Why should we care about software patents?
Software patents have attracted a disproportionate amount of attention about the patent system.
Software patents have attracted a disproportionate amount of attention about the patent system.
Is the attention on software patents warranted?

Yes. *Software patents are behind a disproportionate share of patent disputes*
As many as 55% of all patent defendants and 82% of PAE ("patent troll") defendants have been sued on the basis of a software patent.

Class-based definition of “software” patent: Graham & Vishnubhakat, Journal of Ec. Perspectives. 27:1 (2013) which notes that this definition may contain false positives and negatives. Based on an analysis by GazelleTech of data provided by RPX Corp. © 2012-current suit #s: 86%/35% PAE/non-PAE, respectively.
Software patents have disproportionately been asserted by PAEs (patent “trolls”). WHY?
Software is abstract. The more abstractly a patent is claimed, the larger its footprint on others.
There is a perception that “bad” software patents are breaking the patent system.
There is a perception that “bad” software patents are breaking the patent system

But “bad” software patents are difficult to weed out
By many measures, PTO examination is just as rigorous of software patents as of non-software (Graham & Vishnubhakat)
Patentable subject matter (101) line-drawing is difficult, impossible?
Novelty and nonobviousness screens (102/103) are costly to apply
Today: If those levers aren’t working how about 112 (the disclosure doctrines)?

Why don’t we more forcefully apply the disclosure law (35 USC 112(b) and 35 USC 112(f)) to rebalance the patent bargain without changing the patent statute?
This Presentation tests the premise that greater application of 112(f) would help. How?

112 (f)  PAE Patents  Technical abstraction framework

We examine how well-supported functionally claimed PAE patents are. Are they “crap”? Or are they actually well-supported?
What we did

1. Develop ways to identify functional claims
2. Apply to PAE and non-PAE patents
3. Look for support for functionally claimed PAE patents
Our analysis creds

Aashish R. Karkhanis  Reg. # 62,572
SCU Law ‘13
B.S., Computer Engineering, Virginia Tech
Patent Prosecutor, 4 Years
Patent Examiner, 2 Years (AU 3714)

Colleen V. Chien  Reg. # 55,062
B.S., Engineering
A.B., Science Technology & Society, Stanford
Full-Time IP Litigator and Patent Prosecutor,
4 Years, Fenwick & West
What we did
Step 1: identify functionally claimed patents

Key words/phrases
[see, e.g. Lemley 2013 & MPEP]

“configured to”, “permitting…”,
“programmable means for,” “capable of
engaging,” “adapted to,” “for…ing,”
“operable to…”, “mechanism”,
“data processing system”
“mechanism for,” “module for,” “device for,”
“unit for,” “component for,” “element for,”
“member for,” “apparatus for,” “machine for,”
or “system for.”

Thanks to Bob Hulse (Partner, Fenwick & West) for help with method based
(step + function) claiming
Step 2: Apply it to PAE and non-PAE litigated patents
The Patent Freedom Dataset – 10 PAE litigated patents, 1 each selected from the following campaigns

<table>
<thead>
<tr>
<th>Defendants</th>
<th>Lawsuits</th>
<th>Patents</th>
<th>Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>GeoTag Inc</td>
<td>435</td>
<td>115</td>
<td>Associating online information with geographic areas</td>
</tr>
<tr>
<td>ArrivalStar</td>
<td>326</td>
<td>211</td>
<td>Vehicle tracking and notification</td>
</tr>
<tr>
<td>PJC Logistics LLC</td>
<td>281</td>
<td>44</td>
<td>Vehicle tracking and monitoring</td>
</tr>
<tr>
<td>Lodsys LLC</td>
<td>106</td>
<td>36</td>
<td>Customer-based product design module</td>
</tr>
<tr>
<td>Blue Spike LLC</td>
<td>79</td>
<td>56</td>
<td>Digital fingerprinting</td>
</tr>
<tr>
<td>Datatem Inc</td>
<td>70</td>
<td>28</td>
<td>Interfacing object oriented software applications with relational database</td>
</tr>
<tr>
<td>Ogma LLC</td>
<td>32</td>
<td>10</td>
<td>Programmable motion-sensitive sound effects device</td>
</tr>
<tr>
<td>Kelora Systems LLC</td>
<td>27</td>
<td>11</td>
<td>Guided parametric search and retrieval</td>
</tr>
<tr>
<td>Project Paperless LLC</td>
<td>3</td>
<td>3</td>
<td>Distributed electronic document management</td>
</tr>
<tr>
<td>Single Touch Systems Inc</td>
<td>1</td>
<td>1</td>
<td>Management and administration of media streaming</td>
</tr>
</tbody>
</table>

