TO INNOVATE OR NOT TO INNOVATE, THAT IS THE QUESTION:[1] THE FUNCTIONS, FAILURES, AND FOIBLES OF THE REWARD FUNCTION THEORY OF PATENT LAW IN RELATION TO COMPUTER SOFTWARE PLATFORMS

Seth A. Cohen[*]

Cite As: Seth A. Cohen, *To Innovate or Not to Innovate, That is the Question: The Functions, Failures, and Foibles of the Reward Function Theory of Patent Law in Relation to Computer Software Platforms*, 5 MICH. TELECOMM. TECH. L. REV. 1 (1999) *available at* <http://www.mttlr.org/volfive/cohen.pdf>

I. INTRODUCTION 2
II. TRADITIONAL FUNCTIONS OF THE PATENT SYSTEM 3
A. *The Reward Theory* 3
B. *Kitch's Prospect Theory* 5
III. THE DYNAMICS OF THE INNOVATION MARKET: REWARD FUNCTION FAILURE 7
A. *Ability to Attain Intellectual Property Rights* 8
B. *Competitive Advantages of First Movers* 9
IV. THE PROBLEMS OF REWARD FUNCTION FAILURE: HARMS TO RELATED PRODUCT INNOVATION 12
A. *Sub-Optimal Innovation* 12
B. *Reward Function Failure: IP-Dependent Markets Have Fewer Competitors* 14
C. *Consumer Welfare Implications* 15
V. WHY THE PROSPECT FUNCTION ANALYSIS DOES NOT REMEDY REWARD FUNCTION FAILURE AND WHY RIVALRY DOES 17
A. *Inadequacies of the Prospect Function* 17
B. *Solving Reward Function Failure: Rivalry in the Innovative Process* 20

1. With apologies to William Shakespeare.

* Seth A. Cohen, J.D., University of Pittsburgh School of Law; B.A. University of Maryland, is a Corporate and Technology Associate at Morris, Manning & Martin, LLP in Atlanta, Georgia. The author wishes to thank Professor Julie E. Cohen for her comments on earlier drafts and her invaluable insight into the law of intellectual property, which helped both inspire and motivate the production of this article.

VI. CURING REWARD FUNCTION FAILURE: THE SPECIAL PROBLEM OF COMPUTER SOFTWARE 23
A. *Technical Aspects of Computer Software* 23
B. *The Special Role of APIs and Software Platforms in the Application Market*.. 24
C. *Innovation Market Implications of Technical Aspects of Computer Software*.. 25
D. *Preventing Failure: Reevaluating Protection for Software Platforms* 26
1. Proprietary Protection: Determining the Scope............. 26
2. Defining the Proprietary Nature of Software Platforms 28
3. Potential Sui Generis Protection for Software Platforms 30
VII. CONCLUSION.. 33

I. INTRODUCTION

The patent system has traditionally been viewed as having two primary functions: the reward function and the prospect function. Although these theories do explain some behavior which results from the practical applications of the patent system, they also overlook some behavior of the patent system which indicates a failure of these functions. In order to properly prevent such failure, this paper proposes that the patent system adopt an orientation that will lead to increased innovative rivalry and competition. In Part I, using the computer operating system software market as an example, I propose a framework for reconceptualizing patent protection as it applies to software operating system platforms. Part II briefly examines both the classical and neoclassical reward function and prospect function theories.[2] Part III defines the innovation market and describes the market dynamics that create disincentives for innovation. These disincentives tend to limit the

2. In the context of patent law, the classical reward function theorists are those who use natural law as a basis of describing the function of intellectual property rights, and argue that patent rights serve as incentives for inventive activity. Although there were some early criticisms of the natural rights/incentives theories, the reward function became entrenched as the predominant view of patent rights. For a historical review, *see* EDITH TILTON PENROSE, THE ECONOMICS OF THE INTERNATIONAL PATENT SYSTEM (1951); Fritz Machlup & Edith Penrose, *The Patent Controversy in the Nineteenth Century*, 10 *J. Econ. Hist.* 1 (1950). Neoclassical reward theory relies on classical theory, but focuses on the type and means of granting rewards. Prospect theory largely developed as a critical response to reward theory. For a discussion of the various theories, *see* Samuel Oddi, *Un-Unified Economic Theory of Patents—The Not-Quite Holy-Grail*, 71 NOTRE DAME L. REV. 267 (1996).

number of competitors in innovation markets and create conditions which reduce the effectiveness of the reward incentives to the extent that the reward function fails in its entirety. Part IV examines the resulting harms of this failure and identifies how reward function failure affects product markets, which are dependent upon the reward function. Part V discusses why the prospect function does not address the problems related to reward function failure and the reasons that the promotion of innovative rivalry would alleviate some of the problems. Part VI uses the relationship between computer programs to illustrate the shortcomings of the patent system's inability to prevent the problems created by non-competitive innovation. This section also considers how the patent system might be better adjusted to prevent reward function failure. Lastly, Part VI also proposes a series of alternative frameworks for creating a competitively oriented approach to the application of the patent system in the case of computer software platforms.

II. TRADITIONAL FUNCTIONS OF THE PATENT SYSTEM

The genesis of the economic analysis of the patent system is a result of the U.S. Constitution's initial economic assertion related to the promotion of the "Progress of Science and useful Arts."[3] Economists and legal scholars, when examining patent scope and innovative improvements in light of this assertion, have long considered the patent system as a framework serving two primary functions: the prospect function and the reward function.[4] Although theorists have used total innovative output as a gauge of the benefits created by the patent system, much of the research in these areas has used economics, specifically allocative efficiency, as a measuring stick in determining the patent system's preference over alternative frameworks that provide protection through trade secrecy.[5]

A. *The Reward Theory*

The earliest and most widely accepted theory related to the patent system is that it serves as a means of providing rewards to those who develop and disclose ideas and products in the areas of science and

3. *See* U. S. CONST. art I, §§ 8, cl. 8; *see also* Kenneth W. Dam, *The Economic Underpinnings of Patent Law*, 23 J. LEGAL STUD. 247, 248 (1994).

4. *See generally*, Mark F. Grady & Jay I. Alexander, *Patent Law and Rent Dissipation*, 78 VA. L. REV. 305, 310–316 (1992).

5. *See, e.g.*, Ward S. Bowman, Jr., Patent and Antitrust Law: A Legal and Economic Appraisal (1973); Edmund W. Kitch, *The Nature and Function of the Patent System*, 20 J.L. & ECON. 265 (1977).

technology.[6] Much of the basis for the patent system has been based on this theory. The exclusive ownership rights are incentives for inventors to disclose their inventions rather than withhold them as trade secrets.[7] Both monopoly rights, as well as laws to protect the infringement of those rights, allow the inventor to recoup the costs of research and development as well as earn a fair amount of remuneration for the disclosure of the idea.[8] Ultimately, the reward function utilizes legal devices to create incentives for technological innovation and disclosure by means of granting rewards which are otherwise unrecognizable by the innovator. These rewards are also essentially a *quid pro quo* exchange for the public benefit based on both the innovation and disclosure of technology.

Building on the general principle of the reward-oriented function of patents, the reward theory has been split into two factions: weak reward theory and strong reward theory.[9] The key difference between the factions is that the supporters of the weak theory view the reward function broadly and view its function as a general inventive stimulant.[10] In contrast, supporters of the strong theory support patent rights for only those inventions which are "patent-induced."[11] A third set of theorists have proposed that rent dissipation is the proper theoretical framework for analyzing the way the reward function provides incentives for innovation.[12]

---

6. *See* Grady & Alexander, *supra* note 4, at 310. *See also* Senate Subcomm. On Patents, Trademarks and Copyrights, Comm. on the Judiciary, *85th Cong., 2d Sess.*, *An Economic Review of the Patent System* (Comm. Print 1958, Study No. 15, Fritz Machlup).

7. *See* Dam *supra* note 3, at 270–71.

8. *See id.* at 247.

9. *See* Grady & Alexander *supra* note 4, at 312–13.

10. *See* Bowman, *supra* note 5, at 21–28.

11. *See* A. Samuel Oddi, *Beyond Obviousness: Inventive Protection in the Twenty-First Century*, 38 AM. U. L. REV. 1097 (1989); *see also* Frederic M. Scherer, Industrial Market Structure and Economic Performance 440–58 (2d ed. 1980).

