Fall 12-17-2020

A Preliminary Analysis of how a Software Organization's Maturity and Size Affect its Intellectual Property Portfolio

Daniel Gifford

A Preliminary Analysis of how a Software Organization's Maturity and Size Affect its Intellectual Property Portfolio

A Thesis Presented to The Faculty of the Software Engineering Department

by

Dan Gifford

Bachelor of Science in Computer Science, Georgia Institute of Technology, 2009

In partial fulfillment of requirements for the degree

Master of Science in Software Engineering

Kennesaw State University

December 2020

Abstract

Intellectual property, commonly known as IP, is complex. The four main types of software IP, which is what this thesis will focus on, are patents, trade secrets, trademarks, and copyright. Patents, trade secrets, and copyrights were all studied by this thesis. Software IP is unique in that it can by copyrighted. Different IP owners, which can be businesses of different types, individuals, and universities, often have different strategies as to how to use their IP portfolio. This thesis studies differences in IP usage between these entities specifically in the field of software. Large and small software companies were analyzed specifically.

This thesis attempted to find the differences in computing IP strategies between different stakeholders and explain these differences in as comprehensive a manner as possible. To find answers to the issues at hand, a systematic literature review was performed. A systematic literature review (SLR) is a research method where multiple peer-reviewed articles are gathered and analyzed in a predetermined way. Usually SLRs do not have limits on the number of considered papers. In this work, we conducted a preliminary analysis and focused on 30 peer-reviewed articles. Ten articles from Software Engineering, Economics, and Law were all reviewed. This was necessary because this research lies at the intersection of all three of these disciplines. The results were tabulated and analyzed both quantitatively and qualitatively.

Our initial analysis shows that there are considerable differences in how different IP holders handle their IP. Among these differences, it was found that large companies are more likely to sell patents to small companies. Furthermore, small businesses often do not honor IP contractual agreements at first and then hope they do not get sued because they are too small to warrant the litigation costs.

The SLR research indicated, that with respect to IP, economists agree about copyright and trade secrets for practicing software entities. In almost all instances, economists stated that trade secret techniques, when combined with copyrighting, are superior to patenting. The research also showed that economists were usually in favor of open-source software. The exact findings of the economists will be expanded on further in this paper.

In addition to the findings recorded, various themes found throughout the research literature were cataloged and analyzed. The themes were then evaluated in what was called a "thematic analysis". These differences are discussed in detail in this thesis.

Contents

Table of Figures

1 Introduction

Intellectual property in software, also known as IP, is complex. Software IP can come in many forms: trade secrets, copyrighting, patents, and trademarks. Software IP is different from other technical IP in that it can by copyrighted. The traditional way of thinking about IP is that it gives the owner a temporary monopoly that can be used to prevent competitors from using the protected idea to their advantage [1]. That is just one way of thinking about IP, but it is certainly not the only one [1].

For the best possible results, IP should be considered a protected asset class by the owning or creating firm [1]. It is best practice to consider all IPs in a company's portfolio a distinct asset class, with its own unique characteristics. This is not without good reason. Over 100 billion USD a year are generated from copyright and trademark sales [1]. Both small and large companies[1] benefit from IP. The approach many firms take towards IP is multifaceted. A firm may choose to keep its monopoly on some IP, license out some of its IP to other companies, and sell full patent rights to patent trolls all at the same time. Patent trolls are companies that only litigate patents but do not create IP, and buy intellectual property rights from other players [1]. To make matters more complex, a firm may choose to give a start-up company a reduced royalty rate on some of their IP. This can be done in the hopes that the start-up will do well in the future and be able to pay higher fees at a later date.

To make proper IP management even more important, investment firms consider IP royalties on established IP to have a greater than normal return on investment vs other company assets [2]. This is largely because there is often no work required to maintain IP. Although determining a fair price for IP can be complex, a good rule of thumb is 5% of gross sales on any product being sold by a licensee [2].

Although IP law may seem simple, in the field of software, it is anything but. Software is unique as an asset class. First, and probably most importantly, software patents have an unusually long lifespan of 20 years. This is a very long time in a field that changes every few years [3]. Second it can take prohibitively long to get a patent, 3 to 4 years [3]. Third, it is also expensive to defend patents.

According to *The Washington Post,* it is hard to find an economist who thinks software patents are a good idea [4]. To make matters even more complex, recently, in October of 2020, the Supreme Court is heard arguments between two major software companies, Google and Oracle over basic copyright infringement [5].

1 In this work, we consider small companies to have less than 500 employees.

Nevertheless, over 500,000 software patents have been issued in the United States [6]. Yet many software companies have little to no software patent portfolio. Software patents may be seen as prohibitively expensive for individuals. Obtaining a patent can easily cost $15,000+ [7]. This can be prohibitive for individual inventors but may be a minor expense for major companies such as IBM. Small businesses and individuals simply may not be able to afford such patenting. Furthermore, the patent success rate, the rate of getting a patent through the application process, is only 33% [7]. Software companies may feel that copyrighting their hard-earned software provides all the intellectual property protection they require. Alternatively, the company creating the software in question may align itself with copyleft, such as the GNU Public License, which makes the inner workings of software known to all, and thus not feel the need to protect their investment. In fact, the Free Software Foundation, the group that created the GNU Public License, is against software patents [8]. In the USA, copyrighting is free. This is a distinct advantage for copyrighting over patenting.

Enforcing one's IP can also be expensive. Firms must choose between expensive patent attorney fees, or becoming experts in something that is not part of their core business. Litigation fees "for one case can range from $500,000 through summary judgment to over $4 million through trial" [SR30, PG 317]. To make matters worse, software patent litigation is painfully slow in software timelines. The average time to trial is 2.5 years [7]. Many companies rely on software copyright and licensing laws but have little to no understanding of the tradeoffs between the intellectual property management systems available to them.

