematical expression of it, is not patentable invention, a novel and useful structure created with the aid of knowledge of scientific truth may be." *Id.* at 67 (internal quotation mark omitted). The Court declined to hold that "a process patent must either be tied to a particular machine or apparatus or must operate to change articles or materials to a 'different state or thing.'" *Id.* at 71. Nonetheless, the Supreme Court found that the claims addressed the abstract idea of converting BCD numerals into binary numerals. The claims were so broad that they would "cover both known and unknown uses," effectively preempting the mathematical formula and constituting "a patent on the algorithm itself." *Id.* at 68, 72.

ii. *Parker v. Flook*: The Rise of Point-of-Novelty Analysis

The Supreme Court again found process claims using mathematical formulas unpatentable in *Parker v. Flook*, 437 U.S. 584 (1978). The case involved a method for updating alarm limits in catalytic chemical conversions.[8] In his majority opinion, Justice Stevens adopted a point-of-novelty approach, evaluating only the

---

[7] Conversion of BCD to binary is relatively simple. The Supreme Court explained that

> [t]he BCD system using decimal numerals replaces the character for each component decimal digit in the decimal numeral with the corresponding four-digit binary numeral . . . . Thus decimal 53 is represented as 0101 0011 in BCD, because decimal 5 is equal to binary 0101 and decimal 3 is equivalent to binary 0011. In pure binary notation, however, decimal 53 equals binary 110101.

Benson, 409 U.S. at 66–67.

[8] The Supreme Court explained the technology as follows:

> During catalytic conversion processes, operating conditions such as temperature, pressure, and flow rates are constantly monitored. When any of these "process variables" exceeds a predetermined "alarm limit," an alarm may signal the presence of an abnormal condition indicating either inefficiency or perhaps danger. Fixed alarm limits may be appropriate for a steady operation, but during transient operating situations, such as start-up, it may be necessary to "update" the alarm limits periodically.

Flook, 437 U.S. at 585.

claim's novel elements for patent eligibility and ignoring elements found in prior art. The claim's only novel element was a mathematical formula, which an operator could use to update an alarm limit by inputting values for a number of variables. *Id.* at 586–87. The patent did not explain how to select any values for variables. *Id.* at 586. All other elements of the claim were in the prior art. *Id.* at 586–87. Justice Stevens noted that "[w]hether the algorithm was in fact known or unknown at the time of the claimed invention, as one of the 'basic tools of scientific and technological work,' it is treated as though it were a familiar part of the prior art." *Id.* at 591–92 (internal citation omitted). Because the Court treated the mathematical formula as "prior art" along with the claim's other elements, the "application . . . contain[ed] no claim of patentable invention." *Id.* at 594.

iii. *Diamond v. Diehr*: The Fall of Point-of-Novelty Analysis

Benson and *Flook* created a strict test for eligibility that meshed § 102 novelty concerns with § 101 eligibility factors. But the Supreme Court changed direction in *Diamond v. Diehr*, 450 U.S. 175 (1981), which found patentable a claim for curing synthetic rubber. The Supreme Court retreated from its point-of-novelty analysis, clarifying that "[i]t is inappropriate to ***dissect the claims into old and new elements*** and then to ignore the presence of the old elements in the analysis." *Diehr*, 450 U.S. at 188 (emphasis added). Further, courts should consider, rather than ignore, the presence of a mathematical algorithm when determining patentability. *Id.* at 189 n.12. In a process claim, "a new combination of steps in a process may be patentable," even if the constituent elements are well known. *Id.* at 188. Thus, "the 'novelty' of any element or steps in a process . . . is of no relevance" in § 101 analysis. *Id.* at 188–89. The Supreme Court determined that the claim did not preempt the use of "a well-known mathematical equation" but foreclosed use of that equation only in conjunction with other steps, including "installing rubber in a press, closing the mold, constantly determining the temperature of the mold, constantly recalculating the appropriate cure time through

the use of the formula and a digital computer, and automatically opening the press at the proper time." *Id.* at 187. These steps in the claim "tranform[ed] or reduc[ed] an article to a different state or thing," making the claim the kind of invention deserving protection. *Id.* at 192. The Supreme Court read *Flook* as holding that an abstract idea does not become patentable merely because it is limited "to a particular technological environment" or because the claim recites "insignificant postsolution activity." *Id.* at 191, 192 n.14. Justice Stevens dissented, faulting the majority for mischaracterizing the invention as "a method of constantly measuring the actual temperature inside a rubber molding press." *Id.*at 206 (Stevens, J., dissenting). Justice Stevens characterized the invention as the abstract idea of an "improved method of calculating the time that the mold should remain closed during the curing process." *Id.* at 207.

iv. *Bilski v. Kappos*: Longstanding Business Method Is Abstract

Following *Diehr*, the Supreme Court did not revisit § 101 for more than a quarter of a century. This period saw the Federal Circuit adopt an expansive view of eligibility in *State Street Bank & Trust Co. v. Signature Financial Group*, 149 F.3d 1368 (Fed. Cir. 1998), where the court said § 101 allowed claims on mathematical algorithms that produced a "useful, concrete, and tangible result." *Id.* at 1373. The Federal Circuit then significantly limited process claim eligibility in *In re Bilski*, 545 F.3d 943, 954 (Fed. Cir. 2008), *rev'd*, 561 U.S. 593 (2010). When the Supreme Court granted certiorari in *Bilski v. Kappos*, 561 U.S. 593 (2010), it did so for a seemingly modest reason: to clarify that a process could be patentable even if it was not tied to a machine or did not transform an article. *Id.* at 601–04. But significantly, the Supreme Court invalidated claims that captured the concept of hedging. *Id.*at 611–12. It noted that hedging was "a fundamental economic practice long prevalent in our system of commerce." *Id.* at 611. The representative claims either described this practice or reduced it to a mathematical formula, and other claims merely limited the concept to a technological area or

added conventional postsolution components. *Id.* at 611–12. Regardless of the form of these claims, they did nothing more than recite an ineligible concept. *Id.*

v. *Mayo Collaborative Services v. Prometheus Laboratories, Inc.*: Conventional Activity Does Not Make Abstract Ideas Patentable

The Supreme Court returned again to § 101 in *Mayo Collaborative Services v. Prometheus Laboratories, Inc.*, 132 S. Ct. 1289 (2012). This instructive decision provided a perspective on *Benson* and the seemingly conflicting *Flook* and *Diehr* decisions. *Mayo* invalidated a claim setting forth "relationships between concentrations of certain metabolites in the blood and the likelihood that a dosage of a thiopurine drug will prove ineffective or cause harm." *Id.* at 1297. Writing for a unanimous court, Justice Breyer noted that § 101 attempts to reconcile two competing concerns. Although allowing patents on abstract ideas and natural laws would "impede innovation more than it would tend to promote it," the Supreme Court recognized that "all inventions at some level embody, use, reflect, rest upon, or apply laws of nature, natural phenomena, or abstract ideas." *Id.* at 1293. As such, "too broad an interpretation of this exclusionary principle could eviscerate patent law." *Id.* Thus, the Supreme Court emphasized that "'an application of a law of nature or mathematical formula to a known structure or process may well be deserving of patent protection.'" *Id.* at 1293–94 (quoting *Diehr*, 450 U.S. at 187). But "stat[ing] the law of nature while adding the words 'apply it'" does not transform unpatentable subject matter into patentable subject matter. *Id.* at 1294.