12. *See* Grady & Alexander, *supra* note 4, at 308–310; 316–21. Rent dissipation theory holds that the difference between an innovation's development costs and the price society will pay for the innovation is a form of rent, an economic reward which is created by the innovation. Proponents of the theory argue that the rent should be awarded to the inventor as a property right, in order to provide incentives for innovation. Rent dissipation occurs when conduct reduces the value of the rent which is awarded, thus limiting the incentives. Dissipation may occur by redundant innovative development, redundant development of improvements, and excessive investment in protecting the secrecy of the innovation. These forms of dissipation adversely affect the incentives for innovation, thus creating the need to limit rent dissipating activity. *See id.*

B. *Kitch's Prospect Theory*

Edmund Kitch has made the seminal argument for the use of the prospect function in determining the patent system's development. Professor Kitch argues that a major function of the patent system is to maximize the efficiency of technological innovation.[13] Kitch identifies patents as "prospects," and argues that the general attributes of "prospecting" form the basis of inventive behavior.[14] Analogizing patents to mineral claims, Kitch reviews the policy implications of treating patents as prospects and presents numerous justifications for the supremacy of the prospect approach.[15]

In his analogy, Kitch argues that the establishment of a patent claim is similar to staking out a mineralization prospect. Both mineralization claims and patents are limited based on breadth and time and are based on a priority system that rewards the first to discover and register the claim.[16] Additionally, commercial significance is not required for either an enforceable mineralization claim or for a valid patent. Although there are systemic similarities, the key analogous characteristics of the mineralization claim and patent systems are the incentives to stake and later exploit the prospects and the exclusive rights which attach to valid claims. Once these exclusive rights are granted, the holder can develop the prospect without fear that other individuals will exploit the claim. Others are on notice to claim other mineralizations or patents which do not overlap or infringe upon the original stakeholder. Furthermore, both systems create a uniform framework for the identification and exploitation of property rights, as well as restrictions on abusive use of such rights.

Analogizing the mineralization claim system to the patent system led Kitch to establish several conclusions. Most significantly, identifying a patent as a prospect establishes a future right to develop the technological prospect. Kitch argues that unification of control over research and development of the particular patent prospect provides the most efficient means for developing the technology.[17] Multiple and redundant inventive efforts may result in wasteful resource usage or infringing developments.[18] The disclosure elements of the patent system also serve to prevent wasteful resource allocation. Under Kitch's model, the establishment of patent claims does not merely provide information

13. *See* Kitch, *supra* note 5, at 245–80.
14. *Id*. at 267.
15. *See id*.
16. *See id*. at 271–75.
17. *See id*. at 285–86.
18. *See id*.

to the public; it also creates a mechanism to direct the allocation of research and development resources away from patented technology and towards pioneer innovations.

Kitch concludes that when an inventor presents a broad scale invention, the patent scope should be granted broadly in order to allow for the coordination of research and development. This coordination will prevent inefficient innovation and redirect inventive resources away from competitive uses. While some scholars have criticized this conclusion,[19] it has been cited as a significant argument against the development of a competitive approach to technological innovation.[20]

The prospect function analysis is consistent with other allocative efficiency-based theories of the law.[21] Nonetheless, the prospect function analysis fails to account for other efficiency models based on non-monetizable qualities such as social efficiencies.[22] Whereas competitive innovation may not be efficient in one economic model, it may be maximally efficient in other models.[23] Therefore, modeling the patent system's functionality on a particular efficiency model incorporates certain value judgements related to the choice of that model. Kitch, by modeling the patent system on monetizable types of allocative efficiencies, expressly rejects social efficiencies and other alternative welfare models. Thus, his analysis fails to account for other economic approaches and, consequently, may deter the creation of beneficial efficiencies which are created by these alternative models.[24]

Despite the concerns related to Kitch's economic modeling, Kitch's theory has significant implications related to the dynamics of the inventive pool. The inventive pool, for the purposes of this paper, is the group of innovators who are primarily focused on pioneering technological innovation as a result of their endeavors in a particular technological

19. *See* Grady & Alexander, *supra* note 4, at 314–16. *See also infra* notes 84–87 and accompanying text.

20. *See* Dam, *supra* note 3, at 264.

21. This efficiency-based approach largely sprouted from the writings of the Chicago School of economics. *See* John P. Henderson, *The History of Thought in the Development of the Chicago Paradigm*, The Chicago School of Political Economy 341 (Warren J. Samuels ed., 1976); Warren J. Samuels, *The Chicago School of Political Economy: A Constructive Critique*, The Chicago School of Political Economy 1 (Warren J. Samuels ed., 1976).

22. *See* Ezra J. Mishan, *The Folklore of the Market: An Inquiry into the Economic Doctrines of the Chicago School*, The Chicago School of Political Economy 95 (Warren J. Samuels ed., 1976).

23. *See id*. at 100–105.

24. *See generally* Mark R. Tool, Essays in Social Value Theory 33–51 (1986). This concern has been shared by several other economic commentators. *See, e.g.*, Howard F. Chang, *Patent Scope, Antitrust Policy, and Cumulative Invention*, 26 RAND J. ECON 34 (1995); Suzanne Scotchmer, *Standing on the Shoulders of Giants: Cumulative Research and the Patent Law*, 5 J. ECON. PERSP. 29 (1991).

area.[25] Innovation in this context is considered the search, discovery, development, and implementation of new or adapted technologies.[26] Despite Kitch's assertions regarding the benefits created by efficiencies in the innovation process, it has been argued that Kitch's efficiency-based approach may be appropriate if the inventive pool is significant enough to withstand a decrease in the protection of incentive-based doctrines.[27] If the inventive pool is small, Kitch's efficiency based doctrines may further decrease the pool by concentrating the rewards on those who hold the original broad or pioneer patents.[28]

III. THE DYNAMICS OF THE INNOVATION MARKET: REWARD FUNCTION FAILURE

The reward and prospect function theories do not explain all behavior in the area of technological innovation. Both theories fail to recognize that other innovation market dynamics create disincentives for pioneer inventive activity in markets which already have a protected invention. These disincentives stem from the interaction of commercial competitive advantages with intellectual property rights. When the competitive advantages of the initial inventor are greater than the rewards which are granted to a subsequent pioneer inventor (second-comer)[29] by the patent system, the potential for optimal innovation is diminished. I term the existence of these innovative disincentives reward function failure.

25. Although improvements of technology are also products of the inventive pool, this paper does not address the economics of improvement incentives, nor their market implications. Unlike pioneer inventions, improvements are predicated on a pre-developed invention and have different incentives than pioneer efforts. Consequently, reviewing both of the inventive efforts together would fail to properly distinguish their independent economic significance. For an overview of the issues related to inventive improvements, see Mark A. Lemley, *The Economics of Improvement in Intellectual Property Law,* 75 TEX. L. REV. 989 (1997).

26. *See* William Kingston, *The Thesis Chapters*, Direct Protection of Innovation 1, 2 (William Kingston ed., 1987).

27. *See* Jerry Green & Suzanne Scotchmer, *On the Division of Profit in Sequential Innovation,* 26 RAND J. ECON. 20 (1995).

28. *See* Scotchmer, *supra* note 24, at 31–35.

29. In the context of this paper, a "second-comer" is a subsequent pioneer innovator and not an improver. Whereas an improver bases his inventive efforts on developing a pre-existing invention, a second-comer bases his efforts on the development of an alternative invention that competes with other pioneer inventions. Although a second-comer may use pre-existing scientific developments, the key distinction between a second-comer and improver is the relative purpose of the inventive efforts. Therefore, if the innovation is meant to improve upon a prior product rather than supplant it altogether, that innovation is not the work of a second-comer.

In reward function failure, potential second-comers are discouraged from allocating inventive and financial resources towards a particular innovative goal, even though competitive development may create various market and consumer benefits. Such failure occurs in two contexts: 1) the limitation of the second-comer's ability to attain intellectual property monopoly rights by legal doctrines, and 2) the second-comer's limitation in gaining competitive advantages in the innovation market.

A. *Ability to Attain Intellectual Property Rights*

In apparent contradiction to the original intent, factors which discourage innovation are fundamental elements of the intellectual property rights granted by the government. Rights granted by the patent system provide both incentives and rewards for inventive efforts. However, in order to provide such incentives and rewards, the legal rights must provide protection from second-comers who are not deserving of their own legal protection. Nonetheless, both the length of the rights and the breadth of the rights may have unintended consequences which are different from their protective intent.[30] Rather than provide incentives for inventors to research and develop inventions diligently in one specific area so as to make inventive improvements or alternative uses, contemporary patent law serves to make potential inventors wary of innovation in an area in which patent rights have already been assigned. Although these second-comers may be able to develop and innovate products or improvements which satisfy the requirements of the Patent Act, some patent law doctrines may raise enough uncertainty to retard the inventors' efforts.

The issue of patent scope is central to the problem of inventive retardation because the determination of scope inherently affects the number of inventions which may be found to be infringing.[31] Depending on the breadth of the patent grant, inventors cannot sufficiently rely on the certainty that their innovation will be recognized in its entirety.[32] Additionally, the doctrine of equivalents adds uncertainty to patent scope by expanding infringement beyond the literal meaning.[33] Although the doctrine of equivalents prevents second-comers from usurping rights of the patent holder, it is equally possible that second-comers are exces-

30. *See* Robert P. Merges & Richard R. Nelson, *On the Complex Economics of Patent Scope*, 90 COLUM. L. REV. 839, 868–871 (1990). For a discussion of how some unintended consequences are manifested in relation to patenting behavior, see Josh Lerner, *Patenting in the Shadow of Competitors*, 38 J.L. & ECON. 463 (1995).

