

Fabian Patents

INTRODUCTION

Intellectual property law has been interacting with software-related inventions in one way or another for over 60 years. Despite the number of judicial decisions, legislative interventions, public inquiries, policy reports, articles, and books that have been devoted to the subject over this time, there are many unanswered questions concerning intellectual property law and its relationship to software-related subject matter. The confusion and uncertainty that characterises this area of law is particularly evident in patent law. As Dennis Crouch wrote in 2012, it ‘is simply ridiculous that after 40 years of debate, we still do not have an answer to the simple question of whether (or when) software is patentable’.¹ The uncertainty about whether or not software is patentable subject matter was compounded by the 2014 decision of *Alice v. CLS Bank* where the US Supreme Court was asked, again, whether software was patent eligible. The uncertainty created by the *Alice* decision was captured in Robert Merges’ comment that to ‘say we did not get an answer’ from the Supreme Court to the question of whether software was patentable ‘is to miss the depth of the non-answer we did get’.² As a 2022 Patent Office report on subject matter eligibility shows, the situation since then has only got worse.³

While a number of explanations have been given for this confusion, three stand out. The first suggests that the confusion arises because of the peculiar nature of software. More specifically, the confusion is said to arise because as software is neither art nor science but a hybrid thereof, it does not fit neatly into intellectual property law, which distinguishes between artistic creative outputs (copyright) and

¹ Dennis Crouch, ‘Ongoing Debate: Is Software Patentable?’ (27 July 2012) *Patently-O*.

² Robert Merges, ‘Symposium: Go Ask Alice – What Can You Patent after *Alice v. CLS Bank*?’, SCOTUSblog (20 June 2014). www.scotusblog.com/2014/06/symposium-go-ask-alice-what-can-you-patent-after-alice-v-cls-bank/

³ USPTO, ‘Patent Eligible Subject Matter: Public Views on the Current Jurisprudence in the United States. A Report to Congress’ (June 2022).

techno-scientific creations (patents).⁴ At the same time, it is also suggested that while intellectual outputs have typically been protected by one form of intellectual property, this is not the case with software, which is afforded both copyright and patent protection, 'making it a unique phenomenon in the law of intellectual property'.⁵

A second explanation attributes the confusion and uncertainty to the ephemeral, non-physical nature of software, to its intangibility. While the incorporeal nature of intellectual property has long created problems for the law, there is thought to be something particularly disturbing about 'the unphysical nature of computer programming'⁶ that makes it 'very different from any property we have ever known'.⁷ In particular it has been suggested that the confusion associated with software arises because it 'is neither tangible or intangible, but something else'.⁸ This is because software 'has both tangible or intangible aspects. Indeed, it seems to have a chameleon nature, undergoing a transition from a tangible to an intangible and back to a tangible object depending upon how it is used or how it is being viewed'.⁹

A third explanation attributes the uncertainty to the law's inability to keep up with the speed of change associated with information technology and of the inevitable gap that this creates between the law and the technology it is meant to regulate.¹⁰ In this sense, it is seen as yet another example of the dilemma that is created when the 'law does not keep pace with the advance of science and industry'¹¹ and of the problems that arise when the law attempts to make sense of complex new technologies.¹²

While these factors are important, the primary reason why patent law's relationship with software-related subject matter has been so fraught is because of the way the subject matter has been construed. Martin Goetz, from Applied Data Research, summed up these problems when in speaking about information technology in the 1960s and

⁴ See Robert W. Wild, 'Computer Program Protection: The Need to Legislate a Solution' (1969) 54(4) *Cornell Law Review* 586, 589: programs were 'part science, part art'. In the 1970s IBM proposed the introduction of a hybrid registration system (copyright and patents) for software. See Elmer Galbi, 'Proposal for New Legislation to Protect Computer Programming' (1970) 17 *Bulletin of Copyright Society* 280. (Galbi was senior patent attorney for IBM).

⁵ Kenneth Nichols, *Inventing Software: The Rise of Computer-Related Patents* (Westport, CT: Quorum Books, 1998), 3.

⁶ Harold L. Davis, 'Computer Programs and Subject Matter Patentability' (1977-78) *Rutgers Journal of Computers and Law* 1, 22. Martin Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry* (Cambridge, MA: MIT Press, 2004), 3.

⁷ Milton R. Wessel, 'Some Implications of the Software Decision' (1973) *Jurimetrics* 110, 111.

⁸ Duncan M. Davidson, 'Common Law, Uncommon Software' (1986) *University of Pittsburgh Law Review* 1037, 1065.

⁹ *Ibid.*, 1064.

¹⁰ Michael A. Duggan, 'Patents and Programs: The ACM's Position' (1971) 14(4) *Communications of the ACM*, 278. Harold L. Davis, 'Computer Programs and Subject Matter Patentability' (1977-78) *Rutgers Journal of Computers and Law* 1, 4 n 14.

¹¹ Philip Stork, 'Legal Protection for Computer Programs: A Practicing Attorney's Approach' (1970) 20 *Copyright Law Symposium* 112, 138.

¹² For a critical account of this way of thinking about law and technology see Allison Fish, *Laying Claim to Yoga* (New York: Cambridge University Press, forthcoming).

1970s he said: ‘It was a very unclear era. There were questions of whether software was tangible or intangible and what was software. Of course IBM was giving it all away for free, and then suddenly they’re selling it. What were they selling and how do you protect it. There was a question of: is software taxable, is it tangible? There was a great deal of confusion all wrapped up in the intellectual property issues.’¹³

While contemporary accounts of patentable subject matter tend to focus on excluded subject matter (laws of nature, natural phenomena, and abstract ideas), what is clear from patent law’s engagement with software is that while these categories of non-patentable subject matter played a role, this was nowhere near as important as the way that the subject matter was construed. In the same way in which the fate of gene patents in the early part of the twenty-first century turned on whether the isolated genes were characterised in chemical or genetic terms, so too the fate of software-related subject matter across the second half of the twentieth century turned on how it was characterised. The problem for patent law at the time, and a key reason for the ongoing confusion about patent law’s relationship to software, was that it was unable to find a suitable way of answering this question.

While the early discussions were framed in terms of the question – *is software patentable?* – the flexibility inherent in the term ‘software’ masked the fact that strictly speaking the debates were not about the patenting of software as such. Rather, what was at stake in these debates was the preliminary question: *what is the subject matter?* That is, the debates were not about how the class of subject matter should be characterized, so much as about what the class of subject matter was or should be.¹⁴ As Leo Keet, former President of the software products group at Dun & Bradstreet said, ‘during the early years of the software industry, we debated a seemingly simple question: What is software? The answer, once we could agree, would help determine our approach to intellectual property, taxation, contracting, and public policy issues’.¹⁵

As we have seen, the primary reason why patent law’s relationship with software has been so troubled was because the computer industry could not agree on what

¹³ ADAPSO Reunion Workshop, ‘Intellectual Property’ Computer History Museum, CHM Ref No. X4589.2008 (Recorded 4 May 2002), 23. For example, it was unclear whether software was a ‘good’ which fell under Article 2 of the Uniform Commercial Code, or whether it was some sort of ‘service’ that fell outside the scope of the code. The industry was said to have been whipsawed by the government on the nature and taxability of software. On the one hand, ‘the Federal Government took the position that software was intangible and, therefore, did not qualify for things like accelerated depreciation the investment tax credit and other favourable federal tax treatment’. On the other hand, however, the States took the view that ‘software was tangible and, therefore, its transfer or sale was subject to sales and use taxation.’ Ron Palenskim, ADAPSO Reunion Workshop, ‘Intellectual Property’, Computer History Museum, CHM Ref No. X4589.2008 (Recorded 4 May 2002), 15.

¹⁴ As a software industry representative reflecting on the 1960s said, ‘during the early years of the software industry, we debated a seemingly simple question: What is software? The answer, once we could agree, would help determine our approach to intellectual property, taxation, contracting, and public policy issues’. Ernest E. Keet, ‘A Personal Recollection of Software’s Early Days (1960–1979): Part 2’ (October–December 2005) *IEEE Annals of the History of Computing* 31.

¹⁵ *Ibid.*

the subject matter was. This was important because patent law usually relies upon science and technology to accommodate new types of subject matter. As a Patent Office examiner wrote in 1969, what was needed to accommodate the new subject matter was a ‘concerted, unemotional effort by the software industry to define its terminology and specific desires’.¹⁶ While in most situations, technical and scientific communities have provided the law with the tools to understand and define the subject matter being considered, this was not the case with software-related subject matter.¹⁷ Indeed, rather than providing an answer to the question of what the subject matter was or the means to allow that subject matter to be assimilated in the law, the industry sought to resolve its own disputes *through* the law. Unlike the case with organic chemicals and biological inventions, the inherently divided nature of the nascent information technology industry meant that the law was forced to develop its own way of dealing with the would-be subject matter. And while there was no particular reason why the legal response to this question should have been so confused, it was and remains so.

One of the challenges that patent law faced when confronted with software-related subject matter in the 1960s and 1970s was that it was not in a position to evaluate or judge the novelty and obviousness of patent applications. A key reason for this was that patent law ‘had no history to look to’.¹⁸ As the US President’s 1966 Commission on the Patent System Inquiry found:

The Patent Office now cannot examine applications for programs because of the lack of a classification technique and the requisite search files. Even if these were available, reliable searches would not be feasible or economic because of the tremendous volume of prior art being generated. Without this search, the patenting of programs would be tantamount to mere registration and the presumption of validity would be all but nonexistent.¹⁹

In response to this problem, members of the patent profession joined with information technical experts to tame the unruly and disorganized public domain. Notably, the National Bureau of Standards, the American Patent Law Association, and the Association of Computer Machinery Patent Committee joined forces in the late 1960s to classify computer software.²⁰ While this initiative was relatively short-lived,

¹⁶ T. Buckman, ‘Protection of Proprietary Interests in Computer Programs’ (1969) *Journal of the Patent Office Society* 135, 151.

¹⁷ Software was ‘not yet a science, but an art that lacks standards, definitions, agreements on theories and approaches.’ Gene Bylinsky, ‘Help Wanted: 50,000 Programmers’ (March 1967) *Fortune* 141.

¹⁸ Leo Keet, ADAPSO Reunion Workshop, ‘Intellectual Property’, Computer History Museum, CHM Ref No. X4589.2008 (Recorded 4 May 2002), 9.

¹⁹ *Report of the President’s Commission on the Patent System* (Washington, DC, 1966), 13.

²⁰ Michael Duggan, ‘Patents and Programs: The ACM’s Position’ (1971) 14(4) *Communications of the ACM* 278, 279. The Patent Office was said to be ‘enthusiastic’ about the work of the Committee. Letter from Gunter A. Haupton (IBM), (Chair) to members of the PLA Subcommittee on the Classification of Computer Programs (24 October 1969).

by 1970 approximately 700 subject areas had been established and defined for classifying an estimated 20,000 prior art publications relating to software.²¹ As with the attempt to develop a test to allow would-be subject matter to be evaluated to determine if it was patent eligible, the attempt to classify computer-related prior art was hampered by the uncertainty as to what was meant by software.²²

Another challenge the law faced in dealing with computer-related subject matter was working out when the subject matter was patentable. With Congress unable or unwilling to assist, there was (and remains) a hope and expectation that the Supreme Court would intervene to resolve this seemingly intractable problem.²³ As commentators have noted, however, the Supreme Court's pronouncements on software-related subject matter have created more problems than they have solved.²⁴ One reason for this is that despite repeated calls for the Supreme Court 'to rule on the broad question of whether "machine processes" that utilize a general purpose computer for their implementation constitute patentable subject matter', the Supreme Court has consistently refused to provide an answer.²⁵ Instead, the Court has tended to limit its findings to the specific facts of the case at hand, leaving it to others to fight over what the decisions meant for software patentability more generally. As Justice Rich said in criticising the approach of the Supreme Court to software patents, this was 'like taking the problem of school segregation to court on a case-by-case basis, one school at a time'.²⁶ And when the Supreme Court eventually did attempt to make a more general ruling (in *Mayo* and *Alice*), it merely restated the problem as a two-step process.

One of the factors that shaped the way the Supreme Court approached software-related subject matter was that it felt uncomfortable dealing with what the Court of

²¹ G. Knight Jr, *Hierarchical Descriptor Classification System for Documents Related to Computer Software: With Scope Notes* (1970) (prepared for the Administrator, Office of Systems and Search Documentation, US Patent Office). This was said to be 10% of prior art documents.

²² The attempt by the joint study by the Patent Office, National Bureau of Standards, and the ACM to classify extant computer literature dealing with programs was said to be 'intractable'. Michael A. Duggan, 'Patents on Programs? The Supreme Court Says No' (1973) *Jurimetrics* 135, 136.

²³ The Supreme Court in *Benson* concluded that the problems relating to software patentability could only be solved by Congress. *Gottschalk v. Benson* 409 U.S. 63, 73 (1972).

