The Hard Side of Software: The Difficulty of Patenting Software Amid Abstract Ideas

# INTRODUCTION

The United States Patent System is rooted in the Constitution and first laid out in law by the Patent Act of 1790.[[1]](#footnote-1) The first patent granted by the United States of America was to Samuel Hopkins for an improved method for creating pot ash and pearl ash.[[2]](#footnote-2) Today, patent law is codified in Title 35 of the United States Code as enacted Congress in 1952[[3]](#footnote-3), the United States Patent and Trademark Office (“USPTO”) has issued its ten millionth patent; computer and software related patents reign supreme, the byproducts of a global economy increasingly dependent on computer technology.[[4]](#footnote-4) Each year half of the patents issued by the USPTO are related to software.[[5]](#footnote-5) Despite the ubiquity of computer technology, software patents can be difficult to obtain and defend.[[6]](#footnote-6) [Per Arielle’s comment, give some context. Let’s make this intro YUGE] In the wake of the *Berkheimer v. HP* decision[[7]](#footnote-7), the USPTO issued the Berkheimer Memo, which severely limited how patent examiners reject patent applications for not meeting the requirements for patent eligibility contained and implied by 35 U.S.C. § 101.[[8]](#footnote-8) This change eased the burden on applicants seeking software patents, but also created a diverging standard between the judicial system and the USPTO’s standard for approval, which could lead to granting invalid patents. The patent system has too long shoe-horned software patents into an antiquated system designed to encourage innovation of only physical inventions. [Thesis. THESIS HERE]. This note will guide readers through the legislative history and standards regarding software patents, the repercussions of the *Berkheimer* decision, and a possible solution to this patent software dilemma: updating the United States patent system.

# BACKGROUND: Hurdles & History of Software Patents

## The Easy Hurdle: Patentability

Patents may be obtained for “any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof.”[[9]](#footnote-9) This broad statement is qualified by the concepts of patentability and patent eligibility, which are defined by both statute and judicial rulings.[[10]](#footnote-10) Patentability describes the standards to receive a patent, requiring an invention to be novel and nonobvious, and disclosed in detail.[[11]](#footnote-11)[Discuss how patentability is actually usually pretty easy to get around and isn’t at issue here]. The second standard, patent eligibility, concerns what subject matter warrants the incentives and protections provided by patent law.[[12]](#footnote-12) These two standards are closely related and are often conflated by courts when patent validity is considered.[[13]](#footnote-13)

## Patent Eligibility: Laws of Nature, Natural Phenomena, and Abstract Ideas

The Supreme Court holds the statutory language “any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof” contains an implicit exception disallowing patents for laws of nature, natural phenomena, and abstract ideas.[[14]](#footnote-14) These three exceptions prevent monopolies on foundational laws and concepts that many technologies may depend on, insuring that the limited monopoly provided for by law does not hinder innovation.[[15]](#footnote-15) However, too broad an interpretation of these exceptions would also impede the patenting of important discoveries, therefore an invention or discovery cannot be unpatentable merely because it contains natural phenomena or algorithm.[[16]](#footnote-16)

These three exceptions not always clearly defined or distinct from one another. Laws of nature include well known laws of physics such as gravity or Einstein’s famous “E = mc2”.[[17]](#footnote-17) Natural phenomena include more physical natural occurrences such as plants, minerals, lightning.[[18]](#footnote-18) The exceptions extend to even newly discovered uses of naturally occurring phenomenon.[[19]](#footnote-19) In *Funk Brothers Seed Co. v. Kalo Inoculant Co.* [finish citation], the disputed discovery was a mixture of bacteria for inoculating the seeds of several different types of plants at once, rather than separately.[[20]](#footnote-20) The court found the mixture ineligible for patent and therefore no infringement because the bacteria had not been altered; it was a natural phenomenon.[[21]](#footnote-21) In contrast, in *Diamond v. Chakrabarty*, a new species of bacteria that digested oil was upheld to be patentable because it was not naturally occurring and it was unpatentable simply because it was a living organism.[[22]](#footnote-22)What counts as an abstract ideas is not always clear.