31. *See* Merges & Nelson, *supra* note 30, at 882.

32. *See id.*

33. *See* Graver Tank & Mfg. Co. v. Linde Air Prods. Co., 339 U.S. 605 (1950); Warner-Jenkinson Co. v. Hilton Davis Chem. Co., 520 U.S. 17 (1997).

sively restricted in their attempts to attain certain legal rights in new technologies.[34] Therefore, both the fundamentals of patent scope and the definitions of infringement may discourage the inventor's incentive for continuing development and research of a discrete invention in a discrete technological area in which pioneer patents have already been granted.[35]

B. *Competitive Advantages of First Movers*

Perhaps the most significant deterrent to second-comer innovation is the competitive benefits which accrue to first-mover innovators. These benefits, which are not without redeeming importance, nonetheless create advantages that are not easily duplicated for second-comers. In fact, certain competitive benefits grant advantages which even the patent system may not grant, especially in terms of time-related benefits and mechanisms for market control. The two most significant competitive benefits are first-mover advantages and compatibility/network externalities.

First-mover advantages are established, not by being first to invent, but by being first to successfully introduce the product to market.[36] Therefore, first mover-advantages do not accrue based purely on innovative efficiency; they accumulate as the result of a combination of research and development advantages and historical market reactions.[37]

The initial advantages that result from early exploitation involve learning benefits.[38] These learning effects, which have the impact of establishing cost advantages, are primarily relevant to creating research

---

34. *See* Merges & Nelson, *supra* note 30, at 853–60.

35. Professors Merges and Nelson term this type of inventive behavior as the "discrete invention model." They write:

> [The discrete invention model] assumes that an invention is discrete and well-defined, created through the inventors insight and hard work. In the standard discussions it may be recognized that the original invention can be improved, or even that improvement of complementary advances may need to be made if the invention is to be of much use. The basic invention may be amenable to tailoring for different uses or customers. But it is implicit that the invention does not point the way to wide ranging subsequent technical advances. It does not define any broad prospect.

See Merges & Nelson, *supra* note 30, at 880.

36. *See* W. Brian Arthur, *Competing Technologies, Increasing Returns, and Lock-In by Historical Events*, 99 ECON J. 116, 123 (1989).

37. *See id*. at 116.

38. *See* William E. Cohen, *Competition and Foreclosure in the Context of Installed Base and Compatibility Effects*, 64 ANTITRUST L.J. 535, 537 (1996) (citing A. Michael Spence, *Competition, Entry, and Antitrust Policy*, in Strategy Predation, and Antitrust Analysis 45, 65–66 (Steven C. Salop ed., 1981).

and development advantages for future innovations.[39] The sooner an inventor enters the learning curve, the less pronounced the curve will be for future innovative developments.[40] Furthermore, although the learning advantages may vary in degree, the first-mover gains a sizable advantage when the learning curve is moderate.[41]

Although the learning benefits may provide first-movers with some competitive advantage, these advantages are more damaging to innovation incentives when they are supplemented by early market introduction and exploitation. Early exploitation of a product with no marketable substitute and the subsequent growth of an installed base creates competitive advantages which shape later development.[42] Because of the lack of substitutability, the market is likely to adopt the first invention more readily than it would if there were competitive products. This market adoption creates switching costs which may, if too high, effectively lock-in consumers to the installed base.[43]

The problem with the installed base from the second-comer's perspective relates to establishing market adoption of the substitute goods. Foremost, if the installed base is significant in size, the lesser likelihood of a successful market entry by a second innovation might deter a second-comer from focusing on developing a rivalrous product. Although a second-comer may be potentially successful in introducing a discrete invention which is substitutable to that of the first-mover, it is only in rare cases where a second product successfully overthrows a large installed base.[44] Therefore, second-comers are likely to be deterred from innovating on the scale of the original developer. Follow-up research and supplemental improvement become the inventive goal, rather than the development of pioneer research and discrete substitute product innovation.

The second competitive advantage which deters second-comer rivalrous innovation is the system of network externalities and compatibility effects which attend to a large installed base, especially in

39. *See* Cohen, *supra* note 38; *Cf.* Spence, *supra* note 38, at 537.

40. *See* Cohen *supra* note 38, at 537–38.

41. *See id*.

42. *See id*. at 538.

43. *See generally* Joseph Farrell & Carl Shapiro, *Dynamic Competition with Switching Costs*, 19 RAND J. ECON. 123 (1988); Michael L. Katz. & Carl Shapiro, *Systems Competition and Network Effects*, J. ECON. PERSP., Spring 1994 at 93.

44. An example of this is the Betamax/VHS competition in the video-taping machine market. *See* S.J. Liebowitz & Stephen E. Margolis, *Should Technology Choice Be a Concern of Antitrust Policy?*, 9 HARV. J.L. & TECH. 283, 314–316 (1996) (hereinafter referred to as *Technology Choice*).

computer and communications technology.[45] Network externalities, as defined by Michael Katz and Carl Shapiro, are "the utility that a given user derives from the good depend[ent] upon the number of other users who are in the same 'network.'"[46] These benefits are usually supplemental to installed base advantages, but the two often forge a symbiotic relationship. The greater the installed base, the greater likelihood that the network externalities will attract more users to the installed base. Network externalities are not only limited to positive effects, but may also generate negative or pecuniary effects.[47] For a second-comer innovator, the introduction of a rival discrete innovation (which is not an improvement, but rather a substitutable innovation) will likely not produce sufficient network externalities to attract new users. Thus, the first innovation to market has significant advantages in expanding the externalities past the point of competitive vulnerability.

While network externalities do not discourage innovation entirely, their existence does affect the way second-comers identify their paths of innovation. The relevancy of path dependence theory in the technical innovation is based on the principle that small-scale advantages of a particular technology can create substantial influences on the market dynamics of that technology.[48] The theory holds, in the case of technological innovation, that a substitutable rival does not easily replace a technology with a large installed base.[49]

In order for network benefits to flow through to second-comer innovations, there must be a degree of compatibility which allows the second-comer's innovation to complement, rather than supplant, the innovation upon which the installed base is predicated.[50] As a consequence of this necessary compatibility, a second-comer will likely be

45. Technologies which are prevalent in network industries such as telecommunications and computer related industries typically reflect the networks externalities. Although there are numerous technologies which don't have high network externalities, these technologies are usually the basis for products which do not involve product compatibility components. *See* S.J. Liebowitz & Stephen E. Margolis, *Network Externality: An Uncommon Tragedy*, J. ECON. PERSP., Spring 1994 at 135. *See also* Dennis W. Carlton & J. Mark Klamer, *The Need for Coordination Among Firms, with Special Reference to Network Industries*, 50 U. CHI. L. REV. 446, 450 (1983).

46. Michael L. Katz & Carl Shapiro, *Network Externalities, Competition, and Compatibility*, 75 AM. ECON. REV. 424, at 424 (1985).

47. *See generally*, S.J. Liebowitz & Stephen E. Margolis, *Are Network Externalities a New Source of Market Failure?*, 17 RES. L. & ECON. 1 (1995). *Cf.* Liebowitz & Margolis, *Technology Choice*, *supra* note 44, at 283, 287.

48. *See* Liebowitz & Margolis, *Technology Choice*, *supra* note 44, at 288.

49. *See id*. at 291.

50. *See* Cohen, *supra* note 38, at 551.

deterred from developing a non-compatible equivalent product.[51] This discouragement from innovation may restrain the introduction of innovations which may, on their own merits, be as equally marketable as the first-mover's product was at the time of its introduction or even after improvement.[52] Therefore, it is arguable that the first-mover's mastery of the product market's relevant externalities may deter an innovator from choosing such a product to innovate.

IV. The Problems of Reward Function Failure: Harms to Related Product Innovation

Although I have identified why certain legal, economic and market factors deter entrants from competitively innovating similar products, it is important to understand how these disincentives are injurious to the development of primary and secondary product markets.[53] The injury manifests in three different ways: 1) products are introduced at a stage of innovation which creates a sub-optimal product, 2) primary product markets which are dependent on intellectual property rights have few large competitors who tend to have significant market power in their product market, and 3) consumer welfare is harmed by limited choice in the primary product markets.

A. *Sub-Optimal Innovation*

In markets in which there is little or no competition, there is a tendency to innovate sub-optimally.[54] Consequently, the product development may occur inefficiently once there is no threat of improved

---

51. *See id*. *See also* Richard J. Gilbert, *Symposium on Compatibility: Incentives and Market Structure*, 40 J. Indus. Econ. 1 (1992); Carmen Matutes & Pierre Regibeau, *"Mix and Match": Product Compatibility Without Network Externalities*, 19 Rand J. Econ. 221 (1988).