²⁴ See John F. Duffy, 'Rules and Standards on the Forefront of Patentability' (2009) 51 *William and Mary Law Review* 609; Dan Burk and Mark Lemley, *The Patent Crisis and How the Courts Can Solve It* (Chicago: Chicago University Press, 2009), 157. At times this led to calls for patent matters to be taken away from the jurisdiction of the Supreme Court. A questionnaire sent out to the members of the APLA Committee on Computer Program Protection to work out the impact of the *Flook* decision asked whether 'Issues of patentability of inventions under the statute should be removed from the jurisdiction of the US Supreme Court'. 'Questionnaire re Impact of *Flook*', sent out by Reed C. Lawlor (Attorney) to members of the APLA Committee on Computer Program Protection (11 August 1978). Charles Babbage Institute, Applied Data Research, Software Products Division records, CBI 154, File: Box 15, folder 7.

²⁵ Martin A. Goetz, 'The *Flook* Patent Opinion Signals that Inventive Software Processes Are Patentable Subject Matter' (n.d.) Charles Babbage Institute, Applied Data Research, Software Products Division records, CBI 154, Box 15, folder 7.

²⁶ *In the Matter of the Application of Glen F. Chatfield* 545 F.2d 152, 162 (CCPA 1976).

Customs and Patent Appeals in *Prater II* described as ‘one of the most technical-legal matters ever appealed to this court’²⁷ (which led to calls, which have been repeated recently, that technological matters should be removed from the jurisdiction of the Supreme Court).²⁸ While it had been hoped that when the Supreme Court was asked in *Benson* to consider the patentability of a general-purpose digital computer programmed with an algorithm that converted binary coded decimals to pure binary numbers that the Court would have provided clarity about how software-related subject matter should be interpreted, this was not to be. While the parties recognized that the outcome of the decision turned on how the technology was construed, the Supreme Court felt that it was ‘not competent to resolve’ ... ‘the vast technological questions’ that had been raised in the fourteen amici curiae briefs.²⁹ That is, the Court felt it was not in a position to decide either what the software-related subject matter was or how it should be interpreted.³⁰

While the Supreme Court may not have offered much assistance in determining when computer-related subject matter might qualify as patentable subject matter, it has played an important role in framing the way this question was asked. The first way it did this was in terms of the way composite inventions should be approached, something that was particularly important with machine-based subject matter. While often overlooked, this is perhaps the most important and enduring contribution made by the Supreme Court to subject matter eligibility.

When the courts first began to consider software-related subject matter, there were two competing ways of approaching inventions that were made up of parts or elements. One approach, often confusingly called the ‘point of novelty test’, requires composite inventions to be separated into parts. Specifically, it requires courts to exorcise and then ignore those parts of the claimed invention that either lack novelty or are deemed to be excluded subject matter (such as a computer program). Motivated by a desire ‘to discourage clever attorneys from using their skill to hide software claims among a sea of irrelevant non-novel limitations’,³¹ the courts

²⁷ *Application of Prater and Wei* 415 F.2d 1378, 1390 (CCPA 1969). Anon, ‘Computer Patent Backed by Court’ (23 November 1968) *The New York Times* 71

²⁸ In its brief amicus curiae in *Diamond v. Diehr*, Applied Data Research argued that the writ should be dismissed because the Supreme Court was not equipped to resolve what the Commissioner of Patents had presented as the key issues in the case, which would require the court to ‘undertake a thorough inquiry into the complex technological facts of the construction of computerized machines’. The problem was that none of the eight computer program cases ... has contained a factual record of the nature of this technology and the ‘Supreme Court was not the appropriated forum for initial fact finding’. ‘Brief Amicus Curiae for Applied Data Research and Whitlow Computer System’ in *Diamond v. Bradley and Diamond v. Diehr* Nos 79–855 and 79–112, 7.

²⁹ Harold L. Davis, ‘Computer Programs and Subject Matter Patentability’ (1977–78) 6 *Rutgers Journal of Computers and Law* 1, 13–14 n 13.

³⁰ *Gottschalk v. Benson* 409 U.S. 63, 73 (1972). The court felt that the technological problems raised in the briefs could only be answered by committees of Congress: which was not forthcoming.

³¹ Keith E. Witek, ‘Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents’ (1996) *Berkeley Technology Law Journal* 363, 375.

were then expected to determine whether what was left of the invention fell within one of the classes of statutory subject matter. When applied to computer-related subject matter, the point of novelty approach meant that the court would ignore the computer program, mathematical method, algorithm, etc. and only consider the parts that remained (the computer).³² Given that this would have excluded many computer-related inventions, it is not surprising that the point of novelty approach was supported by hardware manufacturers. The second more straightforward test, which was sometimes known as the ‘whole contents approach’, requires the courts to evaluate the invention as a composite entity without breaking it down into parts. That is, the courts were expected to consider whether the invention as a whole was statutory subject matter.

While the Supreme Court briefly flirted with the point of novelty test in *Parker v. Flook*, it changed course in *Diamond v. Diehr* and came out in favour of the whole contents approach: a position which it has consistently adhered to subsequently.³³ As the court said in *Diehr*, a claim was not unpatentable merely because it included a step(s) or element(s) directed to a law of nature, mathematical algorithm, formula, or computer program so long as ‘the claim as a whole is drawn to subject matter otherwise statutory’.³⁴ In doing so, the Supreme Court reinstated the long-held view that the ‘practice of dissecting a machine and rejecting it piecemeal is without sanction of either reason or law’.³⁵

A second change instigated by the Supreme Court that helped to frame the way computer-related subject matter was evaluated concerned the way the excluded subject matter was categorised. While contemporary accounts of patentable subject matter tend to treat laws of nature, natural phenomena, and abstract idea as timeless, ahistorical categories, they have a much more recent history. Until the 1980s or thereabouts, the language used to describe excluded subject matter was fluid,

³² Although the appellant in *Noll* had couched his invention as an apparatus claim and argued that the invention should be scrutinised as a whole, this was rejected by the Patent and Trademark Office Board of Appeals. The reason for this was that the applicant perceived his invention to lie in the computer program. Paraphrasing Gertrude Stein, they added ‘a program is a program is a program’ and to have allowed protection would have allowed protection over programs per se. *In re Noll* (18 November 1976) as cited in *In re Noll* 545 F2d 141, 148 (CCPA 1976) who rejected the approach by Board of Appeals holding that it was necessary to focus on the claimed subject matter as a whole.

³³ *Diamond v. Diehr* 450 U.S. 175, 188 (1981) (‘It is inappropriate to dissect the claims into old and new elements and then to ignore the presence of the old elements in the [35 U.S.C. § 101] analysis’).

³⁴ George H. Knight, *Patent-Office Manual: Including the Law and Practice of Cases in the United States Patent Office and the Courts Holding a Revisory Relation Thereto, also, an Appendix of Copyright Decisions* (Boston: Little Brown, 1894), 135.

³⁵ Anon, ‘Timely Hints for Patent Office Examiners’ (25 May 1872) 26(22) *Scientific American*, 353. ‘A machine may be either a single organism or a combination or organisms so related to each other as to co-operate, successively or simultaneously, in the production of the required result. When it is composed of parts, none of which without all the others constitute a machine, or when certain of its parts form a complete machine but the other portions, whether taken singly or together, are incapable of organic action the machine is a single organism.’ William C. Robinson, *The Law of Patents for Useful Inventions: Vol 1* (Boston: Little Brown, 1890), 262.

inconsistent, and changing. This was particularly the case with computer-related inventions, where a number of different overlapping terms were used to describe the excluded subject matter including software, computer programs, algorithms, mathematical formula, mental methods, and a range of variations thereof.

Over the course of the 1980s, the way excluded subject matter was categorised began to change. We can get a sense of some of the reasons for and the nature of these changes from the letter that the patent attorney, Reed C. Lawlor, sent to the American Patent Law Association's Committee on Computer Program Protection in 1978 complaining about the impact of the Supreme Court decision of *Parker v. Flook*. As Reed said, the 'Flook case arose because the patent profession as a whole has neglected the computer program allegedly "because it involves special interests"'. To remedy this, Reed said it was 'time ... to re-examine the fundamental principles of patent law concerning scientific principles, laws of nature, and mathematical formulas and algorithms, remembering that computer programming as merely one example, so that we can avoid another Flook'.³⁶ While the process may have been unscripted, sentiments such as these, combined with a string of decisions dealing with subject matter eligibility and a consequential growing academic interest in subject matter, had an impact on the way excluded subject matter was categorised.

Motivated by the legal impulse to codify, there were various attempts across the 1980s to synthesise the unwieldy and inconsistent list of subject matter that had been excluded by the courts over the last 150 or so years into a smaller number of more coherent categories. While there was some success, many issues were left unsettled. This was particularly the case with computer-related subject matter. In the early 1980s, there were many in the patent community who believed, for example, that in addition to the (now familiar) categories of 'laws of nature, natural phenomena, and abstract ideas' that *Benson*, *Flook*, and *Diehr* had created a fourth category of unpatentable subject matter, namely a general mathematical-algorithm exception.³⁷ This argument was considered and rejected by the Court of Appeals in *Alappat* where the court said, a 'close analysis of *Diehr*, *Flook*, and *Benson* reveals that the Supreme Court never intended to create an overly broad, fourth category of subject matter excluded from § 101'.³⁸ As the Court of Appeals explained, the reason for this was that 'at the core of the [Supreme] Court's analysis in each of these cases lies an attempt by the Court to explain a rather straightforward concept, namely, that certain types of mathematical subject matter, standing alone, represent nothing more than *abstract ideas* until reduced to some type of practical application, and thus that subject matter is not, in and of itself, entitled to patent protection'.³⁹

³⁶ Letter sent by Reed C. Lawlor (Attorney at Law) to the members of the APLA Committee on Computer Programming Protection, 11 August 1978, 2.

³⁷ Charles A. Damschen, 'Patentable Subject Matter: Do the 2005 USPTO Interim Guidelines Intersect State Street at a Roundabout?' (2008) 93 *Iowa Law Review* 1889, 1901.

³⁸ *In re Alappat* 33 F.3d 1526, 1543 (Fed Cir 1994).

³⁹ *Ibid.*

As well as providing a useful review of the Supreme Court decisions in the 1980s dealing with computer-related subject matter, the *Alappat* decision also highlights some of the changes that occurred in the way excluded subject matter was categorised at the time. The most obvious was that the different types of excluded subject matter were now subsumed within three general categories: laws of nature, natural phenomena, and abstract ideas.⁴⁰ Importantly, as part of this process the excluded subject matter that had previously been associated with computer-related inventions – software, computer programs, algorithms, mathematical formula, and mental methods – were now subsumed within the newly anointed overarching category of excluded subject matter labelled ‘abstract ideas’. As a result, instead of asking whether a computer-related application was really for a computer program or a mathematical formula, the courts now asked whether it was for an abstract idea. This brought about a change in the way excluded subject matter was interrogated, from the situation previously where excluded subject matter was described in technical or quasi-technical terms⁴¹ to a situation where excluded subject matter was defined in terms of the thing that was presumed to unite the different types of excluded subject matter, namely, as the Court of Appeals said in *Alappat*, that they ‘represent nothing more than *abstract ideas*’. And while this was certainly not the first time when a pre-emption argument was made – this is the argument that protection should correspond to what was invented – pre-emption took on a new prominence at the time as a means of justifying the shift to the more general principal-based categories. As the Federal Circuit said in *In re Bilski*, the question ‘before us then is whether Applicants’ claim recites a fundamental principle and, if so, whether it would pre-empt substantially all uses of that fundamental principle if allowed’.⁴² Or as the Supreme Court said in *Alice*, ‘while pre-emption is not the test for determining patent-eligibility’ it is certainly the ‘concern that undergirds our § 101 jurisprudence’ dealing with subject matter eligibility.⁴³

The adoption of the technologically neutral ‘abstract ideas’ category brought about a number of subtle but important changes in the way excluded subject matter was thought about. As we saw earlier, during the 1960s and 1970s patent professionals were aware that when thinking about subject matter eligibility, it was important to decide what the technology was and how it was to be interpreted. With the shift to a principle-based mode of categorisation, subject matter eligibility was decoupled from its technological origins to be replaced by debates about the meaning of abstract ideas, a process which accelerated following the 2014 Supreme Court

⁴⁰ ‘The subject matter courts have found to be outside of, or exceptions to, the four statutory categories of invention is limited to abstract ideas, laws of nature and natural phenomena’. USPTO, *Interim Guidelines for Examination of Patent Applications for Patent Subject Matter Eligibility*, Official Gazette of the United States Patent Office Notices (22 November 2005), 6.

⁴¹ With the exception of mental steps.

⁴² *In re Bilski* 545 F.3d 943, 954 (Fed Cir 2008).

