Intellectual Property Rights in the IT Sector – A bane or a boon

Chirag Tanna
B-72, Pereira Nagar No. 7,
Khopat, Thane (W) 400 601
Maharashtra, INDIA
chirag981@googlemail.com

***Abstract* — With emerging trends in the field of Information Technology, research and development (R&D) efforts are being employed to develop new innovations for enhancing existing systems. With the increasing R&D efforts and cutthroat competition, there is a requirement to understand the potential of Intellectual Property Rights, especially of patents, to gain competitive edge and reap benefits / incentives for the efforts made in that direction.**

THE EARLY TRAIL:

At its inception, the patent system dates back to the Greeks, as far back as 3rd century B.C.[1], the item of the monopoly by virtue of a 'patent' being a recipe. This evidence of granting monopoly provides a seed for the thought process that grant of monopoly was an honour and a reward for creating something 'new'.

Evolving inter-societal relations and anthropological advancements brought about the concept of trade. In England, this trade business was given an incentive in the form of an exclusive marketing right. Also, a reward system was coined for bringing in new products to the state, such reward system being the grant of monopolies to the first individuals who would import a new product into the state.

This effusion of knowledge may well be cited as growth of mankind in areas where their knowledge seemed deficient.

Centuries later, the process of making salt was granted a letters patent[2] by the Crown (England) for a royalty to be paid to the Crown for awarding such a right. With the honest and earnest intention of granting monopoly in exchange of some sort of royalty to the State as well as disclosing such new item / process brought about the fabric of the Patent System, as we know today.

Legislatures and legalities came into picture. Governing bodies dictated policies and jotted down requirements in favour of the public good; in order to set the Patent system in place.

The first US Patent[3] was granted in 1790 for the process of making potash, and since then the gigantic USPTO 'machine' has been credited with prosecuting and granting well over 7 million patents. The federal circuit well works in oiling and regulating this machine by providing judgements and arguments in order to streamline the system.

In this entire scheme of things, patents started encompassing various and all technologies of sciences from mechanical to electrical to electronic to chemical to pharmaceutical to business

[1] Stobbs, Gregory A., *SOFTWARE PATENTS*, Aspen Publishers, 2000 (at 3)

[2] *Id.* at 4

[3] Hernandez, Maria V., *United States Patent and Trademark Office*, 2001 See also http://www.uspto.gov/news/pr/2001/01-33.jsp

method to biomedical to telecommunication to networking to the Information Technology industry. Amidst all this brouhaha and setting and improving the various Patent systems in place, the more recent Information Technology patents, colloquially called as 'software patents', took intensive jolts; from the legislative point of view, from the developer point of view, from the point of view of benefits to SMEs, from the point of view of hindrance to corporations and / or individual developers. All in all, the greatest target were IPR practitioners (patent agents, patent attorneys, IPR counsel) who were touted to be lobbying for software patents by suggesting that their mere wordplay in respect of drafting patent specification (in exchange of hefty fees) provides protection for something which is more harmful than useful. In this paper, we aim to delineate these arguments in a simple step-by-step format, mainly to ward off any fear that the antagonists of the pro-software patent story may have propounded.

THE WEIRDLY INTERTWINED RUBRIC:

The driving logic for a hardware component or a plurality of sub-components in order to achieve a defined or visualized output is SOFTWARE. Phonetically speaking, the term 'software' channels one to think towards the guiding mechanism of hardware(s). It is this software, then that gives the hardware its logic to perform its functions as well as to interact with other hardware components. And it is this very logic or scheme or mechanism that we seek to protect by virtue of IT patents.

Drawing parallels from another field of engineering; the chemical engineering, patents have always been non-contentious in respect of process patents, for chemical processes. This chemical process is in every way an algorithm to arrive at a defined result. The steps of an algorithm which include inputs, enabling means, and subsequent outputs are similar to a chemical process which includes inputs, their reactions enabled by some means, and subsequent outputs. This argument steers the IT patent clear off the anti-IT patent propaganda.

Another contentious subject that most non-IT-patent aficionados put forth is that software is internationally protected by copyrights; then what do patents have to offer in software. What they fail to understand is that the structure of one code merely protects the code from being copied, per se. It is easy to design around existing codes to achieve the exact same result, thus defeating the purpose of the copyright. Even assuming the copyleft scenario, where an owner wants to divulge the invention into free domain, it would make more sense to have it patented just to establish the right to ownership before providing free access. This establishment serves as an important milestone for the creator and the owner.