52. *See* Liebowitz & Margolis, *Technology Choice*, *supra* note 44, at 289, 307. The reluctance to innovate may create adverse consequences which result from path dependence. It has been argued that path dependence is created when one technological product's advantages in a particular market has significant ramifications for the development of the economies of that market. *See* Arthur, *supra* note 36, at 126; *See generally*, S.J. Liebowitz & Stephen E. Margolis, *Path Dependence, Lock-In, and History*, 11 J.L. Econ. & Org. 205 (1995).

53. In the context of this discussion, primary products are those technological developments which are used to enable other compatible products or technologies. A primary product may possess some individual functionality, but is best thought of as an enabling product. Secondary products are those which are dependent upon a primary product for implementation or execution. Without a primary product, secondary products would have no independent use.

54. *See* Merges & Nelson, *supra* note 30, at 873.

substitutable products.[55] Similar concerns exist in relation to product innovation prior to market introduction.[56] Nonetheless, unlike the improvement market in which there is already a baseline product, the members of the inventive pool do not have a baseline product upon which to base their innovation.[57] As mentioned above, it is possible, because of the first-mover advantages, that an inventor will attempt to establish legal rights to his product and introduce his product to market prior to the optimal technological innovation, or optimally commercial innovation. Although Professor Kitch has argued that this prospect function is the most efficient way to analyze the patent system, many notable scholars disagree.[58]

A negative consequence of the prospect analysis is that a product which is sub-optimal usually gains enough competitive advantages to deter the innovation of superior substitute products.[59] Therefore, the prospects which may have the strongest market presence may also be the products which bring the least-optimal effectiveness or use. A greater number of competing innovators would arguably force the most optimal innovation of the baseline product, upon which successive competitive efforts would create optimal improvements. Nonetheless, without such rivalry in the innovative process, inventors are likely to attempt to commercialize innovations as early as possible, believing that the first-mover advantages will compensate for any disadvantages that a sub-optimal product may create.[60]

---

55. *See id*. at 874–75.

56. *See* Janusz A. Ordover, *Economic Foundations and Considerations in Protecting Industrial and Intellectual Property*, 53 ANTITRUST L.J. 503, 510 (1985).

57. Although members of the inventive pool possess baseline scientific information and know-how this information is functionally different from the information attendant to a baseline product. Although the aggregate of the scientific information may help the innovator in the development of the invention, the information only creates a scientific framework for the development of the innovations as opposed to the design framework which is provided by a baseline product.

58. *See infra* notes 84–87 and accompanying text.

59. *See supra* note 52. Professors Liebowitz and Margolis refer to this market occurrence as "getting stuck." *See Technology Choice*, *supra* note 44, at 309. One of the most well known examples of locked-in sub-optimal innovation is the QWERTY keyboard. *See* Paul A. David, *Clio and the Economics of QWERTY*, 75 AM. ECON. REV. 332 (1985). *But see Technology Choice*, *supra* note 44, at 312; S. J. Liebowitz & Stephen E. Margolis, *The Fable of the Keys*, 33 J. L. ECON. 1 (1990) (arguing that it is possible to get 'unstuck').

60. *See* Richard J. Gilbert & David Newbery, *Pre-emptive Patenting and the Persistence of Monopoly*, 72 AM. ECON. REV. 514, 515 (1982).

B. *Reward Function Failure: IP -Dependent Markets Have Fewer Competitors*

The lack of competitors in certain IP-dependent markets results from the deterrence of innovators to develop substitute primary products.[61] Therefore, instead of developing rival primary products, innovators choose to develop secondary products that are compatible to the unrivaled primary products.[62] The secondary market of compatible products may be competitive, while leaving a relatively non-competitive primary product market. The increase in size of the compatible secondary product market has the effect of reinforcing the primary product's dominance and the primary product's dominance has the effect of redirecting inventive resources to the development of compatible secondary products.[63] I term this redirection of inventive resources "inventive tipping."

Inventive tipping is similar to consumer tipping, which occurs as a result of network externalities. Whereas network externality based tipping occurs in the dynamics of the user/consumer market, inventive tipping occurs in the innovation market. Tipping occurs generally when a baseline product (or in this context, a primary product) has a large installed base which creates significant network externalities.[64] These externalities/benefits become so great that the use of the product is widely accepted based on the product's inherent use, as well as the sum of its externalities. Essentially, the market has *tipped over* to the side of the product with the greatest externalities.[65] Inventive tipping occurs in a similar manner, with the exception that acceptance of the product and its network benefits are examined from the inventor/supplier side, as opposed to the user/consumer side.[66] For example, there comes a point where a primary product is sufficiently adopted by the market to reasonably indicate that it will continue to have a significant market presence.[67] Once inventors or developers determine that the baseline product is the more accepted product (and the one most likely to accrue network benefits), they will tailor their products to be compatible with

61. *See* Ordover, *supra* note 56, at 510–11.
62. *See* Cohen, *supra* note 38, at 550.
63. *See* Mark A. Lemley & David McGowan, *Legal Implications of Network Economic Effects*, 86 CAL. L. REV. 479 (1998).
64. *See id*. at 496–97.
65. *See id*.
66. *See* David McGowan, *Regulating Competition in the Information Age: Computer Software as an Essential Facility Under the Sherman Act*, 18 HASTINGS COMM. & ENT. L.J. 771, 841–43 (1996).
67. *See* Cohen, *supra* note 38, at 541–42.

that product, essentially adopting the product as a standard.[68] An example of such tipping can be found in the production of videotapes. When tape providers had readily determined that VHS recorders had sufficiently overtaken the market from rival Betamax recorders, producers sold videos in VHS format.[69] This tipping, and the consequent reduction in sales of Betamax videos, helped to further increase sales of VHS recorders and strengthen the VHS installed base.[70]

Although the inventors/producers are the ones who effectuate the supply-side dynamics of the inventive tipping, it is closely related to demand-side causation.[71] Compatibility is a characteristic of technology products which affects consumer choice and helps define the development of markets. Consumers are less likely to demand non-compatible goods when there is a substitution compatible good.[72] To respond to this demand, inventors and producers allocate more of their resources to satisfy this consumer demand. Although non-compatible secondary goods may be optimally developed and designed, their incompatibility may effectively negate any consumer desirability.[73]

The sum result of this inventive tipping and compatibility demand is that first-mover innovations can narrow product development simply by establishing an installed base of their initial primary product innovation. Even if second-comers attempt to innovate around a first-mover's product, the second-comer may find that the ancillary compatible goods markets which developed around the first-movers baseline good have the effect of suffocating the competitive ability of the second-comer's product. Secondary market developers further strengthen the installed base of the baseline product and establish a new set of compatibility-based network externalities.

C. *Consumer Welfare Implications*

A third problem related to non-rivalrous innovation is the impact on consumer welfare. Apart from the economic efficiencies, the issues pertaining to the monopoly powers of intellectual property rights holders and the attendant concerns of optimal patent life, consumers may suffer as a result of 1) sub-optimal development of the useful sciences, and 2) monopolistic control of the development of the useful sciences.

---

68. *See id.* S*ee also*, David Friedman, *Standards as Intellectual Property: An Economic Approach*, 19 U. DAYTON L. REV. 1109, 1119–29 (1994).

69. *See* Liebowitz & Margolis, *Technology Choice*, *supra* note 44, at 314–16.

70. *See id*.

71. *See id* at 295.

72. *See id*.

73. *See id*. *See also* Katz & Shapiro, *supra* note 46, at 424.

Although it has been argued that intellectual property rights should be granted only for those innovations which appear to be optimal, the Patent Act only demands novelty, utility and nonobviousness.[74] The Constitution only grants power to Congress to promote the "useful Arts," not only the *most* useful.[75] Nonetheless, the underlying goal of the clause is to maximize the public welfare by allowing Congress to establish a system that will encourage innovations to be shared with the public.[76] For example, the Patent Act already sets a bar that the innovation must be useful.[77] Although this requirement has been severely weakened, it is evidence of one of the core values of the intellectual property system; useful developments are favored over frivolous and non-useful developments.

An intellectual property system that encourages sub-optimal innovation and discourages individuals from innovating is inconsistent with the goal of promoting useful sciences generally. Although the patent system does reward innovation, it rewards innovation only up to a point, and then, as discussed in the above sections, discourages further innovation that may in fact be optimal. Consequently, the systemic disincentives are contrary to both the letter and the spirit of the Constitution and fail to protect the public welfare as it was originally construed.

Another implication for consumer welfare is the lack of competitive choice that results from the innovation disincentives which limit the size of the primary product market.[78] Assuming that a first-mover's installed base is significant enough to establish inventive tipping, a consumer may find that the differential between the externalities of the products gives her no real choice in determining which technology she will use.[79] Rather than choosing among competing substitutable products within the same field of technology, she may find the need to choose between very different technologies because no innovator developed an intermediate substitute product.[80] Furthermore, the same consumer may find

74. *See* 35 U.S.C. §§ 101–103 (1994). *See also* Dam, *supra* note 3, at 257–261.