Processes are expressly included within § 101: “any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof.”[[23]](#footnote-23) However, process claims face great scrutiny under the judicial exception of abstract ideas. Machines, new manufactures and compositions of matter are physical and easily patentable, providing the invention meets requirements for patentability.[[24]](#footnote-24) Processes are not as tangible and can easily be classified as an abstract idea therefore are more difficult to patent.[[25]](#footnote-25)

## Abstract Ideas: Inconsistent Case Law

Abstract ideas often overlap with the other judicial exceptions of natural phenomena and laws of nature. There is no legal definition of “abstract ideas” and what is abstract can be difficult to describe. Algorithms and mathematical formulas are common examples of an abstract idea, but can also fall under any or all of the three judicial exceptions, as certain natural laws are easily describe mathematically.[[26]](#footnote-26) With no definition, applicants and litigants are left with vague statements as guidance, such as the following:

Limiting formulas to a particular technological environment will not make them patentable.[[27]](#footnote-27) Patents are not ineligible merely because they rely on a law of nature or algorithm.[[28]](#footnote-28) To be patentable, inventors must do more than describe the idea or law and append the words “apply it”.[[29]](#footnote-29) Patents must have additional steps outside of the patent ineligible concept that integrate the equation into the process as a whole.[[30]](#footnote-30) They must have an inventive concept outside of the patent ineligible idea.[[31]](#footnote-31)

Statements like these do give some clarity on what an applicant must do to meet the requirement of patent eligibility. However, the lack of a formal legal definition has led to contradictory application of these guidelines and inconsistent judicial decisions.

[Discuss Allapat – see state street - ] math allowed, how is it not just putting it in a technological environment? The formula is the process

[Discuss Arrhythmia Research Technology Inc. v. Corazonix Corp – see state street] – math applied. Contrast with benson

In 1972, before the rise of the digital age, the Supreme Court heard arguments for *Gottschalk v. Benson* regarding a method for converting binary coded decimal to pure binary. In binary coded decimal (BCD) each digit of a number is represented by a four-digit binary segment e.g. 534 would be represented as 0101-0011-0100, where 0101, 0011, and 0100 are the numbers 5, 3, and 4 respectively in pure binary; in pure binary 534 would be represented as 1000010110.[[32]](#footnote-32) With no definition of abstract ideas, the Court compared the claims with previous cases and facts.[[33]](#footnote-33) However, at the time the patentability of computer programs was still in debate, and the patent was held invalid because it has no practical exception outside of a digital computer.[[34]](#footnote-34) Clearly the Court did not understand the importance digital (the more advanced counter-part to analog) computers would play in the future, and so they dismissed the patent for having no use outside of computers. Although the claims described hardware components executing this algorithm, it was not enough to meet the Court’s requirements for patentability.

In 1980 the Supreme Court considered claims for a process for curing synthetic rubber in *Diamond v. Diehr*. This process relied heavily on a formula known as the Arrhenius equation. By continually taking temperature measurements, a digital computer would use the formula to provide an accurate cure time.[[35]](#footnote-35) Here the Supreme Court upheld the patent, stating the claims must be considered as a whole, and that use of a mathematical formula did not disqualify a patent. This decision seemly reversed *Benson*, but the Supreme Court, distinguished it from *Benson*, characterizing the *Diehr* claims as a method for curing rubber, rather than a math formula. The court emphasized examining the claims as a whole, rather than individually. Upon examination, the Court found that the additional steps integrated the equation into a process and therefore patentable. The patent did not seek to protect the formula, but rather the process of how the formula was used.[[36]](#footnote-36) Despite similar dependencies on mathematical formula, two different outcomes occurred. This suggests that physical transformation is required for patent eligibility[[37]](#footnote-37) and how an invention is described and labelled is critical to patent survival.