At most companies, an all or nothing approach is taken to IP protection. The source code, the code programmers write, is created either with all rights reserved, or made available to the public via a copyleft, such as the GNU Public License. Little to no information is available to companies wishing to determine what the proper license and patent strategy are for their firm. The information available about what licensing options are available, just not which is "best" for a firm right now, is hard to find.

Firms need measurable and quantifiable predictions about what might arise from whatever IP management plan they utilize. Firms also need to understand the risk of patent trolls, and countries/entities that do not respect western intellectual property laws.

This thesis seeks to understand why these shortcomings happen, and under what circumstances software companies can benefit from increased patent usage and licensing selection. The fees associated with self-patenting software may also be decreased as companies, or individual inventors could decide not to patent their software. On the other hand, this research could help companies

decide to patent and license their software more thoroughly and license it out to other companies at a handsome fee.

This thesis completes a partial SLR of among three disciplines, software engineering, economics, and law and found information relating to differences between small and large companies and how they managed and benefitted from different IP portfolios. This was necessary because this research is a mix of all three disciplines. The research questions section of this thesis brings up four key questions that were investigated by the research. The research method section explains how the research was structured, and what needed to be done to execute said research. The analysis and findings section explained what the results of the research were and gave deep insights into the data itself. It also summarizes all the themes that were found by doing the SLR itself. Next, the limitations of the research are addressed. After that, the conclusion section ties everything together and the future work section guides researchers as to what can be explored next.

2 Research Questions

With all of the complexities surrounding software IP, it was decided that some research into the subject would be worthwhile. The purpose of this thesis is to determine the return on investment of a software patent, and the effects of various factors on a software company. As previously stated, IP portfolios can be complex. Stakeholders must juggle temporary monopolies from the IP itself, decide if they want to license out part of their IP portfolio, and determine if they want to buy anything from other firms. There is a large body of work available on the economics of software patents from economists in economics journals, but the literature available usually does not take into account company size, practical uses of different software licensing methods, IP infringement, and company goals.

This thesis will correlate software licenses and company goals vs licensing / patent strategies. It is hoped that the results of this research will be useful for both software researchers and software practitioners. For the reasons mentioned above, it is believed that this research is both timely and relevant.

In this thesis, the researcher wanted to answer the following research questions:

Research Question 1: Is there a relationship between software company size and its usage of patents?

Research Question 2: Is there a relationship between company size and how much a company benefits from patents?

Research Question 3: Is there a relationship between the size of a software company and the type of licensing it uses?

3.1 Do large or small companies benefit from a particular type of software licensing?

Note: these questions specifically pertain to the US patent system. The European Union's system, which does not formally allow software patents as per article 52 of the European Patent Convention, provides for some loopholes, but was not researched [8, 12]. Other governing bodies have different systems. None of these were covered by the research in question.

3 Research Method

Considering the huge body of literature in the field, the variety of sources significant for this work, and the kind of questions considered in this thesis, we decided to conduct a systematic literature review.

3.1 How to conduct a systematic literature review

A systematic literature review is a research method that allows existing research findings to be systematically gathered and quantitatively and qualitatively analyzed. Using this technique, trusted literature sources are searched through, the results are gathered, and then the results are presented via a report. This is an established research method with excellent results [9, 10, 11]. Many published papers showcasing SLR in use were located. The following two articles were used as examples of this. The first article was "A Systematic Literature Review Protocol for User Involvement in Software Development and System Success" [9]. The second article was "Guidelines for Performing Systematic Literature Reviews in Software Engineering" [10].

There are different ways to conduct SLRs, but they all follow analogous steps. In this research, the researcher followed the guidelines provided in the previously mentioned articles [9]. In particular, the following steps were provided:

- Create one or more research questions to answer.
- Derive major search terms from the research questions.
- Conduct pilot testing on major terms to identify relevant terms, synonyms, and alternative spellings that are used in published literature.
- Connect the resulting terms using boolean operators to construct a search string.
- Select a range of online databases, journal archives, and conference proceedings for searching. Customize the search string for the online databases' interfaces, to be applied to the abstracts.
- Retrieve the citations and abstracts from the results and manage these using a citation manager.

- Analyze the articles gathered and keep track of the results via a spreadsheet.
- Compile the findings and present the results to the greater research community.
    - The results of this research can have great impact in letting software companies know the optimal IP route to take. Without said research, companies often are often left in the dark.

3.2 SLR execution

The next few sections explain the keyword identification, the actual searching of the online journal databases, and the data analysis itself. All the activities previously mentioned are critical to the research process.

3.2.1 Keyword identification

After the questions were formed, search keywords were then generated that would help find research articles that address the questions indirectly and directly. There were four key search terms that were used by this study. They were put into research journal search engines for further analysis. The terms are as follows: "software licensing", "software licensing economics", "software patent return on investment", and "software patent economics". These search terms were carefully selected and fine-tuned to yield the best results for the questions at hand. The articles were then read and analyzed to answer all three of the major research questions.

3.2.2 Select the relevant databases

Once the research search terms were decided on, they were entered into the scholastic online search engines, and the results were merged from the different databases that were queried. The law and economics articles were combined into one search result using EBSCOHOST, an aggregator offered by the Kennesaw State University library. For the technical articles, the following search engines were used: ACM Digital Library, IEEE Xplore, Science Direct, Google Scholar, Citeseerx, Springerlink, and MIS Quarterly. The appendix has the full search information used.

The one variation on the guidelines was that the searches on the online search engines were done using full text or default mode. This was done because researching abstracts did not give a perceivable advantage for finding meaningful content about the project at hand. The first 50 technical journal articles in question were reviewed from the merged results of the various technical search engines laid out by the systematic literature review guides.

3.2.3 Exclusion and inclusion criteria

The exclusion/inclusion criteria were formed to sift through the search results. Exclusion/inclusion criteria are a series of simple if statements (in English) that allow a researcher to determine if the findings on hand were valuable [9]. The criteria were applied to the research abstracts that were found. Pre-determining the exclusion criteria also helped prevent researcher bias from being introduced into the research project itself [10].