The Supreme Court engaged in a first step of analysis, in which it determined that the claims set forth laws of nature, a § 101 ineligible concept. *Id.* at 1296–97. The Supreme Court then engaged in a second step of analysis, in which it analyzed whether "the claims do significantly more than simply describe these natural relations." *Id.* at 1297. The Supreme Court determined that the claim elements "inform a relevant audience about certain laws of nature; any additional steps consist of well understood, routine, conventional activity already engaged in by the

scientific community." *Id.* at 1298. These other elements were insignificant and could not save the claim from ineligibility. Either they merely limited the law of nature to a technological area or constituted "[p]urely conventional or obvious [pre]solution activity." *Id.* (internal quotation marks omitted). The Supreme Court said its holding was consistent with *Flook* and *Diehr*, treating both as binding. It distinguished *Diehr* from *Flook* because in *Diehr* the Supreme Court never stated that the claimed "steps, or at least the combination of those steps, were in context obvious, already in use, or purely conventional." *Id.* at 1299.

vi. *Alice Corp. Pty. Ltd. v. CLS Bank International*: A Missed Opportunity to Clarify Computer Software Patentability

Alice Corp. Pty. Ltd. v. CLS Bank International, 134 S. Ct. 2347 (2014), presented the Supreme Court with the opportunity to clarify when computer software is patentable, but the Supreme Court left the question mostly unanswered. Admittedly, the Supreme Court clarified some aspects of the doctrine. First, the Supreme Court determined that the two-step test in *Mayo* governed all eligibility questions. *Id.* at 2355. Second, it clarified that a claim cannot satisfy step two of *Mayo* by reciting a generic computer. *See id.* at 2358 ("[T]he mere recitation of a generic computer cannot transform a patent-ineligible abstract idea into a patent-eligible invention. . . . Stating an abstract idea while adding the words 'apply it with a computer' . . . [creates a] deficient result.") Third, the Supreme Court clarified that reframing a method claim as an apparatus claim does not avoid eligibility issues, when the apparatus claimed is a generic computer. *Id.* at 2360. Fourth, and perhaps most significantly, it left open the possibility that claims which "improve the functioning of the computer itself" or "any other technology" are patentable. *Id.* at 2359.

Yet *Alice* did not answer the bigger questions, only incrementally clarifying § 101.[9] Perhaps the patent in *Alice* was the improper vehicle for clarifying the law: the patent claimed the age-old business method of mitigating settlement risk by using a third party intermediary, and the role of the computer in the claims was limited to generic functions like creating electronic records and tracking multiple transactions. *Id.* at 2359. *Alice* held only that abstract business methods do not become automatically patentable when implemented on a computer. *Id*. *Alice* failed to answer this: when, if ever, do computer patents survive § 101?

B. Is Computer Software Patentable?

Although the Supreme Court has never declared that software is patentable subject matter, software must be eligible under § 101. A bright-line rule against software patentability conflicts with the principle that "courts should not read into the patent laws limitations and conditions which the legislature has not expressed." *Bilski*, 561 U.S. at 602 (internal quotations marks omitted). One could argue that eliminating software patents is desirable public policy, but Congress has spoken on the patentability of software. The America Invents Act ("AIA") contemplates the existence of software patents explicitly in Section 14, which states in relevant part:

> (a) IN GENERAL.--For purposes of evaluating an invention under section 102 or 103 of title 35, United States Code, any strategy for reducing, avoiding,

---

[9] Regardless of *Alice*'s actual holding, *Alice* has brought about a wave of decisions finding software patents ineligible. *See, e.g., Eclipse IP LLC v. McKinley Equip. Corp.*, No. 8:14-cv-742-GW(AJWx), 2014 WL 4407592 (C.D. Cal. Sept. 4, 2014) (finding unpatentable claims reciting methods for communications); *Tuxis Techs v. Amazon.com, Inc.*, No. 13-1771, 2014 WL 4382446 (D. Del. Sept. 3, 2014) (finding unpatentable claims on upselling); *Loyalty Conversion Sys. Corp. v. American Airlines, Inc.*, No. 2:13-cv-655, 2014 WL 4364848 (E.D. Tex. Sept. 2, 2014) (Bryson, J.) (finding unpatentable claims on converting one vendor's loyalty credits into another's). This Court has found few district court decisions finding software claims patentable post-*Alice*. *See, e.g., Card Verifications Solutions, LLC v. Citigroup Inc.*, No. 13 C 6339, 2014 WL 4922524, at *5 (N.D. Ill. Sept. 29, 2014) (refusing to find unpatentable claims at motion to dismiss stage but allowing defendant to renew its challenge at a later time); *Helios Software, LLC v. SpectorSoft Corp.*, No. 12-081, 2014 WL 4796111, at *16–18 (D. Del. Sept. 18, 2014) (finding eligible claims directed to "remotely monitoring data associated with an Internet session and controlling network access").

> or deferring tax liability, whether known or unknown at the time of the invention or application for patent, shall be deemed insufficient to differentiate a claimed invention from the prior art. . . .
>
> (c) EXCLUSIONS.--This section does not apply to that part of an invention that--
>
> (1) is a method, apparatus, technology, ***computer program product***, or system, that is used solely for preparing a tax or information return or other tax filing, including one that records, transmits, transfers, or organizes data related to such filing; or
>
> (2) is a method, apparatus, technology, ***computer program product***, or system used solely for financial management, to the extent that it is severable from any tax strategy or does not limit the use of any tax strategy by any taxpayer or tax advisor.

Leahy-Smith America Invents Act, 112 P.L. 29, § 14, 125 Stat. 284, 327–28 (2011) (emphasis added); *see* Mark J. Patterson & M. Andrew Pitchford, *First to File*, 47 Tenn. B.J. 14, 16 (November 2011) ("[T]ax strategies are no longer patentable, but . . . computer implemented methods and computer program products (e.g., software) have been implicitly affirmed as patentable subject matter."). By excluding computer programs from subsection (a), Congress contemplated that some computer programs were eligible for patent protection. Courts should not read § 101 to exclude software patents when Congress has contemplated their existence. Similar reasoning was used in *Bilski* with regard to business method patents. In pre-AIA § 273(b)(1), an alleged infringer of a method in a patent could assert a defense of prior use, where for this defense, method was defined as "a method of doing or conducting business." *See* 35 U.S.C. § 273(a)(3) (2006). Thus, the Supreme Court determined a categorical exclusion against business method patents would "violate the canon against interpreting any

statutory provision in a manner that would render another provision superfluous." *Bilski*, 561 U.S. at 595.

Perhaps Congress did not intend to affirm that software was patentable. Maybe Congress was merely acknowledging that software patents exist without approving of their existence. But this speculative reasoning was rejected by *Bilski* with regard to business method patents. *Compare* 561 U.S at 644–45 (Stevens, J. dissenting) (arguing that Congress enacted § 273 to limit the damage caused by *State Street Bank* but did not intend to adopt its holding), *with id.* at 608 (rejecting Justice Stevens' reasoning because an "established rule of statutory interpretation cannot be overcome by judicial speculation as to the subjective intent of various legislators in enacting the subsequent provision").

Moreover, the Supreme Court has implicitly endorsed the patentability of software. *Alice* seems to acknowledge that software may be patentable if it improves the functioning of a computer. *See Alice*, 134 S. Ct. at 2359 ("The method claims do not, for example, purport to improve the functioning of the computer itself. Nor do they effect an improvement in any other technology or technical field." (internal citations omitted)). The Supreme Court could have resolved *Alice* and provided clarity to patent law by declaring all software patents ineligible.[10] However, the Supreme Court did not do this. This is some evidence of the continuing eligibility of software.