75. U. S. Const. art. I, § 8, cl. 8.

76. *See* Margaret Chon, *Postmodern "Progress": Reconsidering the Copyright and Patent Power*, 43 DEPAUL L. REV. 97, 134–44 (1993).

77. *See* 35 U.S.C. § 101 (1994).

78. *See supra* notes 63 to 73 and accompanying text. *Cf.* Joseph F. Brodley, *The Economic Goals of Antitrust: Efficiency, Consumer Welfare, and Technological Progress*, 62 N.Y.U. L. REV. 1020 (1987) (consumer choice in the secondary product market will likely not be limited if inventive tipping has occurred).

79. *See* Michael L. Katz & Carl Shapiro, *Product Introduction with Network Externalities*, 40 J. INDUS. ECON 55, (1992) [hereinafter *Product Introduction*].

80. *See id. But see* Liebowitz & Margolis, *Technology Choice, supra* note 44, at 317 (arguing that an apparently inefficient alternative selection process may still be efficient

that she has no substantial choice because the secondary compatibility markets make the choice of the non-dominant technology too costly.[81] A system which fails to encourage competitive innovation may result in protecting innovators while passing along limited and restrained choice in the use of such useful sciences, thus harming consumer welfare.[82]

As the above assessment of the problems which result from reward function failure indicates, the disincentives to innovate competing primary products has the effect of limiting technology and market development and harms consumer welfare. Although these problems have a significant impact on the dynamics of the inventive pool, and further exacerbate reward function failure, both the disincentives and resulting problems are ignored by the prospect function theorists.

V. WHY THE PROSPECT FUNCTION ANALYSIS DOES NOT REMEDY REWARD FUNCTION FAILURE AND WHY RIVALRY DOES

A. *Inadequacies of the Prospect Function*

The prospect theory has already been criticized as understating the benefits of competitive rivalry. Professors Merges and Nelson have criticized Kitch's prospect theory, arguing empirical evidence suggests that competitive development of technological prospects is preferable to coordinated development.[83] They address the limitations of Kitch's mining model analogy and argue that Kitch's model ignores the tendency of prospectors to exploit their prospects inadequately, as well as the likelihood that rightholders will sub-optimally develop potential improvements.[84] Additionally, they suggest that both the legal and theoretical limitations which are created by property rights also undermine the effectiveness of the prospect theory. In this argument, Merges and Nelson refute Kitch's expansive view of property rights.[85] Doubting that unrivaled rightholders would exploit their rights as actively and efficiently as they would in a competitive environment, they are wary of expanding the property rights and endorse rivalry in the innovation process. Additionally, unlike Kitch, they provide historically-based empirical evidence which supports their theory.[86]

when considering consumer choice); Joseph Farrell & Garth Soloner, *Standardization & Variety*, 20 ECON. LETTERS 71 (1986).

81. *See Product Introduction*, *supra* note 79, at 72–74.
82. *See* Brodley, *supra* note 78, at 1032–35.
83. *See* Merges & Nelson, *supra* note 30, at 872.
84. *See id*.
85. *See id*. at 871–75.
86. *See id*. at 884–915.

In addition to the limitations of the Kitch model stated by Professors Merges and Nelson, Professor Kitch's theory is also limited in that it does not anticipate the interrelation of several discrete prospects.[87] Kitch's model anticipates that each prospect, like a mineralization, is a singular and discrete claim that can be exploited without any interrelation to a separate discrete claim.[88] This analysis breaks down when one prospect is relied upon by other prospects for purposes of compatibility and interoperability. Whereas Kitch anticipates that the prospects are primarily exploited in a manner which improves the original broad claim, he fails to foresee the impacts the exploitation of that prospect will have on the behavior of other prospectors. The behavioral impacts occur in two ways: 1) increased innovative activity in areas of technology which are compatible to the original claim, and 2) increased activity in the development of compatible technologies.

To borrow Kitch's mineral analogy, when a prospector located mineralizations in a certain area, the usual result was increased prospect activity in that particular area. This increased activity did not affect the prospector's original claim, so long as he had filed the proper claims. Nonetheless, although there may have been other areas with greater amounts of mineralization, the tendency was to prospect the region which had already proved to be mineralized.[89] The greater the permitted claim sizes, the fewer the prospectors could benefit from the find. Although it was understood that a prospector could develop his own prospect, it was uncommon to find one claim holder organizing the exploitation of the entire region, even if he was the first to locate the mineralized area.[90] Each prospector could exploit his particular claim in any manner, creating rivalry in the development of claims. Consequently, Kitch's theory overlooks the fact that this rivalry in exploiting the prospect is deterred when the permissible claim size is too big.[91]

87. *See* Cohen, *supra* note 38, at 551–53 (discussing complementary components).

88. *See* Kitch, *supra* note 5, at 285–86. Although Kitch argues that two unrelated claims may be more efficiently developed under unified control, he does not posit any other substantial interrelation.

89. *See* Grady & Alexander, *supra* note 4, at 313–16. For historical and economic summaries of the California Gold Rush, one of the most famous periods of mineralization activity in U.S. history, see Paula Mitchell Marks, Precious Dust: The American Gold Rush Era, 1848–1900 (1994); John R. Umbeck, A Theory of Property Rights with Application to the California Gold Rush (1981).

90. Railroads provided much of the ability to coordinate mineralization exploitation by developing the regional networking of mineral prospects. *See generally*, Robert William Fogel, Railroads and American Economic Growth: Essays in Econometric History (1964); Sarah H. Gordon, Passage to Union: How the Railroads Transformed American Life, 1829–1929 (1996); George Rogers Taylor & Irene D. Neu, The American Railroad Network, 1861–1890 (1956).

91. *See* Grady & Alexander, *supra* note 4, at 315–16.

The patent system works much the same way. When patent rights are claimed in a particular technology, and that technology is adopted by consumers, there is increased prospecting activity in that technological field.[92] Innovators prospect compatible technological innovations and claim patent rights. The rivalry in developing these complementary or compatible products helps create optimal innovation and consumer choice.[93] If the original patent holder was granted too expansive a claim, there would be no ability to establish any technological prospects, nor would rivalrous innovation occur among compatible products because the patent holder's rights would foreclose the development of all compatible technologies.[94]

Secondary innovations that evolve from the primary prospect are also foreclosed in Kitch's theory. This foreclosure can be exemplified by the mineralization theory as well. Rather than using mineralization generally, let us use a high-nitrate mineralization such as limestone, which is a type of mineralization that enhances the soil for cultivation. A prospector searching for mineralizations of limestone could make a claim for the area in which the mineralization could be found. Nonetheless, there are limitations to the property rights which he claimed. He has no rights in determining how the surface land is used. Therefore, if a farmer determines that the land is highly arable, the prospector has no ability to prevent the farmer from using the land above the prospect for farming. Significantly, the claim holder has the ability to extract the minerals but not to control the information regarding the location of the mineralization, which is useful in determining the most arable land.

In patent theory, the information pertaining to the claim is also released to the public by means of public disclosures.[95] But unlike the limitations in the mineralization example, a patent holder may be able to control the use of the information that is not expressly granted by the patent. If the patented technology gains numerous competitive advantages, the patent holder may be able to control the entire area of technology, thus limiting the prospecting of the technology for alternative or rivalrous purposes.[96] Therefore, the distinction in the analogy, which Kitch overlooks, is that patent law lacks the division that exists between mineral rights and real estate rights. Rather, patent law creates a bundle of rights, which affects the foreclosure of certain innovation. Whereas in the mineral prospect analogy, a mineral rights holder cannot

92. *See* Merges & Nelson, *supra* note 30, at 873–74.
93. *See id*. at 873.
94. *See generally* Oddi, *supra* note 11.
95. *See* 35 U.S.C. § 112 (1994).
96. *See supra* notes 30–52 and accompanying text.

foreclose competition for the real estate in the area of the claim, a patent holder does have the ability to foreclose competition in the relevant innovation market.

Nonetheless, the foreclosure of competition in the innovation market is not likely to occur in every technological field.[97] It is more likely to occur when there are two innovation markets which are closely related and contain interconnected products.[98] These markets, which contain the primary and secondary products, are those which are most susceptible to the harms of innovation disincentives. Because of the symbiotic relationship of the primary and secondary products, either a decrease in innovation or an occurrence of inventive tipping in one market has significant effects in the development of the other market. Therefore, one way to address some of the problems resulting from reward function failure is to spur innovative competition. The following section addresses the theoretical framework of why encouraging innovational rivalry can be an important and functional element remedying reward function failure.