The complete exclusion/inclusion criteria from the journal article's abstract can be seen below. The article:

- Must be in English.
- Must be a published journal article
- Must not be specific to a foreign country or the EU patent systems.
- Must cover either patent or licensing law.
- Will automatically be accepted if the abstract covers a blend of patents or licensing technologies, economics, and law.
- These abstracts almost always covered big/small company advantages.
- Must use one or more keywords in the abstract.
- Many times the abstract would be irrelevant. This must be filtered for and accounted for by the research.
- Will automatically be included if large/small company size is mentioned.
- Must not be a duplicate article from other search results.

3.2.4 Selection of the papers to analyze

Once the exclusion filtering had taken place, the search terms were combined in the technical and non-technical searches, and the results were excellent. Over 10,000 articles were found to research. There was no shortage of journals to analyze. Links and a ZIP file containing the extra results can be found in the Appendix section of this thesis.

Because this research lies at the center of software engineering, economics, and law, journal articles from all three disciplines were chosen for review. The initial search has identified several related works (all the data will be analyzed in the thesis). The initial goal was to analyze the 30 most relevant works from the three different areas of influence: 10 from economics, 10 from law, and 10 from technical articles.

3.2.5 Data analysis

Once the articles were read and analyzed, the findings in each article were tallied up in a spreadsheet. The findings will be explained in detail later in this thesis paper. This allowed for compiling the findings from multiple documents with ease. The spreadsheet had a list of findings (Y/N, reasons, etc), a citation section, and a notes section. It is worth noting that many of the journals had found similar results. Findings were counted once per paper, with multiples only appearing on different papers in question. In the classification of the papers, we will also perform a thematic analysis

In particular, the research questions were answered using the following analysis techniques:

Research Technique 1: **The regular features were extracted through the labeling of relevant terms in the selected papers. We considered the frequency, correlation, and other traditional statistics to analyze such data.**

Research Technique 2: A thematic analysis was performed to extract the relevant terms that correlate the size of the company and the benefits perceived. The themes were categorized using the guideline papers' categories. The results obtained in this analysis were complemented using quantitative data.

Research Technique 3: Research Technique 3: An analysis was conducted for answering RQ2, but using different goals from the labeling procedure.

3.2.6 Limitations

The research for this thesis did not include a secondary method, but in the last phase of this thesis, we analyzed possible techniques to be used in future work to refine the results obtained with the SLR and address new questions that might arise from the study conducted in this thesis.

4 Analysis and Findings

We conducted our research following the steps describe in chapter 3 and this produce interesting preliminary results that will be described in this chapter. The obtained results contribute to the state of the art as follows:

- Identification of the terms and rules to conduct a multidisciplinary SLR around the problem investigated in this work (already discussed in section 3.2);
- A comprehensive list of papers to be analyzed to conduct a complete SLR. The papers where selected applying the parameters described in section 3.2 (section 4.1);
- A detailed analysis of the paper selected for our analysis (Section 4.2);
- Additional relevant themes which help to better understand the complex situation of IP in software engineering (section 4.3);
- Preliminary answers to the research questions introduced in chapter 2 (section 4.4).

4.1 Repository of selected papers

Thirty articles were found to be analyzed. Ten economics articles, ten legal articles, and ten technical articles were all reviewed. The appendix goes into detail with various titles being given. When a researched article is referenced in this thesis, it will be done with the prefix 'SR'. A summary can be found in the next section. Many more articles could have been found, but the 30 in question represent what this research was concerned with. While limited, the results indicated the research was fruitful.

4.2 Analysis of the selected papers

From the list of papers described in section 4.1, we selected 30 papers by the SLR process mentioned previously in this thesis.

Table 1 reports the basic information about the papers analyzed in this work. The selection includes 28 research journal articles, 1 report and 1 conference paper. The appendix has the full information on the articles in this table.

Citation #	Authors	Article Type	Name of Journal / Conference
1	F. Cugno, E. Ottoz	Journal	*Review of Law & Economics*
2	J. Bess, R. Hunt	Journal	*Business Review*
3	S. Kologlugil	Journal	*Journal of Economic Issues*
4	W. F. Shughart, D. W. Thomas	Journal	*Supreme Court Economic Review*
5	N. B. Niman	Journal	*American Journal of Economics and Sociology*
6	S. J. Shapiro	Journal	*J. Legal Econ.*
7	R. Mazzoleni and R. R. Nelson,	Journal	*Journal of Economic Issues*
8	M. S. Clancy and G. Moschini	Journal	*Appl Econ Perspect Policy*
9	M. Stürmer, G. Abu-Tayeh	Journal	*Sustainability science*
10	B. Coriat and O. Weinstein	Journal	*Socioecon Rev*
11	B. L. Smith and S. O. Mann	Journal	*The University of Chicago Law Review*
12	M. Xin	Report	*Social Science Research Network*
13	A. Leiponen and H. Delcamp	Journal	*Research Policy*
14	B. H. Hall	Conference	*Science Policy Workshop, International Institute for Applied Systems Analysis in Laxenburg, Austria*
15	M. A. Lemley and D. W. O'Brien	Journal	*Stanford Law Review*
16	A. Fosfuri, M. S. Giarratana, and A. Luzzi	Journal	*Organization Science*
17	R. J. Mann and T. W. Sager	Journal	*Research Policy*
18	H. Huang, G. Parker, Y. Tan, and H. Xu	Journal	*Management Information Systems Quarterly*
19	R. J. Mann	Journal	*Tex L. Rev.*
20	D. S. Evans and A. Layne-Farrar	Journal	*Va. J.L. & Tech*
21	C. V. Chien	Journal	*Berkeley Technology Law Journal*
22	C. V. Chien	Journal	*Hous. L. Rev.*
23	C. V. Chien	Journal	*Mich. Telecomm. & Tech. L. Rev.*
24	R. W. Gomulkiewicz	Journal	*Ariz. L. Rev.*
25	M. Risch	Journal	*Iowa L. Rev.*
26	R. G. Bone	Journal	*Colum. J.L. & Arts*
27	V. N. Vasudeva	Journal	*The Journal of World Intellectual Property*

28	M. Risch	Journal	*Berkeley Technology Law Journal*
29	R. A. Hillman and M. O'Rourke	Journal	*U. Chi. L. Rev.*
30	S. P. Miller	Journal	*Stan. Tech. L. Rev.*

Table 1 – Papers analyzed in this thesis

In Table 2, we show how the selected papers are distributed over time. More than half of the articles analyzed were published in 2011 or later (the average publication here is 2011 with standard deviation 6.473).