Includes a small number of DJ cases where the operating company is a plaintiff

© 2013 PatentFreedom. All Rights Reserved.
The Patent Freedom Dataset – control group of 20 non-PAE litigated patents

Half highly litigated, half randomly selected
Submission will include details
Step 3: Evaluate per a textbook technical abstraction framework

Software Construct

- Functional Abstraction
- Abstract Data Type
- Pseudocode/Native Code
- Data Structure
- Source Code

Carrano and Prichard, Chapter 3: “Data Abstraction, the Walls”
Case Study Examples – 5 litigated PAE patents
### Step 3: Evaluation per a textbook technical abstraction framework

<table>
<thead>
<tr>
<th>Software Construct</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Functional Abstraction</strong></td>
<td>Conceptually, what the software program will do.</td>
</tr>
<tr>
<td><strong>Abstract Data Type</strong></td>
<td>A collection of data and set of operations on them.</td>
</tr>
<tr>
<td><strong>Pseudocode/Native Code</strong></td>
<td>A set of instructions that specifies the operations that collectively achieve the function.</td>
</tr>
<tr>
<td><strong>Data Structure</strong></td>
<td>A programming language construct that stores a collection of data.</td>
</tr>
<tr>
<td><strong>Source Code</strong></td>
<td>Human-readable computer code before it is compiled into machine readable object code.</td>
</tr>
</tbody>
</table>
Case Study Examples – 5 litigated PAE patents
“Geolocation/ Where’s the closest Starbucks?”
U.S. 5,930,474 Asserted by GeoTag

City of Los Angeles, Ca.
Folders

Our Town (27 of 27)
- Amusement Parks
- Beaches & Harbors
- Calendar
- Chamber of Commerce
- City Government
- Clubs & Organizations
- Convention Center

435 defendants
115 lawsuits
1 patent
45 pages
delivering info “such as business services, entertainment, news, consumer goods” for a user’s local area

See U.S. Patent No. 5,930,474 at col. 9, lines 28-35.
“... if a user is interested in finding an out-of-print book, or a good price on his favorite bottle of wine, but does not want to travel outside of the Los Angeles area to acquire these goods, then the user can simply designate the Los Angeles area as a geographic location for which a topical search is to be performed ... the geographic topical organization format provided in accordance with the preferred embodiment provides the user with a valuable Internet organizing tool”

U.S. Patent No. 5,930,474 at col. 7, lines 5-29.
U.S. Patent No. 5,930,474 at fig. 2C.
“This parameter may be used by the **Read subroutine** 320 whenever there are more than 50 entries in a list and scrolling is to be supported. In a preferred embodiment, the **first search has this value always entered as zero**, and subsequent scroll searches increment this value to **support scrolling**. Finally, the **NameKey parameter** indicates the name of the folder to display … Any **entry whose parent folder name matches the name specified** will be **returned** by the search.”