C. Software Patentability After *Alice*

Although computer software is patentable generally, neither *Alice* nor any other Supreme Court precedent defines when software is patentable. This has proven detrimental to the patent system. The purpose of patents is "promote the Progress of . . . useful Arts, by securing for limited Times to . . . Inventors the exclusive right to their . . . Discoveries." U.S. Const. art. I, § 8, cl. 8; *see also Siemens Med.*

[10] Hughes implies that in order for software claims to survive § 101, claims must recite specifically designed, non-generic hardware. *See, e.g.*, Defs.' Mem. in Supp. of Invalidity at 22, Dkt. No. 126. This Court does not read *Alice* to require this.

Solutions United States, Inc. v. St.-Gobain Ceramics & Plastics, Inc., 647 F.3d 1373, 1375 (Fed. Cir. 2011) ("At its heart, the patent system incentivizes improvements to patented technology."). In order to best incentivize innovation, however, patent law must be predictable, consistent, and uniform. *See Lighting Ballast Control LLC v. Philips Elecs. N. Am. Corp.*, 744 F.3d 1272, 1282 (Fed. Cir. 2014). *Alice* does not achieve this goal, leaving the boundaries of § 101 undefined. *See McRO, Inc. v. Sega of America,, Inc.*, No. 2:12-cv-10327, 2014 WL 4749601, at *5 (C.D. Cal. Sept. 22, 2014) (Wu, J.) ("[T]he two-step test may be more like a one step test evocative of Justice Stewart's most famous phrase ['I know it when I see it'].").

If an issue is significant or complicated, the Supreme Court may not announce definitive rules on its first pass at an issue. Instead, the Supreme Court may allow the issue to percolate, which permits lower courts the opportunity to offer their views. By allowing "a period of exploratory consideration and experimentation by lower courts," the Supreme Court can have "the benefit of the experience of those lower courts" when it revisits the issue. *California v. Carney*, 471 U.S. 386, 400 n.11 (1985) (quoting Samuel Estreicher & John E. Sexton, *A Managerial Theory of the Supreme Court's Responsibilities: An Empirical Study*, 59 N.Y.U. L. Rev. 681, 716 (1984)). When the Supreme Court leaves questions open, lower courts have a duty to offer their views and develop the law. Lower courts have endeavored to fulfill this responsibility with regard to § 101, but the resulting decisions demonstrate the continuing uncertainty surrounding software patentability.

i. Federal Circuit's Post-*Alice* Decisions

The task of clarifying and developing patent law is primarily assigned to the Federal Circuit. Indeed, these concerns motivated the formation of the Federal Circuit. *See Lighting Ballast*, 744 F.3d at 1282 ("The purposes of consistency and stability that underlie stare decisis led to the formation of the Federal Circuit, now thirty years past, to provide consistency and stability to the patent law."). Thus far,

the Federal Circuit has had three opportunities to clarify the application of § 101 to computer software. In these cases, the Federal Circuit has taken two routes: either it has said as little as possible or announced rules that are seemingly at odds with judicial precedent and congressional intent.

In *Digitech Image Technologies, LLC v. Electronics for Imaging, Inc.*, 758 F.3d 1344 (Fed. Cir. 2014), the Federal Circuit first invalidated claims for a device profile composed of data. *Id.* at 1349. Because the claims were directed to "information in its non-tangible form," the claims were not a machine or manufacture within the meaning of § 101. *Id.*[11] The Federal Circuit then invalidated method claims that involved generating data sets for a device profile and combining the data sets, stating that the claims "recite[ ] an ineligible abstract process of gathering and combining data that does not require input from a physical device." *Id.* at 1351. Writing for the panel, Judge Reyna stated a general principle that "[w]ithout additional limitations, a process that employs mathematical algorithms to manipulate existing information to generate additional information is not patent eligible." *Id.* The court passed on the question of "whether tying the method to an image processor would lead us to conclude that the claims are directed to patent eligible subject matter in accordance with the Supreme Court's *Mayo* test." *Id.*

Digitech seems to set forth a bright-line rule: if a claim consists of mathematical algorithms that transform data, the claim is not patentable. But that cannot be what *Digitech* means. There are two problems with this interpretation of *Digitech*. The first problem is that this interpretation results in the incorrect conclusion that software is not patentable. The essence of software is manipulating existing data and generating additional data through algorithms. *See Oplus Techs. Ltd. v. Sears*

[11] Software necessarily exists in a non-tangible form, and although the court observed that the claims do "not describe the device profile as a tag or any other embodiment of hardware or software," it is unclear why patentability depends on an explicit recitation of software. *Digitech*, 758 F.3d. at 1349.

Holding Corp., No. 2:12-cv-5707, 2013 WL 1003632, at *12 (C.D. Cal. Mar. 4, 2013) ("All software ***only*** 'receives data,' 'applies algorithms,' and 'ends with decisions.' That is the *only* thing software does. Software does nothing more."); *see also* Brookshear, *supra* at 168–70. This simplistic take on *Digitech* would eviscerate all software patents, a result that contradicts Congress's actions and the Supreme Court's guidance that software may be patentable if it improves the functioning of a computer.

The second problem with *Digitech* relates to the first one. By passing on the question as to whether the invention would be patentable if it were connected to a machine, the Federal Circuit perhaps inadvertently suggested that method claims need to meet the machine-or-transformation test, which is merely an "important and useful clue." *Bilski*, 561 U.S. at 603. A better reading of the Federal Circuit's statement is that some abstract ideas may become patentable if they are tied to uniquely designed machines with specific purposes. But courts must remember that generic recitation of hardware will not save a claim. *See Alice*, 134 S. Ct. at 2360 ("Put another way, the system claims are no different from the method claims in substance. The method claims recite the abstract idea implemented on a generic computer.").

Federal Circuit panels have spoken two other times on § 101 post-*Alice*.[12] In *Planet Bingo, LLC v. VKGS LLC*, 576 Fed. App'x 1005 (Fed. Cir. 2014) (nonprecedential), the court invalidated claims directed to a computerized bingo game, which recited "storing a player's preferred sets of bingo numbers; retrieving

[12] Judge Mayer, in his *I/P Engine* concurrence, expressed his view that *Alice* recited a technological requirement for § 101. He wrote that "*Alice* thus made clear that abstract ideas untethered to any significant advance in science and technology are ineligible for patent protection." *I/P Engine*, 576 F. App'x 982, 992 (Mayer, J., concurring). This view overstates *Alice*'s holding. *Alice* held that an abstract business method remains abstract even if it is implemented on a generic computer performing generic functions. But *Bilski* refused to hold that business method patents are ineligible, and a technological requirement seems to contradict this holding. At the very least, reading a technological requirement into § 101 is inconsistent with the section's plain language. *See Bilski*, 561 U.S. at 603.