Software IP is a fairly new occurrence. The graph below shows that most of the papers reviewed were published after 2011. This information shows that the articles analyzed were mostly recently published. Software legal issues are constantly in flux. Having recent data makes getting relevant results easier. If the results had been older, they may have been out of date. Filters for article publication date were not included in the inclusion / exclusion criteria.

Date	# of occurrences
1981-1990	0
1991-2000	3
2001-2010	7
2011-2020	20

Table 2 – Temporal distribution of the selected papers

Table 3: This table shows the type of journals the articles were published in. This table shows the type of research journals that contained the articles to be analyzed. The most popular type of journals were economics journals. This finding makes sense. Often the technical and legal search engines turned up economics articles. Often, technical professionals would be co-publishers in economics journal articles as well. Interdisciplinary journal articles were also common. For example, five journal articles were published in journals that contained a mix of technology and legal issues.

Journal Type Frequency	
Econ	10
Law	8
Tech	5
Econ & Law	2
Tech & Law	5

Table 3 – Journal Type Frequency

Table 4: This table shows which journals had more than one publication in the analyzed group. The *Journal of Economic Issues, The University of Chicago Law Review, Berkeley Technology Law Journal,* and *Research Policy* all had two articles taken from their catalogs. All the other journals only had one article taken from them. The fact that there were repeats is a good thing. It clearly demonstrates that the researchers found journals that had the desired knowledge being pooled.

Journals with multiples	Count
Journal of Economic Issues	2
The University of Chicago Law Review	2
Berkeley Technology Law Journal	2
Research Policy	2
All others	1

Table 4 – Multiple Journal Findings

4.3 Themes

Next is table 5: This table shows the frequency of the themes in the research findings in alphabetical order. Twenty themes, with varying frequency were found in the articles analyzed. If a theme was especially strong in a paper, or essentially came up twice, that was only counted on time. Hence, each theme could only count once per paper. Multiple themes per paper were allowed. For the thematic analysis, whenever a theme was prevalent in each paper, it was noted and cataloged. This allowed for a running count of themes, and their prevalence, to be presented to the reader. The strength of any given theme in a paper was not recorded. This can be further researched but was outside the scope of the work at hand.

Theme	Unique Count

Companies are often sued when they are most vulnerable	1
Economists are against IP	2
Finding a settlement amount is complex	1
GPL / Free Software is best	5
Litigation costs are extremely high for small businesses.	1
More money may make for better patents	1
Open standards are best	2
Patent thickets are not a problem	1
Patent thickets are real and help big companies	3
Patents are a mixed bag	4
Patents are more likely in large companies	1
Patents help small companies gain funding	2
Prize / patent combo is best	1
Pro copyright / trade secret	7
SaaS is most profitable	1
Shorter software patents terms	1
Software patents decrease research intensity	2
Software patents work	2
Tried and true patents are more valuable.	1

Trolls are here to stay	2

Table 5 – Frequency of Themes

Twenty unique themes were found in the thematic analysis of the selected papers. Figure 5 shows the frequency of each theme. Themes that came up more than once in an article were only counted once. Hence, the strength of the theme within the same paper was not taken into account. In the following we analyze the themes more deeply and report some significant quotes.

One research article found that companies were most often sued when they were most vulnerable. For instance, small companies were often sued by patent trolls when they were unable to cover the litigation costs [SR23]. The smaller companies usually settled out of court and paid the trolls to avoid future issues. Twice, economists were found to be against intellectual property [SR4, SR5]. The economists in this camp were against intellectual property because it creates a monopoly. Although this idea may not be within the mainstream, it should not be ignored. Another article found that determining litigation settlement amounts was complex [SR6]. Fifteen factors were identified that should be considered by judges and juries. Five papers sided with free software, such as Linux [SR3, SR4, SR5, SR16, SR18]. These papers sided with free software primarily because the authors believe that information should be free, as this was best for society as a whole. Surprisingly, only one paper had the theme that intellectual property litigation costs were too high for small businesses [SR22]. This is especially true if the losing party had to pay the winning party's litigation costs. The following quote bolsters that finding: "[they] cannot bear the risk of paying the opposing party's costs if, despite the strength of the case, they nonetheless lose in court" [SR22, pg 372]. Risch simply found that "more money may make for better patents" [SR25, pg 1576].

One paper found that more established companies with deeper pockets often had better success litigating their patents, even when the patents were sold to trolls [SR25}. Two articles found that open standards for software were best [SR15, SR20]. Patents disclose how inventions work, so these authors thought that was a good thing. Building software on top of other software is much easier with open standards. One paper found that software patent thickets, while an issue theoretically, were not an issue in practice [SR20]. On the other hand, three papers found that patent thickets benefitted large companies and were a very real threat to small companies [SR7, SR10, SR20]. Two of those papers specifically stated that large companies were using patent thickets to manipulate the market to their advantage [SR7, SR10]. Four articles found that software patents had good and bad traits [SR2, SR11, SR15, SR19]. For instance, one article found a direct correlation between a firm's

research quality and its IP portfolio health [SR19]. Another paper stated that increasing software IP strictness might hurt society as a whole [SR2].

Another paper found that large companies were much more likely to have patents in their IP portfolio as opposed to smaller firms [SR2]. Researchers in two articles stated that patents helped software companies gain venture capital funding during a company's early years [SR17, SR21]. The table below explains things in greater detail.