B. *Solving Reward Function Failure: Rivalry in the Innovative Process*

Numerous theorists have offered alternative means to moderate the problems which develop as a result of non-competitive innovation. Adjusting both patent life and scope has been suggested,[99] as has instituting compulsory licensing.[100] Therefore, much of the literature focuses on altering the elements which are generally considered part of the reward function. Rather than modifying the length and scope patent grants, I propose the patent system should be modified in a manner which remedies reward function failure by modifying the nature of the patent grants and technologies for which they are granted. Whereas the modification of patent scope affects the development of follow-on innovations and improvements, it does not address the disincentives to innovate primary products. Overbroad patent scope may deter innovation, but modifying the scope will not remove all of the deterrence created by reward function failure.[101] Rather, the promotion of innovation rivalry, by modifying the application of patent grants, helps counter the systemic problems of

---

97. *See* Merges & Nelson, *supra* note 30, at 881.

98. *See id*.

99. *See id. See also* F. M. Scherer, *Nordhaus's Theory of Optimal Patent Life: A Geometric Reinterpretation*, 62 AM. ECON. REV. 422 (1972).

100. *See generally* Cole M. Fauver, Comment, *Compulsory Patent Licensing in the United States: An Idea Whose Time Has Come*, 8 NW. J. INT'L L. & BUS. 666 (1988). *See also* Ralph Oman, *The Compulsory License Redux: Will It Survive in a Changing Marketplace?*, 5 CARDOZO ARTS & ENT. L.J. 37 (1986).

101. *See* discussion *supra* Parts III.A–B.

reward function failure, and consequently promotes, rather than discourages, innovation.

Competitive innovation also prevents efficiency benefits in the innovative process and protects consumer welfare. In recent years, the use of rivalry as an economic policy goal has been subordinated by the use of efficiency-based standards of competition.[102] Nonetheless, there have been some attempts to establish rivalry as the dominant goal in economics-based law.[103] This resurgence of rivalry, based largely on the arguments that rivalry creates efficiencies in innovation, helps eliminate X-inefficiencies, and increases consumer welfare.[104]

X-inefficiencies occur when firms do not face regular competition in their industry, and thus do not face the same production pressures which occur in a fully competitive market.[105] Although these inefficiencies are normally considered to exist primarily in the production process, there is evidence that such inefficiencies may also exist in the innovative process.[106] In fact, there may be more to lose from X-inefficiencies in the innovative process. Rather than higher unit costs, the harm may be sub-optimal innovation that may have numerous multiplying effects in the innovation process.[107] Although a thorough evaluation of the existence of X-inefficiencies is difficult,[108] some empirical evidence has illustrated that rivalry limits the existence of innovative characteristics that are similar to X-inefficiencies.[109]

Rivalry also enhances consumer welfare by promoting innovative efficiency and choice. The specific determination of what constitutes consumer welfare is varied.[110] In the innovation context, optimal innovation at the lowest cost is beneficial to consumer welfare.[111] By increasing competitive tensions, rivalry forces innovators to maximize their efforts so as to withstand the rigors of the market.[112] Although the costs which are incurred by increased innovative efforts may reduce

102. *See* Dam, *supra* note 3, at 264. *See also* Harry S. Gerla, *Restoring Rivalry as a Central Concept in Antitrust Law*, 75 NEB. L. REV. 209, 210 (1996).

103. *See* Gerla, *supra* note 102, at 211.

104. *See id*. For a definition of X-inefficiencies see *infra* notes 105–109 and accompanying text.

105. *See* ROGER S. FRANTZ, X-EFFICIENCY: THEORY, EVIDENCE AND APPLICATIONS 2–3 (1988).

106. *See* Gerla, *supra* note 102 at 228.

107. *See id.* at 238–39.

108. *See id*. at 227–28.

109. *See id. See also* Merges & Nelson, *supra* note 30, at 873 n. 143. *See generally*, F.M. SCHERER & DAVID ROSS, INDUSTRIAL MARKET STRUCTURE AND ECONOMIC PERFORMANCE 668–72 (3d ed. 1990).

110. *See* Brodley, *supra* note 78, at 1020, 1032.

111. *See id*.

112. *See generally* Merges & Nelson, *supra* note 30.

some allocative efficiencies, innovative efficiencies increase the aggregate consumer welfare benefits above those created by the allocative efficiencies.[113] Consumers benefit foremost from increased choice among innovations of variable price and quality.[114] Another benefit is that rivalry creates the incentive for numerous competitors to explore innovative development in areas which they would not otherwise explore, thus creating the potential for even greater innovative efficiencies.[115] These new innovations create even more consumer choice. Therefore, the promotion of rivalry and innovative efficiencies created by such rivalry provides benefits exceeding pure industrial-based allocative efficiencies.[116]

Although rivalry appears to provide reasonable benefits that create both efficiencies and public welfare, introducing rivalry to innovation markets raises significant questions related to the nature and scope of intellectual property protection. Specifically, the type of patent protection, if there is any to be granted at all, may depend greatly on the type of technology.[117] Additionally, the close relationship between primary and secondary technologies and products may also affect the nature of the protection to be provided. Therefore, when determining the nature of patent protection in light of the policy goal of innovative rivalry, the technologies which raise the greatest number of problems are those which have 1) a high propensity for product success based on the aggregate network externalities which the product creates, and 2) a proximate relationship between primary and secondary products. One technology which matches these criteria is computer software. The following section reviews the dynamics of the computer software market and proposes a framework for evaluating potential types of patent grants which may best enhance innovative rivalry and prevent reward function failure in this particular technology.

---

113. *See* Gerla, *supra* note 102, at 229–30. *See also* Brodley, *supra* note 78, at 1027–33.

114. *See* Brodley, *supra* note 78, at 1033.

115. *See generally* Merges & Nelson, *supra* note 30.

116. *See* Brodley, *supra* note 78, at 1027.

117. *See* Merges & Nelson, *supra* note 30, at 883.

VI. CURING REWARD FUNCTION FAILURE: THE SPECIAL PROBLEM OF COMPUTER SOFTWARE

A. *Technical Aspects of Computer Software*

Computer software programs, at a very basic level, can be divided into two categories: 1) operating systems and 2) applications.[118] Each of these categories provides a different function in computer processing but are co-related. Although the division between what constitutes an application and what constitutes a function of an operating system has been blurred, for the initial purposes of this discussion it is important to understand the distinctions between these types of programs.[119]

An operating system is the program which coordinates all activity on a computer, essentially providing a basic roadmap for the functionality of the hardware.[120] Its key tasks are to recognize and control input and output functions, manage file directories, and manage the behavior of the external peripheral functions of the computer.[121] Additionally, the operating system manages the processing schedule of the microprocessor and controls resource usage.[122]

One of the most essential attributes of the operating system is the software platform.[123] The platform is the basic framework which is used to run application programs. For an application to run successfully on a particular operating system, the program must be designed in a manner that makes it compatible with the operating system's software platform.[124] Normally this design is based on the use of a common set of application programming interfaces (APIs) which allow the application to communicate with the operating system in the same language and can create compatibility.[125] Therefore, the development of a platform and its APIs is a necessary element for the development of applications.

---

118. For an overview of computer technology and the relevant market attributes, see Peter S. Menell, *Tailoring Legal Protection for Computer Software*, 39 STAN. L. REV. 1329, 1334 (1987) [hereinafter *Computer Software*] (discussing computer technology in general as well as the relevant market attributes).

119. Although this paper reviews the traditional distinctions between operating systems and applications, recent action by the Department of Justice and the litigation which has ensued has challenged the definition of an operation system.

120. *See Computer Software*, *supra* note 118, at 1334.

121. *See PC Webopedia, Operating System* (visited October 1, 1998) <http://webopedia.internet.com/Operating_Systems/operatingsystem.html>.

122. *See id*.

123. *See PC Webopedia, Platform* (visited October 1, 1998) <http://webopedia.internet.com/TERM/p/platform.html>.

124. *See id.*

125. *See PC Webopedia, API* (visited October 1, 1998) <http://webopedia.internet.com/TERM/A/API.html>.

Applications are programs that create end-user functionality,[126] such as word processors, spreadsheets, and design programs.[127] Applications are essentially high level sets of commands that use the underlying computer processing ability to serve specific functions. Unlike operating systems which manage system resources, applications are the higher level products of those resources. As stated above, all applications contain Application Programming Interfaces (APIs) which communicate with the software platform. Without APIs, applications will not initialize or function with the software platform, will be unable to request system resources, and will possess no independent use in regard to that particular computer platform.