⁴³ *Alice Corp. v. CLS Bank International* 134 S. Ct. 2347, 2358 (2014).

decision of *Alice v. CLS Bank*. This not only further distanced patent law from the information technology industry (and with it the possibility that the industry would help the law to deal with computer-related subject matter), it also shifted attention away from the way the subject matter as technology was interpreted and, in turn, the role this played in deciding the fate of many types of subject matter. The shift away from a subject matter that was described technically to one based on more general criteria (abstract ideas) also undermined the role computer programs played as boundary objects in patent law. While the computer program continued to operate as a boundary object in copyright law and in patent law in other countries, patent law in the United States moved in a different direction.

The decision to subsume the excluded subject matter associated with computer-related inventions within ‘abstract ideas’ also had an impact on the way computer-related subject matter was evaluated. While many issues were unsettled in the 1960s and 1970s, when thinking about subject matter eligibility patent law tended to focus on whether the (unpatentable) two-dimensional computer program had been transformed into novel three-dimensional machine. As Morton Jacobs said at the time, the key issue for patentability was whether a ‘machine invention has been made, or merely a discovery in mathematics, a mental process or the like’.⁴⁴ There were two notable features of this short-lived approach. The first was that it tended to see subject matter through a technical lens. The second was that the fate of computer-related subject matter depended on an applicant being able to show that they had brought about a change of kind, created a new kind of thing, or as the Commissioner of Patents said in 1966 transformed a general-purpose computer into a new type of specific-purpose machine.⁴⁵

The decision to subsume excluded subject matter within the rubric of ‘abstract ideas’ changed the mode of questioning that was used to interrogate computer-related subject matter. At the heart of the new approach that took shape in the 1980s was the simple idea that a claim drawn to a fundamental principle such as an abstract idea was unpatentable because it risked ‘disproportionately tying up the use of the underlying ideas’.⁴⁶ The problem with this however was, as the Supreme Court recognised, that because ‘all inventions at some level embody, use, reflect, rest upon, or apply laws of nature, natural phenomena, or abstract ideas’,⁴⁷ to exclude an invention simply because it touched on an abstract idea would have run the risk that it would ‘eviscerate patent law’.⁴⁸ To ensure that this did not happen, the Supreme Court was forced to qualify the idea that an abstract idea was unpatentable because

⁴⁴ Morton C. Jacobs, ‘Commissions Report (re: Computer Programs)’ (1967) *Journal of the Patent Office Society* 372, 374–75.

⁴⁵ Edward J. Brenner, ‘Guidelines to Examination of Programs’ (9 August 1966) Vol 829(2) *Official Gazette of the United States Patent Office* 441, 442.

⁴⁶ *Alice Corp. v. CLS Bank International* 134 S. Ct. 2347, 2354 (2014).

⁴⁷ *Mayo Collaborative Servs. v. Prometheus Labs.* 566 U.S. 66, 71 (2012).

⁴⁸ *Ibid.*, 71.

it risked disproportionately limiting use of the underlying ideas. As the Supreme Court said in *Alice* and *Mayo*, ‘an invention is not rendered ineligible for patent simply because it involves an abstract concept’.⁴⁹

To ensure that the subject matter exclusion did not eviscerate patent law, it was necessary to work out some way of distinguishing legitimate and illegitimate uses of abstract ideas within patents. Patent law’s response to this problem was to fall back onto the idea of invention as a transformative process to draw a distinction between applications that claimed the ‘building blocks’ of human ingenuity and those that integrated the building blocks into ‘something more’. While the former disproportionately tied up use of the underlying ideas and were therefore ineligible for patent protection, the latter posed no comparable risk of pre-emption and therefore remained eligible for patent protection.⁵⁰ As the Supreme Court said, while a claim drawn to an abstract idea was unpatentable, the *application* of abstract ideas ‘to a new and useful end’ remained eligible for patent protection.⁵¹

The structure of the questions used to interrogate computer-related subject matter in the 1960s and 1970s was similar to the questions asked from the 1980s: both distinguished between subject matter that was ineligible (whether computer programs, algorithms, etc. or abstract ideas) and inventions that applied or used that ineligible subject matter to create something new. Where they differed, however, was in the way ineligible and eligible subject matter were distinguished. As we saw earlier, for an applicant to satisfy the subject matter eligibility requirement in the 1960s and 1970s, they had to show that they had brought about a change of kind – that they had created a new kind of thing. With computer-related subject matter this meant that they had to convince the Patent Office and the courts that they had created a specific-purpose machine, rather than a mere computer program.

With the shift in the 1980s away from an excluded subject matter described technically to a principle-based excluded subject matter, this approach was no longer possible. The reason for this was that once it was accepted that all computer-based inventions embody, use, reflect, rest upon, or apply abstract ideas (or that all patents pre-empt to some degree), this meant that the existence of an abstract idea in an application could not be used as a litmus test for deciding eligibility (without running the risk of eviscerating patent law). Unlike the situation previously, where machine-like status signalled patent eligible subject matter, there was no obvious end (or kind) that could be used to distinguish a legitimate (patentable) use of an

⁴⁹ *Alice Corp. v. CLS Bank International* 134 S. Ct. 2347, 2354 (2014). *Mayo Collaborative Servs. v. Prometheus Labs.* 566 U.S. 66, 71 (2012). *Diamond v. Diehr* 450 U.S. 175, 187 (1981).

⁵⁰ *Mayo Collaborative Servs. v. Prometheus Labs.* 566 U.S. 66, 81 (2012).

⁵¹ *Gottschalk v. Benson* 409 U.S. 63, 67 (1972); *Diamond v. Diehr* 450 U.S. 175, 187 (1981); *Mayo Collaborative Servs. v. Prometheus Labs.* 566 U.S. 66, 72 (2012).

abstract idea from an illegitimate unpatentable use. As a result, patent law was forced to find a different way of evaluating computer-related subject matter.⁵²

Patent law's response to this task was shaped by the fact that while some degree of pre-emption or limitation on use was seen to be inevitable and thus permissible, too much was not. In light of this, instead of asking whether the subject matter was of the type that could and should be protected, patent law found itself in a situation where it had to decide what limitations on the use of an idea it was willing to accept, or how broadly the exclusionary principle should be applied. As a result, while subject matter eligibility for computer-related subject matter in the 1960s and 1970s had been a question of kind, it changed in the 1980s to become one of degree. The problem with this, however, as the courts repeatedly said, is that deciding where and how the law is to be drawn between legitimate and illegitimate use of an abstract idea is a challenging task.⁵³ As the court said in *Bilski*, the inquiry into 'whether Applicants' claim recites a fundamental principle and, if so, whether it would preempt substantially all uses of that fundamental principle if allowed' ... 'is hardly straightforward. How does one determine whether a given claim would pre-empt all uses of a fundamental principle?'⁵⁴ Rather than helping to resolve the question of how the eligibility of computer-related subject matter was to be decided, the approach developed by the courts in the 1980s only served to compound the problems patent law faced when dealing with computer-related subject matter. This is because instead of helping to determine subject matter eligibility, it merely added a new question and an extra layer of complexity to the subject matter inquiry: namely, *where and how was the line to be drawn between a (non-patentable) abstract idea and an application of an idea that produces eligible subject matter?*

The upshot of this was while the Supreme Court may have set out the parameters that framed the way questions about the eligibility of computer-related subject matter were asked, it failed to provide any real guidance about how this question was answered. As Justice Stevens said in his dissent in *Diamond v. Diehr*, the cases considering the patentability of program-related inventions had not established 'rules that enable a conscientious patent lawyer to determine with a fair degree of accuracy which, if any, program-related inventions will be patentable'.⁵⁵ Instead,

⁵² The underlying legal question that had to be decided was: 'what test or set of criteria governs the determination by the Patent and Trademark Office ... or courts as to whether a claim to a process is patentable under § 101 or, conversely, is drawn to unpatentable subject matter because it claims only a fundamental principle.' *In re Bilski* 545 F.3d 943, 952 (Fed. Cir. 2008).

⁵³ 'The subject matter courts have found to be outside of, or exceptions to, the four statutory categories of invention is limited to abstract ideas, laws of nature and natural phenomena. While this is easily stated, determining whether an applicant is seeking to patent an abstract idea, a law of nature or a natural phenomenon has proven to be challenging'. USPTO, *Interim Guidelines for Examination of Patent Applications for Patent Subject Matter Eligibility*, Official Gazette Notices (22 November 2005), 6.

⁵⁴ *In re Bilski* 545 F.3d 943, 954 (Fed. Cir. 2008).

⁵⁵ *Diamond v. Diehr* 450 U.S. 175, 219 (1981).

as often happens in patent law, this was left to others to do. As we will see, the response to the question of how patent law should deal with computer-related subject matter emerged out of an iterative process that moved between patentees, the Patent Office, patent examiners, and lower-level courts, both in response to each other, to technological innovations, and to pronouncements by the Supreme Court. While all of these factors played a role, ultimately it was the way that patentees and their attorneys drafted their patent applications that drove the way that patent law responded to computer-related subject matter.

FABIAN DRAFTING STRATEGIES

The techniques used by patentees to describe computer-related subject matter changed constantly over the twentieth century. As well as responding to changes in technology and drafting in order to future-proof claims, patentees also had to work with a Patent Office that was at best finding its feet in terms of how it dealt with computer-related subject matter or at worst ambivalent or hostile to their inventions. Patentees also had to navigate case law and Patent Office practice regarding software patenting that was ‘vague, largely form over function, constantly in flux and inconsistent’.⁵⁶ At the same time, patentees also had to deal with a judiciary that was inherently suspicious of them. In judging computer-related subject matter, the courts repeatedly warned that they needed to ensure that they were not being hoodwinked by patent attorneys who were using their nefarious drafting skills ‘to evade the recognized limitations on the type of subject matter eligible for patent protection’.⁵⁷ As the court said in *In re Noll*, it was important to recognise that ‘claims may be drafted in the *form* of one of the statutory classes but in *substance* be directed to non-statutory subject matter’.⁵⁸ Underpinning judicial warnings of this nature was a concern that patent attorneys were using their dark arts to obtain patent protection over computer programs. As the Supreme Court said in *Benson*: ‘Direct attempts to patent programs have been rejected on the ground of nonstatutory subject matter. Indirect attempts to obtain patents and avoid the rejection by drafting claims as a process, or a machine or components thereof programmed in a given manner, rather than a program itself, have confused the issue further and should not be permitted.’⁵⁹

⁵⁶ Keith E. Witek, ‘Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents’ (1996) *Berkeley Technology Law Journal* 363, 367.

⁵⁷ *Diamond v. Diehr* 450 U.S. 175, 191 (1981).

⁵⁸ *In re Noll* 545 F.2d 14 (CCPA 1976). The ‘current status of the law requires patent practitioners to be particularly artful in drafting software patent applications, to engage in limited legal fiction in certain instances, and to inform their clients of the uncertainty that still exists in this area of patent law’. Lawrence Kass, ‘Computer Software Patentability and the Role of Means-Plus-Function Format in Computer Software Claims’ (1995) 15(3) *Pace Law Review* 787, 791–92.

⁵⁹ *Gottschalk v. Benson* 409 US 63, 72 (1972).

While the courts and the Patent Office have periodically attempted to follow through on this threat to deny indirect protection to computer-related subject matter, applicants have consistently managed to find ways around the judicial hurdles that were imposed on them (which is reflected in the large number of computer-related inventions that have been patented since the 1980s).⁶⁰ A key reason for this was that in dealing with computer-related subject matter, the courts effectively backed themselves into a corner, which made it difficult for them to exclude indirect attempts to patent computer-related subject matter, a situation that patent attorneys skilfully exploited when drafting patents. As a result, the art of software patent drafting became ‘an exercise in form over function mastery, for which software clients would pay their attorneys dearly.’⁶¹

While patent attorneys adopted a number of different drafting strategies in order to get around the judicial objections to computer-related subject matter that had been raised, they tended to coalesce around a shared goal, namely to ‘disguise software innovations as hardware inventions by disclosing significant computer hardware details along with the software code within the patent specification.’⁶² As a patent attorney explained, ‘to fool the courts and the USPTO, practitioners needed to hand-craft and custom tailor the entire software patent application to look and feel like hardware’.⁶³

Patent attorney adopted a number of different techniques to ensure that their patents looked, smelt, and felt like hardware.⁶⁴ One strategy that was adopted in the 1960s and 1970s was to avoid mentioning anything about ‘algorithms’ or ‘software’ in a patent.⁶⁵ Using what the Patent Office solicitor called the Fabian strategy ‘of presenting the invention as though implemented by hardware programming not software’⁶⁶ patents were also drafted to ‘show the software as a hardware system both textually and graphically.’⁶⁷ One way this was done was to draft applications in such a way that the software code appeared as part of the structure of a computer. As a result, software patent applications typically ‘disclosed the computer hardware or electrical computer system which incorporated the software in a manner similar to

⁶⁰ For an overview of the ‘exponential growth’ in software patents from 1971 to 1994 see Keith E. Witek, ‘Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents’ (1996) *Berkeley Technology Law Journal* 363.

⁶¹ *Ibid.*, 375–76.

⁶² *Ibid.*, 371–72.

⁶³ *Ibid.*, 375

⁶⁴ *Ibid.*

⁶⁵ *Ibid.*, 376.