THE ANGEL MONOLOGUE:

First Software Patent:

On 21 May 1962, a British patent application entitled "A Computer Arranged for the Automatic Solution of Linear Programming Problems"[4] was filed. The invention was concerned with efficient memory management for the

[4] Beresford, K (2000) *Patenting Software under the European Patent Convention*, London: Sweet & Maxwell, page 4. See also http://www2.warwick.ac.uk/fac/soc/law/elj/jilt/2003_1/kretschmer/

simplex algorithm, and may be implemented by purely software means. The patent was granted on August 17, 1966 and seems to be one of the first software patents.
However, it is known (arguably) that, the first patent for software was granted to S. Pal Asija[5] who was a programmer and a patent lawyer. He battle with the courts for 7 years before he was awarded a patent (U.S. Patent No. 4,270,182) for his software, 'SwiftAnswer'. This set up a precedent of sorts.

First Computer (Hardware):
The Atanasoff–Berry Computer (ABC) was the first electronic digital computing device. Conceived in 1937[6], the machine was not programmable, being designed only to solve systems of linear equations.

First Patented Computer (Hardware):
J. Presper Eckert and John Mauchly were the first to patent a digital computing device, the ENIAC[7]. ENIAC short for Electronic Numerical Integrator And Computer, was the first general-purpose electronic computer.

It is evident that hardware works without software, and the working of hardware is a tangible and visible functionality. On the other hand, the invisible functionality of the software brings it under the cannons of the anti-software-brigade. But the effects and results of this software are evidently felt. The causal effect of this software is actuated by the logic governing this software and hence, it is a laudable aspect to give credit to the creator of this logic (which drives the hardware) for improvements which are new and not obvious.

On the public front:
IT patents have assumed a nebulous form, even in the mind-frames of IT professionals. Prima facie, the single factor that drives people against patents in the Information Technology (IT) sector is fear; fear of the unknown. This fear embeds itself in the minds of inventors and persons serving the software industry. The term 'gridlocks' or 'patent-mines'[8] are techno-babble terminologies which these people fear. They fear that eventually any 'act' of procreating would result in treading into the muddy lurking waters of patent infringement, court cases, penalties, punishments and the entire routine. What they fail to understand is that the field of patents in a specific subject (a.k.a. patent-mines) is freely available as study and research material for each and every person, especially for ones skilled in the art to upgrade their knowledge and come to be at par with the very best in the industry and to understand the new frontiers of science. This exercise not only increases ones knowledge quotient, but helps to achieve levels of automation, levels of robustness, levels of quality and quantity, and the like betterment standards. And it is only right if then that they have to pay a royalty to the owner of the patent in order to use or duplicate the concept. More often than not, this entire exercise is what

[5] Ganapati, Priya, *This Day in Tech*, See also http://www.wired.com/thisdayintech/2009/05/dayintech_0526/
[6] Ralston, Anthony; Meek, Christopher, eds. (1976), Encyclopedia of Computer Science (second ed.), pp. 488–489, ISBN 0-88405-321-0
[7] "The ENIAC Story". Ftp.arl.mil. http://ftp.arl.mil/~mike/comphist/eniac-story.html. Retrieved 2008-09-22.
[8] L. Gordon Crovitz, *Patent Gridlock Suppresses Innovation*, Wall Street Journal, July 14, 2008, Page A15

stimulates the grey cells into one of the following two options:

1) Design around existing IP; or
2) Procreate new IP.

Eventually, upgrades happen, albeit at a price.

On the personal front:
The classical 'grapes are sour' philosophy seems to apply amongst some lobbyists opposing the IT patent regime. These are people who may have missed the bus in protecting their ideas. It is then only logical for such people to try to rig the system into believing that their bad can be overturned by making freely available what ought to have been their right assertion. This ignorance and / or bitterness can easily be overturned by appropriate management, as explained later.