one such set upon demand, and playing that set; while simultaneously tracking the player's sets, tracking player payments, and verifying winning numbers." *Id.* at 1006. The court determined that managing a bingo game "consists solely of mental steps which can be carried out by a human using pen and paper" and was abstract. *Id.* at 1007. Because the computer elements recited were purely generic and conventional, there were no meaningful limitations at step two of the *Mayo* test. *Id.* at 1008–09. In *buySAFE, Inc. v. Google, Inc.*, 765 F.3d 1350 (Fed. Cir. 2014), the patent claimed "methods and machine-readable media encoded to perform steps for guaranteeing a party's performance of its online transaction." *Id.* at 1351. The court determined that the claims recited an abstract idea because they were "squarely about creating a contractual relationship—a 'transaction performance guaranty'—that is beyond question of ancient lineage." *Id.* at 1355. At the second step of *Mayo*, the Court determined there was no inventive concept added. The computer functions were generic, because the claims recited the mere use of a computer to receive and send information over a network. *Id.*

All three decisions reach the correct result, based on Supreme Court precedents. In *Digitech*, the claims were so broad as to capture a large amount of inventive activity and in effect impede innovation. In *buySafe* and *Planet Bingo*, the claims were broad and directed to age-old concepts. But these decisions provide either false guidance to district courts, or no guidance at all. *Digitech* risks eviscerating software patents, while *Planet Bingo* and *buySAFE* provide little help because they involved obvious examples of ineligibility. Although these cases reveal examples of software patents that the Federal Circuit deems ineligible, the cases do not explain when other kinds of software patents survive.

ii. *McRO v. Sega of America, Inc.*

District courts, too, have expressed their views on § 101 in an effort to clarify this area of law. Courts in the Central District of California have been particularly active in offering their views on § 101. *See, e.g., Wolf v. Capstone Photography,*

Inc., No. 2:13-cv-09573 (C.D. Cal. Oct. 28, 2014) (Snyder, J.) (finding unpatentable a computerized process of providing event photographs); *Eclipse IP LLC v. McKinley Equip. Corp.*, No. 8:14-cv-742, 2014 WL 4407592 (C.D. Cal. Sept. 4, 2014) (Wu, J.) (finding unpatentable claims reciting methods for communications).

One Central District of California decision deserves special attention: *McRO, Inc. v. Sega of America, Inc.*, No. 2:12-cv-10327, 2014 WL 4749601, (C.D. Cal. Sept. 22, 2014) (Wu, J.). In *McRO*, the court found unpatentable claims addressed to "automatically animating the lip synchronization and facial expressions of 3D characters." *Id.* at *1. The court acknowledged that at first glance the claims seem tangible and "do not seem directed to an abstract idea." *Id.* at *8. *Id.* Nonetheless, the court found the claims unpatentable. The court observed that *Mayo* requires it to "factor out conventional activity," which it interpreted to include all elements found in prior art. *Id.* at *10. Applying this approach, before performing step one of *Mayo*, the court filtered out all tangible elements found in prior art and focused on the invention's point of novelty. *See id.* at *10. The court determined that the point of novelty was "the idea of using rules, including timing rules, to automate the process of generating keyframes." *Id.* But this idea was abstract. The claims merely recited "obtaining ***a first set of rules*** that define output morph weight set stream as a function of phoneme sequence and time of said phoneme sequence." *Id.* at *8 (emphasis added). The claims did not specify what the rules should be or how the user should choose the rules. *See id.* at *11 ("[T]he user, not the patent, provides the rules."). As a result, the claims "cover[ed] all such rules," preempting the field of "lip synchronization using a rules-based morph target approach." *Id.* at *11. Because the claims did not recite inventive concepts but only an abstract idea, the court found all asserted claims unpatentable. *Id.* at *13.

McRO offers an interesting but problematic interpretation of § 101. *McRO* reads § 101 as requiring a point-of-novelty approach, in which courts filter out claim elements found in the prior art before evaluating a claim for abstractness. The merit to this approach is that it provides a clear test for determining patentability. But ultimately, *McRO* seems to misread the law. Despite its convenience, courts should not apply the point-of-novelty approach when examining claims under § 101.

This Court finds this methodology improper for three reasons. The first reason is that the Supreme Court has held that novelty "is of no relevance" when determining patentability. *See Diehr*, 450 U.S. at 189. In so noting, the Supreme Court rejected *Flook*'s point-of-novelty approach.[13] *McRO* applies this abrogated form of § 101 analysis, despite the fact that the Supreme Court did not revive this approach in *Bilski*, *Mayo*, or *Alice*. Admittedly, *Mayo* does require courts to ignore "well understood, routine, conventional activity" at step two, 132 S. Ct. at 1298, but neither *Mayo* nor any other precedent defines conventional elements to include ***everything*** found in prior art. Rather than relying on *Flook*, courts must follow the guidance of *Diehr*, which discourages courts from "dissecting a claim into old and new elements" when searching for an abstract idea. *Diehr*, 450 U.S. 189 n.12.

The second objection to *McRO*'s methodology is that it conflates step one and step two of *Mayo*. At *Mayo*'s second step, the court must determine whether there is something more than an abstract idea, and conventional elements do not

[13] Justice Stevens' dissent in *Diehr* is proof that the Supreme Court abandoned this methodology. Justice Stevens faults the majority for not focusing on the point of novelty—that is, what the patentee newly invented, as opposed to what the patentee borrowed from the prior art. *See Diehr*, 450 U.S. at 211–12 (Stevens, J., dissenting) ("[I]f the only concept that the inventor claims to have discovered is not patentable subject matter, § 101 requires that the application be rejected without reaching any issue under § 102; for it is irrelevant that unpatentable subject matter -- in that case a formula for updating alarm limits -- may in fact be novel. Proper analysis, therefore, must start with an understanding of what the inventor claims to have discovered -- or phrased somewhat differently -- what he considers his inventive concept to be.").

constitute something more. From this principle, the court in *McRO* determined it must filter out elements found in prior art ***before*** performing step one. This appears to be incorrect, because according to *Alice*, courts should not even consider whether elements are conventional unless the court determines that the invention is abstract at step one. Courts must filter out elements only at step two.[14] *McRO* therefore conflates *Mayo*'s two steps in the face of binding precedent rejecting that approach.

Finally, it is difficult to imagine any software patent that survives under *McRO*'s approach—most inventions today build on what is known in the art, and an improvement to software will almost inevitably be an algorithm or concept which, when viewed in isolation, will seem abstract. This analysis would likely render all software patents ineligible, contrary to Congress's wishes.

Although *McRO* offers valuable contributions to the discussion around § 101, it ultimately appears to reach the wrong conclusion. Federal Circuit precedents likewise offer little guidance for this Court to follow. As such, this Court must look to Supreme Court precedents to properly apply § 101 to computer software.

D. Themes in § 101 Precedents

The decisions discussed above demonstrate the difficulty of interpreting and applying § 101 to software inventions. Given the opacity of *Alice*, it is unsurprising that courts have struggled to define the boundaries of software patentability. Nonetheless, Supreme Court precedents offer broad themes on

---

[14] Judge Wu has observed that *Mayo*'s two-step inquiry is a one-step inquiry:

> Describing this as a two-step test may overstate the number of steps involved. If the claim is not "directed" to a patent-ineligible concept, then the test stops at step one. If the claim is so directed, but we find in step two that the claim contains an "inventive concept" that "transforms" the nature of the claim into something patent eligible, then it seems that there was a categorization error in finding the claim—which is considered "as an ordered combination"—"directed to an abstract idea" in step one.

McRO, 2014 WL 4749601 at *4. But step one does not determine whether the claim as a whole is abstract; rather, it determines whether the claim's purpose is directed to an abstract idea.

software patentability and patentability generally. These themes underlie both steps of the § 101 inquiry and clarify the types of inventions that courts should find patentable.