[Summary Paragraph, contrast these 4 different cases and discuss how they’re inconsistent with the “guidelines” in Bilski, Benson, Diehr, Flook. The Courts decisions are arbitrary]

## Abandoned Tests

[Discuss abandoned state street test] – process produced a “useful, concrete and tangible result” - abandoned in favor of m-o-t (in re bilski). An early test

The “machine or transformation test” was used as the sole test by the Federal Circuit to determine patent eligibility of a process.[[38]](#footnote-38) This test, articulated in *Benson* and affirmed in *Diehr*.[[39]](#footnote-39), required an applicant to show the claim was tied to a machine, or that it transformed an article.[[40]](#footnote-40) If either of these criteria was met, then the process was not an abstract idea and therefore patent eligible. A claim “tied to a particular machine” is only implemented on a specific machine. The second path of the machine-or-transformation test deems a process patent eligible if it transforms “particular article into a different state or thing”.[[41]](#footnote-41) In *Diehr*, the process included a mathematical formula, a perfect example of an abstract idea. Despite this, it met both criteria of the machine-or-transformation test and was therefore patent eligible.[[42]](#footnote-42) The formula calculated the time to cure rubber and was integrated into a process tied to a specific oven and technological set up. With these additional elements it was deemed patent eligible[[43]](#footnote-43) In contrast a few years prior to *Diehr*,in *Parker v. Flook*, the applicant had claimed a “Method for Updating Alarm Limits” which was a mathematical formula for updating alarm limits during catalytic conversion processes. There was no novel machine or physical connection and was deemed not patent eligible.[[44]](#footnote-44) [Concluding sentence summarizing and relating back to thesis and why this is a problem].

When first introduced this test was not the sole determining inquiry it became. In *Benson* the Supreme Court stated it was “*the* clue” to patent eligibility.[[45]](#footnote-45) Initially used with the caveat that a process may be valid even without meeting the machine-or-transformation tests, the test was reaffirmed several times and lost the caveat until eventually the Federal Circuit used it as the sole test.[[46]](#footnote-46) Soon after the Supreme Court stated the machine-or-transformation test was never intended to be an exhaustive or exclusive test, arguing the machine-or-transformation test would create uncertainty as to the patentability of software.[[47]](#footnote-47) Today there is no test for determining abstract ideas and each patent must be considered on a case-by-case basis, comparing the invention at hand to others rather than applying a set of factors.[[48]](#footnote-48)

## Software Patents & Abstract ideas (Why is software difficult to patent)

Software is not easily sorted into the categories of 35 U.S.C. § 101. It not a “machine, manufacture, or composition of matter”, and must therefore be a “process”.[[49]](#footnote-49) Processes are not always physical and can easily be called abstract ideas. This difficulty extends to software, which is often difficult to patent for several reasons: 1) it is intangible; 2) software frequently is made up of algorithms (a classic example of an abstract idea); and 3) it can be seen as a mere representation of an abstract idea.[[50]](#footnote-50) Many, including a past director of the USPTO and a former Federal Circuit Judge, believe that this eligibility requirement is stifling innovation with its chilling effect on patents.[[51]](#footnote-51) These inherent difficulties require inventors to claim their inventions in specific ways. Early decisions relating to software patents seemed to require physical effects to make the software appear mechanical and more easily fit into the other statutory categories[[52]](#footnote-52). The outcomes of *Diehr* and *Benson* exhibit this requirement. In *Diehr*, the process for curing rubber was upheld despite of its reliance on a mathematical formula, because of the physical transformation that occurred in the rubber. *Benson*’s conversion from BCD to binary was not overtly physical was invalidated.[[53]](#footnote-53)

## The State of the Practice Recent Methods & Cases

To help root software into the physical realm, applicants will often claim software with a Beauregard claim.[[54]](#footnote-54) These claims began in response to *In re Beauregard*, which quoted the Commissioner of Patents and Trademarks, who stated that software embodied in a tangible medium was patentable.[[55]](#footnote-55) Thus claims often contain a variation on the following, “A computer readable medium containing program instructions…”[[56]](#footnote-56) in an effort to connect the abstract nature of software with something tangible and real.