TABLE ONE: POSITIVE EFFECTS OF PATENTS

	Prerevenue Startups	*Later-Stage Startups*	*Large Firms*
Litigation	Impractical	Useful	Unnecessary
Resources	Scarce	Available	Bountiful
Exclusion Benefits	None	Large	Minimal
Licensing Benefits	Rare	Occasional	Varied
Cross-Licensing Benefits	None	Potential	Large
Information Benefits	None	Significant	Trivial

[SR19, Table 1]

As seen in the table above, large firms often find litigation unnecessary, have "bountiful" resources, and can license out part of their IP portfolio. On the other side of the spectrum, small startups cannot afford litigation, have little resources, and do not benefit from cross-licensing.

It was also found that small software companies do not usually patent their work unless they specifically need venture capital funding [SR17]. One group of researchers stated that IP should be done away with entirely, and all intellectual property should be funded publicly [SR8]. That paper appears to be in the minority. Seven papers found that trade secrets and copyrighting represented the best way for software firms, especially smaller firms, to guard their work [SR1, SR4, SR5, SR11, SR21, SR28, SR29]. It is worth mentioning again that, in America, copyrighting is free. This can be great for many firms. One group of researchers found that "trade secrets and code were more important than patents for transferring software innovation between firms" [SR21, pg 1721]. Two such articles stated that trade secret violations were much easier and more profitable to litigate than

patents [SR4, SR28]. Although it may be considered a stretch, one paper found that software as a service was the optimal choice for selling software [SR12].

One paper advocated for shorter software patents terms, which are currently at 20 years [SR4]. That paper supported different patent lengths for different types of inventions across the board. Two papers found that software patents decrease research intensity [SR2, SR9]. Hoarding of knowledge via a monopoly can essentially create a patent thicket [SR9]. Two articles found that software patents in their current state, while not perfect, benefit society [SR14, SR21]. One of the articles found that mobile phone apps that were patented did significantly better in the marketplace [SR21] One article found that patent trolls were more likely to purchase, and litigate, older patents that had been sustained over time [SR13]. Lastly, two articles found that overall, patent trolls are here to stay [SR13, SR26].

4.4 Findings for research questions

Table 6: This table shows the frequency of the features found in the research literature. The two most common findings were that copyright and trade secrets were better than patenting for small firms and large companies are often pro patent. Findings summarized in this figure were directly used to answer the research questions posed previously in this document.

Finding Description	Unique Count
Companies (small) are often sued when they are most vulnerable	1
Copyright / trade secret is best for guarding work - especially in small companies	5
Copyright is the future of licensing	1
Free software (GPL) is best for small businesses	3
Large companies have IP advantage due to increased resources	1
Large companies pro patent	5
More money may make for better patents	2

Open Source Software is best for society	4
Patent thickets are real and help big companies	4
Patent wars are not new	1
Patents are a mixed bag	3
Patents help small companies gain funding	2
Patents may help smaller firms	2
Pro open standards / re-usable code	2
SaaS is most profitable	1
Small companies and universities see no need for patents	1
Small companies pay less and cheat because they are too small to bother suing	1
Small offenders get off easy (not worth suing)	2
Status quo seems to be working	1
Trade secret is better than patenting	1
Trade secret is easier for large businesses who can obfuscate their code	1
Universities have different rules due to Bayh-Dole effect	1

Table 6 – Frequency of Findings / Features

Research Question 1: Is there a relationship between software company size and its usage of patents?

In general, yes. The research indicated some real differences. Big companies tend to use patents in a defensive manner [SR21]. Large firms create patent thickets to defend themselves, sell off their patents to younger companies, and are less likely to sell their patients to patent trolls [SR7, SR20, SR26]. Smaller companies tend to use patents in an offensive manner, get squeezed out of markets due to patent thickets, buy patents from more established firms, and are more likely to sell their patents to patent trolls. Startups also use patents to secure funding for their ventures [SR17, SR21]. Successful patent litigation attempts seem to be mainly for bigger firms. As one paper put it, “the

most egregious troll suits have opportunistically been brought against small companies , and end users who are ill-equipped to play the expensive 'sport of kings' of patent litigation." [SR23, pg 7].

Research Question 2: Is there a relationship between company size and how much a company benefits from patents?

The research gave a varied answer. Large software companies have a higher probability of getting software patents in the first place [SR25]. Large companies are more likely to sell patents to small companies [SR21]. Large software companies often create patent thickets to manipulate the market to their advantage [SR7, SR20, SR26]. Small startups often gain funding for their work using patents. All of this shows how much patents help large and small companies alike, but no definite answer was found to this question via the research. This is an area where additional research could be most profitable.

Research Question 3: Is there a relationship between the size of a software company and the type of licensing it uses?

The research gave some insight here. One article found that big companies, when in offensive mode, were more likely to use copyright laws than patents to protect themselves [SR24]. This is in keeping with the idea that, patent thickets aside, large companies primarily use patents in a defensive manner. Another paper found a relationship between the age of a company, and its effectiveness at creating good patents [SR25]. This naturally leads to the conclusion that large companies are more effective at patenting their ideas.

Research Question 3.1: Do large or small companies benefit from a particular type of software licensing?

One article clearly showed that copyrighting software and selling it as a service was the ideal solution for software of all types [SR12]. This approach is new. Until halfway through the 2000s, over 90% of all commercial software was sold with an upfront cost with an unlimited lifetime [SR12]. Web apps use the software as a service technique extensively, do not have a problem with software piracy, and make up a very profitable sector of the economy. Economists also showed that GPL / free software was the best technique for small companies [SR3, SR4, SR5]. There are a variety of reasons for this, but the key reason is that there is little to no barrier to entry with GPL software. Firms can couple their GPL product with hardware and/or support and make a satisfactory return. It was also

shown that large companies can benefit from GPL when the costs to enter a particular market are exceedingly high [SR16]. In instances where IP protection is desirable, copyright combined with trade secrets are the recommended path by economists [SR1, SR4, SR5, SR21, SR28]. Once again, in America, copyrighting is free. This is especially helpful for smaller firms because the techniques proposed have little to no legal feeds involved. Being able to afford legal fees is one of the advantages large firms have.