First, the concern underlying § 101 is preemption. Preemption is the idea that allowing a patent on the invention will impede innovation rather than incentivize it. This preemption concern underlies both steps of the analysis. The court must be wary about overstating this concern. By definition, every patent preempts an area of technology. A patentee with a groundbreaking invention is entitled to monopolize a segment of technology, subject to the limits of the Patent Act.[15] Moreover, the court must be wary of litigants who exaggerate preemption concerns in order to avoid developing innovative workarounds. *See McRO*, 2014 WL 4749601 at *7 ("[W]e must be wary of facile arguments that a patent preempts all applications of an idea. It may often be easier for an infringer to argue that a patent fails § 101 than to figure out a different way to implement an idea, especially a way that is less complicated." (internal quotation mark omitted)). Nonetheless, § 101 prevents patentees from too broadly claiming a building block of research. Building blocks include basic tools of mathematics, as in *Benson*, or formulas describing preexisting natural relationships, as in *Mayo*. But "a novel and useful structure created with the aid of knowledge of scientific truth" may be patentable. *Mackay Radio & Tel. Co. v. Radio Corp. of America*, 306 U.S. 86, 94 (1939).

Second, computer software and codes remain patentable. The Supreme Court approved a patent on computer technology in *Diehr* and suggested that software and code remain patentable in *Alice*. The America Invents Act further demonstrates the continuing eligibility of software. Moreover, *Alice* did not

[15] Justice Stevens in *Flook* expressed skepticism at the notion of preemption as a § 101 concern, perhaps for this reason. *Flook*, 437 U.S. at 590 n.11 ("[T]he formula [in *Benson*] had no other practical application; but it is not entirely clear why a process claim is any more or less patentable because the specific end use contemplated is the only one for which the algorithm has any practical application.").

significantly increase the scrutiny that courts must apply to software patents. It held only that an ineligible abstract idea does not become patentable simply because the claim recites a generic computer. Courts must not extend the reach of *Alice* too far, lest they read in § 101 limitations that do not exist. *Cf. Bilski*, 561 U.S. at 603 ("This Court has not indicated that the existence of these well-established exceptions gives the Judiciary *carte blanche* to impose other limitations that are inconsistent with the text and the statute's purpose and design.").

Third, the Supreme Court has been more skeptical of bare attempts to patent mathematical formulas, as opposed to algorithms generally. An algorithm is not necessarily expressed as a mathematical formula. Rather, an algorithm is a series of steps for accomplishing a goal. *Compare Benson*, 409 U.S. at 65, *and Flook*, 437 U.S. at 585 n.1 (finding patents on algorithms abstract, where Court defined algorithm as a "procedure for solving a given type of mathematical problem"), *with Diehr* 450 U.S. at 186 n.9 (finding algorithm for curing rubber patentable, where Court defined an algorithm as "[a] fixed step-by-step procedure for accomplishing a given result"). Mathematical formulas that describe preexisting relationships or symbolize longstanding ideas create significant § 101 concerns, but not all computerized procedures evoke the same concerns. *See, e.g.*, *Mayo*, 132 S. Ct. at 1298 (finding unpatentable claim expressing natural relationship); *Bilski*, 561 U.S. at 611–12 (finding unpatentable claim expressing hedging risk as mathematical formula). The court should not ignore mathematical formulas in its § 101 analysis, because a formula combined with other elements may transform an abstract idea into patentable subject matter.

Fourth, a claim is more likely to be abstract if it stands for a fundamental practice with a long history, like the method in *Bilski* for hedging risk. However, § 101 does not preclude a claim directed to a longstanding practice that adds something more. The Supreme Court left open the possibility that innovative

elements, rather than "token postsolution components," could make such a claim patent eligible. *See Bilski*, 561 U.S. at 612.

E. Determining Patentability Post-*Alice*

Keeping those observations in mind, this Court must conduct § 101 analysis using the two-part *Mayo* test in the following manner.

i. The First Step of *Mayo*

First, the court must identify whether a claim is directed to an abstract idea. To do this, the court must identify the purpose of the claim—in other words, what the claimed invention is trying to achieve—and ask whether that purpose is abstract. For example, in *Alice*, the court concluded that the claims were directed to mitigating settlement risk using a third party, but the claims recited more. They outlined an entire process, including creating shadow records, obtaining from an exchange institution a start-of-the-day balance, and so on. *See Alice*, 134 S. Ct. at 2359. But these steps were meant to achieve the purpose of mitigating settlement risk. The Supreme Court took the same approach in *Bilski* and *Mayo* by characterizing the claims in terms of the inventions' purposes: hedging risk and applying a natural law, respectively. *See Bilski*, 561 U.S. at 611; *Mayo*, 132 S. Ct. at 1296–97. As discussed above, prior art plays no role in this step.

The characterization of the claim is essential to the § 101 inquiry. In *Diehr*, the dispute boiled down to *what* the majority and dissent were evaluating for abstractness. The *Diehr* majority took the correct approach of asking what the claim was trying to achieve, instead of examining the point of novelty. Courts should recite a claim's purpose at a reasonably high level of generality. Step one is a sort of "quick look" test, the object of which is to identify a risk of preemption and ineligibility. If a claim's purpose is abstract, the court looks with more care at specific claim elements at step two.

After determining the claim's purpose, the court then asks whether this purpose is abstract. Age-old ideas are likely abstract, in addition to basic tools of research

and development, like natural laws and fundamental mathematical relationships. *See Mayo*, 132 S. Ct. at 1296–97; *Bilski*, 561 U.S. at 611–12; *Benson*, 409 U.S. at 71–72. In evaluating whether a purpose is abstract, the court can rely on Supreme Court precedents.

ii. The Second Step of *Mayo*

If the court finds the claim's purpose abstract at step one, it must then determine whether there is an inventive concept that appropriately limits the claim such that it does not preempt a significant amount of inventive activity. In performing this second step of analysis, the court must be wary of making patentability "a draftsman's art." *See Mayo*, 132 S. Ct. at 1294. But inevitably, drafting plays a key role. Patents that claim too broadly or prohibit a vast amount of future inventive activity are suspect. *See Benson*, 409 U.S. at 68; *O'Reilly*, 56 U.S. at 113. Thus, the second step should provide "additional features that provide practical assurance that the process is more than a drafting effort designed to monopolize [the ineligible concept] itself." *Mayo*, 132 S. Ct. at 1297. A claim cannot avoid this preemption concern by limiting itself to a particular technological environment. *See Alice*, 134 S. Ct. at 2357–58 (limiting an abstract idea to computer environment does not mitigate preemption concerns).

With this concern in mind, the court must disregard "well-understood, routine, conventional activity" at step two. *Mayo*, 132 S. Ct. at 1299.[16] A conventional element may be one that is ubiquitous in the field, insignificant or obvious. *See Mayo*, 132 S. Ct. at 1298 ("Purely 'conventional or obvious' '[pre]solution activity' is normally not sufficient to transform an unpatentable law of nature into a patent eligible application of such a law."); *Diehr*, 450 U.S. at 191–92 ("Similarly, insignificant postsolution activity will not transform an unpatentable principle into a patentable process."). A conventional element may also be a necessary step, which a person or device must perform in order to implement the abstract idea.

[16] This Court will refer to this concept as "conventional elements."