In 2012, the Supreme Court heard a petition concerning processes that help doctors who administer thiopurine drugs determine if a dosage is too low or too high. The Mayo framework introduced a two-step process to determine if an invention claims “building blocks of human ingenuity, which are ineligible for patent protection” or if the patent integrates building blocks into something more. This case, while not software related, introduced the standard that is now the method of considering abstract ideas. [What was the impact?]

[explain the meaning of “directed to”] – https://www.uspto.gov/web/offices/pac/mpep/s2106.html

[alice] In 2014, the patent world was rocked by *Alice Corp. Pty. Ltd. v. CLS Bank Intern*, in which the Supreme Court ruled that escrow software was a patent ineligible invention. The opinion created a two-step framework, based largely on the *Mayo* holding, to determine if an invention was patent eligible. In step one of the framework the Court must determine if the claims at issue are directed to a patent-ineligible concept; if yes then they proceed to step two which asks: “what else is there in the claim before us?”[[57]](#footnote-57) This second step is determining whether additional elements transform the nature of the claim into patent eligible application.[[58]](#footnote-58) The claim elements must be considered individually and in combination. [Insert more about the mechanics of the decision, and how the factors weighed in the case at hand]. In the month after *Alice*, 830 patent applications were withdrawn.[[59]](#footnote-59) In the year following the decision, the Federal Circuit used the two-step framework in ten cases[[60]](#footnote-60) and only found one to contain patent eligible subject matter.[[61]](#footnote-61)

[Discuss DDR Holdings].

Enfish (2016)

Core Wireless

# The Problem at Hand: Berkheimer

In early 2018 the Federal Circuit heard oral arguments for *Berkheimer v. HP*. The Appellant, Steven Berkheimer, was the patentee and brought action for infringement of his patent that described methods for digital file processing and archiving.[[62]](#footnote-62) The claimed invention parsed files into objects and tags the objects to create relationships. The objects are then compared to archived objects to determining variation. The system eliminates redundant storage of common text and graphic element, improving operating efficient and storage.[[63]](#footnote-63) Finding several claims indefinite, the Court then considered patent eligibility of the claims.[[64]](#footnote-64)

Following the questions presented in Alice/Mayo, the Court found that the claims were directed to abstract ideas of parsing, comparing, storing and editing data. Some genuine issue of material fact remained concerning whether some of the claims contained transformative invention concept and was summarily remanded.[[65]](#footnote-65)

