

COPYRIGHTABILITY OF JAVA APIs

Pamela Samuelson, Berkeley
Law

Stanford EECS Colloquium
Feb. 25, 2015

PRELUDE

- I have been writing about SW IP issues for more than 30 years
- CONTU Revisited on SW © (1984), Benson Revisited on SW patents (1990), Manifesto on SW as IP misfit (1994)
- I've written briefs in several SW IP cases, including one in support of G's appeal to SCT
- Write Legally Speaking column for CACM
- I am an independent thinker on the issues, not a spokesperson for any company or organization

OVERVIEW

- Setting the stage: programs as IP misfits
- Review of the key legal issues in the *Oracle v. Google* case, how decided by lower courts
- What Google & Oracle are arguing to the SCT
- Main precedents on which G & O rely
- What might the SCT do with the case?

SW AS MISFITS WITH ©

- Source code falls within “literary work” category
- But object code = functional process
 - Is it a literary work too? Congress decided yes
 - So literal copying code = infringement
- What if any nonliteral elements of programs are eligible for © protection?
 - Dramas may be “substantially similar” even if words are different (e.g., sequence of events within scenes)
 - Structure, sequence & organization (SSO) of SW?
 - Command structures of UIs, of APIs?

SW AS MISFIT WITH PATENTS

- Because SW is a technology, it would seem to be patent SM
- But programs are also writings, and “printed matter” doctrine arguably excludes them
 - Besides, patent on program would not be worth much; too easy to rewrite & avoid infringement
- More abstract elements of programs
 - Algorithms, data structures, APIs = are they mental processes? Too abstract for patents?
 - Recent SCT cases seem to ? SW patents

U.S. CONSTITUTION

Article I, sec. 8, cl. 8:

- Congress has the power
- To promote the *progress of science & **useful arts***
- By granting to *authors & **inventors***
- Exclusive rights for limited times
- In their respective *writings & **discoveries***

CONVENTIONAL CATEGORIES

- © is for “authors” of artistic & literary works
 - Only function of ©’d works is to convey information or display an appearance
 - Authors get automatic protection for life +70 yrs
- Patents are for “inventors” in “useful arts”
 - Machines, manufactures, compositions of matter, & processes
 - Must apply for protection, claim specifically, have claims examined, pay maintenance fees
 - No more than 20 yrs of protection
 - Subject to invalidation (e.g., not novel, obvious)

BAKER v SELDEN (1880)

- Roles of patent & © discussed in this case
- Selden devised a new bookkeeping system & published a book
 - Some textual explanation, but mostly forms
 - Columns and headings to illustrate the system
 - Baker published a book with very similar forms
 - Selden's widow sued for infringement
 - Claimed © in the system as well as the book
 - Lower court ruled in Selden's favor

SUPREME COURT IN *BAKER*

- Selden's © protects his *explanation* of the bookkeeping system
- System itself is a “useful art” for which no exclusive rights except from patent law
- Preface to Selden's book mentioned his application for a patent on the system
- SCT: would be a “surprise & fraud on the public” if creator got patent-like protection from © without satisfying patent law requirements
- Takes categorical exclusivity approach:
 - Creations are either writings or useful arts
 - If writing, then ©; if useful art, then patent
 - (But of course, computer programs are a bit of both!)

1976 © ACT

- Sec. 102(a): © subsists in original works of authorship (OWA) fixed in a tangible medium of expression
- Sec. 102(b): in no case does © for OWA extend to any idea, **procedure, process, system, method of operation**, concept, principle, or discovery, regardless of how it's embodied in the work
 - Congress added (b) to ensure © in SW would not be too broad: what do the PPSMO words mean?
- When merger of idea & expression, no © prot'n

3 PHASES FOR SW PATENTS

- 1970s-mid-1980s: SW rarely pat'ble
 - *Benson*: algorithm for transforming binary-coded decimals to pure binary form unpatentable
 - *Diehr*: rubber-curing process using SW patentable
- Mid-1980s-mid-2000s: virtually always pat'ble
 - Some patents on APIs, many on other SW innovations (many may be invalid now)
- Mid-2000s-now: not pat'ble if for an “abstract idea” or “mental process”
 - *Alice v. CLS Bank*: SW-implementation of financial settlement process held unpatentable

DISTRICT COURT IN *OvG*

- Oracle sued Google in 2010 for copying the structure, sequence, & organization (“SSO”) of 37 Java API packages (shortly after Oracle acquired Sun)
- DCt (2012): these APIs are not protectable by © law:
 - SSO of Java APIs = command structure = 102(b) system or method of operation
 - APIs more appropriate for patent than © protection
 - Idea of the APIs merges with expression because rules of Java syntax dictate use of same names to invoke Java functions
 - Google used Java APIs so Java apps could run on Android (i.e., APIs needed to achieve compatibility)
 - Prior cases had treated APIs as unprotectable elements of SW
 - Java programmers expect to be able to use these APIs to write apps for the Android platform

CAFC RULING IN *OvG* (2014)

- No merger of idea & expression because Java APIs are creative
 - Sun engineers had many choices in designing APIs
 - Merger is only available as defense when there's no other way to express an idea, or implement a function
- 102(b) restates distinction between ideas & expressions
 - G's 102(b) argument would make all programs uncop'ble because all are processes, so it's wrong
 - Interoperability issues may be relevant to Google's fair use defense, but not to availability of © protection

OTHER ISSUES IN CAFC'S RULING

- CAFC accepted Oracle's argument that the 6000+ method headers are source code & G exactly copied those lines of code
 - Plus G copied "SSO" of Java classes in Android
- G copied these APIs after negotiations with Sun to use Java failed (i.e., knew it needed a license)
- G intentionally designed Android to be incompatible with Java technologies
- Patentability of APIs is irrelevant to © in them