For example, the claim elements in *Mayo* were steps all doctors needed to perform in order to apply the natural law. *See Mayo*, 132 S. Ct. at 1298 ("Anyone who wants to make use of these laws must first administer a thiopurine drug and measure the resulting metabolite concentrations, and so the combination amounts to nothing significantly more than an instruction to doctors to apply the applicable laws when treating their patients."). However, as discussed above, conventional elements do not constitute everything in the prior art, although conventional elements and prior art may overlap. *But see McRO*, 2014 WL 4749601 at *9–11 (using prior art to identify conventional elements).

The court must also consider claim elements as a combination. A combination of conventional elements may be unconventional. *See Diehr*, 450 U.S. at 188 ("[A] new combination of steps in a process may be patentable even though all the constituents of the combination were well known and in common use before the combination was made."). For example, in *Diehr*, the combination of steps, which the Supreme Court characterized as unconventional, ensured the claim was patentable. Courts should consider mathematical formulas as part of the "ordered combination," even though, in isolation, the formulas appear abstract. *See Diehr*, 450 U.S. at 189 n.12.

V. Discussion

Caltech's patents recite methods of encoding and decoding data in accordance with an IRA code. At step one, this Court determines that all asserted claims are directed to the abstract idea of encoding and decoding data for the purpose of achieving error correction. Nonetheless, at step two, this Court finds that the claims contain elements that provide an inventive concept. When claims provide a specific computing solution for a computing problem, these claims should generally be patentable, even if their novel elements are mathematical algorithms. That is the case with all of Caltech's asserted claims, which the Court has concluded are patentable.

The Court begins by analyzing only the independent claims of the patents. If the independent claims are patentable, so are the dependent claims. Logically, adding additional elements to non-abstract claims will not make them abstract.

A. Step One: Caltech's Asserted Claims Are Directed to Abstract Ideas

At step one, the Court finds that all the claims at issue are directed to abstract ideas. First, the Court must ask what these claims are trying to achieve. The Court determines that the purposes of the claimed inventions are to encode and decode data to achieve error correction. The claims explicitly recite the fundamental concepts of encoding and decoding data. *See, e.g.*, '032 Patent, 9:57–58 (reciting "device comprising a message-passing decoder"); '710 Patent, 7:14 (reciting "method of encoding a signal"). The concepts of encoding and decoding are longstanding steps in the process of error correction. *See* Sarah J. Johnson, *Iterative Error Correction: Turbo, Low-Density Parity-Check and Repeat-Accumulate Codes* 1, 34 (Cambridge University 2010). *See generally* Robert G. Gallager, *Low-Density Parity-Check Codes* (1963). A patent on these essential concepts, without something more, would threaten to preempt the entire field of error correction. *See* Johnson, *supra*, at 34 (describing use of "parity bits as a means to detect and . . . correct errors in digital data" as theorized by Gallager in 1962 thesis); *id.* at 71 (discussing emerging prevalence of Gallager's ideas).[17]

As such, the purpose of these claims—encoding and decoding data for error correction—is abstract. These ideas, stated at this level of generality, existed long before the patents and were well known in the field. This fact compels the Court's conclusion. Also buttressing the Court's conclusion is the prevalence of these error correction techniques in the field. The primary method of error correction is

[17] For § 101 analysis, it does not matter that certain claims recite "devices" or "coders." Courts must ignore generic recitation of hardware at step one, when the claimed hardware essentially performs a method. *See Alice*, 134 S. Ct. at 2360 ("Put another way, the system claims are no different from the method claims in substance. The method claims recite the abstract idea implemented on a generic computer; the system claims recite a handful of generic computer components configured to implement the same idea.").

encoding and decoding data. Admittedly, this patent claims specific methods of encoding and decoding data for error correction. But at step one, the Court looks only to the general purpose of the claims, as the Supreme Court did in *Bilski*, *Mayo*, and *Alice*. At step two, the Court focuses on specific limitations.

B. Step Two: Caltech's Asserted Claims Are Patentable Because They Contain Inventive Concepts

Despite being generally directed to abstract concepts, the asserted claims contain meaningful limitations that represent sufficiently inventive concepts, such as the irregular repetition of bits and the use of linear transform operations. Although many of these limitations are mathematical algorithms, these algorithms are narrowly defined, and they are tied to a specific error correction process. These limitations are not necessary or obvious tools for achieving error correction, and they ensure that the claims do not preempt the field of error correction. The continuing eligibility of this patent will not preclude the use of other effective error correction techniques. Therefore, all of the asserted claims are patentable.

i. '032 Patent

The claims of the '032 patent contain inventive concepts that makes them patentable. Claim 1 of the '032 patent recites generating a parity bit by accumulating two values: (i) the value of the previous parity bit and (ii) the sum of a number of randomly chosen irregular repeats of message bits.[18] As Hughes correctly notes, the claim's other limitations recite generic steps such as receiving and transmitting message bits. Such limitations are conventional both here and in all other asserted claims, because they are necessary for achieving error correction.

[18] This concept is expressed in claim 1 of the '032 patent through the mathematical formula

$$x_j = x_{j-1} + \sum_{i=1}^{a} v_{(j-1)a+i}$$

in which the first term on the right side is the "value of parity bit j-1" and the second term on the right side is the "value of a sum of 'a' randomly chosen irregular repeats of the message bits."

Therefore, whether viewing the claim's elements as a combination or individually, the patentability of claim 1 depends greatly on its recited formula.

One of Hughes' arguments deserves special attention. Hughes argues that calculating parity bit values involve "mental steps [that] can be performed by a person with pencil and paper." Therefore, Hughes, argues the claim is not patentable. Defs.' Mem. in Supp. of Invalidity at 14, Dkt. No. 126. The Court finds this mode of analysis unhelpful for computer inventions. Many inventions could be theorized with pencil and paper, but pencil and paper can rarely produce the actual effect of the invention. Likewise, with regard to software, a human could spend months or years writing on paper the 1s and 0s comprising a computer program and applying the same algorithms as the program. At the end of the effort, he would be left with a lot of paper that obviously would not produce the same result as the software.[19]

The problems of pencil-and-paper analysis are heightened in the context of software, which necessarily uses algorithms to achieve its goals. Pencil-and-paper analysis can mislead courts into ignoring a key fact: although a computer performs the same math as a human, a human cannot always achieve the same results as a computer. Hughes' statement is theoretically correct. A human could perform the calculations that would yield the value of a parity bit. But Hughes' statement is literally wrong. It states the obvious to say that a pencil and paper cannot actually produce parity bits. Hughes' proposed analysis oversimplifies § 101 and ignores the fact that the '032 patent creates an algorithmic solution for a computing problem—the corruption of data during transmission.

[19] Courts should not view software as abstract simply because it exists in an intangible form. It is as fruitless to say that a human could use pencil and paper to perform the same calculations as a computer, as it is to say that a human could use pencil and paper to write down the chemical structure of a DNA strand. In either case, any effort on the part of a human will only be a symbolic representation. The effort will not produce the same effect as executing a computer program or isolating a DNA strand.

The pencil-and-paper test is a stand-in for another concern: that humans engaged in the same activity long before the invention of computers. *See, e.g.*, *Enfish, LLC v. Microsoft Corp.*, 2:12-cv-07360 (C.D. Cal. Nov. 3, 2014) (finding unpatentable claims addressed to storing information in logical tables on computers). This concern is highly relevant, but courts should scan patents for this concern by using a test that creates false positives. In the case at hand, it is clear that Caltech's error correction codes were not conventional activity that humans engaged in before computers, and the codes do not become conventional simply because humans can do math. Pencil-and-paper analysis is inappropriate at least for this area of technology.