⁶⁶ ‘Actually [Applied Data Research’s Autoflow patent] was the first one that *candidly* presented as implemented by software programming. Prior to that, hardware companies had obtained such patents by the stratagem (called Fabian Strategy by the Patent Office Solicitor) of presenting the invention as though implemented by hardware programming not software’. See Brief Amicus Curiae for Applied Data Research, *Gottschalk v. Benson*, No. 71–485 (Oct Term 1971), 2 n 2.

⁶⁷ Keith E. Witek, ‘Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents’ (1996) *Berkeley Technology Law Journal* 363, 375–76.

a typical electrical system patent'.⁶⁸ To do this, a practitioner would divide the program into different code segments, function calls, procedures, etc. Once this was done the practitioner would then draft the claim 'to make these software routines appear as hardware'.⁶⁹ In some cases, practitioners would illustrate the software to include other hardware components such as a printer, modem, keyboard, mouse, display screen, disk drive, register, sensors, motors, controllers, machinery, assembly line, or some other tangible object in order to properly process information, manufacture items, receive input, provide output, or execute software code. Claiming these 'tangible structural items via a structure software claim format rendered the mysterious and intangible software subject matter statutory as an apparatus'.⁷⁰

The practice of drafting computer-related subject matter as hardware – whether as a machine, an apparatus, or a computer that included software as a component – was widely adopted by patentees at the time to enhance their chances of protection.⁷¹ As Martin Jacobs said, Applied Data Research avoided the objections that were raised about their Autoflow and sorting system patents by defining software as a machine device.⁷² As Martin Goetz wrote in his petition to the Patent Office to expediate the examination of Applied Data Research's application for an 'Automatic system for constructing and recording display charts' (which was a continuation in part of a 1965 application), the objections made to the initial application had been overcome by arguing that the application had 'disclosed a machine or apparatus'.⁷³ This meant that instead of claiming the algorithm that underpinned the invention, the patent claimed the material parts of the computer, the electronic components, and circuitry (see Figure 7.1).

Another technique used by patentees to ensure that computer-related subject matter met the subject matter requirements was to claim conventional computer technology with the software stored in the memory. Under this approach, a claim would 'recite the conventional and widely used structure of a computer which executes the novel software from memory locations'. As Witek said, the software patent practitioner of the 1970s illustrated and claimed the software with a central processing unit (CPU) to execute instructions; memory (either magnetic tape, a magnetic drum, magnetic disks, CDs, optical storage, RAM, ROM, EEPROM, EPROM,

⁶⁸ Ibid., 367.

⁶⁹ Ibid., 380.

⁷⁰ Ibid., 372.

⁷¹ 'In the applications that arrived at the Patent Office' in the mid 1960s, 'software became tangible, and in the texts of patents such as Autoflows, it became hardware.' Gerardo Con Diaz, 'Embodied Software: Patents and the History of Software Development, 1946–1970' (July–September 2015) *IEEE Annals of the History of Computing* 8, 16.

⁷² Brief Amicus Curiae for Applied Data Research, *Gottschalk v. Benson* No. 71–485 (Oct Terms 1971), 15, n 20.

⁷³ The patent office admitted as much when in responding to the suggestion that Goetz's 1968 Autoflow patent appeared to cover a computer program, the Office said the patent was not for a computer program: instead it 'involved a combination of equipment and program.' William D. Smith, 'Fighter for Computer-Program Patents' (29 December 1968) *The New York Times* 19.

Oct. 6, 1970

M. A. GOETZ
AUTOMATIC SYSTEM FOR CONSTRUCTING
AND RECORDING DISPLAY CHARTS

3,533,086

Filed Dec. 24, 1968

38 Sheets-Sheet 1

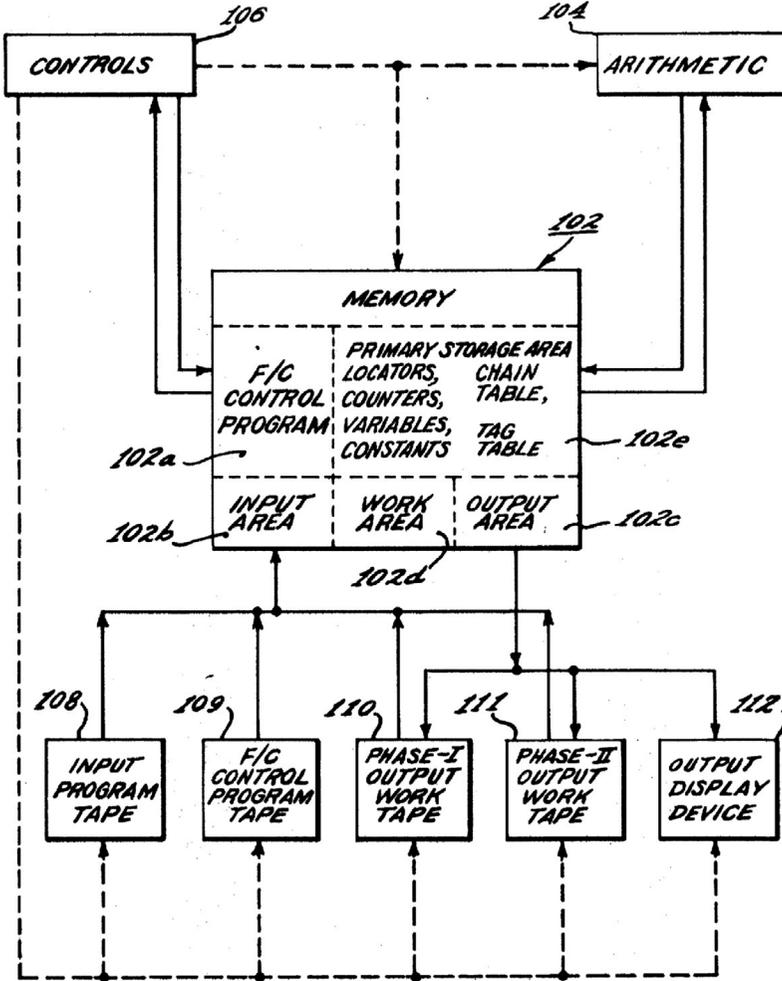


Fig. 1.

INVENTOR.
Martin A. Goetz
 BY
Millman and Jacobs
 ATTORNEYS.

FIGURE 7.1 Schematic block diagram of a data processing system in accordance with the invention
 Martin Goetz, 'Automatic System for Constructing and Recording Display Charts'
 US Patent No. 3,533,086 (6 Oct 1970). Courtesy of the United States Patent and Trademark Office.

flash memory, and/or like storage media) to store executable code and data. Where software was executed in a larger system containing more hardware than just central processing unit and memory, the software patent practitioner would claim the larger hardware system (see Figure 7.2). In these instances, the claim would 'recite a [central processing unit] or computer, memory, peripherals, and/or other computer or system technology, and then recite that the memory coupled in the computer system contains novel software that is executed within the computer system'.⁷⁴

Another technique used to claim computer-related subject matter was to draft applications using the so-called 'means-plus-function claim', which allowed patentees to claim combined elements as 'a means for performing a specified function'.⁷⁵ Historically, the US Patent Office had viewed the means-plus-function claim as a permutation of a process claim. Accordingly, examiners would state that the claim recited an 'algorithm' and reject the means-plus-function claim outright. Following a series of decisions in the 1970s, which upheld the patentability of means-plus-function claims directed to physical apparatus, inventors began using 'means-plus-function claims to link software to generic computer hardware, making the claims appear to recite structure or machine'.⁷⁶ This increased the ability for software practitioners to hide algorithms and programs in a structure-like claim format.

One of the reasons why hardware claims were important for software producers was because although applications were evaluated by the Patent Office on the basis that they were hardware inventions, the protection that these patents provided extended to include software. This was because of the longstanding rule that patent protection for machines not only covered the machine's precise form but also extended to cover other forms that embodied the invention. Specifically, protection extended beyond the specific way the machine was described to include 'equivalent' machines. Under the doctrine of equivalents, two devices were equivalent if they did 'the same work in substantially the same way and accomplish substantially the same result, even though they differ in form, scope, and proportion'.⁷⁷ The purpose of the doctrine was to protect the patent by preventing competitors from making simple changes in the patented machines – for example using a cam instead of a lever or rearranging the constituent mechanisms – and thereby securing separate patents. Thus, 'to copy the principle or mode of operation embodied in an apparatus is an infringement, even though the copy is different in form or proportion'.⁷⁸ This principle was also applied where someone replaced a machine containing special purpose hardware controls with a machine containing software that performed the same function.

⁷⁴ Keith E. Witek, 'Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents' (1996) *Berkeley Technology Law Journal* 363, 382.

⁷⁵ Max W. J. Graham, 'Process Patents for Computer Programs' (1968) 56(2) *California Law Review* 466, 477.

⁷⁶ Keith E. Witek, 'Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents' (1996) *Berkeley Technology Law Journal* 363, 390.

⁷⁷ Max W. J. Graham, 'Patents for Computer Programs' (1968) 56(2) *California Law Review* 466, 476.

⁷⁸ *Ibid.*

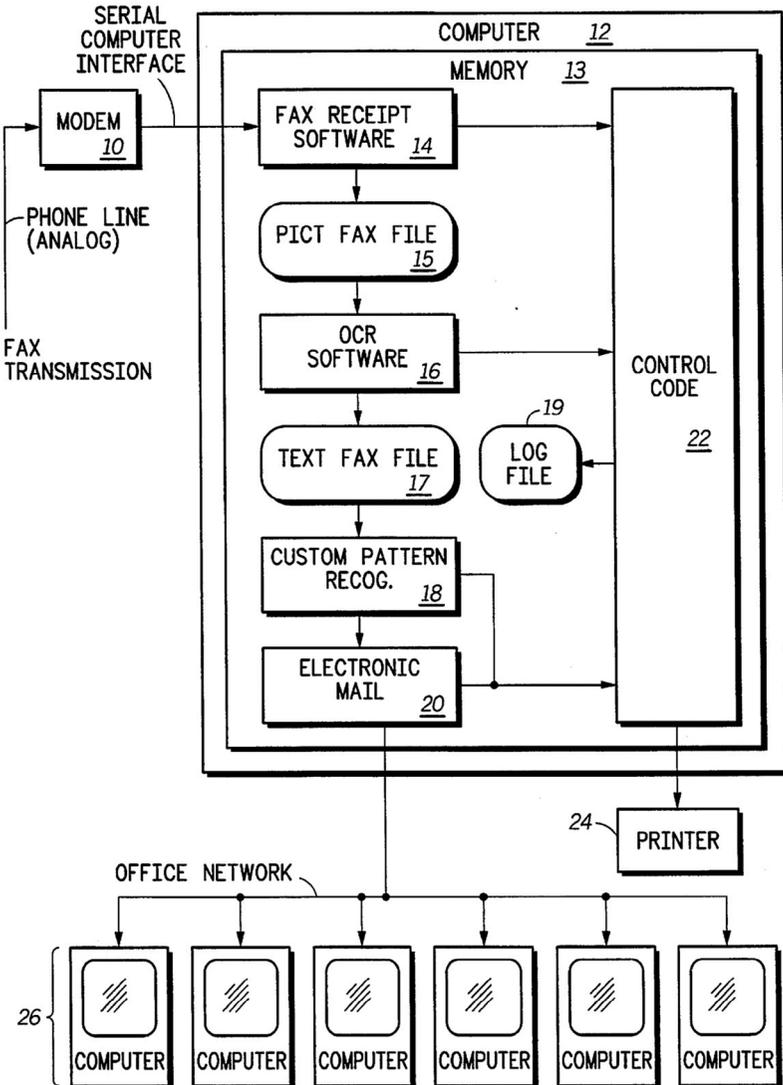


FIG. 1

FIGURE 7.2 Block diagram of fax data processing system in accordance with the invention
 Keith Witek, 'Computerised Facsimile (Fax) System and Method of Operation'
 US Patent No. 5,461,488 (24 Oct 1995). Courtesy of the United States Patent and
 Trademark Office.

We can get a sense of how the doctrine of equivalents operated to protect software-related subject matter from the advice given by a patent practitioner in 1968 about how to draft computer-related applications. In order to maximise the chances of registration, inventors were advised to design 'a fixed wire circuit that performs the same functions as would a computer operating according to this program'.⁷⁹ Once this was done, to avoid a patent application being rejected on the basis that it was for a computer program, inventors were advised to use hardware structural claims that described 'the operation both of the fixed wire system and the programmed computer'.⁸⁰ Importantly, to avoid being rejected on the basis that computer programs were not patentable, applicants were advised that the patent should only describe the fixed wire circuits. While the invention outlined in the patent would be limited to hardware (the fixed wire system), the doctrine of equivalents meant that patentees could 'argue that a computer, programmed to function in the same manner as his patented fixed wire circuit, is an equivalent device'. As a result, no one could 'use the program, which the patentee originally sought to protect, without infringing the patent on the fixed wire circuit. By using this scheme, then, the patentee is able to protect and monopolize the use of his computer program'.⁸¹ That is, a patentee would indirectly protect a computer program by drafting a hardware claim disclosed in terms of a fixed wire system.