5 Limitations of the Research

While the results of this research are promising, the research has a few limitations:

As discussed in section 3, SLRs are prone to bias. Indeed, researcher bias is always an issue, the search terms might not be all inclusive, the journal databases might return narrow results, and different factors might have more influence than others in the research literature. Moreover, in this work we limit ourself to 30 papers, and for the SLR to be complete we would need to analyze the full list of papers presented in Section 4.1.

In this case, the search terms would need to be refined further to give greater detail to the researcher for the data being analyzed. One finding that was noticed later in the research stage was that adding "software" as a mandatory word in the abstract in EBSCOHOST made the search results much easier to search through. Using the search techniques originally planned on meant that dozens of articles had to be sifted through, when sifting through the new and improved search results would have provided a faster research experience.

Another area that could use improvement was in double checking to make sure all the articles being researched online were actual journal articles. A report and a conference paper made their way into the results. This only happened during the technical searches. The search results for the technical articles were often hard to decipher. In the future, extra care must be taken to ensure that inappropriate and misleading articles do not make their way past the inclusion / exclusion criteria.

There are two special types of businesses that were never investigated independently, the patent troll (NPE) and the research university. NPEs generally have many patents that they own. Because they are professional litigating companies, they should have been treated as a separate business type and researched independently. After all, they operate very differently than product and consulting based businesses.

Since 1980, universities have been allowed to own the patents for the public research they perform [SR14]. Prior to this time, universities had little to no incentive to make sure the research they performed made it to market [13]. This change in business is attributed to be a factor in shifting the technical edge from Japan to the USA in the 80s [14]. From both an employee and revenue

standpoint, universities qualify to be counted as large businesses. Classifying universities as large businesses is problematic, however. They produce research and degrees, which are non-traditional outputs for a business. As such, it is believed that universities should be treated as a separate business class and researched as such. Doing so allows researchers to further investigate the unique IP properties of research universities.

There are also issues with SLR as a technique. The first and most obvious challenge is the possibility of researcher's biases. The researcher's advisor and committee act to offset this. Doing an identical study with a second independent party is another way of double-checking the results. Other typical challenges associated with SLR studies are as follows:

The search terms might not be all-inclusive. This was mitigated by doing research using a committee and advisor to assist in the process.

The inclusion criteria might be too broad or too narrow. This was mitigated by testing the inclusion / exclusion criteria in a dry run.

The journals searched through might return narrow results. This was mitigated by using multiple journal search engines.

All these issues are mitigated by the revision of the advisors and the input of the thesis committee.

Another challenge to the validity of the factors identified in this research is that these factors do not have equal weight in the patent / licensing arena. This issue is difficult to mitigate and will require further research.

6 Conclusion

The research summarized in this thesis clearly shows there are structural differences between how large and small businesses handle their IP portfolios. Although IP should be treated as a distinct asset class, with about 5% of gross profits coming in as royalties from consulting and product based businesses, it has been shown that is not always the case [1]. Based on the research findings, many companies take a rather undisciplined approach to managing their IP portfolio.

Although not in mainstream thought, some economists found that patents may harm society as a whole [SR1]. Niman found the following: "The copyright is therefore in accordance with the moral law—it gives to the man who has expended the intangible labor required to write a particular book or paint a picture security against the copying of that identical thing" [SR5, pg 912]. This warrants further research.

Small and large businesses both bend the rules to their liking. Big businesses create patent thickets that consist of, oftentimes, thousands of patents. These tickets make it nearly impossible for small businesses to compete against large players in the area in which the patent was set up. Strict copyright laws do not have these downsides [SR15]. On the other hand, small businesses often ignore IP protocols entirely and use something of a renegade approach until the company grows to a size worth suing.

The SLR study also provided some light on emerging themes in the industry. Ten technical articles, ten economic articles, and ten legal articles were chosen via a rigorous selection process. The articles were analyzed for findings and themes. The results were entered into a spread sheet and then put into this thesis to be presented to the readers. The research found that small and large companies have substantial differences in how they treat software IP.

Open source software was seen in a positive light and was deemed by most economists to be the gold standard and should be the model of how software should be made. One downside is that GPL software cannot be protected by trade secrets [SR28].

Many economists proposed a prize system, summarized earlier, to reward software professionals for their work. As shown previously, economists have been largely ignored with regards to all forms of IP. This is unfortunate.

Another theme worth mentioning is that trade secrets, when combined with copyright protection, can be beneficial to software companies, especially smaller firms. This is because the litigation costs in trade secret infringement are substantially lower than the costs for patent infringement. Copyrighting software in America is free. This theme is important. It should not be ignored [SR22].

Once again, this research yielded real data as to the difference between large and small companies and how they treat their software IP portfolios. These differences should not be ignored. It is hoped that this research will be beneficial to large and small businesses.

7 Future Work

As is always the case, more research can shed more light on the issues at hand. First and foremost, the research performed could be expanded on by refining the searching criteria and allowing for additional articles to be analyzed. This critical step should yield excellent results.

As stated previously, economists have largely been ignored by the IT community. Despite excellent mathematical proofs, and sound theoretical underpinnings, little that economists have proven about patents has been taken to heart by the industry.

Economists are almost uniform in their support of GPL software [SR3]. This quote shows economists' support for open source development: "For smaller companies, 75 percent indicated that open source promotes additional opportunities for innovation" [SR5, pg 919]. Economists even found a unique motivation system for GPL software, which can be found in *Henry George and the Intellectual Foundations of the Open Source Movement* [SR5]. Economists have clearly shown that open source products can have a place within for profit firms [SR16, SR18]. All of this warrants an investigation. How could one class of thought be completely excluded from the discussion? Surely economists have something to contribute. Once researched, the findings as to why economists have been left out may help prevent this issue in other fields as well.