# The Nay Sayers (Those that think there’s not a problem)

# The Solution (Analysis)

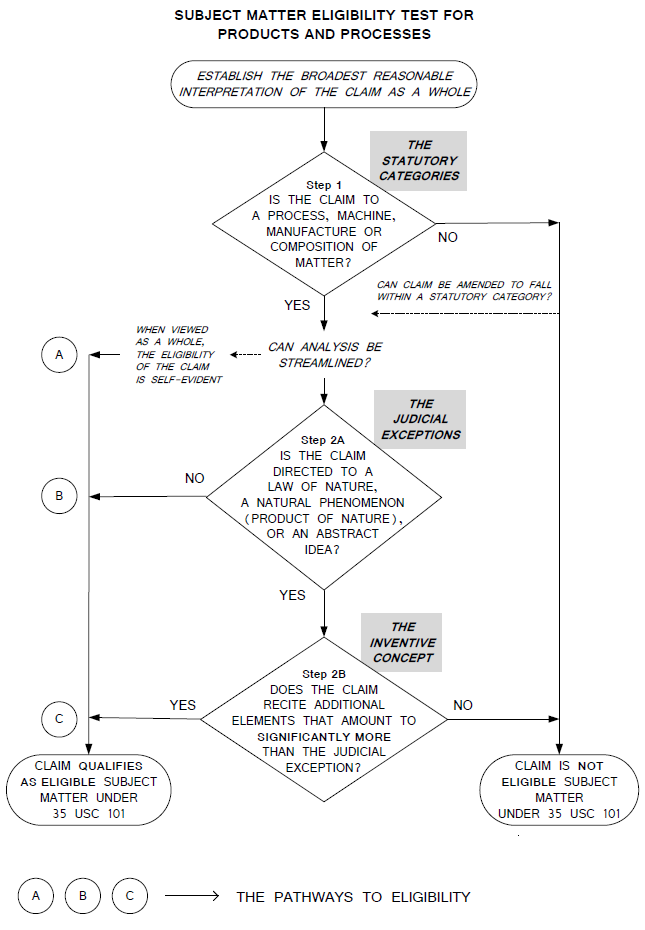


Figure 1 - https://www.uspto.gov/web/offices/pac/mpep/s2106.html

Pledge of Honesty

On my honor, I submit this work in good faith and pledge that I have neither given nor received improper aid in its completion.

/s/ Seth Guthrie

Good start. Watch out for ways to improve sentence structure and pay attention to your grammar. Your explanation of software patent history could use a bit more work for clarity. Make sure you include your prescription for patent system updates. Your analysis should include the argument of those who view the patent system as fine for software needs.