The strategy of representing computer-related subject matter as hardware proved to be an effective way of circumventing the objections that had been raised about software-related subject matter.⁸² There were two reasons for this. The first was that by framing their inventions as hardware, software producers were able to connect their applications to the patents that had been granted since the 1940s for hardwired computers. More specifically, software producers relied upon the fact that to deny protection to claims with hardware limitations, the Patent Office and the courts would have set a precedent that would have invalidated 'every hardware/computer patent ever issued in U.S. history'.⁸³ If this happened, it 'would have invalidated tens of thousands of electrical systems, circuits, and like patents consistently issued by the USPTO for decades'.⁸⁴ As a commentator accurately predicted, it was 'unlikely that the USPTO or the [courts] would ever go that far'.⁸⁵ As a result, the courts and

⁷⁹ *Ibid.*, 477.

⁸⁰ *Ibid.*

⁸¹ *Ibid.*

⁸² As patent examiners, attorneys and their clients became more comfortable with structural and means-end claims, patentees shifted their attention to look to claims that would maximise protection, rather than merely meet the subject matter threshold. While hardware structure claim remained the predominant claim format for U.S. software patenting, applicants began to experiment with other types of claims including method of manufacture claims and method claims. Stephen A. Becker, 'Drafting Patent Applications on Computer-Implemented Inventions' (1991) 4 *Harvard Journal of Law and Technology* 237, 255–56.

⁸³ Keith E. Witek, 'Developing a Comprehensive Software Claim Drafting Strategy for US Software Patents' (1996) *Berkeley Technology Law Journal* 363, 406.

⁸⁴ *Ibid.*, 372.

⁸⁵ *Ibid.*, 406.

the Patent Office had little choice but to accept that when drafted as hardware that software-related subject matter was potentially patentable.

The second reason why hardware claims were successful was because they built upon a longstanding drafting practice that patentees used when machine-based inventions included intangible subject matter: namely, one in which the subject matter was tied to something material or physical. As we saw earlier, to qualify for protection in the 1960s and 1970s, applicants had to show that they had brought about a change of kind, that they had created a new kind of thing, or that they had created a specific-purpose machine rather than a mere computer program.⁸⁶ With the shift away from technologically specific excluded subject matter to the more general ‘abstract ideas’ category, subject matter eligibility for computer-related subject matter changed to become one of degree. As a result, patent law found itself asking: where and how was the line to be drawn between a (non-patentable) abstract idea and an application of an idea that produces eligible subject matter? By building on the idea that a ‘machine is a concrete thing’⁸⁷ applicants offered patent law with a relatively straightforward way of answering this question that was subsequently endorsed by the Patent Office and the courts.

Prompted by the drafting strategies initiated by applicants and building on the idea that ‘the opposite meaning of “tangible” is “abstract”’,⁸⁸ subject matter eligibility was recast in terms of materiality. As part of this process, excluded subject matter was characterised in terms of its lack of physicality: it was intangible, ephemeral, and immaterial.⁸⁹ Albert Walker captured the long-standing view of the immaterial nature of excluded subject matter in his 1887 patent law treatise when in writing about laws of nature, scientific principles, and scientific facts he said ‘by whatever name it is called’, it is ‘certain that the thing referred to is not a material substance. It is not to be apprehended by the sense of touch, but when discovered finds a lodgement in the mind as a mental conception only.’⁹⁰ In contrast to the ephemeral intangible excluded subject matter, eligible subject matter was characterised in terms of its physicality. As the Court of Customs and Patent Appeals said in the 1969 decision of *In re Bernhart*, a computer programmed with a new and unobvious program was physically different from the same computer without that program; the

⁸⁶ For a discussion of natural kinds see Ian Hacking, ‘A Tradition of Natural Kinds’ (1991) 63 *Philosophical Studies* 109.

⁸⁷ *Burr v. Duryee* 68 U.S. (1 Wall) 531, 570 (1863).

⁸⁸ USPTO, *Interim Guidelines for Examination of Patent Applications for Patent Subject Matter Eligibility*, OG Notices (22 November 2005), 9. ‘Information as such is an intangible’. *Microsoft Corp. v. AT & T Corp.* 550 U.S. 437, 451 n 12 (2007). The Oxford Dictionary defines abstract as ‘[e]xisting in thought or as an idea but not having a physical or concrete existence’.

⁸⁹ ‘The legal fiction of attributing physicality to software’ allowed ‘software developers to obtain patent protection where none was previously available’. Lawrence Kass, ‘Computer Software Patentability and the Role of Means-Plus-Function Format in Computer Software Claims’ (1995) 15(3) *Pace Law Review* 787, 850.

⁹⁰ Albert Walker, *Text Book of the Patent Laws of the United States of America* (New York: L.K. Krouse & Co, 1887), 7.

programmed computer was a new machine, or at least a new improvement over the unprogrammed computer.⁹¹

With subject matter reframed in terms of its materiality, physicality now functioned as a litmus test for determining the eligibility of computer-related subject matter. This was reflected in the comment that '[w]ithout some stated relationship to something tangible, such as a computer on which the software can be run, software is merely an abstract idea, not useful itself, and thus not patentable.'⁹² Conversely, 'where the process does not employ and affect physical elements, but is concerned solely with intangibles, it is not patentable.'⁹³ Or, as the Supreme Court explained in *Benson*, the difference between gravity, which was non-patentable subject matter, and a pendulum, which relies on gravity for proper operation, which was patentable subject matter, was that the former was math-like and intangible while the latter was a tangible apparatus.⁹⁴

While the focus on physicality as a way of dealing with the eligibility of computer-related subject matter is often traced to the Supreme Court decisions of *Parker v. Flook* and *Diamond v. Diehr*, it has a much longer lineage.⁹⁵ An early example where physicality was used to indirectly protect excluded subject matter was in the patent reissued to Samuel Morse in 1848 for an 'Electromagnetic Telegraph' that was subject to the 1854 Supreme Court decision of *O'Reilly v. Morse*. As well as rejecting Morse's attempt to claim 'the use of the motive power of the electric or galvanic current' which he called 'electro-magnetism' and upholding what has been described as one of the earliest examples of software-like claims,⁹⁶ the Supreme Court also upheld Morse's claim to use machinery (a register, recording instrument) that embodied the excluded subject matter (See Figure 7.3). To use the language of the patent, Morse's patent laid out a physical 'apparatus for and system of transmitting intelligence between distant points by means of electro-magnetism'; that is, it laid out the material circuitry rather than the logic of a machine.⁹⁷

⁹¹ *In re Bernhart* 57 CCPA (Pat.) 737, 417 F.2d 1395 (1969).

⁹² *In re Alappat* 33 F.3d 1526 (Fed. Cir. 1994).

⁹³ Max W. J. Graham, 'Patents for Computer Programs' (1968) 56(2) *California Law Review* 466, 482.

⁹⁴ *Gottschalk v. Benson* 409 U.S. 63, 71–73 (1972).

⁹⁵ Robinson defined a machine as 'an instrument composed of one or more of the mechanical powers and capable, when set in motion of producing by its own operation certain predetermined physical effects. It is an artificial rule of action, receiving crude mechanical force from the motive power and ... transforming ... it according to the mode established by that rule, William C. Robinson, *The Law of Patents for Useful Inventions: Vol 1* (Boston: Little Brown, 1890), 237. Physicality was used to decide the eligibility of other types of subject matter. For example, where some type of printed matter was at stake, it was held that only by showing a physical relationship between the printed matter and the material structure which effects a new and physical result does a claimant show patentability.' Max W. J. Graham, 'Patents for Computer Programs' (1968) 56(2) *California Law Review* 466, 474.

⁹⁶ Claim 5 of Morse's patent provides 'My system of characters consists of dots, spaces, and lines variously combined to form letters and other characters'. On this see Adam Mossoff, 'O'Reilly v. Morse' *George Mason University: Antonia Scalia Law School Working Papers* (2014), 6.

⁹⁷ See Gerardo Con Diaz, *Software Rights: How Patent Law Transformed Software Development in America* (New Haven: Yale University Press, 2019), 19–20.

No. 117.

REISSUED JUNE 13, 1848.

S. F. B. MORSE.
ELECTROMAGNETIC TELEGRAPH.

4 SHEETS—SHEET 3.

Example 10
Register

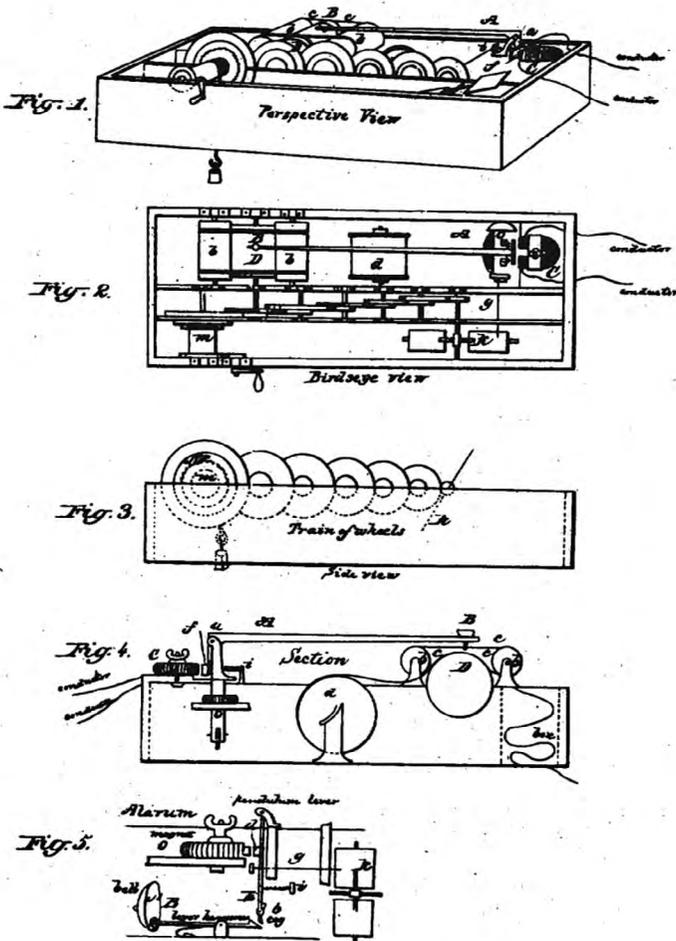


FIGURE 7.3 Register for telegraphic signs
Samuel Morse, 'Improvement in Electro-Magnetic Telegraphs' US Patent No. 1,647
(13 June 1848). Courtesy of the United States Patent and Trademark Office.

The decision to use materiality as a touchstone for deciding the eligibility of computer-related subject matter was justified on the basis that it ensured that patents were only ever granted for practical inventions with ‘real world’ value.⁹⁸ Materiality also ensured that subject matter that was otherwise illusive, undefined, and difficult to delineate was confined within ‘definite bounds’.⁹⁹ Physicality also aligned with a particular vision of property that had long held sway in intellectual property law. As Waite wrote in a 1917 article on the patentability of mental processes, the ‘fact that possession has so correlated with the theory of property that it is difficult to disassociate ownership from the possibility of physical possession.’¹⁰⁰ The use of materiality as a litmus test for determining subject matter eligibility was also explained on the basis that it ensured that the claims did not reach beyond what was disclosed. The reason for this was that a ‘claim that is tied to a particular machine or brings about a particular transformation of a particular article does not pre-empt all uses of a fundamental principle in any field but rather is limited to a particular use, a specific application. Therefore, it is not drawn to the principle in the abstract’.¹⁰¹

As well as being used as a guide for determining subject matter eligibility within patent law, materiality was also used to explain the way different types of intellectual property interacted with computer-related subject matter. In this sense, tangibility replaced the computer program as a boundary object within intellectual property law. This is reflected in the comment that while ‘[h]ardware, because tangible, receives its primary protection from the legal standards of patent law’, ‘[s]oftware, because intangible, receives its primary protection from copyright law, although patent law provides some protection for software linked to physical manifestations. Algorithms, unless tied to a physical process, receive no protection at all’.¹⁰² The explanation given for the ‘different treatment of hardware, software, and algorithms lies in the Court’s focus on the physical manifestations of property. Despite the inextricable bonds among them, hardware is tangible whereas software and algorithms are not’.¹⁰³

While physicality was initially used as a touchstone to examine the eligibility of computer hardware and computer-related inventions that produced physical change outside the computer, it was versatile enough to accommodate many of the changes

⁹⁸ USPTO, *Interim Guidelines for Examination of Patent Applications for Patent Subject Matter Eligibility*, OG Notices (22 November 2005), 8.