Both patent trolls and universities also warrant further research. For reasons stated previously, both entities represent special "business types" that are constrained by different rules than the rest of the industry. To recap, universities should be studied as a separate business type because they produce research and degrees. Patent trolls often buy patents, sit on them, and then sue other entities for trying to use the techniques in question [SR26]. Although common, this practice is frowned upon in the industry. For these reasons, individual studies of both entities make sense.

One article found that, in software, having a head start and sales structures having a bigger impact on a firm's profitability than its patent portfolio [SR7]. It was also found that more established companies created higher quality patents [SR30].

As mentioned, the SLR study done for this thesis can serve as the foundation for future research work. A foundation has been laid, and it should be interesting to see if this research aids other researchers in their goal for the truth about intellectual property portfolios.

It is also worth noting that software scales. Often, small software businesses have extreme power in the industries in which they operate with minimal staff. Cataloging businesses by gross revenue as opposed to a headcount may yield even more interesting research.

8 References

[1] J. G. Palfrey, *Intellectual property strategy*. Cambridge, MA: MIT Press, 2012.

[2] A. Poltorak and P. Lerner, *Essentials of intellectual property: law, economics, and strategy*. Hoboken, NJ: Wiley, 2004.

[3] M. A. Lemley, "Software Patents and the Return of Functional Claiming," *SSRN Electronic Journal*, 2012.

[4] T. Lee, "Here's why economists hate software patents," *The Washington Post*, 23-Apr-2019. [Online]. Available: https://www.washingtonpost.com/news/the-switch/wp/2013/07/31/heres-why-economists-hate-software-patents/. [Accessed: 28-Oct-2020]

[5] Hamm, Andrew. "Google LLC v. Oracle America Inc." SCOTUSblog, www.scotusblog.com/case-files/cases/google-llc-v-oracle-america-inc/. [Accessed: 26-Nov-2020]

[6] "Patent Counts By Class By Year JANUARY 1977 -- DECEMBER 2015," *Patent Counts By Class By Year, CY 1977 - 2015*, 01-Dec-2015. [Online]. Available: https://www.uspto.gov/web/offices/ac/ido/oeip/taf/cbcby.htm. [Accessed: 19-May-2020].

[7] "2018 PWC Patent Litigation Study," *2018 PWC Patent Litigation Study*, 01-May-2018. [Online]. Available: https://www.pwc.com/us/en/forensic-services/publications/assets/2018-pwc-patent-litigation-study.pdf. [Accessed: 19-May-2020].

[8] R. Stallman, "gnu.org," *[A GNU head]* . [Online]. Available: https://www.gnu.org/philosophy/software-patents.en.html. [Accessed: 19-May-2020].

[9] M. Bano, D. Zowghi, and F. da Rimini, "User satisfaction and system success: an empirical exploration of user involvement in software development," *Empirical Software Engineering*, vol. 22, no. 5, pp. 2339–2372, Oct. 2017, doi: 10.1007/s10664-016-9465-1.

[10] B.A. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," *Tech. Rep*., EBSE-2007-001, UK (July 2007). URL.

[11] N. Bencomo, S. Götz, and H. Song, "Models@run.time: a guided tour of the state of the art and research challenges," *Software & Systems Modeling*, vol. 18, no. 5, pp. 3049–3082, Jan. 2019.

[12] E. P. Office, "The European Patent Convention," *Article 52 – Patentable inventions - The European Patent Convention, Convention on the Grant of European Patents – (European Patent Convention), Part II – Substantive patent law, Chapter I – Patentability*. [Online]. Available: https://www.epo.org/law-practice/legal-texts/html/epc/2016/e/ar52.html. [Accessed: 07-Sep-2020]

[13] Suddeath, Daniel. Saturday Marks 40th Anniversary of Bipartisan Bayh-Dole Act, www.newsandtribune.com, 7 Dec. 2020, www.newsandtribune.com/news/saturday-marks-40th-anniversary-of-bipartisan-bayh-dole-act/article_beedcd68-38d9-11eb-8a48-ab26a9f59406.html

[14] Loise, Vicki, and Ashley Stevens. The Bayh-Dole Act Turns 30, vol. 2, no. 52, 6 Oct. 2010, pp. 1–5. Research Gate, www.researchgate.net/publication/47357655_The_Bayh-Dole_Act_turns_30.

9 Appendix - Articles Read and Analyzed

[SR1] F. Cugno and E. Ottoz, "Trade Secret vs. Broad Patent: The Role of Licensing," *Review of Law & Economics*, vol. 2, no. 2, pp. 209–221, Sep. 2006, doi: 10.2202/1555-5879.1069.

[SR2] J. Bessen and R. Hunt, "The software patent experiment," *Business Review*, no. Q3, pp. 22–32, 2004.

[SR3] S. Kologlugil, "Free Software, Business Capital, and Institutional Change: A Veblenian Analysis of the Software Industry," *Journal of Economic Issues*, vol. 46, no. 4, pp. 831–858, Dec. 2012, doi: 10.2753/JEI0021-3624460401.

[SR4] W. F. Shughart and D. W. Thomas, "Intellectual Property Rights, Public Choice, Networks, and the New Age of Informal IP Regimes," *Supreme Court Economic Review*, vol. 23, pp. 169–192, Jan. 2015, doi: 10.1086/686477.

[SR5] N. B. Niman, "Henry George and the Intellectual Foundations of the Open Source Movement," *American Journal of Economics and Sociology*, vol. 70, no. 4, pp. 904–927, 2011, doi: 10.1111/j.1536-7150.2011.00802.x.

[SR6] S. J. Shapiro, "Pitfalls in Determining the Reasonable Royalty in Patent Cases," *J. Legal Econ.*, vol. 17, p. 75, 2011 2010.

[SR7] R. Mazzoleni and R. R. Nelson, "Economic Theories about the Benefits and Costs of Patents," *Journal of Economic Issues*, vol. 32, no. 4, pp. 1031–1052, Dec. 1998, doi: 10.1080/00213624.1998.11506108.