Therefore, in light of platforms and applications, APIs are essentially the technical program specifications which need to be present for applications (which in this case constitute secondary products) to run properly on a platform (the primary product). The designer of the operating system/platform determines the APIs. An application designer can use any API of his choice, but if he intends for the application to be run on a specific platform, he needs to use those platform-specific APIs.[128]

B. *The Special Role of APIs and Software Platforms in the Application Market*

When a software programmer designs an application program, he must make a choice as to the software platform on which the program will run.[129] This decision is likely to be affected by numerous factors. Foremost, the developer will want to insure that he chooses the operating system which will maximize the commercial success of his application. Therefore, the developer will most likely choose the operating system with the highest commercial success or is popular in the market in which the application is most likely to be used.[130] The network externalities of the most successful operating system provide powerful

126. *See Computer Software*, *supra* note 118, at 1334. *See also* Peter S. Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 STAN. L. REV. 1045, 1050–57 (1989) [hereinafter *Application Programs*]; *PC Webopedia*, *Application*, (visited October 1, 1998) <http://webopedia.internet.com/TERM/A/Application.html>.

127. *See Application Programs*, *supra* note 126, at 1051.

128. Recent technological developments have led to the creation of cross-platform functionality by means of the Java language. Java essentially creates another thin software platform over all other platforms, and allows for applications written in Java to run across any platform supporting Java. The cross-platform quality allows application designers to "write once, run everywhere." Although this technology has been extremely popular in the application industry, it has not been without controversy. For review of the Java controversy and the related legal issues, see Lemley & McGowan, *supra* note 63.

129. *See Computer Software*, *supra* note 118, at 1341–42.

130. *See id*. at 1341.

incentives for the application developer to choose that particular system over others both because of its strong market position and because of the consumer benefits stemming from the network externalities. From the application developer's viewpoint, the application's success is partially dependent on the continued success of the operating system. A commercially unpopular operating system, although technically superior, may neither have sufficient network benefits nor the potential network benefits to attract applications.[131] Therefore, the choice of operating system, from the developer's perspective, is more dependent on market dynamics rather than technical functionality.[132] Although other factors may affect the developer's decision, empirical evidence indicates that developers tend to design applications which are compatible with the most successful system on the market, even if that system is not the most technically superior alternative.[133]

Once the choice of operating system has been made, the developer must acquire the specific APIs which will create interoperability. The proprietary nature of the APIs thus creates the transactional issues which are raised in application development. In almost all cases, the programmer will need to undertake measures to ensure that the application contains all the elements necessary to run on the specific platform. These measures usually consist of the licensing of patent or copyright rights as well as the licensing of trademarks.

C. *Innovation Market Implications of Technical Aspects of Computer Software*

As noted above, a significant implication of the relationship between operating systems and platforms is that the development of applications depends largely on the software platforms of operating systems. The market success of operating systems consequently has an impact on the development of application programs. Those developers who seek to maximize the commercial success of their application must choose compatibility with the most commercially successful operating system. Alternatively, developers of alternative operating systems will find increased difficulty in attracting application developers to choose compatibility with rival systems. The circular result here is that the growth of the compatible application innovation market further expands the competitive advantages of the particular operating system, and the enhanced competitive advantage of the operating system encourages

131. *See Computer Software*, *supra* note 118, at 1342–43.
132. *See id*.
133. *See id.* at 1341–43. *See also* Friedman, *supra* note 68, at 1120.

growth in the compatible application innovation market. The number of rival operating systems is limited by the innovation in the application market. Therefore, to encourage rivalry in the operating system innovation market, one of the roots of the dynamic must be addressed—the intellectual property protection of software platforms.

D. *Preventing Failure: Reevaluating Protection for Software Platforms*[134]

In order to promote rivalry in the development of operating systems, the special nature of software platforms should be reflected by the intellectual property protection which is granted. The key question is how platforms should be protected considering their special nature. Tailoring a type of protection for platforms raises a host of questions, including the determination of what constitutes a software platform and what, if any, proprietary protection should be granted. The answers to these questions raise the possibility that the best way to protect the market regulation function of the patent system in regard to operating systems is to design *sui generis* or hybrid protection for software platforms.

1. Proprietary Protection: Determining the Scope

The first question relating to the special case of computer platforms is how to define the platform itself. One way to perceive the software platform is to recognize it as the central component of the operating system. If this is the case, then the operating system itself may be the property which needs protection. Alternatively, if the software platform is merely one discrete element in the operating system, without which the operating system would still serve a majority of its functionality, perhaps only the code which makes up the routines involved in the software platform should be considered as the platform.

The latter alternative raises another significant question related to which portions of the actual code should be affected by alternative treatment of platforms. Not all of the code which creates the software platform is specifically assignable to the interoperability function of the platform. Portions of the platform code relate to the functions of the user interface, and may direct system resources toward these functions.[135] Although applications usually manipulate the user interface, the

---

134. The following framework for altering the protection of software platforms does not cure the present platform monopoly, rather it provides forward-looking analysis. If the following framework would have been adopted, it would likely have prevented the high concentration of competitors currently existing in the operating system market.

135. The user interface consists of both the look and feel of the computer as well as the user/computer interaction. The interface of a software program may include the way the

platform may synthesize the application interface commands with the operating system's user interface commands. Therefore, when determining the relevant platform code, a distinction may be drawn between the code which creates the interoperability functions of software platforms and the code which creates or directs the user interface.

However, drawing such a distinction may raise problems because interface compatibility may be a significantly desirable commercial attribute of applications. For example, when using the Microsoft Windows platform, a computer can seamlessly combine the Windows interface with the interfaces of applications designed to be compatible with Windows. Conversely, when a user attempts to run a MS-DOS application, the computer must reset the user interface to DOS standards. This reset usually results in the computer switching interfaces during the time that the non-Windows application is being used. Although the change in interface may not have significant functional implications, it does alter the seamless interface created by Windows-compatible applications. Nonetheless, determining the distinction between interoperability code and interface code may be a significant issue in determining the components of a software platform.

Although interface compatibility may not be as technically significant as functional interoperability, dividing the two may have significant commercial effects. By separating the platform code in such a manner, it is possible that application developers could develop functionally interoperable software. However, consumers would still not respond to the applications because of the interface incompatibility. Thus, in this case, excessively limiting the boundaries of the operating system could have drastic commercial effects.

The aforementioned problems all raise one significant issue: the boundaries of the operating system are not easily determined. Perhaps technological innovation will make this distinction clearer. Recent events have highlighted the fact that operating systems can perform much more functionality than basic organizational processing decisions and incorporate new functions which are more commonly associated with software applications. If this is the case, although it may be more difficult to distinguish the applications from the operating system, it may be easier to distinguish the software platform from the rest of the application-type functions of the operating system.

---

program is displayed, the way the program is prompted by commands, as well as the means by which the user can alter or customize the program.

2. Defining the Proprietary Nature of Software Platforms

In addition to the questions regarding scope are the questions related to defining the proprietary nature of software platforms. As a result of their importance in the innovation markets of applications and operating systems, software platforms have several attributes which make it difficult to define their proprietary nature. In fact, software platforms have more similarities to non-proprietary concepts such as languages, algorithms, and industry know-how. Therefore, one possible alternative is to determine whether or not the software platform elements of an operating system are unprotectable because of their underlying importance to the functionality of all other software.

This alternative is rooted in the argument that software platforms serve functions similar to those of algorithms, scientific principles and languages. Although software platforms are not inherently self-limiting in their expression, they are the principal building blocks to software interoperability. As a result, the same justifications which are used to deny patentability to the aforementioned concepts apply to software platforms. For example, an algorithm is not proprietary because it is a mathematical function and not a machine or process and therefore is unpatentable under the Patent Act.[136] The algorithm, as a mathematically created function, is inseparable from the larger structure of mathematical science.[137] By appropriating an algorithm, a patent holder could exclude others (who do not license the algorithm) from performing mathematical functions which may depend on the use or variation of the algorithm. This exclusion not only limits the development of mathematical theory, but also limits the development of technologies which could use the algorithm to perform processes or mechanical functions.

Similarly, a person appropriating a software platform would have the ability to foreclose the development of other software programs and functions. But substitute platforms, market dynamics and innovation disincentives may make the development of such a substitute less likely. By excluding others from the software platform's basic interoperability function, a patent holder would be denying all software developers half of the equation necessary for the development of functional software.[138] Essentially, the interoperability function of a software platform transcends a mere idea or process. It is the principal means of operating

---

136. *See* Parker v. Flook, 437 U.S. 584, 594–96 (1978) (holding that granting a patent to Flook would have the result of patenting a mathematical formula). *But see* Donald S. Chisum, *The Patentability of Algorithms*, 47 U. PITT. L. REV. 959 (1986).

137. *See* Parker, 436 U.S. at 591–92.

138. *See* Pamela Samuelson, *Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer-Related Inventions*, 39 EMORY L.J. 1025 (1990).

software, just as algorithms are a principal means of mathematical analysis. Therefore, rejecting patent protection for software platforms would be both theoretically and legally consistent.