GOOGLE WANTS SCT REVIEW

- Google has asked the SCT to hear its appeal from the CAFC ruling, arguing:
 - Case of exceptional importance to the SW industry
 - Conflict among appellate courts as to
 - Interpretation of 102(b)
 - Interpretation of merger doctrine
 - Significance of interoperability
 - APIs are patent, not ©, subject matter
 - SCT itself split 4-4 on 102(b) issue in *Lotus v. Borland*; failure to resolve issue then has caused split to deepen
 - 6 amicus curiae briefs in support of G's petition

ORACLE SAYS “DON’T TAKE”

- Oracle’s brief contends:
 - No circuit split on ©’ty of source code, which O claims Java API method headers are
 - G literally copied 7000 lines of this code
 - G also copied SSO of Java API package classes
 - CAFC correctly interpreted 102(b) & merger doctrine
 - Case was remanded for fair use determination, so SCt review not ripe because no final judgment
 - Just because APIs may be patentable doesn’t mean they can’t be copyrightable too

AMICUS BRIEFS TO SCT

- In support of Google:
 - 77 Computer Scientists (EFF)
 - Public Knowledge
 - Hewlett-Packard, Red Hat, & Yahoo!
 - Open Source Initiative, Mozilla, & Engine
 - Computer & Communications Industry Association
 - Intellectual Property Law Professors (me)
- In support of Oracle:
 - Free SW Foundation (huh? afraid what SCT wd do)

SCT REVIEW?

- Unlike intermediate appellate courts, SCT doesn't have to take appeals
- More likely to take when a circuit split exists
- SCT sometimes asks the Solicitor General to review case and recommend whether SCT should take or not
 - Referral to SG makes SCT review 46X more likely
 - SCT asked SG to review *GvO*, but review will take time
 - SCT likely to decide whether to take this spring, but case will almost certainly be heard next fall if the SCT decides to take G's appeal

WHAT HAPPENS IF SCT SAYS NO?

- Case would be remanded to DCT for consideration of G's fair use defense
 - G came within one vote of fair use to jury in 2010
- CAFC decision won't bind any other court
 - May not be followed because it's inconsistent with other appellate precedents
 - May give rise to more litigation, though
 - *Cisco v. Arista*: © infringement for A to reuse command line interface (i.e., same way to transmit networking instructions to routers)?

WHAT WILL SCT DO?

- I think they'll take the case:
 - 4-4 split on same legal issue in *Lotus v. Borland* in 1995
 - Circuit split has not been resolved, deepened
 - Two titans of the SW industry
 - Huge impact on Android mobile phone market, app developers
 - Even huger impact on SW industry
 - Long-time assumption that OK to reimplement APIs
 - SCT reverses CAFC a lot lately
- I predict the SCT will reverse if they take G's appeal
 - But far from a sure thing

HOW COURTS DECIDE

- Look at the statute
 - 102(a) & 102(b) are most relevant provisions
- Look at the legislative history
 - If statute isn't clear on its face, what guidance did Congress provide through reports on law
- Look at the caselaw
 - *Baker* & its progeny; prior SW cases
 - Pattern-matching exercise
- Look at amicus briefs, treatises, & commentary
 - What impacts will the SCT's decision in the case have on third parties?

CASES ON WHICH O RELIES

- *Whelan v. Jaslow* (1986):
 - J infringed by copying SSO (file & data structures + some subroutines) from W's dental lab program
 - SW = literary work; © protects SSO of literary works; so © protects SSO of SW too
 - Merger only if no other way to perform function
- *American Dental Ass'n v. Delta Dental* (1997)
 - Delta copied taxonomy of names of dental procedures & #s representing those names from ADA for its guide to dentists
 - Although ADA acknowledged these were part of a “coding system,” CT AP said names & #s were ©'ble because other choices available; not dictated by function

CASES ON WHICH G RELIES

- *Lotus v. Borland*:
 - Borland developed an emulation interface for QP that allowed users experienced with 1-2-3 to run macros constructed in Lotus macro language
 - Command hierarchy for emulation IF had exactly the same commands in exactly the same order
 - Commands had to be the same and in the same order for the macros to execute properly
 - DCT (1992): selection & arrangement of command words was creative; so B was infringer, relying on *Whelan*
 - CT AP (1995): command hierarchy was “method of operation” excluded from © by sec. 102(b), *Baker v. Selden*
 - SCT: 4-4 split (1995)

CASES ON WHICH G RELIES

- *Computer Associates v. Altai* (1992):
 - Substantial similarities between CA's & Altai's parameter lists & list of services for their scheduling programs for IBM OSs
 - No infringement because of constraints due the need to achieve compatibility
- *Sega v. Accolade* (1992):
 - A reverse-eng'd Sega SW to get access to information needed to adapt its videogames so they could run on Sega's Genesis platform
 - CT AP: Reverse eng'g copies were fair use, only way to get IF information
 - + Sega IF was unprotectable "procedure" under 102(b)

LOOKING AT BIG PICTURE

- Sun's goal in creating Java was to overcome MS monopoly by making APIs open
- Sun was the foremost proponent of no © (or patent) protection for APIs in U.S. & abroad
- Sun cultivated a Java developer community
- Huge success: 6.5 M Java programmers worldwide
- Oracle says G should have developed own Java APIs, but that would fracture Java much more than G's use of the 37 APIs has
- Java developers have created a very large number of Android apps, using the 37 Java APIs
- If Oracle's claim succeeds, everybody who owns an Android device is an infringer, not just Google + app developers + device mfrs
- More importantly, SCT ruling in Oracle's favor may change rules of the road for all programmers: reimplement API = infringement!