U.S. Patent No. 5,930,474 at fig. 20.
<table>
<thead>
<tr>
<th>TABLE 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-type: text/html</td>
</tr>
</tbody>
</table>
| <HTML> <HEAD> <TITLE> City of Irvine, CA: City 1.6.3 </TITLE> </HEAD> <BODY background="/gifs/rock1.gif"> <center> <B> City of Irvine, CA </B> <br> <B> Folders </B> </center> <br> <DL> <DT> <IMG SRC="/icons/folder_open.gif" hspace=10 border=0> <B> School Listing </B> </DT> <DD> <IMG SRC="/Bullets/yellow.gif" hspace=10 border=0 ALT=""> <A HREF="/search/yp2/nap/us/ca/\n\n\n\nse\n\n\ncity\n\n\nSTDYP=KeywordListing+0+Colleges__&__Universities"> Colleges & Universities </A> </DD> <DD> <IMG SRC="/Bullets/yellow.gif" hspace=10 border=0 ALT=""> <A HREF="/search/yp2/nap/us/ca/\n\n\n\nse\n\n\ncity\n\n\nSTDYP=KeywordListing+0+Preschool__&__Kindergarten"> Preschool & Kindergarten </A> </DD> <DD> <IMG SRC="/Bullets/yellow.gif" hspace=10 border=0 ALT=""> <A HREF="/search/yp2/nap/us/ca/\n\n\n\nse\n\n\ncity\n\n\nSTDYP=KeywordListing+0+Secondary__&__Elementary"> Secondary & Elementary </A> </DD> <br> </DL> <p> Send your letters and comments to: <A href="mailto:webeditor@mail.zland.com"> webeditor@mail.zland.com </A> </p> Copyright &copy; 1995 Z Land, LLC. All rights reserved.  
</BODY> 
</HTML>  

“The Light Saber”
U.S. 6,150,947 Asserted by Ogma

32 defendants
10 lawsuits
1 patent
14 pages
U.S. Patent No. 6,150,947 at col. 7, lines 17-25; cols. 9-10.
“Is my Train on Time?”
U.S. 5,223,844 Asserted by PJC Logistics

281 defendants
44 lawsuits
1 patent
41 pages

Ex Parte Reexam
Claims Amended
“In a preferred embodiment mapping 182A displays a general area coverage map a relatively large area, such as the 14 counties around the Dallas/Fort Worth metroplex area. **Mapping displays 182B, 182C, and 182D may be used display vehicle locations for both stolen vehicle and motorist assistance calls on much smaller maps.**”
Functional Abstraction in ‘078 (Lodsys)

Pseudocode Describing Handling of Variables

Functional Description of User Interaction

Functional Description of User Interaction Preferences

U.S. Patent No. 7,222,078 at fig. 23.
“Printing e-Documents”
U.S. 6,185,590 Asserted by Project Paperless

3 defendants 3 lawsuits 2 patents 40 pages
“Loading and unloading the engine (DLLs provided into and out of memory)*

**Mapping original functions to engine object counterparts**

Adding general error detection and correction*

**Determining and matching arguments and return values** for mapping the original functions to their engine object counterparts In order to add assertion and error detection and correction, the original function must be wrapped and called from within the engine object version of the original function.

Managing error feedback. All APIs have their own way providing error feedback. Since one of the goals of the Engine Management layer is to generically manage error detection, correction, and feedback, it must handle all errors identically … **By creating specific classes of APIs the process of generating Layer 1 engine management may be expedited manually and/or automatically.**

U.S. Patent No. 6,185,590 at col. 17, lines 29-50.

*source code disclosed: U.S. Patent No. 6,185,590 at cols. 15-16.
Our findings: all 10 PAE patents were functionally claimed, but the supporting disclosure varied

We found (N=30):

PAE litigated patents were always functionally claimed (100%), but functional claiming was also prevalent among non-PAE litigated patents (50%)

Among the 10 PAE patents, the supporting disclosure varied significantly, 40% of the patents contained only functional abstraction, but the other 60% contained more, e.g. pseudocode and ADT type disclosure

“Not all code is created equal” the contribution conferred via pseudo or source code varied. Source code over generic steps didn’t add much.
Implications

Does functional claiming correctly identify the problem?
Yes but may be overinclusive? Applies to non-s/w patents too. Narrow to PoN FC?

What is the payoff for construing more claims as 112(f)?
Existing patents and applications likely to be invalidated – 40% of PAE patents didn’t include more than functional abstraction. Others will be narrowed in scope.

How should supported claims be construed?
Need clarity around this to avoid creating even more uncertainty. What are equivalents of ADT, pseudocode, source code?

What would heightened application of 112(f) do to filing incentives?
Better disclosure. Delayed application.

Recommendation: if guidelines, phased introduction of them to allow prosecutors time to change their practices.
Thank you!