1. U.S. Const. art 1, § 8, cl. 8. [↑](#footnote-ref-1)
2. https://10millionpatents.uspto.gov/ [↑](#footnote-ref-2)
3. MPEP – INTRODUCTION. <https://www.uspto.gov/web/offices/pac/mpep/mpep-0020-introduction.html>. [HOW TO CITE MPEP?] [↑](#footnote-ref-3)
4. https://10millionpatents.uspto.gov/ [↑](#footnote-ref-4)
5. <https://www.ipwatchdog.com/2017/05/21/alice-over-half-u-s-utility-patents-issued-annually-software/id=83367/>; https://www.ificlaims.com/rankings-trends-2017.htm [↑](#footnote-ref-5)
6. *See e.g.* Benson, Berkheimer [↑](#footnote-ref-6)
7. Berkheimer [↑](#footnote-ref-7)
8. How in the world do you cite the Berkheimer memo? [↑](#footnote-ref-8)
9. 35 U.S.C. § 101 [↑](#footnote-ref-9)
10. 35 U.S.C. §§ 102, 103, 112; *Mayo Collaborative Servs. v. Prometheus Labs., Inc.*, 566 U.S. 66, 70, 132 S. Ct. 1289, 1293, (2012). [↑](#footnote-ref-10)
11. 35 U.S.C. §§ 102, 103, 112 [↑](#footnote-ref-11)
12. CITATION NEEDED [↑](#footnote-ref-12)
13. CITATION NEEDED [↑](#footnote-ref-13)
14. *Mayo Collaborative Servs. v. Prometheus Labs., Inc.*, 566 U.S. 66, 70 (2012) (restating the long-held exceptions 35 U.S.C. § 101: laws of nature, natural phenomena, and abstract ideas). [↑](#footnote-ref-14)
15. *Gottschalk v. Benson*, 409 U.S. 63, 67 (1972) (holding a method for converting binary-coded-decimals to binary unpatentable) [↑](#footnote-ref-15)
16. *Mayo Collaborative Servs. v. Prometheus Labs., Inc.*, 566 U.S. 66, 70 (2012) (restating the long-held exceptions 35 U.S.C. § 101: laws of nature, natural phenomena, and abstract ideas) [↑](#footnote-ref-16)
17. Diamond v. Chakrabarty, 447 U.S. 303, 309 (XXXX) [↑](#footnote-ref-17)
18. Diamond v. Chakrabarty, 447 U.S. 303, 309 (XXXX) [↑](#footnote-ref-18)
19. *Funk Bros. Seed Co. v. Kalo Inoculant Co*., 333 U.S. 127, 131 (1948) [↑](#footnote-ref-19)
20. *Funk Bros. Seed Co. v. Kalo Inoculant Co*., 333 U.S. 127, 131 (1948) [↑](#footnote-ref-20)
21. *Funk Bros. Seed Co. v. Kalo Inoculant Co*., 333 U.S. 127, 131 (1948) [↑](#footnote-ref-21)
22. *Diamond v. Chakrabarty*, 447 U.S. 303, 304 (1980) [↑](#footnote-ref-22)
23. 35 U.S.C. § 101 [↑](#footnote-ref-23)
24. Randall Rader, Benjamin Christoff, Patent Law in a Nutshell 56 (3rd ed. 2018). [↑](#footnote-ref-24)
25. *See e.g.* [Alice, Mayo, Berkheimer, etc.]; *see* Kathleen Chapman, Esq. & Stephen Ball, Esq*., Challenges with Patenting Software,* Vt. B.J., Winter 2007/2008, at 36 [↑](#footnote-ref-25)
26. *See Diamond v. Dieher;* [↑](#footnote-ref-26)
27. In re Bilski [↑](#footnote-ref-27)
28. Diehr [↑](#footnote-ref-28)
29. Mayo, Quoting Benson [↑](#footnote-ref-29)
30. Mayo, quoting Diehr [↑](#footnote-ref-30)
31. Mayo, Quoting Parker v. Flook [↑](#footnote-ref-31)
32. *Gottschalk v. Benson* [↑](#footnote-ref-32)
33. *Gottschalk v. Benson* [↑](#footnote-ref-33)
34. *Gottschalk v. Benson* [↑](#footnote-ref-34)
35. *Diamond v. Diehr* [↑](#footnote-ref-35)
36. *Diamond v. Diehr* [↑](#footnote-ref-36)
37. *Diamond v. Diehr,* 450 U.S. 175, 184 (1981) (noting several times that the respondents claims involve transforming of raw rubber into a different state). [↑](#footnote-ref-37)
38. *Bilski v. Kappos*, 561 U.S. 593, 603 (2010). [↑](#footnote-ref-38)
39. *Gottschalk v. Benson*, 409 U.S. 63, 70 (1972); *In re Bilski*, 545 F.3d 943, 955 (Fed. Cir. 2008). [↑](#footnote-ref-39)
40. *In re Bilski*, 545 F.3d 943, 961 (Fed. Cir. 2008). [↑](#footnote-ref-40)