The Court should not ask whether a human can calculate parity bit values using pencil and paper. Instead, the Court must ask whether the formula in claim 1 constitutes an inventive concept that sufficiently limits the claim's preemptive effect. It does. Hughes argues that the Supreme Court has endorsed a bright line rule against patenting mathematical formulas. *See* Defs.' Mem. at 8; *see also Mayo*, 132 S. Ct. at 1303 ("[C]ases have endorsed a bright-line prohibition against patenting laws of nature, mathematical formulas and the like, which serves as a somewhat more easily administered proxy for the underlying 'building block' concern."). But this dictum is misleading. On the contrary, Supreme Court precedent allows mathematical formulas to be considered in § 101 analysis. *See Diehr*, 450 U.S. at 189 n.12. Error correction codes depend on algorithms that may be reduced to mathematical formulas. Hughes' rule would make all error correction codes, and much of computer software, ineligible subject matter.

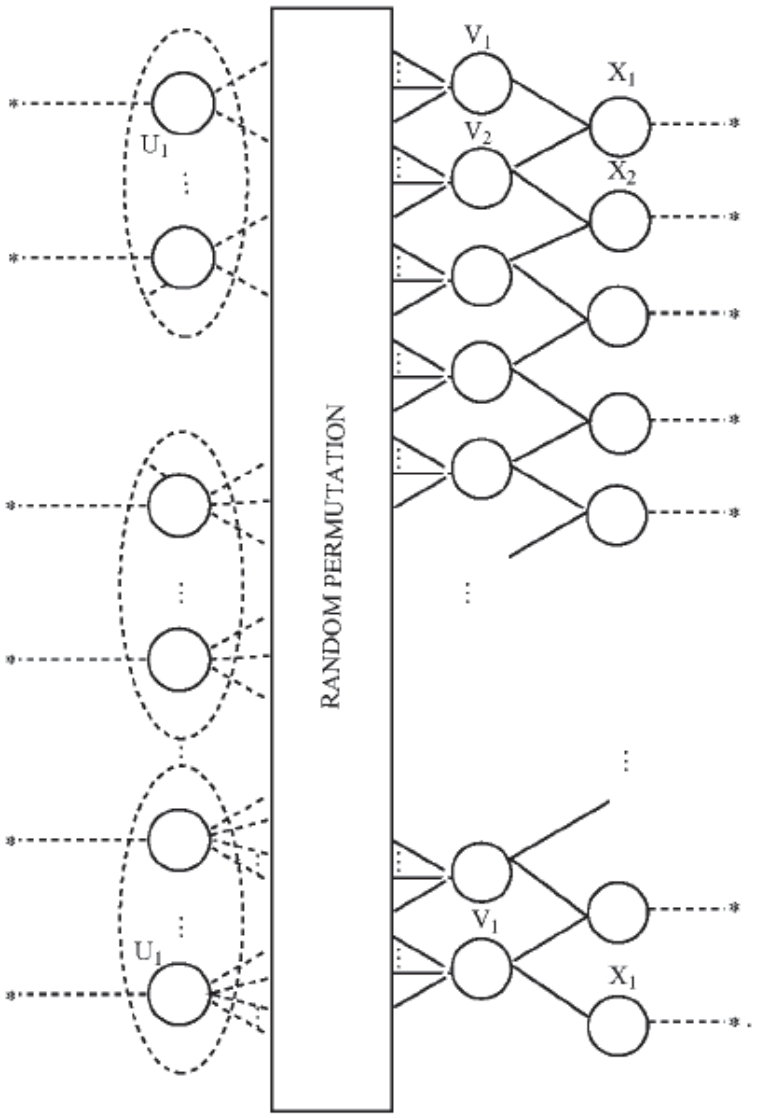
Hughes' other cited cases are inapposite. The claims in *Benson* essentially described a natural relationship between two well-known number systems, BCD and pure binary, and reduced that relationship to a formula. Thus, the claim set forth a formula for converting one well-known numerical representation to another. This kind of discovery is not eligible for patent protection. *See Benson*, 409 U.S.

at 65 ("[The procedures] are a generalized formulation for programs to solve mathematical problems of converting one form of numerical representation to another."). Likewise, the claim in *Flook* recited a formula that captured the process of updating an alarm limit—a process that operators engaged in long before the claims existed. The formula was written broadly as to capture a swath of situations where an operator updated an alarm limit; the claim did "not purport to explain how to select the appropriate margin of safety, the weighting factor, or any of the other variables." *Flook*, 437 U.S. at 586.

But in claim 1 of the '032 patent, the mathematical formula reflects inventive concepts: namely, the irregular repetition of message bits and the use of a prior parity bit for calculating a subsequent parity bit. Irregular repetition is a significant benefit of this invention, as it balances the goals of efficiency and accuracy in error correction. The innovative use of a prior parity bit further improves efficiency. The mathematical formula in claim 1 does not describe a preexisting relationship but rather sets forth unconventional steps for achieving error correction.

These two claim elements are not necessary for achieving error correction, and Hughes has not suggested they were ubiquitous or obvious. In fact, these steps greatly limit the scope of the claim. The claim does not capture many forms of error correction, including turbo codes and regular repeat-accumulate codes. As such, the claims do not preempt the field of error correction but capture only one effective form of error correction.

Similar analysis applies to claim 18. Claim 18 recites a message-passing decoder that decodes data encoded according to the depicted Tanner graph:



The left side of the Tanner graph depicts subsets of information nodes (designated with the letter U on the graph). The subsets repeat a different number of times, as shown by the edges exiting the subsets. These edges enter a "Random Permutation" box, which represents the scrambling of the edges joining the information nodes and check nodes (designated with the letter V). The right side of the graph depicts parity bits (designated with the letter X) that are connected to two check nodes. Each check node has a value of 0 or 1. By summing all the bits connected to a check node, the encoder can determine the value of the next parity

bit.[20] *See* '032 Patent, 10:10–40. Again, this claim recites unconventional steps that constitute inventive concepts—irregular repetition and the use of a prior parity bit to calculate the next parity bit. Either individually or in combination with the claim's other elements (including the scrambling of bits), these unconventional steps sufficiently limit preemption concerns.

The other asserted claims from the '032 patent are dependent on these independent claims. Because the independent claims are patentable, these dependent claims are patentable as well.

ii. '781 Patent

Claim 1 of the '781 patent contains inventive concepts that make it patentable. It recites a method of encoding a signal by (i) performing a linear transform operation on information bits to produce "L transformed bits," and (ii) accumulating the L transformed bits to produce at least a portion of a codeword. The claim's other recited limitation is a conventional step of receiving a block of data to be encoded.

The claim contains two elements that provide an inventive concept: a linear transform operation to produce L transformed bits and the accumulation of these bits to produce a codeword. Hughes does not argue that these elements alone or in combination were ubiquitous in the field or obvious. Instead, Hughes argues that a linear operation is a mathematical algorithm, and *Digitech* states that mathematical algorithms are not patentable unless there are other limitations. But the breadth of the claims at issue in *Digitech* far exceeds the breadth of this claim.[21] The

---

[20] To understand this concept, imagine a check node has the value 0. It is connected to three information bits, all with the value 1. It is connected to two parity bits, one with the value 1 and one with a value to be determined. Let us call the value of the undetermined parity bit *y*. To solve for *y*, the encoder would use the following formula: 0=1+1+1+1+*y*. Using mod-2 addition, the encoder would determine that the value of *y* is 0.