⁹⁹ T. Buckman, ‘Protection of Proprietary Interests in Computer Programs’ (1969) *Journal of the Patent Office Society* 135, 151. (Buckman was a Patent Office examiner). As Supreme Court said in *Benson*: ‘[T]he arts of tanning, dyeing, making waterproof cloth, vulcanizing India rubber, smelting ores ... are instances, however, where the use of chemical substances or physical acts, such as temperature control, changes articles or materials. The chemical process or the physical acts which transform the raw material are, however, sufficiently definite to confine the patent monopoly within rather definite bounds’. *Gottschalk v. Benson* 409 U.S. 63, 69 (1972).

¹⁰⁰ John Waite, ‘The Patentability of a Mental Process’ (1917) 15(2) *Michigan Law Review* 660.

¹⁰¹ *In re Bilski* 545 F.3d 943, 957 (Fed. Cir. 2008).

¹⁰² Note, ‘Computer Intellectual Property and Conceptual Severance’ (1990) 103 *Harvard Law Review* 1046, 1049.

¹⁰³ *Ibid.*

that occurred in information technology across the later part of the twentieth century. This can be seen, for example, in the way that patent law responded to attempts to patent information embodied on a computer-readable medium (such as a floppy disc). In thinking about this new type of subject matter, patent law built upon the intangible/tangible dichotomy that underpins the physicality requirement to draw a distinction between non-functional and functional descriptive material. Because non-functional descriptive material such as music, literary works, and compilations of data recorded on a computer-readable medium was merely carried on rather than structurally and functionally interrelated to the medium, the subject matter was not a physical thing.¹⁰⁴ As such, non-functional descriptive material embodied on a computer-readable medium could not be protected. In contrast, functional descriptive material was deemed to be patent eligible.¹⁰⁵ The reason for this was that when functional descriptive material was recorded on a computer-readable medium it became structurally, functionally, and physically integrated into that medium.¹⁰⁶ Even as patent law extended its reach beyond programs embodied with a computer (as a machine) to recognise programs embodied on a computer-readable medium, it did so by focusing on the physicality of the subject matter.

While patent practitioners were largely successful in their efforts to draft patent applications for new types of computer-related subject matter in a way that highlighted their physicality and thus rendered them patent eligible, the physicality requirement did pose some problems. This can be seen, for example, in the decision of *In re Nuijten*, which concerned a technique for reducing the distortion caused when digital watermarks were introduced into signals. As well as claiming the process and apparatus for generating, receiving, processing, or storing signals, the applicants also attempted to patent the signals themselves. While the process and apparatus claims were allowed, the claims for the signals were not. The reason for this was that the signal claims ‘were not limited by any specified physical medium, nor do the dependent claims add any physical limitations.’¹⁰⁷ As the Board said, the signal ‘has no physical attributes and merely describes the abstract characteristics of the signal and, thus, it is considered an “abstract idea” unpatentable under *Diamond v. Diehr*’.

¹⁰⁴ USPTO, *Examination Guidelines for Computer-Related Inventions* (29 March 1996) 61 Fed. Reg. 7478, 5.

¹⁰⁵ *Ibid.*

¹⁰⁶ In response to the examiner and Board who had held that ‘the provision of new signals to be stored by the computer does not make it a new machine, i.e., the computer is structurally the same, no matter how new, useful and unobvious the result, the court replied: ‘To this question we say that if a machine is programmed in a certain new and unobvious way, it is physically different from the machine without that program; its memory elements are differently arranged’. Importantly, the court added that the fact that these physical changes were ‘invisible to the eye should not tempt us to conclude that the machine has not been changed’. *In re Lowry* 32 F.3d 1579, 1583 (Fed. Cir. 1994) citing *Application of Bernhart* 417 F.2d 1395, 1400 (CCPA 1969).

¹⁰⁷ *In re Nuijten* 500 F.3d 1346, 1353 (Fed. Cir. 2007).

The Board and the Federal Circuit Court of Appeals also took the unusual step of framing subject eligibility in terms of the statutory categories of patentable subject matter (process, machine, manufacture, or composition of matter), rather than in terms of the judicially created excluded categories, which is usually the case with computer-related subject matter. Importantly in doing so the Board and the Federal Circuit drew upon the physicality requirement to find that the subject matter did not fall within the statutory categories of patentable subject matter. In particular, it was held that the signal claim did not qualify as a ‘machine’ (the possibility that they were processes or composition of matter were effectively dismissed out of hand) because it had ‘no concrete tangible physical structure’. More specifically, it was held that a propagating electromagnetic signal was not a machine as that term is used in § 101 because while a transitory signal made of electrical or electromagnetic variances ‘is physical and real, it does not possess concrete structure. No part of the signal – the crests or troughs of the electromagnetic wave, or perhaps the particles that make it up (modern physics teaches that both features are present simultaneously) is a mechanical device or part’.¹⁰⁸

The Board and the Federal Circuit also looked to physicality when considering whether the signal was a ‘manufacture’. In denying that it was the Board said that as the ‘signal does not have any physical structure or substance’ it ‘does not fit the definition of a “manufacture” which requires a tangible object’. The Federal Circuit adopted a similar approach in denying that a signal was a manufacture. A key reason for this was that a signal, which was a transient electric or electromagnetic transmission, was neither a tangible article or a commodity. As the Federal Circuit said: ‘While such a transmission is man-made and physical – it exists in the real world and has tangible causes and effects – it is a change in electric potential that, to be perceived, must be measured at a certain point in space and time by equipment capable of detecting and interpreting the signal. In essence, energy embodying the claimed signal is fleeting and is devoid of any semblance of permanence during transmission. Moreover, any tangibility arguably attributed to a signal is embodied in the principle that it is perceptible – e.g., changes in electrical potential can be measured. All signals within the scope of the claim do not themselves comprise some tangible article or commodity. This is particularly true when the signal is encoded on an electromagnetic carrier and transmitted through a vacuum – a medium that, by definition, is devoid of matter. Thus, we hold that Nuijten’s signals, standing alone, are not “manufacture[s]” under the meaning of that term in § 101.’¹⁰⁹

Whether it was the transformation of an article from one state or thing to another state or thing,¹¹⁰ the existence of a physical step,²⁰ a ‘useful, concrete and tangible result’,¹¹¹ or a physical or tangible form,¹¹² the result was the same: subject matter

¹⁰⁸ *Ibid.*, 1355–56.

¹⁰⁹ *Ibid.*, 1356–57.

¹¹⁰ *Parker v. Flook* 437 U.S. 584 (1978).

¹¹¹ *State Street Bank v. Signature Financial Group* 149 F.3d 1368 (Fed. Cir. 1998).

¹¹² *Digitech Image Techs v. Elecs. for Imaging* 758 F.3d 1344, 1348 (Fed. Cir. 2014).

eligibility of computer-related subject matter was dependent on the existence of a tangible material trace that the examiner or court could latch on to as proof of invention. That is, evidence of physical change (or some equivalent thereof) was treated as proof of the transformation of an abstract intangible computer program into a novel three-dimensional machine and thus of its patent-worthiness.¹¹³ As a result, subject matter eligibility again became a question of kind, the difference now being that it now turned on the tangibility of the subject matter, rather than in terms of its machine-like status, which it had been previously. By calibrating subject matter eligibility in terms of materiality, patent law ‘enunciated a definitive test to determine whether a ... claim is tailored narrowly enough to encompass only a particular application of a fundamental principle rather than to pre-empt the principle itself.’¹¹⁴

Recognising patent laws ongoing reliance on physicality has a number of ramifications for how we think about patent law; one of the most important is that it forces us to question the suggestion I made earlier that the unbundling of hardware and software that took place in the early 1970s brought about a dematerialisation of computer-related subject matter. While from a commercial perspective, software products may have been separated from the hardware they interacted with, from a technical or engineering perspective they were still connected and intertwined (at least potentially) with material machines.¹¹⁵ The situation was similar in patent law where the ongoing use of materiality as a touchstone for distinguishing ephemeral immaterial non-patentable subject matter from potentially patentable tangible computer-related inventions suggests, at least in this context, that the unbundling did not lead to the dematerialization of the subject matter. While the partisan characterisation of the subject matter as an unbundled dematerialised computer program, on the one hand, and a bundled material computer-driven machine on the other, may have served the ends of software and hardware producers, it did not translate well into the subject matter inquiry in patent law. The reason for this was that the material and immaterial are not separate and distinct as these arguments presupposed. Rather, as the notion of informed materials reminds us, the material

¹¹³ The 1996 Guidelines addressed the rationale for excluding claims to software alone from the realm of statutory subject matter as follows: ‘[C]omputer programs claimed as computer listings per se, i.e., the descriptions or expressions of the programs, are not physical “things,” nor are they statutory processes.’ See *Bilski v. Kappos* 561 U.S. 593 (2010).

¹¹⁴ *In re Bilski* 545 F.3d 943, 954 (Fed. Cir. 2008). Even though part of the invention in *Diamond v. Diehr* was data transformation (process conditions data into rubber cure time data), ‘an integral part of the invention was the physical transformation of uncured physical material or chemical compounds into cured rubber. It was this physical transformation that the Court found dispositive in rendering the process or method claims patentable. By focusing on the tangibility of inventions, the Court recognised a legal framework that provides protection for abstract inventions such as software and algorithms where they are linked to a physical process. Note, ‘Computer Intellectual Property and Conceptual Severance’ (1990) 103 *Harvard Law Review* 1046, 1051.

¹¹⁵ One of the factors that Judge Rich relied upon in his dissent in *In re Johnson*, where he rejected the idea that a programmed computer was a unique machine, was that the invention was being sold as a computer program. *In re Johnson* 502 F.2d 765, 773 (CCPA 1974).

and the immaterial constantly blend into and inform each other; the immaterial (nearly) always has a material context.

FROM MATERIALITY TO SPECIFICITY

While physicality proved to be a versatile and resilient tool for deciding the eligibility of computer-related subject matter, it eventually ran up against a number of problems as the technology advanced. One reason for this was that while physicality may have provided courts, patent examiners, and lawyers with a relatively straightforward and easy-to-apply touchstone for determining the eligibility of the special-purpose machines of the 1970s where the change occurred outside of the computer, it was more difficult to apply when the changes occurred within the computer. As Judge Newman said in *Bilski*, the physicality test was difficult to apply where the subject matter was for processes that dealt ‘with data and information, whose only machinery is electrons, photons, or waves, or whose product is not a transformed physical substance’.¹¹⁶ Problems also arose where applicants claimed advanced diagnostic medicine techniques and where inventions were based on linear programming, data compression, the manipulation of digital signals as well as other processes that handle data and information in novel ways.¹¹⁷

As the information technology industry progressed and the subject matter moved further away from the programmed computers of the 1970s, the courts and examiners increasingly found themselves struggling when applying the physicality test to answer difficult ‘esoteric and metaphysical’¹¹⁸ questions such as whether there were any limits on the type or amount of physical transformation that was needed to guarantee eligibility,¹¹⁹ or whether a material trace was transitory, electronic, virtual, and so on. These problems were exacerbated by the fact that there was always some form of physical transformation whenever a computer functions (signals are transformed and the computers components are changed during execution of a computer program). As the Patent Office admitted, one of the consequences of this was that in these cases physicality could not be determinative of whether the computer-related subject matter was patentable.¹²⁰

As technological developments moved subject matter even further away from the programmed machine, the physicality test became even more difficult to apply.

¹¹⁶ *In re Bilski* 545 F.3d 943, 976, 985 (Fed. Cir. 2008).

¹¹⁷ See *Ibid.*, 964; *Bilski v. Kappos* 561 U.S. 593 (2010).

¹¹⁸ *In re Nuijten* 500 F.3d 1346, 1353 (Fed. Cir. 2007).

¹¹⁹ There was uncertainty about the type and extent of the functional relationship needed between software and a tangible object for the claimed invention to qualify. See Elizabeth A. Richardson, ‘Toward a Direct Functional Relationship Requirement for Claims to Software Encoded on a Computer-Readable Storage Medium’ (2006) 3 *Oklahoma Journal of Law and Technology* 30.

¹²⁰ Nancy J. Linck and Karen A. Buchanan, ‘Patent Protection for Computer-Related Inventions’ (1996) 18 *Hastings Communications and Entertainment Law Journal* 660, 669 (Linck and Buchanan were solicitors for the US Patent Office).