The alternative that software platforms be considered unprotectable does raise a difficult question of what platform components can still be protected by other intellectual property protection. For example, if the APIs are the components that are essential to interoperability, it is possible that the remainder of the platform may still qualify for other types of intellectual property protection such as copyright. A careful distinction must be drawn between these components so that the lack of protection of the platform will still provide incentives for platform innovation while reducing the harms that result from the exclusivity of proprietary rights.[139]

The exclusion of the software platform's compatibility components from patent protection removes two significant barriers to the development of rivalry in the innovation of software. First, the relationship between applications and platforms will cease to be a limiting factor in the development of software platforms. Application developers will not need to make a choice regarding platform compatibility because all of the APIs and other compatibility information will be costless. Therefore, a developer will be able to acquire all APIs for use in an application and not be limited by compatibility costs. Since application developers will not need to make such a choice, the potential for inventive tipping will be mitigated. Consequently, platform developers will no longer be subject to a significant disincentive for inventing a rival software platform.

The second barrier which is removed is that the innovation market dynamics will be altered to recreate benefits which would otherwise be created by the reward function. Although the lack of rewards created by patentability may be dissipated, rival developers may have other sufficient rewards for rivalrous development. If the non-compatibility related components are still protected, a developer may have sufficient justification for developing a new operating system based on an unprotected software platform. Specifically, the benefits received from developing a new operating system may outweigh the lost benefits of releasing a new

139. Such a distinction has been analogized to the idea-expression dichotomy in copyright law. *See* Maximilian R. Peterson, Note, *Now You See It, Now You Don't: Was it a Patentable Machine or an Unpatentable "Algorithm"? On Principle and Expediency in Current Patent Law Doctrines Relating to Compute-Implemented Inventions*, 64 GEO. WASH. L. REV. 90, 122 (1995). Analogizing the machine-computer dichotomy to the idea expression dichotomy, Peterson suggests the application of an abstractions-like test. *See id*. at 123–124. Similar reasoning could be used in supporting an *Altai*-like test in relation to the distinctions between compatibility and interface components.

and improved non-proprietary platform component. Application developers, no longer limited by inventive tipping, will be able to easily enhance the marketability of any new software platform because they can add compatibility easily and without cost. Therefore, operating systems will be a stronger market competitor regardless of whether or not it is the first to market.

Essentially, by leaving software platforms unprotected, the first-mover advantages are significantly limited, and the disincentives they create are removed. As a result of such a removal, second comers are not subjected to the dynamics of reward function failure. Rivalry among platforms will develop because path dependence and lock-in will no longer limit the potential for second-comer success. Rather, the open aspects of software platforms will create more choices for platforms without sacrificing application compatibility.[140] As a result, leaving software applications unprotected will create competition in both the innovation market as well as the product market, and will provide a remedy for the harms of reward function failure.

3. Potential Sui Generis Protection for Software Platforms

The creation of *sui generis* protection for computer programs has been a subject of much academic discussion in recent years.[141] These arguments resonate in the particular situation of software platforms. As noted above, the current protections available for computer programs do not prevent reward function failure in the operating system innovation market. Consequently, alternative protections must be designed to better protect innovation market competition. These protections may be either modified proprietary protections, compulsory licensing requirements, or a limited general public license requirement. Although these three protections are diametrically different (one grants property rights, whereas the other two create a type of public good), they all serve the function of creating competition in the innovation market.

An example of a modified proprietary protection for software platforms is a patent system which leaves certain components unprotected, while designating other components to be patentable. Another alternative is the creation of an altogether new protection which is a hybrid of

---

140. Software platform components may be standardized by industry participants through licensing or other means. *See generally*, Friedman, *supra* note 68, at 1123 to 24; Mark A. Lemley, *Antitrust and the Internet Standardization Problem*, 28 CONN. L. REV. 1041, 1043–54 (1996).

141. *See generally Application Programs*, *supra* note 126; *Computer Software*, *supra* note 118; J.H. Reichman, *Legal Hybrids Between the Patent and Copyright Paradigms*, 94 COLUM. L. REV. 2432 (1992); Pamela Samuelson et al., *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 COLUM. L. REV. 2308 (1994).

patent and copyright laws.[142] For both alternatives, the proprietary rights that are created would need to be sufficiently limited to prevent reward function failure in the innovative market, but be sufficiently broad to promote innovative rivalry. Nonetheless, the *sui generis* protection should be limited only to those software platform components which create compatibility (e.g. APIs) and should not necessarily be extended to other non-essential components.

The second alternative, compulsory licensing of the compatibility elements, is not novel, but it does demand a closer examination in regard to software platforms. The compulsory license may be limited only to those elements of the operating system which compose the software platform. The remaining portion of the operating system code will retain the attributes of a patented invention. Again the difficulty in this hybrid may be in determining the make-up of the platform. This problem, however, does not make it impracticable. The compulsory license may be structured so that the actual design of the compatibility component need not be licensed, but the code needed to make applications compatible with the platform is subject to a license. Such a license would prevent the duplication of the platform but would force the function of the platform to be licensed. This license is similar to the "Automatic Anti-Cloning Protection Followed by an Automatic Royalty-Bearing License" as discussed by Professor Samuelson.[143]

Compulsory licensing does not completely cure reward function failure and the disincentives for rivalrous innovation. In one respect, the compulsory licensing element will prevent reward function failure by limiting the excess competitive advantages of a first-mover. Competing innovators will be allowed to incorporate rival software platforms into their operating system. The overall result of this practice will be to create cross platform compatibility by default.

Nonetheless, the compulsory licensing framework does not solve the problems related to inventive tipping. Although application developers would have the platform's compatibility codes available for licensing, the developer would still need to make cost-based decisions regarding the choice of platform. Assuming application developers cannot license every platform code (unless the licensing fees are insignificant), the application market will still tilt towards one particular platform. An alternative would be to create a license which covers all the software platform compatibility components of all the rival software platforms. Such a universal license fee could then be split among all the software platform owners covered by the license. This license is similar

142. *See* Reichman, *supra* note 141, at 2502–503.
143. *See* Samuelson et al., *supra* note 141, at 2414–15.

to a system proposed for software innovations based on registration and automatic licensing.[144] Furthermore, this alternative might solve the cost-based problems associated with the compulsory license, thus preventing a significant innovation disincentive.

The last type of alternative to a modified proprietary system is a non-proprietary general public license (GPL) system for software platforms similar to the one developed by the Free Software Foundation.[145] The GPL is a hybrid of contract and copyright law, creating proprietary rights while also granting a license which allows the modification of the copyrighted work. The hybrid's goal is to further development of the material covered by the GPL while preventing its appropriability.[146] Although arguments against such a private propertization of intellectual property rights exist, these arguments are based on the claim that the overall benefits to society are based on the underlying principal of property rights. Contrary to this argument, in the context of software platforms, the GPL may produce optimal and beneficial results.

In such a system, the software platform (or at least its compatibility components) would be subject to a GPL and freely distributed. If developers improved the platform, they would be required to disclose the improvements by means of documentation. Although the platform developers would not be able to control the development of the platform, they would be certain that no other developer could appropriate the proprietary rights either. The mandatory disclosure would assure all developers that trade secrets will not balkanize the platform.

Furthermore, there would still be incentive to innovate platforms because the benefits would accrue to the development community as a whole. As a result of such a distribution of benefits, there would be a community-wide effort to develop the platform to its most superior form, thus preventing sub-optimal innovation.[147] While the essential nature of the reward would be different, it would still serve the same function, providing incentives to innovate.[148] An operating system developer, possessing all the compatibility-related information, could then focus on developing other aspects of the operating system. Consequently, competition in the operating system market could be enhanced

144. *See id*. at 2426–29.

145. *See The GNU Project and the Free Software Foundation (FSF)* (visited Nov. 19, 1998) <http://www.gnu.org/home.html>.

146. *See GNU Library General Public License* (visited Nov. 19, 1998) <http://www.gnu.org/copyleft/lgpl.html>.

147. *See* discussion *supra* Part IV.A.

148. Since the nature of the reward function is different under the GPL, coordinated development as opposed to rivalrous innovation would be a more preferable dynamic for the innovation market.

by placing all the developers on the same level in regard to compatibility. Additionally, inventive tipping would become meaningless because all platforms would share the same components, making the applications universally compatible.

Although these alternatives are not flawless, they serve as a starting point for examining what type of legal protection should be tailored for software platforms. In order to prevent the reward function failure in the operating system market, steps must be taken to prevent the occurrence of disincentives to innovate, as well as a basis for rivalry in development.

VII. CONCLUSION

Reward function failure has been a long-neglected occurrence within the patent system and devices to remedy it have been grossly underrepresented in the current workings of patent regulation. As legal and economic dynamics have had the effect of limiting the effectiveness of the reward functions of the patent systems to the point where even the prospect function cannot explain the results, the importance of reestablishing the reward function has become paramount. In order to correct these failures, innovative rivalry must be encouraged in intellectual property systems, whether by modifying the proprietary systems, or by creating a new non-proprietary paradigm.