41. *See* *Benson*, 409 U.S. at 70. [↑](#footnote-ref-41)
42. *In re Bilski*, 545 F.3d 943, 954 (Fed. Cir. 2008). [↑](#footnote-ref-42)
43. *Diehr*, 450 U.S. at 184, 187; *In re Bilski*, 545 F.3d 943, 954 (Fed. Cir. 2008). [↑](#footnote-ref-43)
44. *Parker v. Flook*, 437 U.S. 584, 585 (1978). [↑](#footnote-ref-44)
45. *Benson*, 409 U.S. at 70. [↑](#footnote-ref-45)
46. *In re Bilski*, 545 F.3d 943, 956 (Fed. Cir. 2008), aff'd but criticized sub nom. Bilski v. Kappos, 561 U.S. 593, (2010) [↑](#footnote-ref-46)
47. Bilski v. Kappos, 561 U.S. 593, 605 (2010). [↑](#footnote-ref-47)
48. Robert Daniel Garza, Software Patents and Pretrial Dismissal Based on Ineligibility, 24 Rich. J.L. & Tech. 1, 41, 87 (2018); See *Amdocs (Isr.) Ltd. v. Openet Telecomm. Inc.*, 841 F.3d 1288, 1293-94 (Fed. Cir. 2016). [↑](#footnote-ref-48)
49. 35 U.S.C. § 101 [↑](#footnote-ref-49)
50. CITATIONS NEEDED [↑](#footnote-ref-50)
51. Paul R. Gugliuzza, Quick Decisions in Patent Cases, 106 Geo. L.J. 619, 622 (2018) [↑](#footnote-ref-51)
52. Kathleen Chapman, Esq. & Stephen Ball, Esq., Challenges with Patenting Software, Vt. B.J., Winter 2007/2008, at 36, 37 [↑](#footnote-ref-52)
53. *Diamond v. Diehr*; *Gottschalk v. Benson* [↑](#footnote-ref-53)
54. Intellectual Property Channeling for Digital Works; Preserving The Value Of Medical Device Patents During The Rise Of Three-Dimensional Printing – Westlaw Journal IP 2013 WL 5808127, at \*4 [↑](#footnote-ref-54)
55. *In re Beauregard*, 53 F.3d 1583, 1584 (Fed. Cir. 1995) [↑](#footnote-ref-55)
56. *See e.g.* *CyberSource Corp. v. Retail Decisions, Inc.*, 654 F.3d 1366, 1373 (Fed. Cir. 2011); *SEVEN Networks, LLC v. Google LLC*, No. 2:17-CV-441-JRG, 2018 WL 5263271, at \*30 (E.D. Tex. Oct. 23, 2018) [↑](#footnote-ref-56)
57. *Alice Corp. Pty. Ltd. v. CLS Bank Intern* [↑](#footnote-ref-57)
58. *Alice Corp. Pty. Ltd. v. CLS Bank Intern* [↑](#footnote-ref-58)
59. *See* Jasper L. Tran, Software Patents: A One-Year Review of Alice v. CLS Bank, 97 J. PAT. & TRADEMARK OFF. SOC'Y, 532, 539-540 (2015) [↑](#footnote-ref-59)
60. Robert Daniel Garza, Software Patents and Pretrial Dismissal Based on Ineligibility, 24 Rich. J.L. & Tech. 1, 28 (2018); *See* Versata Dev. Grp., Inc. v. SAP America, 793 F.3d 1306 (Fed. Cir. 2015); Intellectual Ventures I LLC v. Capital One Bank (USA), 792 F.3d 1363, 1367-68 (Fed. Cir. 2015); Internet Patents Corp. v. Active Network, Inc., 790 F.3d 1343, 1345 (Fed. Cir. 2015) OIP Techs., Inc. v. Amazon.com, Inc., 788 F.3d 1359, 1362-63 (Fed. Cir. 2012); Content Extraction & Transmission LLC v. Wells Fargo Bank, Nat'l Ass'n, 776 F.3d 1343, 1346-47 (Fed. Cir. 2014); DDR Holdings, LLC v. Hotels.com, L.P., 773 F.3d at 1245, 1256; Ultramercial, Inc. v. Hulu, LLC, 772 F.3d 709, 721-22 (Fed. Cir. 2014); buySAFE, Inc. v. Google, Inc., 765 F.3d 1350, 1351 (Fed. Cir. 2014); Planet Bingo, LLC v. VKGS LLC, 576 F. Appx. 1005, 1006 (Fed. Cir. 2014); Digitech Image Techs. v. Elecs. for Imaging, 758 F.3d 1344, 1348-51 (Fed. Cir. 2014). [↑](#footnote-ref-60)
61. *See DDR Holdings*, LLC, 773 F.3d at 1245; Part IV (A) - A. DDR Holdings: Rooted in Computer Technology - The First Victory. [↑](#footnote-ref-61)
62. Berkheimer v. HP, p. 1360 [↑](#footnote-ref-62)
63. B v. HP, 1362-1363 [↑](#footnote-ref-63)
64. B v. HP, 1364 [↑](#footnote-ref-64)
65. B v. HP, 1360 [↑](#footnote-ref-65)