[SR8] M. S. Clancy and G. Moschini, "Incentives for Innovation: Patents, Prizes, and Research Contracts," *Appl Econ Perspect Policy*, vol. 35, no. 2, pp. 206–241, Jun. 2013, doi: 10.1093/aepp/ppt012.

[SR9] M. Stürmer, G. Abu-Tayeh, and T. Myrach, "Digital sustainability: basic conditions for sustainable digital artifacts and their ecosystems," *Sustainability science*, vol. 12, no. 2, Art. no. 2, Mar. 2017, doi: 10.1007/s11625-016-0412-2.

[SR10] B. Coriat and O. Weinstein, "Patent regimes, firms and the commodification of knowledge," *Socioecon Rev*, vol. 10, no. 2, pp. 267–292, Apr. 2012, doi: 10.1093/ser/mwr019.

[SR11] B. L. Smith and S. O. Mann, "Innovation and Intellectual Property Protection in the Software Industry: An Emerging Role for Patents?," *The University of Chicago Law Review*, vol. 71, no. 1, pp. 241–264, 2004.

[SR12] M. Xin, "The Impact of Customer Valuation Uncertainty on Software Licensing," Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3481475, Jun. 2018. doi: 10.2139/ssrn.3481475.

[SR13] A. Leiponen and H. Delcamp, "The anatomy of a troll? Patent licensing business models in the light of patent reassignment data," *Research Policy*, vol. 48, no. 1, pp. 298–311, 2019.

[SR14] B. H. Hall, "On Copyright and Patent Protection for Software and Databases: A Tale of Two Worlds," Paper presented at an ESF-IIASA-NSF," in *Science Policy Workshop, International Institute for Applied Systems Analysis in Laxenburg, Austria on*, 1999, pp. 3–5.

[SR15] M. A. Lemley and D. W. O'Brien, "Encouraging Software Reuse," *Stanford Law Review*, vol. 49, no. 2, pp. 255–304, 1997, doi: 10.2307/1229298.

[SR16] A. Fosfuri, M. S. Giarratana, and A. Luzzi, "The Penguin Has Entered the Building: The Commercialization of Open Source Software Products," *Organization Science*, vol. 19, no. 2, pp. 292–305, Mar. 2008, doi: 10.1287/orsc.1070.0321.

[SR17] R. J. Mann and T. W. Sager, "Patents, venture capital, and software start-ups," *Research Policy*, vol. 36, no. 2, pp. 193–208, Mar. 2007, doi: 10.1016/j.respol.2006.10.002.

[SR18] H. Huang, G. Parker, Y. Tan, and H. Xu, "Altruism or Shrewd Business? Implications of Technology Openness on Innovations and Competition," *Management Information Systems Quarterly*, vol. 44, no. 3, pp. 1049–1071, Sep. 2020.

[SR19] R. J. Mann, "Do Patents Facilitate Financing in the Software Industry," *Tex L. Rev.*, vol. 83, p. 961, 2005 2004.

[SR20] D. S. Evans and A. Layne-Farrar, "Software Patents and Open Source: The Battle over Intellectual Property Rights," *Va. J.L. & Tech.*, vol. 9, p. 1, 2004.

[SR21] C. V. Chien, "Software Patents As A Currency, Not Tax, On Innovation," *Berkeley Technology Law Journal*, vol. 31, no. 3, pp. 1669–1723, 2016.

[SR22] C. V. Chien, "Reforming Software Patents," *Hous. L. Rev.*, vol. 50, p. 325, 2013 2012.

[SR23] C. V. Chien, "Holding up and Holding out," *Mich. Telecomm. & Tech. L. Rev.*, vol. 21, p. 1, 2015 2014.

[SR24] R. W. Gomulkiewicz, "Is the License Still the Product," *Ariz. L. Rev.*, vol. 60, p. 425, 2018.

[SR25] M. Risch, "The Layered Patent System," *Iowa L. Rev.*, vol. 101, p. 1535, 2016 2015.

[SR26] R. G. Bone, "Of Trolls, Orphans, and Abandoned Marks: What's Wrong With Not Using Intellectual Property," *Colum. J.L. & Arts*, vol. 42, p. 1, 2019 2018.

[SR27] V. N. Vasudeva, "A Relook at Sui Generis Software Protection Through the Prism of Multi—Licensing," *The Journal of World Intellectual Property*, vol. 16, no. 1–2, pp. 87–103, 2013, doi: 10.1002/jwip.12009.

[SR28] M. Risch, "Hidden in Plain Sight," *Berkeley Technology Law Journal*, vol. 31, no. 3, pp. 1635–1668, 2016.

[SR29] R. A. Hillman and M. O'Rourke, "Defending Disclosure in Software Licensing," *U. Chi. L. Rev.*, vol. 78, p. 95, 2011.

[SR30] S. P. Miller, "Fuzzy Software Patent Boundaries and High Claim Construction Reversal Rates," *Stan. Tech. L. Rev.*, vol. 17, p. 809, 2014 2013.

10 Appendix - Search Findings

Technical Articles:

The technical articles were chosen using the steps listed previously in this document. No one link performs the search. The search results for the top articles will be attached as a ZIP file when this thesis is published.

Economics Articles:

https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,shib&bquery=software+patent+economics+OR+software+patent+return+on+investment+OR+software+licensing+OR+software+licensing+economics&cli0=DISCIPLINE&clv0=LO+system.dis-econ&cli1=FT1&clv1=Y&type=1&searchMode=And&site=eds-live&scope=site&custid=ken1

Legal Articles:

https://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,shib&bquery=software+patent+economics+OR+software+patent+return+on+investment+OR+software+licensing+OR+software+licensing+economics&cli0=DISCIPLINE&clv0=LO+system.dis-lawx&cli1=FT1&clv1=Y&type=1&searchMode=And&site=eds-live&scope=site&custid=ken1