Off the tripod that any IPR hoists its flag, the counts[9] are:

- Ethical Justification;
- Empirical Justification; and
- Economical Justification

On the count of 'Ethical' justification, it is imperative to justify the morally significant interest that the authors may have in controlling the disposition of the contents of their creations. Whether such 'holdback' (as the opponents may tag) hinders subsequent inventions is purely speculative. It is an insult to belittle the human mind which ought not to be bogged down by claiming such authorship. The ethical justification works because, on the counts of ethical obligation from the state to the author, it ensures that the content creators continue to devote their intelligence, time, labour, and money to continually innovate.

On the count of 'Empirical' justification, the classic argument which stands in favour of IPR rights is that, 'if one does not know whether a system 'as a whole' is good or bad, the safest 'policy conclusion' is to 'muddle through'. If we did not have a (software) patent system, it would be irresponsible to recommend instituting one. But since we have a (software) patent system for a long time, it would be irresponsible, on the basis of our present knowledge, to recommend abolishing it.'[10]

On the count of 'Economic' justification[11], theories proposed by Nelson and Mazzeloni cite the Invention – Inducement Theory, the Disclosure Theory, the Development & Commercialisation Theory, the Prospect Development Theory. All of these define the function of patents, namely, stating that:

1. The prospect of patent protection provides a motivation for useful invention; this model is called the *invention motivation* theory[12].
2. Patent protection for inventions may be needed to induce the investment required to develop and commercialize them; this model is called the *induce commercialization* theory[13].
3. Patents are awarded to induce inventors to disclose their inventions; this model is referred to as the *information disclosure* theory[14].
4. Patents may be needed to permit the orderly exploration of a broad

[9] Hope, J. *Open-Source Biotechnology*. Ph.D. Thesis, Australian National Univ. (2004).

[10] Machlup 1958: *An Economic Review of the Patent System*

[11] Hope, J. *Open-Source Biotechnology*. Ph.D. Thesis, Australian National Univ. (2004).

[12] *Id.*

[13] *Id.*

[14] *Id.*

prospect of inventions; this is called the *exploration control* theory[15].

On each of the above-mentioned counts of justification of IPR, it is unclear as to why 'IT patents' are a subject matter of belligerence. This belligerence of the free – world protagonists is aplenty. But what we wish to achieve with the patent system, is the attempt to piggy back on a procreator's (read 'inventor') intelligence, labour, time, and money.

Antagonists have been saying through the years, in the recent past, against the existence of patents in the IT industry that there is an epidemic of patent litigation in respect of allowance of software claims in the USA. However, this 'epidemic' of litigation is nothing but a forging process to shape the system into place. Every system is an evolutionary system, and this evolution happens as mankind progresses, as new problems come to the fore, as new solutions are sought, and betterment being thus achieved. The family tree of software patents traces its roots to 1840s[16]. However, as mentioned above in this article, the first 'patent' for a recipe, for all practical matters, is also a software patent (based on the algorithm of executing one step after another), albeit lacking a computational system for executing it.

In a first of sorts, the Supreme Court (USA) first ruled on a modern software patent in 1972, *re Gottshalk v. Benson*[17], The Benson patent claimed conversion of Binary-Coded Decimal numerals into pure Binary numbers. The claim 8 of this invention could entirely be done by pen and paper. Similarly, the step performed by the re-entrant shift register' of claim 2 could also, otherwise, be performed with pen and paper. The Supreme Court rejected the patentability of the information processing algorithm with "insignificant post-solution activity", thus defining non statutory subject matter.

Further ahead, the fate of software patents took a temporary turn for the worse in the 1970s, *re Parker v. Flook*[18], in respect of the subject matter of Flook's patent for updating the alarm limit (number) used in catalytic chemical conversion of hydracarbons. The cause of rejection of Flook's patent was on the grounds of novelty, that being, 'claimed method was a mathematical algorithm or formula, and further the Court opined that, "*Respondent's process is unpatentable S101 of the Patent Act, not because it contains a mathematical algorithm as one component, but because once that algorithm is assumed to be within prior art, the application, considered as a whole, contains no patentable invention.*"

A major milestone, which opened the door for software patents was the *Diamond v. Diehr* case[19]. Claim 1 recited a method of operating a rubber-molding press with the aid of a digital computer comprising a database and a variety of constants governing the equation:

$$ln\ v = CZ + x$$

The Patent Examiner & USPTO rejected the claims under 35 U.S.C. 101 that they were drawn to non-statutory subject matter under *Gottschalk v. Benson*. The Court of Customs and Patents Appeal

[15] *Id.*
[16] 56 U.S. 62, (1854)
[17] 409 U.S. 63 (1972)
[18] 437 U.S. 584 (1978)
[19] 450 U.S. 175 (1981)

disagreed[20] and reversed the decision, stating that it involved use of a computer and hence does not become non-statutory.