One of the consequences of this was as Justice Mayer said in *Bilski*, ‘although [the Federal Court] has struggled for years to set out what constitutes sufficient physical transformation to render a process patentable, we have yet to provide a consistent or satisfactory resolution of this issue’.¹²¹

These problems were compounded by the fact that patent applicants rarely attempted ‘to patent (let alone succeed in obtaining a patent for) an abstract idea per se. Instead, where a patent implicates the abstract idea exception’ the claim ‘typically involves some concrete or tangible implementation or application of that idea’. One of the consequences of this was that when deciding whether a claim was ‘directed to an abstract idea’, a court had to ‘dissect the underlying abstract idea from the integrated claim, an inevitably subjective undertaking’.¹²² As a result, deciding whether a particular claim was abstract was ‘subjective and unsystematic, and the debate often trends toward the metaphysical, littered with unhelpful analogies and generalizations’.¹²³

There were a number of different responses to the problems that arose in attempting to apply the physicality test to newer forms of computer-related subject matter. One response was to place limits on when physicality could be used to determine patent eligibility. While it was recognised that materiality (in the form of the machine-or-transformation test) was a ‘useful and important clue’ for determining patent eligibility,¹²⁴ technological change, which ensured that ‘not all machine implementations [were] created equal’, meant that the physicality test could no longer be applied automatically to all computer-related subject matter. As the Supreme Court said in *Mayo*, the reason for this was that ‘not all transformations or machine implementations infuse an otherwise ineligible claim with an inventive concept’.¹²⁵ This qualification as to when physicality could be used to decide subject matter eligibility meant that simply using off-the-shelf technology for its intended purpose,¹²⁶ introducing generic computer limitations, or ‘implementing a mathematical principle on a physical machine, namely a computer’¹²⁷ was not enough to ensure that the subject matter eligibility threshold was met. As Justice Chen said, the bare fact that a computer exists in the physical rather than purely conceptual realm was ‘beside the point’.¹²⁸

Another response to the problems that arose in using tangibility as a touchstone for eligibility was to expand what was meant by ‘physicality’. This can be seen for example *in re Lowry* where the Federal Circuit was called on to evaluate the eligibility of an application for a data processing system that provided an efficient, flexible

¹²¹ *In re Bilski* 545 F.3d 943, 1010 (Fed. Cir. 2008).

¹²² See *CLS Bank Int'l v. Alice Corp.* 717 F.3d 1269, 1277 (Fed. Cir. 2013).

¹²³ *Ibid.*

¹²⁴ *Bilski v. Kappos* 130 S. Ct. 3218, 3227 (2010).

¹²⁵ *Mayo Collaborative Servs. v. Prometheus Labs.* 132 S.Ct. 1289, 130.

¹²⁶ *Chamberlain Group v. Techtronic Industries* 935 F.3d 1341 (Fed. Cir. 2019). See also *In re Marco Guldenaar Holding* 911 F.3d 1157, 1161 (Fed. Cir. 2018).

¹²⁷ *Gottschalk v. Benson* 409 U.S. 63, 64 (1972).

¹²⁸ *DDR Holdings v. Hotels.com* 773 F.3d 1245 (Fed. Cir. 2014).

method of organizing stored data in computer memory.¹²⁹ In upholding the patent, the Federal Circuit said that it did not matter that the stored data did not adopt a physical structure per se. The reason for this was that ‘if a machine is programmed in a certain new and unobvious way, it is physically different from the machine without that program; its memory elements are differently arranged’. In a move which extended the meaning of physicality (and certainly moved it beyond Walker’s idea that patentable subject matter was defined by its ability to be apprehended by the sense of touch) the court added that ‘the fact that these physical changes are *invisible to the eye* should not tempt us to conclude that the machine has not been changed’.¹³⁰ The definition of physicality was expanded further *in re Abele* where the Federal Circuit said that physicality reached ‘beyond physical objects or substances themselves to include *representations* of physical objects or substances.’¹³¹ This meant that a claim providing for the electronic transformation of x-ray data or data ‘clearly representing physical and tangible objects’ into a particular visual depiction on a display was patentable.

While the decision to extend the meaning of physicality to encompass subject matter that brought about non-visible physical changes or produced representations of physical objects may have provided some relief to patentees, there were still situations where the ‘focus on tangible physical inventions’ meant that ‘many abstract advances in computer technology remain[ed] unprotected’.¹³² This led commentators to complain that physicality tied patent law to an outdated worldview that ‘acted as a substantial obstacle to software inventors seeking patent protection’,¹³³ or that in drawing ‘an arbitrary distinction between the tangible and the abstract’ it left ‘abstract innovations either completely unprotected or distorted and “shoehorned” into some tangible expression’, which resulted in ‘high transaction costs and uncertain protection’.¹³⁴

¹²⁹ *In re Lowry* 32 F.3d 1579, 1580 (Fed. Cir. 1994).

¹³⁰ *Ibid.*, 1582–3 quoting *In re Bernhardt* 417 F.2d 1395, 1400 (CCPA 1969).

¹³¹ *In re Abele* 684 F.2d 902, 908–9 (CCPA 1982) (emphasis added).

¹³² Note, ‘Computer Intellectual Property and Conceptual Severance’ (1990) 103 *Harvard Law Review* 1046. With some technologies that the physicality test was ‘too easily circumvented’ that, for example, ‘[t]hrough clever draftsmanship, nearly every process claim can be rewritten to include a physical transformation’ *In re Bilski* 545 F.3d 943, 1008–9 (Fed. Cir. 2008). ‘The fact that a computer “necessarily exist[s] in the physical, rather than purely conceptual, realm,” ... is beside the point. There is no dispute that a computer is a tangible system (in §101 terms, a “machine”), or that many computer-implemented claims are formally addressed to patent-eligible subject matter. But if that were the end of the §101 inquiry, an applicant could claim any principle of the physical or social sciences by reciting a computer system configured to implement the relevant concept. Such a result would make the determination of patent eligibility “depend simply on the draftsman’s art”, thereby eviscerating the rule that “[l]aws of nature, natural phenomena, and abstract ideas are not patentable,” *Alice Corp. v. CLS Bank International* 134 S. Ct. 2347, 2358–9 (2014).

¹³³ Lawrence Kass, ‘Computer Software Patentability and the Role of Means-Plus-Function Format in Computer Software Claims’ (1995) 15(3) *Pace Law Review* 787, 868–69.

¹³⁴ Note, ‘Computer Intellectual Property and Conceptual Severance’ (1990) 103 *Harvard Law Review* 1046, 1060.

While the physicality threshold might have worked for inventions from ‘the brick and mortar world’ of the Industrial Age and even been effective when applied to the special purpose programmed machines of the 1960s and 1970s, which were ‘grounded in a physical or other tangible form’,¹³⁵ it excluded many new information-age innovations such as electronic signals and electronically manipulated data.¹³⁶ The problem that patent law faced was that many of the advances in computer technology that had taken place since the 1970s consisted ‘of improvements to software that, by their very nature, may not be defined by particular physical features but rather by logical structures and processes.’¹³⁷ Or, as Justice Radar wrote in 2008, ‘[t]oday’s software transforms our lives without physical anchors.’¹³⁸ The situation was summed up by the comment in the amicus curiae brief for the United States in the *Alice* decision that the ‘abstract-ideas exception should not encompass innovations in technology, science, or industry’ ... ‘that improve computer function, including those “based on linear programming, data compression and the manipulation of digital signals”’.¹³⁹ Instead of being excluded it was argued that ‘those invention should be patent-eligible because they disclose concrete technological applications and fall within patent law’s traditional bailiwick of the scientific, technological, and industrial arts. That is so even if the advancement in computing technology is not grounded in “tangible form”’.¹⁴⁰ Building on the idea that it was not appropriate to freeze ‘patents to old technologies, leaving no room for the revelations of the new, onrushing technology’¹⁴¹ and that there was a need to make the subject matter eligibility test ‘responsive to the needs of the modern world’,¹⁴² there was a growing sense in which the physicality test was antiquated and in need of change.

Patent law initially responded to this challenge by downplaying the role that physicality played in deciding subject matter eligibility. In rethinking how the eligibility of information-age subject matter was to be decided the courts said that while in some circumstances ‘physical transformation’ was a ‘useful clue’ for deciding subject matter eligibility,¹⁴³ they stressed that it was ‘not an invariable requirement’. Instead, physicality was presented as an example of how excluded subject matter could bring about a useful application.¹⁴⁴ As a result, it was argued that physicality

¹³⁵ *Bilski v. Kappos* 561 U.S. 593, 605 (2010). See also *In re Bilski* 545 F.3d 943, 1015 (Fed. Cir. 2008) (Rader Circuit Judge, dissenting).

¹³⁶ *In re Bilski* 545 F.3d 943, 962 (Fed. Cir. 2008).

¹³⁷ *Enfish v. Microsoft Corp* 822 F.3d 1327, 1339 (Fed. Cir. 2016).

¹³⁸ *In re Bilski* 545 F.3d 943, 1015 (Fed. Cir. 2008).

¹³⁹ Brief for the United States as Amicus Curiae in Support of Respondents No 13–298 *Alice Corporation v. CLS Bank*, 16.

¹⁴⁰ *Ibid.*

¹⁴¹ *Gottschalk v. Benson* 409 U.S. 63, 71 (1972).

¹⁴² *ATT Corp v. Excel Communications* 172 F.3d 1352, 1356 (Fed. Cir. 1999).

¹⁴³ Brief for the United States as Amicus Curiae in Support of Respondents No 13–298 *Alice Corporation v. CLS Bank* 1.

¹⁴⁴ *ATT Corp v. Excel Communications* 172 F.3d 1352, 1358 (Fed. Cir. 1999).

should not be the sole criterion for determining the patentability of newer forms of computer-related subject matter.¹⁴⁵ At the same time the courts also began to distance themselves from the decisions of the 1970s and 1980s, which had promoted the use of materiality to prove eligibility. In light of changes in technology, these earlier decisions and with them the physicality test that they relied upon were now said to be ‘of limited usefulness because the more challenging process claims of the twenty-first century are seldom so clearly limited in scope as the highly specific, plainly corporeal industrial manufacturing process of *Diehr*, nor are they typically as broadly claimed or purely abstract and mathematical as the algorithm of *Benson*.’¹⁴⁶

Freed up from the ability or need to find physicality as a pre-condition for eligibility, the courts returned to focus (again) on the abstract nature of the excluded subject matter. As had been the case previously, the problem with abstract subject matter was that it provided too much protection (or at least too much protection in relation to what was being disclosed). As the Supreme Court said in *Mayo*, the concern underlying the exceptions to subject matter eligibility ‘is not tangibility, but pre-emption.’¹⁴⁷ While some pre-emption was permissible, too much was not. As a result, the task that the law set for itself in dealing with computer-related subject matter was working out how to differentiate abstract ineligible subject matter which pre-empted too much from eligible subject matter, which did not.

Building on the idea that the ‘preemption concern arises when the claims are not directed to a specific invention and instead improperly monopolize “the basic tools of scientific and technological work”’,¹⁴⁸ subject matter eligibility was recast in terms of the *specificity* of the invention. Unlike the situation previously where abstractness was framed in terms of materiality, eligibility was now evaluated in terms of the specificity of the subject matter. With abstractness and specificity treated as opposites, the specificity of the subject matter came to be treated as a proxy for its eligibility. Conversely, the absence of specificity gave rise to a presumption that the subject matter was abstract and thus ineligible. Framed in terms of pre-emption this meant that while patenting a specific or particular invention ‘would incentivize further innovation in the form of alternative methods for achieving the same result’, allowing more abstract claims would ‘inhibit ... innovation by prohibiting other inventors from developing their own solutions to problem without first licensing the abstract idea’.¹⁴⁹

¹⁴⁵ *Bilski v. Kappos* 561 U.S. 593, 604 (2010).

¹⁴⁶ *In re Bilski* 545 F.3d 943, 954 (Fed. Cir. 2008). *AT&T* 172 F.3d at 1358–59, 50 USPQ2d, 452 (physical transformation is only one example of a practical or useful application of an abstract idea).

¹⁴⁷ *Mayo Collaborative Servs. v. Prometheus Labs.* 132 S.Ct. 1289, 1301 (2012). On this see *McRo v. Bandai Namco Games Am.* 837 F.3d 1299, 1314 (Fed. Cir. 2016).

¹⁴⁸ *Gottschalk v. Benson* 409 U.S. 63, 67, 93 S.Ct. 253, 34; *Alice Corp. v. CLS Bank International* 134 S. Ct. 2347, 2354 (2014); *Association for Molecular Pathology v. Myriad Genetics* 133 S. Ct. 2107, 2116 (2013).

¹⁴⁹ *Electric Power Group v. Alstom* 830 F.3d 1350 (Fed. Cir. 2016).

The use of specificity as a guide for deciding the eligibility of computer-related subject matter can be seen in *DDR Holdings*,¹⁵⁰ a 2014 Federal Circuit decision, which concerned the eligibility of a system that allowed website owners who advertised third party goods and services to prevent visitors who wanted to purchase such goods and services from leaving their site. The invention did this by directing visitors who clicked on links to third-party vendors to a hybrid webpage that combined information for the third-party product with the look-and-feel of the host website. In finding the claims to be eligible, the court noted that the invention did ‘not merely recite the performance of some business practice known from the pre-Internet world along with the requirement to perform it on the Internet’, as was the case in many of the situations where software claims had been held to be ineligible. Instead, the invention was ‘rooted in computer technology in order to overcome a problem specifically arising in the realm of computer networks.’¹⁵¹ A key reason why the claims were allowed was because they specified how interactions with the Internet were manipulated to yield a desired result – a result that overrode the routine and conventional sequence of events ordinarily triggered by the click of a hyperlink.¹⁵² As the court said, the claim was calculated to improve sales in a very specific manner. Importantly it did so without pre-empting all applications of the idea to increase sales by making two web pages look the same. Essentially, the abstract idea was narrowly tailored to increase sales in a specific application without broadly claiming ownership over a societal building block like the computer or the Internet.’ Because the patent only claimed ‘a specific way to automate the creation of a composite web page’ the court felt it would only ‘have a limited preemptive effect’ and, as such, was eligible.