[21] The method claim at issue in *Digitech* recited:

> A method of generating a device profile that describes properties of a device in a digital image reproduction system for capturing, transforming or rendering an image, said method comprising:

mathematical operation here greatly limits the claim's scope. As with the claims of the '032 patent, claim 1 of the '781 patent does not preempt a significant number of error correction techniques.

The Court is not required to ignore the linear transform operation simply because the operation is mathematical. *See Diehr*, 450 U.S. at 189 n.12. Again, if courts could not consider mathematical operations in § 101 analysis, error correction codes and most software would be unpatentable. Using a linear transform operation to produce bits, which are accumulated to produce a codeword, is an innovative application of a mathematical principle. *See Mackay*, 306 U.S. at 94 ("[W]hile a scientific truth, or the mathematical expression of it, is not patentable invention, a novel and useful structure created with the aid of knowledge of scientific truth may be."). As such, claim 1 is patentable.

Claims 16 and 19 recite methods of encoding a signal that do not require a linear transform operation but are nonetheless patentable. Claim 16 recites, in part,[22] a method of encoding a signal by (i) accumulating mod-2 or exclusive-OR (XOR) sums of bits in subsets of information bits, to generate at least a portion of a codeword, (ii) where the information bits appear in a variable number of subsets.

> generating first data for describing a device dependent transformation of color information content of the image to a device independent color space through use of measured chromatic stimuli and device response characteristic functions;
>
> generating second data for describing a device dependent transformation of spatial information content of the image in said device independent color space through use of spatial stimuli and device response characteristic functions; and
>
> combining said first and second data into the device profile.

Digitech, 758 F.3d at 1351. Claim 1 of the '781 patent is not so broad. If claim 1 instead recited a method of encoding a signal by performing ***any*** mathematical operation on data to produce a codeword, the claim would be akin to the one in *Digitech*, and Hughes would have a much stronger argument against patentability.

[22] Claim 16 is dependent on claim 13 of the '781 patent. In this analysis of claim 16, the Court first analyzes the elements of independent claim 13. If claim 13 is patentable, the Court need not analyze the added elements in claim 16.

Claim 19 recites the same method, except that it specifies that two of the information bits should appear in three subsets of information bits.

Claim 16 recites unconventional steps of using, as input, information bits from a variable number of subsets and accumulating mod-2 or XOR sums of bits to produce a codeword. Hughes has not shown that these steps were ubiquitous or obvious but only raises the objection that the Court cannot consider mathematical operations. Although mod-2 arithmetic alone is a conventional idea, the accumulation of a selection of bits from a variable number of subsets is not. Likewise, claim 19 recites an inventive concept of accumulating mod-2 sums of bits and requiring at least two information bits to appear in three subsets. Claim 16 and 19 both recite inventive concepts that satisfy step two.

The other asserted claims from the '781 patent are dependent on claim 1. Because this independent claim is patentable, the other asserted claims are patentable as well.

iii. '710 Patent

The asserted claims of the '710 patent contain inventive concepts that make them patentable. Claim 1 of the '710 patent recites a method of encoding a signal by (i) partitioning a data block into sub-blocks, (ii) repeating the data elements in different sub-blocks a different number of times, (iii) interleaving the repeated data elements, and (iv) using an encoder to encode the data block with a rate close to 1. Claim 15 of the '710 patent recites a coder that performs substantially the same process. The other asserted claims from the '710 patent are dependent on the above claims. As such, the patentability of all asserted claims in the '710 patent rises and falls with claim 1.

Like the asserted claims of the '032 patent, claim 1 contains the inventive concept of repeating data elements irregularly. As discussed above, the irregular repetition of bits is an innovative feature that balances efficiency and accuracy. Moreover, the claim requires the encoder to encode the data block with a rate close

to 1, which means that the encoder is restricted in the number of extra bits it can produce. This coding rate requirement is a feature that ensures the code is efficient and does not produce a significant number of unnecessary bits. This requirement is unconventional and significantly limits the breadth of the claim. At least in combination with the claim's other elements, including the irregular repetition of bits, this element constitutes an inventive concept. Therefore, claim 1 is patentable.

Claim 15 is likewise patentable. In fact, claim 15 specifically requires the coding rate to be within 10 percent of a coding rate of 1.[23] This requirement constitutes an inventive concept and sufficiently limits the claim's breadth. The other asserted claims from the '710 patent are dependent on claims 1 and 15 and therefore are also patentable.

iv. '833 Patent

Finally, the asserted claims of the '833 patent contain inventive concepts that make them patent eligible. Claim 8 recites the elements of (i) combining[24] bits in one set of memory locations to other bits in a second set of memory locations, based on a corresponding index, (ii) accumulating these bits in the second set of memory locations, and (iii) requiring a permutation module to read two or more of the memory locations in the first set at different times from each other. Hughes skims over these limitations and characterizes the elements as mathematical processes, but as discussed above, the Court can and must consider mathematical processes in § 101 analysis. Even if the Court could not consider mathematical processes, Hughes makes no argument that element (iii) is conventional. Again, given the claim's limitations, the claim does not have a significant preemptive effect in the field of error correction. There is no basis for the Court to conclude that these elements were ubiquitous or obvious in the field or are necessary for

[23] For an explanation of coding rate, see section II, *supra*.

[24] The Court has construed "combining" to mean "performing logical operations on." *Cal. Inst. of Tech.*, 2014 WL 3866129 at *10–11.

encoding data to achieve error correction. In sum, the recited algorithm constitutes an inventive method of encoding data, making claim 8 patentable. Claim 1 of the '833 patent recites an apparatus that performs the steps of claim 8 and is also patentable.

The other asserted claims from the '833 patent are dependent on claims 1 and 8. Because these independent claims are patentable, the dependent claims of the '833 patent are as well.

VI. Conclusion

Section 101 must strike a precise balance in the context of software patents. On the one hand, patent law should not protect inventions that simply apply longstanding ideas to a computer environment. On the other hand, patents should encourage inventors to create new computing solutions to today's computing problems. Caltech's patents improve a computer's functionality by applying concepts unique to computing (like using a linear transform operation to encode data) to solve a problem unique to computing (data corruption due to noise).[25] The Supreme Court in the future may provide a clearer outline for applying § 101 to software, but to this Court, it at least must be true that § 101 protects a unique computing solution that addresses a unique computing problem.

Today, the Court decides only that the asserted claims are patentable under § 101. Whether these claims survive § 102, § 103, or other requirements of the Patent Act is a separate question for another day, and the Court expresses no views on these issues. Because the asserted claims are patentable under § 101, Hughes' motion for summary judgment is denied.

---

[25] At least one other court has recently found claims for an error correction code eligible under § 101. In *France Telecom S.A. v. Marvell Semiconductor Inc.*, No. 12-cv-04967, 2014 WL 1478850 (N.D. Cal. Apr. 14, 2014), Judge Orrick upheld as patentable claims for a turbo code, a type of error correction code. *See id.* at *7–12. This decision was released before the Supreme Court's decision in *Alice* and relies in part on the Federal Circuit's language in *CLS Bank Int'l v. Alice Corp. Pty. Ltd.*, 717 F.3d 1269, 1282 (Fed. Cir. 2013), *aff'd* 134 S. Ct. 2347 (2014).

//

//

IT IS SO ORDERED.

DATED: November 3, 2014

Hon. Mariana R. Pfaelzer
United States District Judge