In perspective, the cases of *Diamond v. Diehr* & *Parker v. Flook*, can be differentiated, although both involve solving a mathematical formula and neither wholly pre-empts its mathematical formula. Yet, *Diehr* is patentable and *Flook* is not. It may be understood, hence, that:

- *Diehr* effects a change in state of a physical thing: a mold is opened.
- *Flook* effects a change in state of a non-physical thing.

And that sets the ground rules and foundation of conforming to (pro)claiming rights in respect of software patents.

Along the lines, further in time, according to USA Supreme Court, in respect of *Expanded Metal Co. v Bradford*[21], the citation reads *"A machine is a thing. A process is an act, or a mode of acting. The one is visible to the eye, - an object of perpetual observation. The other is a conception of the mind, - seen only by its effects when being executed or performed. Either may be the means of producing a useful result.*" This laid the plinth of the structure of validity of software patents.

THE DEVIL MONOLOGUE:

On the system front:

Most Patent Offices in the world are not equipped to understand and appreciate the art of IT patents. Understanding and appreciating this art borders on 2 factors:

1) Understanding the technology:
   The highest qualified patent examiners may be employed by the patent office. But each examiner may cater to a variety of technologies. E.g. 18933[22] patents are granted in the USA, which have the technology, 'XML'. Similarly, 6517[23] patent exist for 'video compression' in the USA. Here, we cannot have dedicated examiners to search only one respective technology.

2) Ability to appreciate and imagine the technology:
   Since, one examiner may cater to a plurality of allied technologies, at best, a range of imagination is at play. However, with a good draft and intelligent background reading, discussions during Examination Reports and their subsequent Replies should enable an examiner to understand and appreciate the claimed invention in respect of the prior art, and approve its grant.

On a similar scale, IT professionals, instead of running away from the educational exercise, should be able to grasp the invention and appreciate the inventor by restricting mere duplication.

One probable aberration in the patent regime vis-à-vis IT patents is the term for which the patent is granted. A grant of a term of 20 years may be futile in the IT patent scenario, as this term outlives the life of the technology itself.

THE BIG BANG CONCLUSION:

The clutter of flawed arguments is expansive. Mainly, because the

[20] 409 U.S. 63 (1972)
[21] 214 U.S. 366, 382 (1909)
[22] *USPTO search*
[23] *Id.*

audience, at large, to whom such arguments are presented are the masses who although may have a generic idea of the patent procedures and effects, but do not fully understand the scope, the need, the exitways, the protection parameters, in full detail, as much as a patent agent would.

Unless we claim a barter system world, one important aspect of any system is money. Why then is the money making mechanism either by patenting IT subject matters, by obtaining license and revenues, or by invoking the judicial system a bad thing?

An overhaul of sorts may be envisaged with respect to IT patent terms, wherein the term of the patent may be a short version in order to free the technology of any frivolous litigation at a time beyond its fruitful life.

One way to exercise diligence and control is to appoint IP managers in each company; one who tends and herds the IT professionals at all points. Typically, this IP manager may work in tandem with a Patent Agent / Attorney and perform the following tasks:

1) Chalk out the service areas of the company;
2) Identify these services areas from a patent-relevance point of view;
3) Provide patents to IT staff relating to identified subject matter;
4) Keep IT staff abreast of worldwide patent applications in respect of identified subject matter;
5) Understand the value of IP with respect to market value and company portfolio value;
6) Invest in protecting such IP;
7) Act upon propagating such IP publicly to generate revenue.

The whole idea, regarding IT patents, is to think of them as machines, too. In *re Alappat*[24], having subject matter as 'means for creating a smooth waveform display in a digital oscilloscope', the Federal Circuit declared the subject matter non-statutory, "*We have held that such programming creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to instructions from program software.*"

Now, since machines concerning mechanical inventions are the undisputed subject matter of patents, it is only logical to award machines concerning Information Technology a fair receipt within the realms of patent protection, especially taking cognizance of the change from industrial era of the 19th century to the Information Technology of the 20th century.

[24] 33 F.3d 1526 (1994)