A similar approach was adopted in *Enfish*, a 2014 Federal Court decision that concerned the eligibility of claims for a ‘method and system for reducing the time it takes for a trader to place a trade when electronically trading on an exchange, thus increasing the likelihood that the trader will have orders filled at desirable prices and quantities.’¹⁵³ To this end the patent claimed a data storage and retrieval system for computer memory, which allowed faster searching and more effective storage of data. To determine whether these claims were eligible, the court said it was necessary to ‘look to whether the claims ... focus on a specific means or method that improves the relevant technology or are instead directed to a result or effect that itself is the abstract idea and merely invoke generic processes and machinery.’¹⁵⁴ In applying this approach, the Federal Circuit held that the claims were not directed to an abstract idea. Rather, they were directed to a specific improvement in the way computers operated. The Court held that the ‘challenged patents do not simply claim information displayed on a graphical user interface’. Nor did they merely

¹⁵⁰ *DDR Holdings v. Hotels.com* 773 F.3d 1245 (Fed. Cir. 2014).

¹⁵¹ *Ibid.*, 1257.

¹⁵² *Ibid.*, 1258–9.

¹⁵³ *Enfish v. Microsoft Corp* 822 F.3d 1327 (Fed. Cir. 2016).

¹⁵⁴ *Ibid.*, 1336.

involve the routine or conventional use of computers or the Internet. Instead, the claims required ‘a specific, structured graphical user interface paired with a prescribed functionality directly related to the graphical user interface’s structure that is addressed to and resolves a specifically identified problem in the prior state of the art.’¹⁵⁵ In doing so, the court distinguished between situations where ‘general-purpose computer components were added post-hoc to a fundamental economic practice or mathematical equation (which were ineligible)’ and situations where ‘the claims were directed to a specific implementation of a solution to a problem in the software arts’ (which were eligible).

Another example of the way specificity was used as a proxy for deciding subject matter eligibility was the 2016 Federal Circuit decision of *McRo*, which considered the validity of US Patent Number 6,611,278, which claimed a method for automatically animating lip movements and facial expressions for 3-D animated characters.¹⁵⁶ To do this, computer software applied a set of rules to control the lip movement and facial expressions of an animated character as it pronounced certain sounds. In evaluating the patent, the Federal Circuit applied *Enfish*’s ‘specific improvement’ test to determine whether the claims were directed to abstract ideas.¹⁵⁷ In doing so, the court said: ‘We look to whether the claims ... focus on a specific means or method that improves the relevant technology or are instead directed to a result or effect that itself is the abstract idea and merely invoke generic processes and machinery’. While the District Court had said that the claims were ineligible because they were drawn to the abstract idea of automated rules, the Federal Circuit disagreed saying that the claims were ‘limited to rules with specific characteristics’. In particular, the Federal Circuit said that the patent described a specific improvement to animation technology through its use of a specific set of rules governing how animated facial expressions should be synchronized with sounds. On this basis, the court concluded that the claimed invention was not drawn to an abstract idea, explaining that ‘[t]he claimed process uses a combined order of specific rules that renders information into a specific format that is then used and applied to create desired results: a sequence of synchronized, animated characters.’¹⁵⁸ Recognising the shift away from physicality, the court added that while ‘the result may not be tangible, there is nothing that requires a method “be tied to a machine or transform an article” to be patentable’.¹⁵⁹

While physicality remains an important touchstone for deciding the eligibility of some types of computer-related subject matter, when dealing with more immaterial inventions physicality has been replaced by a concern with the relative specificity

¹⁵⁵ *Ibid.*

¹⁵⁶ *McRo v. Bandai Namco Games* 837 F.3d 1299, 1314 (Fed. Cir. 2016).

¹⁵⁷ *Ibid.*

¹⁵⁸ The claim ‘does not preempt approaches that use rules of a different structure or different techniques’. *McRo v. Bandai Namco Games* 837 F.3d 1299, 1316 (Fed. Cir. 2016).

¹⁵⁹ *McRo Inc. v. Bandai Namco Games* 837 F.3d 1299, 1315 (Fed. Cir. 2016) citing *Bilski v. Kappos* 561 U.S. 593, 603 (2010).

of the subject matter. Whether it was the specific way sensors operate,¹⁶⁰ the ‘specific method of filtering Internet content’,¹⁶¹ specific improvements in the way computers operate,¹⁶² or a ‘specific way of enabling a computer to monitor data from multiple sources across an electric power grid’,¹⁶³ or some variation thereof, the fate of information-based computer-related subject matter in the early part of the twenty-first century turned on how precisely the subject matter had been claimed. As Mark Lemley said at a 2016 roundtable on subject matter eligibility organised by the US Patent Office, ‘the Federal Circuit is beginning to define a “set of standards” to distinguish between an ineligible invention and one that is directed to a specific algorithm or improvement in computer technology’.¹⁶⁴

While the decision to use the specificity of the subject matter as a proxy for eligibility was presented as a logical extension of the use of physicality to decide whether the subject matter threshold had been met and as a continuum of pre-existing practice, it did bring about a number of subtle but important changes in the way patent law interacted with computer-related subject matter. The first and most obvious was that it ensured that subject matter that would have otherwise been excluded because it lacked tangibility was now able to be protected. The decision to use specificity as a litmus test for deciding eligibility also changed the way subject matter was evaluated. Previously, eligibility had been treated as a question of kind: subject matter either had a physical dimension and was eligible or it didn’t. With the shift to specificity, there was a sense in which subject matter eligibility was again a question of kind: subject matter was either classified as specific, non-abstract, and eligible, or it was abstract and ineligible. Working with these binary categories, there were no grey areas, no difficult questions of degree, and no problematic lines to be drawn: subject matter was either specific thus eligible, or abstract thus ineligible. As patent law confronted more and more information-based subject matter, however, this neat binary distinction began to break down.

Building on the realisation that it was possible to claim subject matter in a ‘highly specific’ way but nonetheless still ‘manipulate abstract concepts’, in their amicus brief for the United States in *Alice* the Solicitor General suggested that ‘the term “abstract” [was] best understood to mean not the opposite of specific, but the opposite of concrete.’¹⁶⁵ While the advice of the Solicitor General was not followed

¹⁶⁰ *Thales Visionix v. United States* 850 F.3d 1343, 1344–5 (Fed. Cir. 2017).

¹⁶¹ *Bascom Global Internet Servs. v. AT&T Mobility* 827 F.3d 1341 (Fed. Cir. 2016). (‘A specific, discrete implementation of the abstract idea of filtering content’).

¹⁶² *Informatica Corp. v. Protegrity* CBM2015-00021 Patent Trial and Appeal Board (31 May 2016).

¹⁶³ *Electric Power Group v. Alstom* 830 F.3d 1350 (Fed. Cir. 2016). *Amdocs (Israel) v. Openet Telecom* 841 F.3d 1288 (Fed. Cir. 2016).

¹⁶⁴ USPTO, ‘Patent Eligible Subject Matter: Report on Views and Recommendations from the Public’ (July 2017), 39.

¹⁶⁵ Brief for the United States as Amicus Curiae in Support of Respondents No 13-298 (February 2014) *Alice Corporation v. CLS Bank*, 24. ‘The opposite of “concrete” is unrepeatable or unpredictable’. USPTO, *Interim Guidelines for Examination of Patent Applications for Patent Subject Matter Eligibility*, OG Notices (22 November 2005), 10.

(either in *Alice* or elsewhere), it is nonetheless still important in so far as it highlights the fact that specific subject matter is not the same as concrete subject matter and that specific subject matter is (potentially) broader and more abstract than concrete subject matter. It also highlights the fact that within the taxonomic framework being developed in patent law, specificity existed somewhere between concrete physical subject matter and abstract subject matter. Because specific subject matter potentially included abstract ideas, it was no longer possible to rely on it as a simple guide to determine eligibility. In doing so, it suggests that using the specificity of the subject matter to decide eligibility may not have been as straightforward as it may first have appeared.

In many ways this was confirmed by *McRo*; the decision of the Federal Circuit about the eligibility of a method for automatically animating lip movements and facial expressions for 3-D animated characters that was discussed above. As we saw, in finding the subject matter eligible the court recognised that the patent did not ‘improperly purport to cover all rules’; nor did it pre-empt ‘the field of rules-based animation’ or ‘all techniques for automating 3-D animation that rely on rules’. Rather, the court found that the claims were ‘limited to rules with specific characteristics’. As such the patent could not be classified as abstract excluded subject matter. So far, so good. The ability to use specificity as a guide to subject matter eligibility was called into question, however, by the fact that while the specificity of the subject matter meant that it was not abstract, the court also found that the subject matter was not restricted to individual, concrete inventions. Rather, the claims were ‘limited to rules with certain common characteristics, i.e., a genus’.¹⁶⁶ While it may have been ‘self-evident that genus claims create a greater risk of preemption, thus implicating the primary concern driving § 101 jurisprudence’ the court stressed that that ‘this does not mean they are unpatentable’.¹⁶⁷ Drawing on *Diamond v. Chakrabarty* (which had recognised the patentability of a bacterium from the genus *Pseudomonas*), the Federal Circuit said that ‘[c]laims to the genus of an invention, rather than a particular species, have long been acknowledged as patentable.’ And while patent law had ‘evolved to place additional requirements on patentees seeking to claim a genus ... these limits have not been in relation to the abstract idea exception to [subject matter eligibility in § 101]’. ‘Rather they have principally been in terms of whether the patentee has satisfied the trade-off of broad disclosure for broad claim scope implicit’ in the requirement of enabling disclosure (in section 112).¹⁶⁸

Had patent law followed the advice of the Solicitor General in *Alice* and used concreteness as a proxy for subject matter eligibility, the eligibility test might have remained a question of kind. By accepting that specificity was potentially broader than a concrete individual invention (akin to a chemical sample) but something less than a patent that

¹⁶⁶ *McRo v. Bandai Namco Games* 837 F.3d 1299, 1313 (Fed. Cir. 2016).

¹⁶⁷ *Ibid.*, 1314.

¹⁶⁸ *Ibid.*, 1313–14.

claimed subject matter at the level of genus and even less than one that claimed abstract excluded subject matter, subject matter eligibility became a question of degree. It also confirms the remark in *Mayo* that ‘all inventions at some level embody, use, reflect, rest upon, or apply laws of nature, natural phenomena, or abstract ideas’ which means that at a certain level of generality all inventions include ineligible subject matter, whether subject matter is judged in terms of physicality or specificity.

By recognising that specific subject matter potentially incorporates abstract ideas, patent law created a situation where it had to work out to what extent abstract ideas could be protected or, to use the language of the Federal Circuit in *McRo*, what degree of risk was the court willing to accept in the granting of a patent? In adopting specificity as a touchstone for deciding subject matter eligibility, rather than answering what the Supreme Court in *Alice* described as a key question in this context, namely, what is needed to ‘transform the nature of the claim into a patent-eligible application’¹⁶⁹ patent law added a new question to the subject matter inquiry, namely where and how was the line to be drawn between a (non-patentable) abstract idea and an application of an idea that produces eligible subject matter?

While an appreciation of the problems that patent law created for itself in dealing with computer-related inventions that could not be made to look, feel, or smell like hardware is important for understanding some of the problems bedeviling patent law today, from my perspective, the most important change instigated by the decision to adopt specificity as a guide for deciding eligibility was that it uncoupled subject matter from its physical roots, that is, it dematerialised the subject matter. In this sense the decision to use specificity as a guide for eligibility allowed patent law to reconceptualise ‘the notion of invention ... not through the form of the machine or organism but through that of information and information processing’.¹⁷⁰ While this dematerialisation changed the way that patent law interacted with computer-related subject matter, it was not as significant as the changes that occurred as a result of the shift to structural formula in chemical subject matter. As we will see in the next three chapters, it was also very different to the way that patent law responded to the dematerialisation of biological subject matter. The key difference being in terms of how the law interacted with science and technology. Unlike the case with organic chemicals and biological inventions where the law consistently looked to science and technology to help it deal with new types of subject matter, the inherently divided nature of the information technology industry meant that the law was forced to develop its own way of dealing with the would-be subject matter. It was this, much more than the process of dematerialisation, that shaped the way that patent law has interacted with computer-related subject matter since the 1960s.

¹⁶⁹ *Alice Corporation v. CLS Bank International* 134 S. Ct. 2347 2355 (2014) (this is the second part of the 2-part test).

¹⁷⁰ Mario Biagioli, ‘Between Knowledge and Technology: Patenting Methods, Rethinking Materiality’ (2012) 22(3) *Anthropological Forum* 285, 286.